UNIVERSITAT DE BARCELONA

Deep Multimodal Learning for Egocentric Storytelling and Food Analysis

Marc Bolaños Solà

UNIVERSITAT DE BARCELONA

DOCTORAL THESIS

Deep Multimodal Learning for Egocentric Storytelling and Food Analysis

Author:
Marc BOLAÑOS SOLÀ

Supervisor:
Dr. Petia RADEVA

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Doctorat en Matemàtiques i Informàtica
Departament de Matemàtiques i Informàtica

October 1, 2020

UNIVERSITAT DE BARCELONA

Abstract

Facultat de Matemàtiques i Informàtica
Departament de Matemàtiques i Informàtica

Doctor of Philosophy

Deep Multimodal Learning for Egocentric Storytelling and Food Analysis

by Marc BOLAÑOS SOLÀ

The world of Machine Learning and Computer Vision has experienced a revolution since the last years. The appearance of Deep Learning algorithms and Convolutional Neural Networks, altogether with the increased processing capabilities provided by modern GPUs and the enormous amounts of annotated data publicly available, have allowed a boost in the field as never seen before.

These notable improvements achieved in the Machine Learning world have led to the appearance of new fields like the Multimodal Learning, which encompasses and learns from many subfields. Additionally, new applications have taken profit of these advancements in order to reach high levels of performance. The huge results improvement of the currently available algorithms have allowed not only revolutionizing the academic world, but also bringing AI-based solutions to the market that looked like science fiction barely 10 years ago.

This thesis, which is written as a papers compendium, focuses on delving deeper into the novel topic of Deep Multimodal Learning by proposing new algorithms and solutions for both already existing and newly defined problems. From the applications perspective, most of the papers presented can be divided in two areas of applicability. From the one hand, Egocentric Vision and Storytelling, which consists in acquiring images from the daily life of a person in order to analyse its behaviour patterns like social interactions, activities and events, interactions with objects, etc. And on the other hand, Food Recognition and Analysis, which consists in visually analysing and recognizing the food appearing on images in multiple contexts and with different levels of complexity, from food groups recognition to nutritional analysis.

In both applications, the final purpose of the proposed papers is building tools that provide information that could lead to a better quality of life of the users.

Acknowledgements

Many days have passed since the beginning of my thesis, many projects and ideas have been developed, and some others have been discarded. I have met many people since then. Some have been close since the beginning, but regarding others, we have seen our professional or personal paths diverge from each other.

Even though, I would like to thank all colleagues, friends and family, both old and new, no matter if they were there at the beginning or at the end of this path, because all and every one of them have shaped in one way or another both my life and the way I have worked in the development of this thesis.

I will not be able to mention all the people that have influenced and have been close to me during all these years, but at least, I would like to mention and thank some of them.

I would like to thank my parents, Imma and Alfredo for all their support and their encouragement since the beginning of my academic and professional career in the university. I would like to thank my girlfriend, Cristina, for her love and for always encouraging me looking up to my work. She has always been very supportive whenever I have been busy during long working hours and never-ending meetings. Many thanks to my grandparents, who have also always looked up to my career. And I would like to thank many other family members and friends that have also been supportive to me or with which I have shared some important moments during this journey: thanks to my sister Sheila, to Nil, to Alex, to my uncles, to Trini and to Pol.

I would like to thank my supervisor, Petia, who has always believed in me and in my professional capabilities. Thanks to Estefanía, who has been a colleague and a friend to me since the very beginning, thanks for all the support and long professional and personal discussions. Thanks to all my colleagues for all the moments and experiences that we lived together, Maya, Bea, Edu, Mariella, Juan Luis, Pedro, Eduardo, Bhalaji, Rupali, Andrés, Álvaro Peris, Martín, Alejandro, Gabriel, Axel and Albert. Thanks to Xavi Giró and to Paco Casacuberta for the professional advice. And thanks to my colleagues now at LogMeal and LogMask for building together these amazing projects, to Marta, Pritomrit, Arnau, Eric and Francesc.

All of you have been important people in some way or another at some point during this journey. Many good experiences and events have happened since the beginning, and some others not that good. But one can never forget that all these experiences are what have built and enriched who I am right now as well as how has this thesis been developed.

Thank you all.

Contents

Chapter 1

Introduction

1.1 Deep Learning and Multimodal Learning

It has not been until recently (Krizhevsky, Sutskever, and Hinton, 2012) that the raise of Machine Learning, and more precisely Computer Vision, has made a qualitative difference in real world data, allowing to create algorithms that can become solutions for problems in our daily life.

This huge step forward has been possible thanks to three main contributions that, although not all were directly associated to the field of research, allowed to make it grow.

The first contribution was the increase in computational power and the appearance of powerful Graphical Processing Units (GPUs) (Raina, Madhavan, and Ng, 2009) that allowed operating over tens and hundreds of images per second.

The second contribution was related to data availability, which with the appearance of crowdsourcing platforms, made possible the creation and distribution of large annotated real-world image datasets that allowed training powerful computer vision algorithms. The most notable example was ImageNet, a dataset with millions of annotated images for object detection and recognition (Deng et al., 2009).

And the last, but not the least, the re-discovery of Convolutional Neural Networks, which were initially proposed in (Fukushima and Miyake, 1982), allowed to automatically extract patterns and learn the structure of these large real-world datasets and thus, develop inference algorithms on real-world scenarios.

In the following subsections the main characteristics of Machine Learning and Computer Vision since its appearance will be analysed as well as the key factors for their recent boost in performance and its technological impact and applications.

1.1.1 Machine Learning and Computer Vision

The field of Machine Learning can be described as the study and development of computer algorithms that allow to automatically learn patterns and structures on a set of data. This learning procedure allows to apply an automated prediction on new data that has never been seen before by the machine.

This learning procedure can be applied on nearly any kind of data as long as it is represented in a numerical way for the computer to read it. Some examples of this could be floods detection (Lopez-Fuentes et al., 2017), medical diagnosis (Marone et al., 2016), face recognition (Turk and Pentland, 1991) or stock trends prediction (Choudhry and Garg, 2008).

The main characteristic that most of the Machine Learning algorithms have in common is that they need to be trained. Supervised learning algorithms are usually trained by, given a set of training samples X and their associated output labels or ideal predictions Y, applying an optimization method that intends to optimize its

prediction function $f(X, W) = Y$ by minimizing the error by adapting the set of parameters W.

Different optimization procedures can be found in the literature (e.g. adam, adadelta, etc.), but the most widely known and that set the basis for working on modern Machine Learning methods was Stochastic Gradient Descent (SGD) (Robbins and Monro, 1951). This method iteratively calculates the gradient of the error for each set of training samples. After the gradient is calculated, the weights are slightly modified iteratively with respect to a learning rate parameter in order to reach the desired minimum point. When running an optimization procedure it is important to properly adjust the training parameters in order to avoid overfitting, which consists in adapting too much the model to the training data and thus, reducing its generalization capabilities in front of testing data. This must be avoided by combining a set of model-dependent measures, some of which are: weights regularization, adding enough variability on the training data and adequately adjusting the complexity of the model.

Traditional Computer Vision algorithms worked on the premise of how to better numerically represent and summarize the images at hand. This process had the purpose of having a representation characteristic enough that allowed to better discriminate and classify the different labels or output predictions at hand. This race for finding the best hand-crafted feature extraction method lead to the development of multiple methods that in most cases only worked relatively well on certain scenarios but did not have the capability to generalize for most of the image recognition problems.

With the appearance of Deep Learning, the Machine Learning paradigm drastically changed. Being the Neural Networks the main tool in this field, Deep Learning works on defining an end-to-end architecture that automatically learns the weights with the capabilities to generate both a meaningful input representation from the raw data as well as a highly performing classifier that maps it into the output space.

1.1.2 Artificial and Convolutional Neural Networks

Artificial Neural Networks (ANNs) are the most extended Machine Learning algorithm nowadays thanks to the boost of Deep Learning. This type of algorithms were initially proposed in (Rosenblatt, 1958), which defined the basic building block of the artificial neural networks, the Perceptron. This architecture was built inspired by the functioning of the human brain. The neurons, the cells in the brain, communicate information with each other in the form of electrical impulses through the axons, which are the nervous connections between them. This communication is transmitted with a variable intensity depending on both the intensity of the input signal as well as the structure of the neuron, causing the signals to propagate along the brain with millions of ramifications depending on what is it perceiving.

Trying to resemble this very same structure, Artificial Neural Networks are composed of nodes or neurons, that capture the input signal and apply an activation function that modulates and transmits the signal to the following set of neurons. ANNs are composed of layers of variable sizes (groups of neurons): an input layer, which connects to a set of internal hidden layers, and finish in a final output layer that applies the final classification, providing the output signal.

Deep Artificial Neural Networks (DNNs) follow exactly the same principle but with the difference that they have a larger number of hidden layers, granting them the condition of 'deep'. Given this structural difference, DNNs allow capturing more

complex structures in the input data and create data representations at different levels of abstraction (the deeper the more abstract). The DNNs that are most commonly used when working with images are Convolutional Neural Networks (CNNs). Its most basic characteristic, which differentiates them from DNNs, is that they are composed of convolutional layers, among others. Convolutional layers apply spatial computations over image pixels, creating multiple level representations of the structures appearing on the images. These structures go from basic edges and colours in the initial layers to more complex object parts or spatial object relationships in the final layers.

One of the first CNNs proposed in the literature was LeNet (LeCun et al., 1989), which was applied to classify hand written digits. Even though, it was after their re-appearance in 2012 with the proposal of AlexNet (Krizhevsky, Sutskever, and Hinton, 2012) when they revolutionized the Computer Vision field. The trigger of this revolution was that they proved to outperform with great margin any other previous proposal at the task of object recognition on the ImageNet challenge (Deng et al., 2009). Since that moment the paradigm of using hand-crafted methods changed drastically, and several network architectures with better and better performance appeared and beated all the previous performance records in Computer Vision problems one after another. Some notable examples are GoogleNet (Szegedy et al., 2015) with its inception structure; ResNet (He et al., 2016) with its residual skip connections or ResNext (Xie et al., 2017) with its multiple towers/branches with a high number of parallel paths.

1.1.3 The Power of Data

Although the appearance of Deep NNs and Deep CNNs made a great difference in performance, we must not forget the importance of data. With the appearance of the dataset ImageNet in the year 2009 (Deng et al., 2009), the revolution of data started, too. ImageNet as of today contains more than 14 million images from more than 20.000 different categories of objects in the wild, making it the largest open dataset with real world images.

After ImageNet many other large-scale datasets have been proposed in the literature. For example Youtube-8M for video categorization (Abu-El-Haija et al., 2016), Food 101 for food recognition (Bossard, Guillaumin, and Van Gool, 2014) or Open Images (Krasin et al., 2016) for object segmentation.

Although the quantity of training data on modern deep learning algorithms is a relevant factor, one must not forget about its quality too. The continuously increasing capabilities of data transfer and data sharing that the internet offers allows to acquire millions of data with relative ease. Even though, the creation of huge public datasets is not possible without adequate sample annotations specifically curated for the problem at hand. This increasing needs for annotated data has boosted the appearance of annotation tools and crowdsourcing platforms that enable collaborative work for building enormous pools of annotated data. At the same time, the use of crowdsourced workforce makes the process of obtaining quality annotations a challenging task (Hsueh, Melville, and Sindhwani, 2009).

This data revolution was the main reason that made possible training deep architectures, which contain millions of parameters, allowing to work on real world problems and, at the same time, preventing them to overfit.

1.1.4 Multimodal Learning

Although images were the main data source treated during the development of this thesis, it is also relevant mentioning the importance of multimodal data and multimodal learning.

The concept 'modality' can have multiple definitions depending on the point of view. One could say that it is:

> A particular mode in which something exists, is expressed or is experienced.

Or that it:

> Refers to a certain type of information and/or the representation format in which information is stored.

From the point of view of a human being, we can refer as "human sensory modalities" to the different data types that provide from our senses: 'sound', 'taste', 'touch', 'appearance' and 'aroma', as well as include additional data modalities referring to the mental representation and conjunction that our brain builds from this perceptions.

From the point of view of a machine, we can also distinguish several data modalities like

- 'natural language', textual data,
- 'visual', both from images and videos,
- 'audio', from voice, sounds or music,
- 'haptics' or 'touch',
- 'emotions' representations,
- 'physiological signals' like electrocardiogram (ECG) or skin conductance
- or other modalities like infrared images, depth images, fMRI, etc.

This thesis is mostly focused on dealing with both natural language and visual data, even though, most of the developed or applied techniques could also deal with other data modalities or applications. For this reason the following sections will focus on describing the challenges of dealing with these modalities.

1.1.5 Multimodal Challenges and Tasks

Taking all this data types into account, the main challenges when working on multimodal data are how to numerically and mathematically represent signals coming from the machine environment or from external data sources and also how to combine them and learn predictive models from them. The main challenges can be described the following way:

- Representation challenge: How to create a unique computational representation of a signal coming from any possible modality.
- Mapping challenge: How to change the representation type of some input data to a different modality.

- Fusion challenge: How to combine information from multiple input modalities to perform a prediction to a certain output modality.

Some of the techniques that have been proposed for dealing with these challenges are: feature extraction techniques, which intend to capture the essence and the most important information depending on the data source modality; text embedding mechanisms like (Tang, Qu, and Mei, 2015) or (Pennington, Socher, and Manning, 2014), which provide semantic representations for operating on textual information; multimodal fusion methods, which propose mathematical transformations for connecting and merging data providing from different modalities; or Recurrent Neural Networks like LSTMs (Hochreiter and Schmidhuber, 1997), which deal with sequential data samples (e.g. sentences, videos, etc.).

From the tasks and applications perspective, there are several problems that have been of high interest in the literature.

One of the earliest deep machine learning problems dealing with multimodal data was image description (Specia et al., 2016). The goal consists in building a model that is capable of, given an input image, providing a human readable sentence as an output. The main challenge in this task is not only how to represent the input image on a rich enough representation, but also how to be able to capture the semantics as well as the key people/objects appearing in it in order to convert it into a textual and human-like representation.

Another application is video description or captioning (Xu et al., 2015a). Although being very similar to the previous one from a conceptual perspective, in this case a greater challenge appears from the input representation perspective. The methods proposed have to be able to capture the key aspects appearing in all the frames present in the input video.

Increasing even more the complexity of the multimodal tasks we can find the visual question answering problem (Antol et al., 2015), which consists in providing both an input image together with a natural language question and develop a system capable of not only understanding the picture but at the same time answering the question with respect to that image.

In Chapter 2, we present the works developed during this thesis that are specifically related to Deep and Multimodal Learning. More specifically, they deal with visual question answering and video description or captioning.

1.1.6 Technological Impact and Applications

The enormous advances achieved in the Deep Learning field has allowed tackling problems that were long thought as science fiction. The accuracy improvement achieved in a lot of artificial intelligence problems has been so huge that it has already overpassed its early stages, where it was only applicable in limited research problems, and now hundreds or thousands of private companies around the world have shifted their internal structure for focusing on research and development in order to create solutions applicable in real world scenarios.

Some of the solutions that have already achieved the market and already are or soon will be in our daily life are Autonomous driving (Tesla); Personal assistants (Amazon Alexa, or Google Home); Machine translation (Google Translate); Face recognition and People counting or Traffic data analysis (Google Maps).

This thesis, which is written as a papers compendium, is organized around two main topics: Deep Learning and Multimodal Learning; and is applied to two different areas: Egocentric Vision and Food Recognition. For these two areas three clear different applications have arised:

- Daily life images analysis and summarization
- Food recognition in the wild
- Food recognition in self-service restaurants

Being the last two, commercial solutions derived from this thesis that are already in the market[1].

In the following section, we will review the first one of the two main applications treated in this thesis.

1.2 Egocentric Vision

The increasing computational capabilities of modern computing devices have recently reached the possibility of wearing micro computers in our pockets that are much more powerful than the ones that allowed the first human to land on the moon on the 1960s.

But smartphones are not the only proof of this. Aside from them, these increased capabilities allowed making Internet of Things (IoT) devices possible. Many companies and research groups are currently working on the topic for producing solutions on different levels of complexity and abilities. These communication and computation capabilities can be present in nearly any device that we buy, from smart home appliances to industrial control systems.

Another interesting area in which these capabilities have been present since the last years is the market of wearable and smart devices. The difference with these devices is that they are worn by the user, either in some part of their body or even in their clothes. Some clear examples of these are the bracelet Fitbit, which tracks the physical activities that you do daily, or the Apple Watch, which communicates to your smartphone and allows accessing some of its features.

Wearable devices have clearly increased their popularity since the early 2010s among sportspeople due to their features oriented to tracking physical activity. Even though, there is a specific type of wearable device that has the power to make a change on our daily life from the Computer Vision perspective. These devices are the wearable cameras (Bolanos, Dimiccoli, and Radeva, 2016). These micro computers have the possibility to take pictures or videos throughout the whole day of the user and thus, acquire valuable data that could lead to analyse the behaviour and the good or bad habits of the person wearing it. This field of study is usually named Egocentric Vision due to the first person point-of-view in which the data is acquired. At the same time, Egocentric Vision is a sub-field of Lifelogging, which deals with analysing information providing from any wearable source that logs and tracks any kind of data that describes the daily living of the user.

One of the most clear applications for these wearable cameras would be preventing cognitive and functional decline in people with degenerative diseases like dementia (Oliveira-Barra et al., 2019) or simply on elderly people (Hodges et al., 2006). Furthermore, they could be applied for detecting unhealthy nutritional or mental

[1] https://www.logmeal.es

trends for preventing or palliating diseases such as obesity (Herruzo et al., 2017) or depression.

Some specific areas in which authors have worked with egocentric pictures from a data organizational perspective are image acquisition, organization, summarization or browsing. From an information extraction perspective, research has focused on answering several questions like **Who?** (people detection and recognition), **When?** (when did some action or event take place), **Where?** (physical localization), or **What?** (object detection, event detection or event classification).

To summarize, some of the researchers that work in the area of Egocentric Vision analysis focus on doing Storytelling. Or, in other words, developing algorithms that intend to discover and tell the story that lies underneath the pictures taken along the day of the person.

1.2.1 Deep Learning on Egocentric Vision

Getting more into the details of how are these questions answered, we can see that most of the authors have proven that the use of Deep Learning and Multimodal Learning techniques have made a huge difference on both performance and in depth analysis capabilities.

The state of the art for organizing and splitting all the daily acquired images or photo streams proposes deep learning techniques for segmenting the different unitary events that happened in the life of the camera wearer. These event segmentation techniques are based on extracting high level or semantic features together with low level features from the environment and the elements appearing on the images. After doing that, unsupervised methods are used in order to temporally clusterize and segment the images on meaningful units (Dimiccoli et al., 2017).

When trying to answer the question **Who?** or, with whom was the camera user interacting with, the authors go a step forward from the classical face detection or recognition. From the egocentric vision perspective, what is really meaningful is being able to detect which are the persons that usually interact with the camera wearer during their daily life. To do so, the authors in (Aghaei, Dimiccoli, and Radeva, 2016) rely both on deep matching techniques to track the persons along different days, and on theories of social interaction to detect who are the people the user interacts with on each of the casual or formal social occasions he/she experiences.

On answering the questions **What?** and **Where?** or, what daily life objects and events are the most relevant for describing my day?, the most usual techniques used in the literature are, from the one hand, object detection and object discovery methods (Bolaños and Radeva, 2015); and from the other hand, activity and event recognition methods (Cartas et al., 2018). All of them using deep learning techniques. Following a similar line of answering the questions What? and Where?, but getting into details about healthy habits. Other authors have focused on detecting nutrition and food-related events and environments (Sarker et al., 2018).

When dealing with the question **When?**, the authors rely on the timestamp of the captured images in order to know at which time of the day they were taken. With this, together with information inferred from the environment, the objects and the people appearing in the pictures, they can infer whenever the user is doing some usual daily life and rutinary actions or, otherwise, they are doing novel activities (Talavera et al., 2020).

Following a very different although very important line from the perspective of detecting and preventing mental diseases or issues, are works that deal with emotion and sentiment analysis (Talavera, Radeva, and Petkov, 2019). Differently from

traditional sentiment analysis, which intends to detect which emotions does a picture or a video produce to the person who sees it, emotion detection on Egocentric Vision tries to infer which emotion could be feeling the camera wearer when living a certain daily life or particular situation. This is a very challenging problem considering that the face of the camera wearer never or rarely appears on the pictures. Furthermore, there are very specific and personal situations that could be perceived very differently depending on the observer and on his/her personal experience.

Finally, another important group of research works are the ones that propose methods for combining information extracted from multiple methods (which could be any of the ones described previously) in order to semantically navigate through the pictures acquired (Oliveira Barra et al., 2016) or to summarize the daily life situations lived based on selecting the key frames or key events (Lidon et al., 2017). Going a step forward, by using multimodal learning techniques, some methods propose automatically infering natural language descriptions from daily life events (Bolaños et al., 2018).

In Chapter 3 we will present the novel methods that we have developed for Egocentric Vision analysis, most of them relying on Deep Learning techniques. Some particular problems that we have tackled are event segmentation; object detection and discovery; visual summarization; event summarization based on natural language descriptions and an application to dementia based on implementing serious games though the analysis of egocentric vision images.

1.2.2 Application to Dementia: Serious Games and Storytelling

Dementia is a cognitive disease with several levels of impairment. The most common and early stage is know as Mild Cognitive Impairment (MCI), and is characterized by a progressive decline in the cognitive capabilities of the patient. The most well-known, aggressive and late state of the disease is Alzheimers' disease.

Although currently there is no cure for dementia, medical specialists have proven that by doing some regular mental activities, the advance of the disease can be slowed and the effects can be palliated. Traditional healthcare practices usually make use of generic activities for these patients, which have been proven to help, although they have low user engagement.

A feasible alternative to the use of generic exercises would be the design of personalized activities based on the daily life of the users acquired by Lifelogging and Egocentric Vision devices. Several studies described that using this data offers a memory support for the patients and greatly lowers the progress of the mild cognitive impairment (Lee and Dey, 2008; Doherty et al., 2012; Gelonch et al., 2020).

Following this line, and working together with healthcare professionals from the Consorci Sanitari de Terrassa (CST), as one of the most relevant outcomes of this thesis, we contributed in combining the outcome of several of the Egocentric Vision-related works presented in this thesis in order to build personalized serious games for MCI patients and contribute to palliate the effects of the disease (Oliveira-Barra et al., 2017; Gelonch et al., 2020).

In the next section, we will review the second application treated in this thesis.

1.3 Food Recognition

Since the appearance of the ImageNet dataset, and with it, the ILSVRC challenge (Deng et al., 2009) dedicated to object detection and recognition, the computer vision field has seen a huge increase in popularity. Furthermore, since 2012, after the reborn of CNNs (Krizhevsky, Sutskever, and Hinton, 2012), its popularity index boosted even more. Several research groups around the world have dedicated a lot of resources to pushing forward the state of the art results on the objects recognition problem, even surpassing the human capabilities (He et al., 2016).

Several computer vision problems have benefited from the progress on the object recognition challenge, seeing an enormous increase on performance, too. One of these applications is Food Recognition or Analysis, which has been a very active topic lately. One of the key differences between object and food recognition is that the latter consists of a fine-grained classification narrowed to a very specific topic, which makes it more challenging. The main challenges of the food recognition problem are:

1. High inter-class similarity.
2. High intra-class differences.
3. Unbounded limit of dishes/classes present in the real world.
4. Large differences in dishes between different regions in the world.
5. Nutritional differences between dishes with minor or invisible changes (e.g. non-visible ingredients).

Another reason that makes food recognition a challenging problem when intending to acquire data to train the algorithms is the visual difference that you can see on the pictures providing from different sources. Some clear examples can be seen when you compare images extracted from the internet to images from food that a normal user might take with their smartphone at home and also pictures taken in restaurants. In the internet case, the images usually provide from advertisements or professional cooking sites, which have a high quality and have a very artistic presentation. In the restaurants case (e.g. Tripadvisor or Yelp) you can usually find images either taken by both professionals or by normal users, where the same dish can have very different aspects depending on the type or restaurant you are in. On the other hand, comparing them to pictures taken by regular smartphone users, you will find images less appealing but more realistic whenever they are of home made food. All these differences, which can be huge in some cases, make the process of large scale image acquisition very challenging.

1.3.1 Deep Learning on Food Recognition

The analysis of food images has been tackled from multiple and very different perspectives in the literature, although most of them have a common denominator, which is that they are usually considered from a Deep Learning perspective. These algorithms intend to differentiate very different aspects of the food images in multiple semantic levels of detail. Going from the simplest one, food detection (Aguilar, Bolanos, and Radeva, 2017; Ragusa et al., 2016), where the goal is detecting whether an image contains some kind of food or not. To possibly one of the most challenging and fine-grained ones, which would be micro and macro nutrients detection from a

food image. A different, yet probably the most extended food image analysis problem is food recognition (Aguilar, Bolaños, and Radeva, 2017; Martinel, Foresti, and Micheloni, 2018), which consists in classifying the main type of food appearing in the image from a pre-defined set of food classes. Another broadly treated problem is food localization (Bolanos and Radeva, 2016), which consists in finding the precise spot of the food items on a picture and separately recognizing each of them. Some fine-grained food image challenges explored in the literature are ingredients recognition (Chen and Ngo, 2016; Bolaños, Ferrà, and Radeva, 2017); calorie counting and monitoring; and volume estimation, like in (Wu and Yang, 2009), where the authors present a mobile phone-based calorie monitoring system to track the calories consumption for the users.

Multi-label learning consists in predicting more than one output category for each input sample. Thus, the problem of food ingredients recognition can be treated as a multi-label learning problem. Some models for ingredients recognition have been proposed, being one example the one presented in (Chen and Ngo, 2016). Their dataset, composed of 172 food types, was manually labelled considering visible ingredients only. With this data as well as with the dish name, they proposed a model with two outputs that provides at the same time the dish together with the ingredients contained in it.

The richness and complexity of the food analysis field makes possible the inclusion of additional context or information apart from the food image. The use of information in multiple contexts enables introducing multiple inputs and/or outputs to the designed models. These additional inputs and outputs, which can be visual or textual, can help on either producing richer predictions or working with more detailed inputs in order to improve the prediction capabilities. This means that some authors have explored the problem from a more challenging perspective which implies developing multimodal algorithms for food analysis. A clear example of these works is the one introduced in (Salvador et al., 2017), where the authors provide a large-scale dataset with more than 800.000 images and 1.000 recipes, and with it they propose a predictive model that joins images and recipes to tackle a food retrieval task. In other cases, multi-task learning (MTL) approaches have been adopted to jointly learn food-related tasks. It has been proven that, by forcing the joint learning of the features from the different tasks within the loss function, the associated tasks can generalize better. Some examples of MTL tasks are (Chen and Ngo, 2016; Zhang, Lu, and Zhang, 2016; Ege and Yanai, 2017; Yin and Liu, 2017) where different combinations of tasks have been learned (e.g. food dish, cooking method and ingredients; food dish and ingredients; food dish and calories, etc.).

In most of the cases, the previously described proposals are working towards defining solutions for personalized food tracking and for building food diaries. Other works have explored food recognition from a very different perspective by developing models specialized for certain contexts on the gastronomic restaurant sector. A clear example might be food tray detection in public spaces (Aguilar et al., 2018; Ciocca, Napoletano, and Schettini, 2016), where the data sample consists of a tray picture that includes the food that a customer is about to consume. In this case, the goal is developing a model to recognize the food items present in that tray. The objectives that could be achieved in this kind of models are very different from the ones tackled so far. A clear example would be developing a self-checkout solution for self-service restaurant, or another one would be creating 'smart trays' that could recommend the users which dishes should he/she choose in order to follow a balanced diet or avoid certain food intolerances. The provided recommendations could be based on calorie counting, healthy food, specific nutritional composition, etc. In

addition, if we also consider a system capable to track the food consumed by every customer, in a long-term perspective it could apply health-related recommendations based on the lack or on the over-consumption of certain macro or micro nutrients. Taking the food recognition problem in restaurants from a different perspective, (Xu et al., 2015b) used menu-based information from restaurants as context in order to recognize the dish appearing in the image. By using GPS information provided by the phone they determined all the nearby restaurants and used this information to limit the search space.

1.3.2 Applications to Nutrition: Health and Leisure

Food plays a very important role in our daily life. First of all, as it is commonly said: "what you eat makes who you are", because your nutritional habits influence the way you physically look as well as the way you feel, but physically and even emotionally. Our nutrition patterns directly influence our health and our quality of life, which means that bad nutritional habits often influence enormously the appearance of non-communicable diseases (NCD), which are any kind of disease that is not transmissible directly from one person to another. Our nutrition habits can effect on the appearance of NCDs like obesity, strokes or heart diseases, most cancers, diabetes, chronic kidney disease and others. Thus, following healthy eating patterns makes a direct effect on your lifespan. We must not also forget people that suffer from food intolerances, that have to be very careful about which ingredients do they consume. In Europe, despite being a first-world region, more than 4 million people die each year due to chronic diseases linked to unhealthy lifestyles. The lack of awareness or basic knowledge is a crucial factor in many of these cases. Most people simply do not pay much attention to their eating habits. Furthermore, a great number of deaths related to coronary heart diseases are caused by a group of major risk factors among which bad eating habits are at the top (Rozin et al., 1999).

Secondly, food also plays an important role on our leisure, commonly being the central element on any type of personal or professional meetings with friends, colleagues or family. Which means that sometimes it is used more as an element for socialization rather than a basic human need. Furthermore, with the appearance of social networks and of easy digital ways to communicate, sharing pictures of food to our acquaintances has become a common habit. Every single day, millions of people use social media to make recommendations, promote a particular place or give their friends a warning sign about a nearby restaurant. The creation of automatic tools for food recognition based on images could enable an easier generation of content, create food diaries for improving nutrition habits or even personal food profiles for offering personalized recommendations.

It is clear that everyday more people is increasing their awareness for following healthy eating habits and nowadays, food does not only cover a basic need, but it has become a really important aspect of our social life. At the same time, we must not forget that most of us are used to have multiple pictures of food in our smartphones. These pictures can be either of food that we ate ourselves or food that some friend sent to us. Taking all these elements into account, it feels natural to make use of computer vision techniques to create tools that can both improve and rise the awareness about our healthy or unhealthy eating patterns and, at the same time, make use of the social element that food has become.

1.4 Goals and Contributions

The main goal since the start of this thesis has been to learn and make advancements in the deep learning field with the greater purpose of developing new technologies and models that are applicable and useful in our daily life. With this goal in mind, 6 main contributions can be drawn from the conclusion of this thesis. Note that each contribution lists its set of associated papers from more to less relevance.

Contribution 1: Egocentric Vision Literature Review.

We presented a complete review of egocentric vision from the first works on the 90s until 2015, highlighting different problems, approaches, available datasets and results. We also highlighted the list of new trends and challenges as well as the open questions and future lines.

- Marc Bolanos, Mariella Dimiccoli, and Petia Radeva (2016). "Toward storytelling from visual lifelogging: An overview". In: *IEEE Transactions on Human-Machine Systems* 47.1, pp. 77–90. **Impact Factor: 5.2. Quartile: Q1.**

Contribution 2: Semantic segmentation framework for egocentric photo-streams.

We tackled the problem of organizing egocentric photo streams acquired by a wearable camera into semantically meaningful and distinct segments. First, contextual and semantic information was extracted. Later, we clustered the semantic data in a vocabulary of concepts. Finally, by exploiting the temporal coherence of concepts in photo streams, images which share contextual and semantic attributes were grouped together. The resulting temporal segmentation is particularly suited for further analysis, ranging from activity and event recognition to semantic indexing and summarization. We also release the EDUB-Seg dataset.

- Mariella Dimiccoli, Marc Bolaños, Estefania Talavera, Maedeh Aghaei, Stavri G Nikolov, and Petia Radeva (2017). "Sr-clustering: Semantic regularized clustering for egocentric photo streams segmentation". In: *Computer Vision and Image Understanding* 155, pp. 55–69 **Impact Factor: 5.3. Quartile: Q1.**
- Estefania Talavera, Mariella Dimiccoli, Marc Bolaños, Maedeh Aghaei, and Petia Radeva (2015). "R-clustering for egocentric video segmentation". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 327–336
- Marc Bolanos, Maite Garolera, and Petia Radeva (2014). "Video segmentation of life-logging videos". In: *International Conference on Articulated Motion and Deformable Objects*. Springer, pp. 1–9

Contribution 3: Visual semantic summarization of egocentric photo-streams.

In this line of research we worked on providing solutions for automatic retrieval and summarization of egocentric photo streams captured through a wearable camera. We combined a non-informative images filter and a ranking method based on semantic diversity and on a novelty criterion in order to find the key-frames of the day.

- Aniol Lidon, Marc Bolaños, Mariella Dimiccoli, Petia Radeva, Maite Garolera, and Xavier Giro-i Nieto (2017). "Semantic summarization of egocentric photo stream events". In: *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*. ACM, pp. 3–11 **Rank A**

- Marc Bolanos, Ricard Mestre, Estefanía Talavera, Xavier Giró-i Nieto, and Petia Radeva (2015). "Visual summary of egocentric photostreams by representative keyframes". In: *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, pp. 1–6 **Rank A1**

Contribution 4: Multimodal textual descriptions generation model for egocentric vision.

We tackled storytelling as an egocentric sequences description problem. We proposed a novel methodology that exploits information from temporally neighbouring events, matching precisely the nature of egocentric sequences. Furthermore, we presented a new method for multimodal data fusion consisting on a multi-input attention recurrent network. We proved that our proposal outperforms classical attentional encoder-decoder methods for video description. We also released the EDUB-SegDesc dataset.

- Marc Bolaños, Álvaro Peris, Francisco Casacuberta, Sergi Soler, and Petia Radeva (2018). "Egocentric video description based on temporally-linked sequences". In: *Journal of Visual Communication and Image Representation* 50, pp. 205–216 **Impact Factor: 3.3. Quartile: Q1.**

- Álvaro Peris, Marc Bolaños, Petia Radeva, and Francisco Casacuberta (2016). "Video description using bidirectional recurrent neural networks". In: *International Conference on Artificial Neural Networks*. Springer, pp. 3–11 **Rank B**

- Marc Bolaños, Álvaro Peris, Francisco Casacuberta, and Petia Radeva (2017). "VIBIKNet: Visual bidirectional kernelized network for visual question answering". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 372–380

Contribution 5: Detection, Localization and Recognition methodologies for food images.

A set of varied methodologies applicable to food images were built. These models provide a wide range of semantic information at different levels of detail, from food/non-food detection to ingredients recognition. We also worked on food recognition and categorization, food groups recognition and simultaneous localization and recognition in conventional and egocentric images. With the presented approaches tools can be built for food monitoring oriented to individuals that have to or want to follow a healthy diet. We also released several food-related datasets.

- Eduardo Aguilar, Marc Bolaños, and Petia Radeva (2019). "Regularized uncertainty-based multi-task learning model for food analysis". In: *Journal of Visual Communication and Image Representation*. Vol. 60, pp. 360–370 **Impact Factor: 3.3. Quartile: Q1.**

- Eduardo Aguilar, Marc Bolaños, and Petia Radeva (2017). "Food recognition using fusion of classifiers based on cnns". In: *International Conference on Image Analysis and Processing*. Springer, pp. 213–224 **Rank B**
- Marc Bolanos and Petia Radeva (2016). "Simultaneous food localization and recognition". In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, pp. 3140–3145 **Rank A1**
- Marc Bolaños, Aina Ferrà, and Petia Radeva (2017). "Food ingredients recognition through multi-label learning". In: *International Conference on Image Analysis and Processing*. Springer, pp. 394–402 **Rank B**
- Pedro Herruzo, Marc Bolaños, and Petia Radeva (2016). "Can a CNN recognize Catalan diet?" In: *AIP Conference Proceedings*. Vol. 1773. 1. AIP Publishing, p. 020002
- Eduardo Aguilar, Marc Bolanos, and Petia Radeva (2017). "Exploring food detection using CNNs". In: *International Conference on Computer Aided Systems Theory*. Springer, pp. 339–347

Contribution 6: Food Recognition techniques in Restaurants.

We proposed novel techniques related to semantic segmentation, food localization and detection as well as multimodal data methods for tackling the problem of food recognition in both self-service and conventional restaurants. These methods can serve as the basis for creating self-checkout systems as well as social sharing tools.

- Eduardo Aguilar, Beatriz Remeseiro, Marc Bolaños, and Petia Radeva (2018). "Grab, Pay, and Eat: Semantic Food Detection for Smart Restaurants". In: *IEEE Transactions on Multimedia* 20.12, pp. 3266–3275 **Impact Factor: 7.8. Quartile: Q1.**
- Marc Bolaños, Marc Valdivia, and Petia Radeva (2018). "Where and What Am I Eating? Image-Based Food Menu Recognition". In: *European Conference on Computer Vision*. Springer, pp. 590–605 **Rank A**

The complete set of papers that are presented in this thesis accumulate a total of more than 560 citations at the date of presentation of this thesis. For a detailed list and description of the journals, papers, commercial solutions and all the contributions of this thesis refer to Appendix A.

1.5 Thesis Organization

In the current Chapter 1, a thorough introduction and state of the art analysis has been presented regarding the general and specific topics of the thesis.

The following chapters include the papers presented divided in three main blocks. In Chapter 2, papers applied to more general topics on Deep Learning and Multimodal Learning are presented. Chapter 3 focuses on articles applied to Egocentric Vision, and Chapter 4 groups all the papers regarding Food Recognition. Each of the three chapters include a brief scheme as an introduction for organizing all the papers that are included in that section.

Chapter 5 and Chapter 6 summarize the results obtained on the thesis, as well as the conclusions extracted.

To complete the thesis, in Appendix A, the list of papers presented together with other contributions and outcomes are listed.

Chapter 2

Deep and Multimodal Learning

In this chapter you can find papers that work on complex multimodal learning topics like Visual Question Answering (VQA) (Bolaños et al., 2017) or understanding and generating video descriptions in natural language (Peris et al., 2016).

In the next two chapters you will also find papers related to multimodal learning together with either Food Recognition or Egocentric Vision. The main difference is that the papers in the current chapter are not related to any of those two topics in particular.

VIBIKNet (Bolaños et al., 2017)
Video Description (Peris et al., 2016)

VIBIKNet: Visual Bidirectional Kernelized Network for Visual Question Answering

Marc Bolaños[1,2( )], Álvaro Peris[3], Francisco Casacuberta[3], and Petia Radeva[1,2]

[1] Universitat de Barcelona, Barcelona, Spain
{marc.bolanos,petia.ivanova}@ub.edu
[2] Computer Vision Center, Bellaterra, Spain
[3] PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain
{lvapeab,fcn}@prhlt.upv.es

Abstract. In this paper, we address the problem of visual question answering by proposing a novel model, called VIBIKNet. Our model is based on integrating Kernelized Convolutional Neural Networks and Long-Short Term Memory units to generate an answer given a question about an image. We prove that VIBIKNet is an optimal trade-off between accuracy and computational load, in terms of memory and time consumption. We validate our method on the VQA challenge dataset and compare it to the top performing methods in order to illustrate its performance and speed.

Keywords: Visual Qestion Aswering · Convolutional Neural Networks · Long short-term memory networks

1 Introduction

Deep learning has proven to be applicable to several problems and data modalities (e.g. object detection, speech recognition, machine translation, etc.). Furthermore, it has been able to set new records, beating the state of the art in several artificial intelligence areas. Now, new machine learning problems may be tackled, taking profit from the capabilities of deep learning methods for combining multiple data modalities and be end-to-end trainable, thus, having potential to enable new research and application areas. Some multimodal problems are image captioning [18], video captioning [12] or multimodal machine translation and crosslingual image captioning [15]. In this work, we address the challenging Visual Question Answering (VQA) [1] problem.

From the visual modality perspective, a clear proposal for processing images are Convolutional Neural Networks (CNNs) [17]. CNNs are a powerful tool, not only for image classification, but also for feature extraction. Nevertheless, they are not fully scale and rotation invariant, unless they have been specifically trained with enough varied examples [3]. Furthermore, this invariance problem

L.A. Alexandre et al. (Eds.): IbPRIA 2017, LNCS 10255, pp. 372–380, 2017.
DOI: 10.1007/978-3-319-58838-4_41

gets more acute in scene images, which are composed of multiple elements at possibly different rotations and scales. In order to tackle this problem, Liu proposed in [9] a Kernelized approach for learning a rich representation for images composed of multiple objects in any possible rotation and scale.

From the textual modality perspective, Recurrent Neural Networks (RNNs) have shown to be effective sequence modelers. The use of gated units, such as Long Short-Term Memory (LSTM) [6], allows to properly process long sequences. In the last years, LSTM networks have been used in a wide variety of tasks, such as machine translation [16] or image and video captioning [12,18].

After the appearance of the VQA dataset [1] and the organization of the VQA Challenge, several models appeared addressing this problem. Some notable examples are the ones by Kim et al. [7], where image and question was separately described by a CNN and by a RNN, and then a Multimodal Residual Network (MRN) was used for combining both modalities. Fukui et al. [4] used a CNN for describing the image and a two-layered LSTM for the question; followed by a Multimodal Compact Bilinear Pooling (MCB) for fusion. Nam et al. [10], after describing the input image and question, applied a powerful Dual Attention Network (DAN) for fusing both modalities.

In this work, we propose a model for open-ended VQA which uses the most powerful state-of-the-art methods for image and text characterization. More precisely, we use a Kernelized CNN (KCNN) for image characterization, which takes profit from detecting and characterizing all objects in the image for generating a combined feature descriptor. For question modeling, we apply pre-trained word embeddings from Glove [11], taking advantage from the transfer learning capabilities of neural networks; and a Bidirectional LSTM (BLSTM), able to learn rich question information by taking into account temporal relationships both in past-to-future and future-to-past manner. Next, we fuse both modalities and finish by applying a classification model for obtaining the resulting answer.

This paper is organized as follows: in Sect. 2, we present the proposed method, VIBIKNet. In Sect. 3, we describe the dataset and the evaluation metrics. We evaluate our model and compare it with the state of the art. Finally, in Sect. 4, we give some concluding remarks and future work directions.

2 VIBIKNet

In this section, we describe our VQA system, named Visual Bidirectional Kernelized Network (VIBIKNet), whose general scheme can be seen in Fig. 1. We also make public the complete source code[1] for reproducing the results obtained.

The VQA problem consists in computing a function f which, having as input an image X and a related question Q, produces a textual answer A:

$$f(X, Q) = A \tag{1}$$

where Q and A are two variable-sized sequences of words, which can be formalized as $Q = q_1, q_2, ..., q_N$ and $A = a_1, a_2, ..., a_M$, respectively.

[1] https://github.com/MarcBS/VIBIKNet.

We formulate the problem under a probabilistic framework. Given the clear multimodality of it, first, we propose to extract independent representations for image and question. For obtaining a rich representation of the image, we apply a KCNN [9] (Sect. 2.1). We process the question with a BLSTM network (Sect. 2.2), which considers the full question context. Next, we need to combine modalities into a single representation. To this purpose we propose using a simple, yet effective, element-wise summation (see Sect. 2.3) after embedding the visual information into the textual one. Finally, we predict the output answer, which can be estimated with a simple classifier for the dataset at hand.

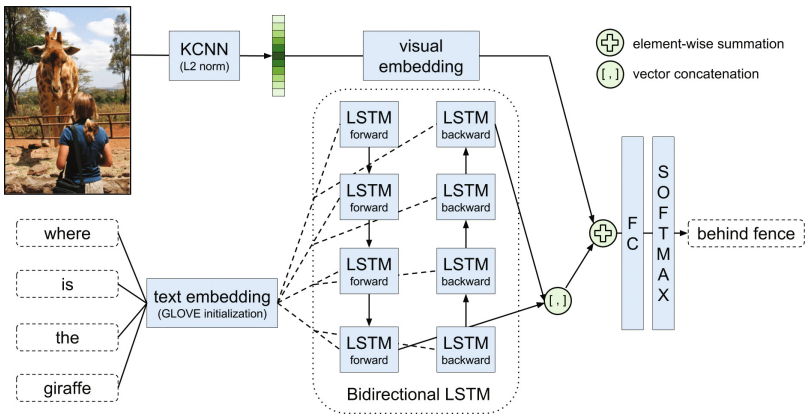

Fig. 1. General scheme of the proposed VIBIKNet model.

2.1 KCNN for Image Representation

A key factor that makes humans able to understand what happens on a picture is the ability to distinguish each of the present elements in it, regarding any possible scale or orientation, together with the relationships and actions that are taking place between them. When we talk about elements we refer to any object, person or animal appearing in the images.

Following this idea, the so-called Kernelized Deep Convolutional Neural Network method [9] has the ability to capture all these aspects. In Fig. 2 we show the general pipeline of steps for extracting KCNN features from images.

More formally, given two images, X and Y, and a set of variable-sized regions for each of them $X = \{x_1, x_2, ..., x_n\}$, and $Y = \{y_1, y_2, ..., y_m\}$, we can define their similarity given by a kernel K as:

$$K(X,Y) = < \sum_{x_i \in X} \psi(x_i), \sum_{y_j \in Y} \psi(y_j) > \; = < \Psi(X), \Psi(Y) > \qquad (2)$$

where the similarity between two regions is computed by their inner product, ψ denotes a linear/non-linear transformation and Ψ denotes the final vectorial image representation composed by the set of initial regions.

Going back to the general scheme applied, initially, an object detector is used for extracting object candidate bounding boxes from each image, x_i. After that,

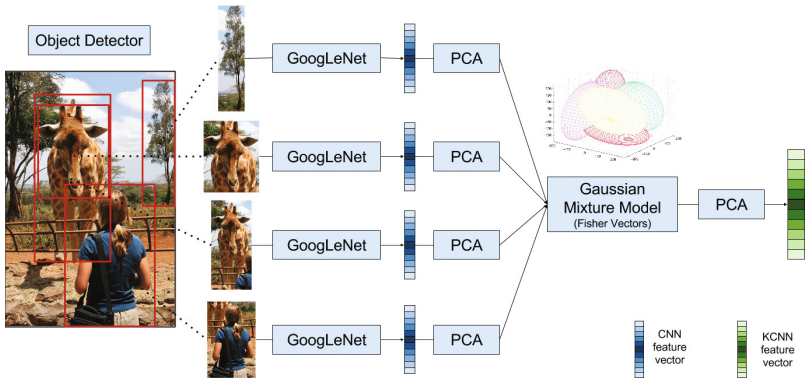

Fig. 2. Steps for the extraction of Kernelized CNN features.

and in order to provide robustness to the point of view, a set of rotations are applied separately to each of the extracted image regions before extracting their image features through a CNN, ψ in Eq. (2). Next, a PCA transformation is applied to the vectors from all image regions. In order to aggregate all vectors from a single image, we learn a Fisher kernel [13] which, similarly to a Bag-of-Words approach [14], jointly models the features distribution by learning a Gaussian Mixture Model (GMM), namely Ψ in Eq. (2). In order to have manageable vector sizes, an additional PCA is applied to the resulting aggregated vectors. This produces an l-size representation of the image, which is finally normalized in order to obtain the final representation of the image ($\Phi(X)$).

2.2 Bidirectional LSTM for Question Representation

As stated above, a question $Q = q_1, q_2, ..., q_N$ is a variable-sized sequence of words. We use a powerful sequence modeler such a RNN for characterizing Q: each word is inputted to the system following a 1-hot codification. Next, we project each word to a continuous space by means of a learnable word embedding matrix. In order to effectively train our word embedding model, we start from pre-trained word vectors provided by Glove [11] and we fine-tune them with the questions corpus. Words not included in Glove are randomly initialized.

The sequence of word embeddings is then inputted to a bidirectional RNN. Bidirectional RNNs are made up of two independent recurrent layers, each of them analyzing the input sequence in one direction. Hence, the forward layer processes the sequence from the left to the right while the backward layer process it from the right to the left. In our case, each recurrent layer is an LSTM layer.

LSTM networks allow to deal with the vanishing gradient problem. These layers maintain two internal states, namely the hidden state ($\mathbf{h}$) and the memory state ($\mathbf{c}$). The amount of information that flows through the network is modulated by the input ($\mathbf{i}$), output ($\mathbf{o}$) and forget ($\mathbf{f}$) gates. Refer to [5] for a more in-depth review of the LSTM networks.

For obtaining a representation of the complete question, we concatenate the last hidden state from the forward and backward layers:

$$\mathbb{Q} = [\mathbf{h}_N^{\mathrm{f}}, \mathbf{h}_N^{\mathrm{b}}] \tag{3}$$

where $\mathbf{h}_N^{\mathrm{f}} \in \mathbb{R}^m$ and $\mathbf{h}_N^{\mathrm{b}} \in \mathbb{R}^m$ are the last forward and backward hidden states, of size m. $[\cdot\ ,\ \cdot]$ denotes vectorial concatenation and $\mathbb{Q}$ is the final representation of Q. Since each LSTM layer processes the complete input sequence in one direction, $\mathbb{Q}$ contains both left-to-right and right-to-left dependencies.

2.3 Multimodal Fusion and Prediction

Multimodal Fusion. Hence, we must combine both image and text representations, given that image X is the KCNN feature vector $\Phi(X)$ of size l, and question Q is represented as $\mathbb{Q}$, of size $2m$.

In order to properly combine both modalities, we first linearly project the image representation to the same space as the question representation, by means of a *visual embedding* matrix:

$$\mathbb{X} = \mathbf{W}_m \Phi(X) \tag{4}$$

where $\mathbf{W}_m$ is a $2m \times l$ matrix, jointly estimated with the rest of the model.

Then, a fusion operation is applied on both modalities, $\mathbb{X}$ and $\mathbb{Q}$:

$$\mathbb{M} = \mathbb{X} \oplus \mathbb{Q} \tag{5}$$

where $\oplus$ is the fusion operator and $\mathbb{M}$ is the joint, multimodal representation of the image and question.

Prediction. Given the nature of the task at hand, a typical answer has few words. More precisely, in the VQA dataset (Sect. 3.1), the 89.3% of the answers are single-worded; and the 99.0% of the answers have three or less words [1].

Therefore, we treat our problem as a classification task over the K most repeated answers. The obtained fusion of vision and text ($\mathbb{M}$) is inputted to a fully-connected layer with the set of answers as output. Applying a softmax activation, we define a probability over the possible answers. At test time, we choose the answer $\hat{a}$ with the highest probability:

$$\hat{a} = \underset{a \in K}{\arg\max}\ p(a|Q, X) \tag{6}$$

3 Experiments and Results

In this section we set up the experimentation and evaluation procedure. Moreover, we study and discuss the obtained results in the VQA Challenge[2].

[2] The VQA Challenge leaderboard is available at http://visualqa.org/roe.html.

3.1 Dataset and Evaluation

We evaluate our model on the VQA dataset [1], on the real open-ended task. The dataset consists of approximately 200,000 images from the MSCOCO dataset [2]. Each image has three questions associated and each question has ten answers, which were provided by human annotators. We used the default splits for the task: *Train* (80,000 images) for training, *Test-Dev* (40,000 images) for validating the model and *Test-Standard* (80,000 images) for testing it. An additional partition, *Test-Challenge*, was used for evaluating the model at the VQA Challenge.

We followed the VQA evaluation protocol [1], which computes an accuracy between the system output ($\hat{a}$) and the answers provided by the humans:

$$Acc(\hat{a}) = \min\left\{\frac{\#\,\text{humans that said}\,\hat{a}}{3}, 1\right\} \tag{7}$$

3.2 Experimental Setup

We set the model hyperparameters according to empirical results. For extracting the KCNN features, we used: EdgeBoxes [19] for proposing 100 object regions, a set of 8 different object rotations of $R = \{0, 45, 90, 135, 180, 225, 270, 315\}$ degrees, the last FC layer of GoogLeNet [17] (1024-dimensional) for extracting features on each object, applied a PCA of dimensions 128 before, and $l = 1024$ after the GMM, respectively, and learned 128 gaussians during GMM training.

Since we used Glove vectors, the word embedding size was fixed to 300. The BLSTM network had $m = 250$ units in each layer. The visual embedding had a size of $2m = 500$. We applied a classification over the 2,000 most frequent answers, covering a 86.8% of the whole dataset. As fusion operator ($\oplus$ in Eq. (5) we tested element-wise summation, concatenation and MCB pooling [4].

We used the Adam [8] optimizer with an initial learning rate of 10^{-3}. As regularization strategy, we only applied dropout before the classification layer.

3.3 Experimental Results

Table 1 shows the accuracies of variations of our model (top) and of other works (bottom) for the *Test-Dev* and *Test-Standard* splits, together with the average μs needed for each of the most relevant methods to perform a forward and backward pass on a Titan X with a batch size of 128. Results are separated according to the type of answer, namely yes/no (Y/N), numerical (Num.) and other (Other) answers. We also report the overall accuracy of the task (All).

It can be seen that both summation and concatenation fusion strategies performed similarly. In terms of performance, MCB was also similar to them. Nevertheless, MCB was much more resource-demanding: while the average time per iteration of summation was 10.25 µs, MCB required 244.14 µs. Such differences come from two different sources. First, the MCB operation is not completely GPU-friendly, which makes it expensive. Second, the MCB network involves the estimation of 22 million parameters, while the number of parameters that

Table 1. Proposed models compared to the state of the art. G stands for GoogLeNet, R for ResNet-152, K for KCNN, L for LSTM, BL for BLSTM, FC for fully-connected layer on text before fusing, sum and cat for fusion by summation and concatenation, respectively, +val for training using train+val. VIBIKNet is "G-K BL sum".

	Test-Dev [%]				Test-Standard [%]				μs/iter
	Y/N	Num.	Other	All	Y/N	Num.	Other	All	
G-K L sum	79.0	33.7	38.2	52.9	–	–	–	–	8.4
G-K BL FC sum	78.6	33.6	36.9	52.1	–	–	–	–	12.7
G-K BL FC cat	79.0	33.6	38.3	53.0	–	–	–	–	16.3
R BL sum	77.8	30.6	38.6	52.3	–	–	–	–	15.2
G-K BL cat	79.0	33.4	38.5	53.0	–	–	–	–	16.3
G-K BL MCB	79.2	33.2	37.5	52.5	–	–	–	–	244.1
VIBIKNet	79.1	33.5	38.3	53.1	78.3	38.9	39.0	54.9	10.2
VIBIKNet +val	–	–	–	–	78.9	36.3	40.3	55.8	–
MRN [7]	–	–	–	–	82.4	38.2	49.4	61.8	–
DAN [10]	83.0	39.1	53.9	64.3	82.8	38.1	54.0	64.2	–
MCB [4]	82.3	37.2	57.4	65.4	–	–	–	–	–
Human [1]	–	–	–	–	95.8	83.4	72.7	83.3	–

Fig. 3. Examples of the predictions provided by VIBIKNet; in green correctly predicted and in red wrongly predicted answers. (Color figure online)

VIBIKNet needs to estimate is below 8 million. For comparison, the MCB architecture proposed in [4] requires the estimation of 48 million parameters, which make us hypothesize that such architecture is notoriously slower than our proposal. Thus, although MCB has the potential to provide a better modalities' fusion, simpler methods like summation are more efficient when dealing with low computationally demanding methods, which have the capacity to learn a question-image embedding representation for applying the summation strategy. Moreover, adding a fully-connected layer after text characterization and before fusion did not help, meaning that the visual embedding mechanism suffices for

providing a robust visual-text embedding. Regarding image characterization, if we compare the results using ResNet-152 vs GoogLeNet-KCNN, we can see that even using a less powerful CNN architecture, the adoption of the KCNN representation provided better results than simply using the ResNet output. Finally, it is worth noting that we used a single model for prediction. The use of network ensembles typically offer a performance boost [4]. In Fig. 3 we can see some qualitative examples of our methodology.

4 Conclusions and Future Work

We proposed a method for VQA which offers a trade-off between the accuracy and the computational cost of the model. We have proven that kernelized methods for image representation based on CNNs are very powerful for the problem at hand. Additionally, we have shown that using simple fusion methods like summation or concatenation can produce similar results to more elaborate methods at the same time that provide a very efficient computation. Nevertheless, we are aware that performing the multimodal fusion at deeper levels may be beneficial.

As future directions, we aim to delve into better fusion strategies but keeping a low computational cost. We extracted KCNN features based on local representations (objects appearance), but using them together with end-to-end trainable attention mechanisms may lead to higher performances [4].

Acknowledgments. This work was partially funded by TIN2015-66951-C2-1-R, SGR 1219, CERCA Programme/Generalitat de Catalunya, CoMUN-HaT - TIN2015-70924-C2-1-R (MINECO/FEDER), PrometeoII/2014/030 and R-MIPRCV. P. Radeva is partially supported by ICREA Academia2014. We acknowledge NVIDIA Corporation for the donation of a GPU used in this work.

References

1. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh., D.: VQA: visual question answering. In: ICCV, pp. 2425–2433 (2015)
2. Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft COCO captions: data collection and evaluation server. arXiv:1504.00325 (2015)
3. Cheng, G., Zhou, P., Han, J.: RIFD-CNN: rotation-invariant and fisher discriminative convolutional neural networks for object detection. In: CVPR, pp. 2884–2893 (2016)
4. Fukui, A., Park, D.H., Yang, D., Rohrbach, A., Darrell, T., Rohrbach, M.: Multimodal compact bilinear pooling for visual question answering and visual grounding. arXiv:1606.01847 (2016)
5. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: continual prediction with LSTM. Neural Comput. **12**(10), 2451–2471 (2000)
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
7. Kim, J.-H., Lee, S.-W., Kwak, D.-H., Heo, M.-O., Kim, J., Ha, J.-W., Zhang, B.-T.: Multimodal residual learning for visual QA. arXiv:1606.01455 (2016)

8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv:1412.6980 (2014)
9. Liu, Z.: Kernelized deep convolutional neural network for describing complex images. arXiv:1509.04581 (2015)
10. Nam, H., Ha, J.-W., Kim, J.: Dual attention networks for multimodal reasoning and matching. arXiv:1611.00471 (2016)
11. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: EMNLP, pp. 1532–1543 (2014)
12. Peris, Á., Bolaños, M., Radeva, P., Casacuberta, F.: Video description using bidirectional recurrent neural networks. In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 3–11. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_1
13. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 143–156. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15561-1_11
14. Sivic, J., Zisserman, A.: Efficient visual search of videos cast as text retrieval. PAMI **31**(4), 591–606 (2009)
15. Specia, L., Frank, S., Sima'an, K., Elliott, D.: A shared task on multimodal machine translation and crosslingual image description. In: Proceedings of the First Conference on Machine Translation, pp. 543–553. ACL (2016)
16. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: NIPS, vol. 27, pp. 3104–3112 (2014)
17. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR, pp. 1–9 (2015)
18. Xu, K., Ba, J., Kiros, R., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: neural image caption generation with visual attention. arXiv:1502.03044 (2015)
19. Zitnick, C.L., Dollár, P.: Edge boxes: locating object proposals from edges. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 391–405. Springer, Cham (2014). doi:10.1007/978-3-319-10602-1_26

Video Description Using Bidirectional Recurrent Neural Networks

Álvaro Peris[1( )], Marc Bolaños[2,3], Petia Radeva[2,3], and Francisco Casacuberta[1]

[1] PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain
{lvapeab,fcn}@prhlt.upv.es
[2] Universitat de Barcelona, Barcelona, Spain
{marc.bolanos,petia.ivanova}@ub.edu
[3] Computer Vision Center, Bellaterra, Spain

Abstract. Although traditionally used in the machine translation field, the encoder-decoder framework has been recently applied for the generation of video and image descriptions. The combination of Convolutional and Recurrent Neural Networks in these models has proven to outperform the previous state of the art, obtaining more accurate video descriptions. In this work we propose pushing further this model by introducing two contributions into the encoding stage. First, producing richer image representations by combining object and location information from Convolutional Neural Networks and second, introducing Bidirectional Recurrent Neural Networks for capturing both forward and backward temporal relationships in the input frames.

Keywords: Video description · Neural Machine Translation · Birectional Recurrent Neural Networks · LSTM · Convolutional Neural Networks

1 Introduction

Automatic generation of image descriptions is a recent trend in Computer Vision that represents an interesting, but difficult task. This has been possible due to the dramatic advances in Convolutional Neural Network (CNN) models that allowed to outperform the state-of-the-art algorithms in many computer vision problems: object recognition, object detection, activity recognition, etc. Generating descriptions of videos represents an even more challenging task that could lead to multiple applications (e.g. video indexing and retrieval, movie description for multimedia applications or for blind people or human-robot interaction).

However, the problem of video description generation has several properties that make it specially difficult. Besides the significant amount of image information to analyze, videos may have a variable number of images and can be described with sentences of different length. Furthermore, the descriptions of videos use to be high-level summaries that not necessarily are expressed in

A.E.P. Villa et al. (Eds.): ICANN 2016, Part II, LNCS 9887, pp. 3–11, 2016.
DOI: 10.1007/978-3-319-44781-0_1

terms of the objects, actions and scenes observed in the images. There are many open research questions in this field requiring deep video understanding. Some of them are how to efficiently extract important elements from the images (e.g. objects, scenes, actions), to define the local (e.g. fine-grained motion) and global spatio-temporal information, determine the salient content worth to describe, and generate the final video description. All these specific questions need the attention of computer vision, machine translation and natural language understanding communities in order to be solved.

In this work, we propose to enrich the state-of-the-art architecture using bidirectional neural networks for modeling relationships in two temporal directions. Furthermore, we test the inclusion of supplementary features, which help to detect contextual information from the scene where the video takes place.

2 Related Work

Although the problem of video captioning recently appeared thanks to the new learning capabilities offered by Deep Learning techniques, the general pipeline adopted in these works resembles the traditional encoder-decoder methodology used in Machine Translation (MT). The main difference is that, in the encoder step, instead of generating a compact representation of the source language sentence, we generate a representation of the images belonging to the video.

MT aims to automatically translate text or speech from a source to a target language. Within the last decades, the prevailing approach is the statistical one [5]. The application of connectionist models in the area has drawn much the attention of researchers in the last years. Moreover, a new approach to MT has been recently proposed: the so-called Neural Machine Translation, where the translation process is carried out by a means of a large Recurrent Neural Network (RNN) [9]. These systems rely on the encoder-decoder framework: an encoder RNN produces a compact representation of an input sentence in the source language, and the decoder RNN takes this representation and generates the corresponding target language sentence. Both RNNs usually make use of gated units, such as the popular Long Short-term Memory (LSTM) [4], in order to cope with long-term relationships.

The recent reintroduction of Deep Learning in the Computer Vision field through CNNs [6], has allowed to obtain new and richer image representations compared to the traditional hand-crafted ones. These networks have demonstrated to be a powerful tool to extract feature representations for several kinds of computer vision problems like on objects [8] or scenes [15] recognition. Thanks to the CNNs ability to serve as knowledge transfer mechanisms, they have also been usually used as feature extractors.

The majority of the works devoted to generate textual descriptions from single images also follow the encoder-decoder architecture. In the encoding stage, they apply a combination of CNN and LSTM for describing the input image. In the decoding stage, an LSTM is in charge of receiving the image information and generating, word by word, a final description of the image [12].

The problem of video captioning is similar. Seminal works applied methodologies inspired by classical MT [7]. Nevertheless, more recent works following the encoder-decoder approach, obtained state-of-the-art performances [11,13].

We present a new methodology for natural language video description that makes use of deeper structures and a double-way analysis of the input video. We propose to use as a base architecture the one introduced in [13]. On the top of it, our contributions are twofold. First, we produce richer image representations by combining complementary CNNs for detecting objects and contextual information from the input images. Second, we introduce a Bidirectional LSTM (BLSTM) network in the encoding stage, which has the ability to learn forward and backward long-term relationships on the input sequence.

3 Methodology

An overview of our proposal is depicted in Fig. 1. We propose an encoder-decoder approach consisting of four stages, using both CNNs and LSTMs for describing images and for modeling their temporal relationship, respectively.

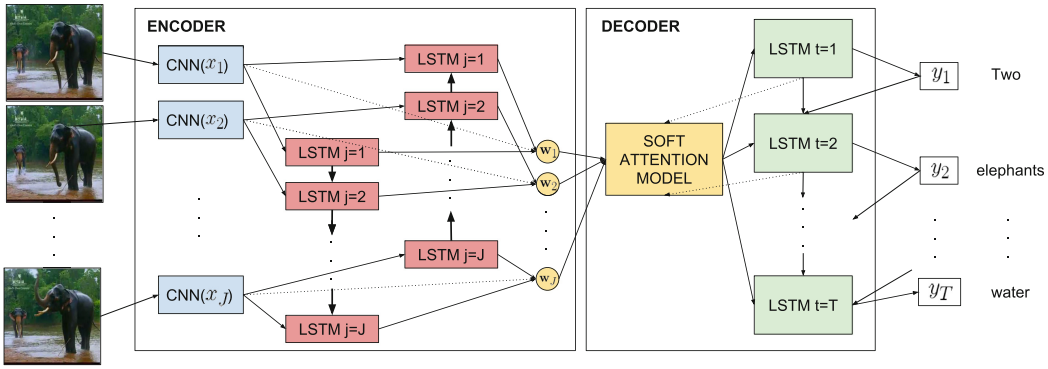

Fig. 1. General scheme of our proposed methodology. (Color figure online)

First (blue in the scheme), we apply two state of the art CNN models for extracting complementary features on each of the raw images from the video.

Second (red in the scheme), considering we need to describe the actions performed in consecutive frames, we apply a BLSTM for capturing temporal relationships and complementary information by taking a look at the action in a forward and in a backward manner.

Third (yellow in the scheme), the two output vectors from forward and backward LSTM models of the previous step are concatenated together with the CNN output for each image and are fed to a soft attention model in the decoder. This model decides on which parts of the input video should focus for emitting the next word, considering the description generated so far.

Fourth (green in the scheme), an LSTM network generates the video caption from the representation obtained in previous stages. The variable-length caption is obtained word by word, using a softmax function on the top of the LSTM.

3.1 Encoder

Given the video description problem, in the encoding stage we need to properly characterize the video for (1) understanding which kind of objects and structures appear in the images, and (2) modeling their relationships and actions along time.

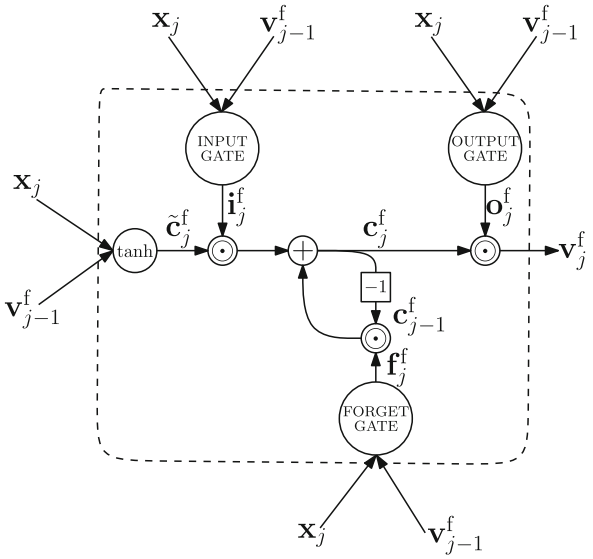

Fig. 2. Forward layer LSTM unit for the encoder. The output depends on the previous hidden state ($\mathbf{v}^{\mathrm{f}}_{j-1}$) and the current feature vector from the video extracted by the CNN ($\mathbf{x}_j$). Input, output and forget gates module the amount of information that flows across the unit.

For tackling the first part of the problem, several kinds of pretrained CNNs may be used for describing the images, which can be distinguished by the different architectures or by the different datasets used for training. Although an extended comparison and combinations of models could be used for applying this characterization, we propose combining object and context-related information. For this purpose we use the GoogleNet architecture [10] separately trained on two datasets, one for objects (ILSVRC dataset [8]), and the other for scenes (Places 205 [15]). The combination of these two kinds of data can inform about the objects appearing and their surroundings, being ideal for the problem at hand. Note that, given the nature of this task, an explicit object or scene segmentation is not required. Additionally, we must note that the features extracted are a representation of the whole image, which means that are not suitable for extracting spatial-related information. For a given video, the CNNs generate a sequence $\mathbf{V}_c$ of J d-dimensional feature vectors, $\mathbf{x}_1, \ldots, \mathbf{x}_J$ with $\mathbf{x}_j \in \mathbb{R}^d$ for $1 \leq j \leq J$, where J is the number of frames in the video.

To solve the second problem, a BLSTM processes the sequence $\mathbf{V}_c$, generating a new sequence $\mathbf{V}_{bi} = \mathbf{v}_1, \ldots, \mathbf{v}_J$ of J vectors. BLSTM networks are composed of two independent LSTM layers namely, forward and backward. Both layers are analogue, but the latter processes the input sequence reversed in time.

LSTM networks have, in addition to the classical hidden state, a memory state. Let $\mathbf{v}_j^{\mathrm{f}}$ be the forward layer hidden state at the time-step j, and let $\mathbf{c}_j^{\mathrm{f}}$ be its memory state. The hidden state $\mathbf{v}_j^{\mathrm{f}}$ is computed as $\mathbf{c}_j^{\mathrm{f}}$ controlled by an output gate $\mathbf{o}_j^{\mathrm{f}}$. The current memory state depends on an updated memory state, and on the previous memory state, $\mathbf{c}_{j-1}^{\mathrm{f}}$, respectively modulated by the forget and input gates, $\mathbf{f}_j^{\mathrm{f}}$ and $\mathbf{i}_j^{\mathrm{f}}$. The updated memory state $\tilde{\mathbf{c}}_j^{\mathrm{f}}$ is obtained by applying a logistic non-linear function to the input and the previous hidden state. Each LSTM gate has associated two weight matrices, accounting for the input and the previous hidden state. Such matrices must be estimated on a training set. Figure 2 shows an illustration of an LSTM unit. The same architecture applies to the backward layer, but dependencies flow from the next time-step to the previous one. Since forward and backward layers are independent, they have different weight matrices to estimate.

Each feature vector $\mathbf{v}_j$ computed by the BLSTM results as the concatenation of the forward and backward hidden states: $\mathbf{v}_j = [\mathbf{v}_j^{\mathrm{f}}; \mathbf{v}_j^{\mathrm{b}}] \in \mathbb{R}^{2 \cdot D}$ for $1 \leq j \leq J$, being D the size of each forward and backward hidden state.

Finally, the encoder combines the sequences $\mathbf{V}_c$ and $\mathbf{V}_{bi}$ by concatenating the vectors from the CNN and from the BLSTM, producing a final sequence $\mathbf{V}$ of J feature vectors $\mathbf{w}_1, \ldots, \mathbf{w}_J$, $\mathbf{w}_j = [\mathbf{x}_j; \mathbf{v}_j] \in \mathbb{R}^{d+2 \cdot D}$ for $1 \leq j \leq J$.

3.2 Decoder

The decoder is an LSTM network, which acts as a language model, conditioned by the information provided by the encoder. This network is equipped with an attention mechanism [1,13]: a soft alignment model, implemented as a single-layered perceptron, that helps the decoder to know *where* to look at for generating each output word. Given the sequence $\mathbf{V}$ generated by the encoder, at each decoding time-step t the attention mechanism weights the J feature vectors and

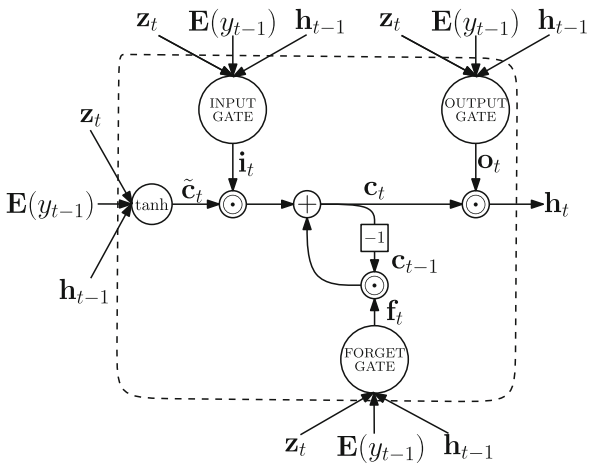

Fig. 3. Decoder LSTM unit. The output depends on the previous hidden state ($\mathbf{h}_t$), the word embedding of the previously generated word ($\mathbf{E}(y_{t-1})$) and the context vector provided by the attention mechanism ($\mathbf{z}_t$).

combines them into a single context vector $\mathbf{z}_t \in \mathbb{R}^{d+2\cdot D}$. Considering that each of our feature vectors describes the scene in a different temporal moment, our dynamic attention mechanism acts as a learnable saliency mechanism applied along time, which is able to weight and emphasize the information of different frames.

The decoder LSTM is defined similarly to the forward layer from the encoder, but it takes into account the previously generated word and the context vector from the attention mechanism, in addition to its previous hidden state. The last word representation is provided by a word embedding matrix $\mathbf{E} \in \mathbb{R}^{m \times V}$, being m the size of the word embedding and V the size of the vocabulary. $\mathbf{E}$ is estimated together with the rest of the model parameters.

A probability distribution over the vocabulary of output words is defined from the hidden state $\mathbf{h}_t$, by means of a softmax function. This function represents the conditional probability of a word given an input video $\mathbf{V}$ and its history (the previously generated words): $p(y_t|y_1, \ldots, y_{t-1}, \mathbf{V})$. Following [9], a beam-search method is used to find the caption with highest conditional probability.

4 Results

In this section we describe the datasets and metrics used for evaluating and comparing our model to the video captioning state of the art.

4.1 Dataset

The **Microsoft Research Video Description Corpus** (MSVD) [2] is a dataset composed of 1970 open domain clips collected from YouTube and annotated using a crowd sourcing platform. Each video has a variable number of captions, written by different users. We used the splits made by [11,13], separating the dataset in 1200 videos for training, 100 for validation and the remaining 670 for testing. During training, the clips and each of their captions were treated separately, accounting for a total of more than $80,000$ training samples.

4.2 Evaluation Metrics

In order to evaluate and compare the results of the different models we used the standardized COCO-Caption evaluation package [3], which provides several metrics for text description comparison. We used three main metrics, all of them presented from 0 (minimum quality) to 100 (maximum quality):

BLEU: this metric compares the ratio of n-gram structures that are shared between the system hypotheses and the reference sentences.

METEOR: it computes the F1 score of precision and recall between hypotheses and references.

CIDEr: similarly to BLEU, it computes the number of matching n-grams, but penalizes any n-gram frequently found in the whole training set.

4.3 Experimental Results

On all the tests we used a batch size of 64, the learning rate was automatically set by the Adadelta [14] method and, as the authors in [13] reported, we applied a frame subsampling, picking only one image every 26 frames for reducing the computational load. The parameters of the network were randomly initialized. An evaluation on the validation set was performed every 1000 updates. The learning process was stopped when the reported error increased after 5 evaluations.

For each configuration we run 10 experiments. At each of them, we randomly set the value of the critical model hyperparameters. Such hyperparameters and their tested ranges are $m \in [300, 700]$, $|\mathbf{h}_t| \in [1000, 3000]$. When using the BLSTM encoder, we performed an additional selection on $|\mathbf{v}_j| \in [100, 2100]$.

Table 1. Text generation results for each model on the MSVD dataset. The results below the horizontal line are our proposals.

Model	BLEU [%]	METEOR [%]	CIDEr [%]
Objects[a]	51.5	32.5	66.0
Objects + BLSTM	**53.6**	**32.6**	66.4
Objects + Scenes	52.6	32.5	67.0
Objects + Scenes + BLSTM	52.8	31.3	**67.2**

[a]Model from [13] only with *Object* features evaluated on our system.

For each configuration, the best model with respect to the BLEU measure on the validation set was selected. In Table 1 we report the results of the best models on the test set. The first row correspond to the result obtained with our system with the object features from [13]. The configurations reported below the horizontal line are our proposals, where *Scenes* indicates we use scene-related features concatenated to *Objects* and *BLSTM* denotes the use of the additional BLSTM encoder.

5 Discussion and Conclusions

Analyzing the obtained results, a clear improvement trend can be derived when applying the BLSTM as a temporal inference mechanism. The BLSTM addition when using *Objects* features allows to improve the result on all metrics, obtaining a benefit of more than 2 BLEU points. Adding scenes-related features also slightly improves the result, although it is not as remarkable as the BLSTM improvement. The combination of *Objects*+*Scenes*+BLSTM offers the best CIDEr performance, nevertheless, this result is slightly below the *Objects*+BLSTM one on the other metrics. This behaviour is probably due to the significant increase on the number of parameters to learn. It should be investigated whether the reduction of the number of parameters by reducing the size of the CNN features, or the use of larger datasets could lead to further improvements.

In conclusion, we have presented a new methodology for natural language video description that takes profit from a bidirectional analysis of the input sequence. This architecture has the ability to infer information from data not only in a past-to-future fashion, but also in the future-to-past direction. Which means that its hidden state will incorporate more confident information, being even more evident in the initial frames where otherwise the result would only take into account a short time-span. On the other hand, the use of a bidirectional model yields doubling the number of parameters on the encoder, which will increase the computational time and the amount of data needed to train the model. Although, in order to extract further conclusions, the presented architecture should be tested on more datasets. Additionally, the use of complementary object and scene-related image features has proven to obtain a richer video representation. The improvements have allowed the method to outperform the state-of-the-art results in the problem at hand.

These results suggest that deep structures help to transfer the knowledge from the input sequence of frames to the output natural language caption. Hence, the next step to take must delve into the application of deeper modeling structures: 3D CNNs allow the recognition of actions and may solve some of the ambiguities existing in the tested methods, which only cope with object and scenes recognition. An additional future step should study the inclusion of spatio-temporal attention models for better coping with the nature of natural videos.

Acknowledgments. This work was partially founded by TIN2015-66951-C2-1-R, SGR 1219, PrometeoII/2014/030 and by a travel grant by the R-MIPRCV network. P. Radeva is partially supported by an ICREA Academia2014 grant. We acknowledge NVIDIA for the donation of a GPU used in this work.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proceedings of the International Conference on Learning Representations, arXiv:1409.0473 (2015)
2. Chen, D.L., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, vol. 1, pp. 190–200 (2011)
3. Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft COCO captions: data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)
4. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
5. Koehn, P.: Statistical Machine Translation. Cambridge University Press, New York (2010)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)

7. Rohrbach, M., Qiu, W., Titov, I., Thater, S., Pinkal, M., Schiele, B.: Translating video content to natural language descriptions. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 433–440 (2013)
8. Russakovsky, O., Deng, J., Hao, S., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vision **115**(3), 211–252 (2015)
9. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. Adv. Neural Inf. Process. Syst. **27**, 3104–3112 (2014)
10. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
11. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4534–4542 (2015)
12. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: a neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164 (2015)
13. Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Describing videos by exploiting temporal structure. In Proceedings of the IEEE International Conference on Computer Vision, pp. 4507–4515 (2015)
14. Zeiler, M.D.: ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)
15. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in Neural Information Processing Systems, pp. 487–495 (2014)

Chapter 3

Deep Learning on Egocentric Vision

In this chapter you can find all the papers related to Deep Learning on Egocentric Vision and Egocentric Storytelling. During the development of this thesis we started treating the topic from a more general and broad approach at first when the paper Toward Storytelling (Bolanos, Dimiccoli, and Radeva, 2016) was presented with the purpose to review the whole literature on the topic.

After that, we continued by working on separate more specific topics related to egocentric vision like video segmentation (Dimiccoli et al., 2017) or object discovery (Bolaños and Radeva, 2015) in order to later present more complex approaches that had the capability to agglomerate the knowledge so far like photo stream semantic summarization (Lidon et al., 2017).

Finally, we ended by working on complex and more directly applicable scenarios like events textual description generation (Bolaños et al., 2018) and development of serious games for mild cognitive impairment patients (Oliveira-Barra et al., 2017).

Literature Review
Toward Storytelling (Bolanos, Dimiccoli, and Radeva, 2016)

Segmentation
Video segmentation (Bolanos, Garolera, and Radeva, 2014)
R-Clustering (Talavera et al., 2015)
SR-Clustering (Dimiccoli et al., 2017)

Object Discovery
Object Discovery (Bolaños, Garolera, and Radeva, 2015)
Ego-Object (Bolaños and Radeva, 2015)

Visual Summarization
Visual Summary (Bolanos et al., 2015)
Semantic Summarization (Lidon et al., 2017)

Complex Topics and Applications
Serious Games (Oliveira-Barra et al., 2017)
Egocentric Temporally-Linked (Bolaños et al., 2018)

Toward Storytelling From Visual Lifelogging: An Overview

Marc Bolaños, Mariella Dimiccoli, and Petia Radeva

***Abstract*—Visual lifelogging consists of acquiring images that capture the daily experiences of the user by wearing a camera over a long period of time. The pictures taken offer considerable potential for knowledge mining concerning how people live their lives; hence, they open up new opportunities for many potential applications in fields including healthcare, security, leisure, and the quantified self. However, automatically building a story from a huge collection of unstructured egocentric data presents major challenges. This paper provides a thorough review of advances made so far in egocentric data analysis and, in view of the current state of the art, indicates new lines of research to move us toward storytelling from visual lifelogging.**

***Index Terms*—Egocentric vision, storytelling, visual lifelogging.**

I. Introduction

LIFELOGGING consists of a user continuously recording their everyday experiences, typically via wearable sensors including accelerometers and cameras, among others. When the visual signal is the only one recorded, typically by a wearable camera, it is referred to as visual lifelogging. This is a trend that is rapidly increasing thanks to advances in wearable technologies over recent years. Nowadays, wearable cameras are very small devices that can be worn all-day long and automatically record the everyday activities of the wearer in a passive fashion, from a first-person point of view. As an example, Fig. 1 shows pictures taken by a person walking down a street while wearing such a camera.

Most wearable cameras on the market like GoPro, MeCam, Looxcie, or Google Glass [see Fig. 2(a) and (c)] are video cameras, which have relatively high temporal resolution (HTR) (e.g., from 25 up to 60 frames/s) and are more suitable to record specific moments, such as cooking or doing sports. A limited number of wearable cameras, such as Narrative Clip and SenseCam [see Fig. 2(b) and (d)] are photographic cameras, which have low temporal resolution (LTR) (2–3 frames/min) and hence are more suitable for acquiring data over long periods of time. On the one hand, data recorded at specific moments with video cameras offer potential for in-depth analysis of daily or special activities, allowing to capture even how something happened. On the other hand, data acquired over long periods of time, commonly called visual lifelogs, offer considerable potential for inferring knowledge about, e.g., behavior patterns, and hence enable many applications that would not be possible with HTR cameras. As shown by Doherty *et al.* [32], visual lifelogs captured through a SenseCam, which as opposed to video cameras can capture the whole day, could be used to prevent noncommunicable diseases associated with unhealthy trends and risky profiles (such as obesity or depression, among others). Additionally, they could also help prevent cognitive and functional decline in elderly people [29], [44], [57]. However, visual lifelogs present a significant challenge for automatic visual analysis. Indeed, due to the free motion of the camera and to its LTR, abrupt changes in lighting conditions and image content are very frequent (see Fig. 1). In such situations, computer vision techniques based on temporal coherence and motion estimation become unreliable. Recognition algorithms have to cope with the huge variety of objects that appear. In addition, due to the nonintentional nature of the pictures captured, they generally contain severely occluded objects, artefacts such as blurring or light saturation

Fig. 1. Example of a sequence acquired by the narrative clip wearable camera while the user is walking down a street. The temporal leaps between neighboring pictures produced by photographic cameras are common in dynamic environments and make the extraction of information from closely spaced images very difficult.

Fig. 2. Examples of wearable cameras on the market. (a) GoPro (2002). (b) SenseCam (2005). (c) Looxcie (2011). (d) Narrative Clip (2013).

Manuscript received December 15, 2015; revised April 4, 2016, May 20, 2016, and July 19, 2016; accepted September 7, 2016. Date of publication October 27, 2016; date of current version January 13, 2017. This work was supported in part by research projects TIN2012-38187-C03-01, TIN2015-66951-C2-1-R (MICINN), SGR 1219 (AGAUR), and 20141510 to Maite Garolera (Fundaci Marat TV3). The work of M. Dimiccoli was supported by a *Beatriu de Pinòs* grant (Marie-Curie COFUND action), and the work of P. Radeva was supported by an *ICREA Academia* grant. M. Bolaños and M. Dimiccoli contributed equally to this work. This paper was recommended by Associate Editor Z. Yu. *(Marc Bolaños and Mariella Dimiccoli contributed equally to this work.)*

The authors are with the Universitat de Barcelona, 08007 Barcelona, Spain, and also with the Computer Vision Center, 08193 Barcelona, Spain (e-mail: marc.bolanos@ub.edu; mariella.dimiccoli@cvc.uab.es; petia.ivanova@ub.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2016.2616296

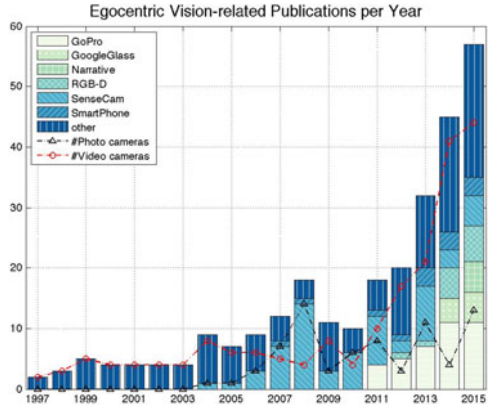

Fig. 3. Histogram of the number of research papers published per year related to egocentric vision. The different colors indicate how many papers used each kind of camera. The dashed blue and black lines make a less specific distinction, showing the number of studies that used photo (LTR) or video (HTR) cameras, respectively.

[89], and a large number of noninformative images that capture nonmeaningful information such as walls, the sky, parts of objects, etc. Furthermore, the sheer number of data that a visual lifelog consists of and the rate at which they increase (up to 2000 images per day or around 800 000 images every year) imposes a need for efficient methods to extract and locate relevant content concerning the wearer from the photo stream. Regarding HTR cameras, if they were employed for a lifelog analysis, the problem of the amount of data would be even more acute and would additionally imply the need of huge computational resources.

In response to the challenges and opportunities introduced by analysis of visual lifelogs, and more generally, by wearable cameras, computer vision scientists have rapidly become more interested in the subject over recent years. By searching for the keywords egocentric vision, first person vision, ego vision, and visual lifelogging, using Google Scholar, DBLP, and visionbib.com, we found 274 papers in total devoted to visual lifelogging. For each of them, we annotated the type of camera used in the study and generated the plot in Fig. 3, which represents all the papers related to egocentric vision up to November 2015. As can be seen, interest grew very fast in the last years and the number of papers published increased by over 50% in 2014 alone. Dotted lines show the comparatively small amount of work devoted to the analysis of image streams captured by photo cameras. This trend seemed to temporally change from 2007 to 2010, when the popularity of SenseCam resulted in a growth in the use of photo streams.

An additional indication of the interest in this emerging field is the fact that in the last years, four surveys of wearable cameras and egocentric vision have been published. One, written by Doherty *et al.* [32], focuses on explaining the ethical and data management issues that must be taken into account when developing some health-related application using wearable cameras. The second one, by Betancourt *et al.* [12], provides a general perspective on egocentric vision and devotes most of its analysis to the egocentric camera hardware, egocentric datasets, augmented reality, algorithm types, and feature types used in the literature from 1997 to 2014. This analysis is focused on providing a historical perspective of egocentric devices and their algorithms in addition to several ways of categorizing the existent papers in this field. The third one, which is a book by Gurrin *et al.* [41], focuses on data management and distinguishes between data storage, organization and visualization, while also provides an overview of potential applications. In the fourth study [42], Harvey *et al.* present their work from the perspective of providing an aid to human memory. They analyze the human memory mechanisms from a psychological perspective and propose a pipeline for enhancing it based on segmentation, context enhancement (recognizing objects and people), and image retrieval.

This paper focuses on addressing the question: How far are we from being able to automatically tell our stories using egocentric photo streams? The process of fully understanding the story behind the pictures is fundamental toward enabling a wide range of applications [27] and user cases [45], especially related to health. As we explained, since these applications require observations over long periods of time, data should be acquired by photographic cameras (e.g., SenseCam, Narrative, etc.) instead of video cameras (e.g., GoPro, GoogleGlass, Looxcie, etc.). To this end, a thorough review of the published advances in egocentric data analysis is presented, and research insights are provided. In contrast with previous surveys, we review and give details of studies that focus on both photographic and video cameras, considering which aspects should be reformulated and modified for their applicability in the LTR domain, and thus for egocentric storytelling.

To summarize, our contributions are as follows:

1) review of methods for acquiring, organizing, summarizing, and browsing large collections of unstructured data;
2) organization of the available literature around the central questions necessary to address the storytelling problem: *Was the user interacting with somebody? How?*, *Where* is he/she?, *When* did the event occur? and *What* is the person wearing the camera doing?;
3) highlights of the weaknesses and strengths of the reviewed techniques with respect to their applicability to the LTR domain (at the end of each subsection);
4) extensive analysis of the available datasets and source code related to the storytelling problems;
5) open problems and challenges in the field of egocentric vision with the final goal of storytelling.

The remainder of this paper is organized as follows. In Section II, we review the most important papers devoted to the task of acquiring, organizing, summarizing, and browsing large and unstructured collections of egocentric data. The solutions to these problems provide a basis to further analyze the data content, as in Section III, where we review papers that claim to construct semantic building blocks for storytelling. Concluding remarks about applicability to the LTR domain are given at the end of each subsection. In Section IV, we summarize the available egocentric datasets with the corresponding annotations, as well as the egocentric vision software. Finally, in Section V, we draw our conclusions and give some possible future directions for the research necessary to fill the gap between raw egocentric data analysis and visual storytelling.

TABLE I
SUMMARY OF ALL THE VISUAL LIFELOGGING PAPERS REVIEWED IN THIS SURVEY RELATED TO ACQUIRING, ORGANIZING, SUMMARIZING, AND BROWSING LARGE COLLECTIONS OF UNSTRUCTURED DATA

TOWARDS STORYTELLING FROM VISUAL LIFELOGGING							
Section II-B: **Informative Images Detection**							
[96]	[61]						
Section II-C: **Temporal Segmentation**							
[60]	[30]	[28]	[62]	[86]	[64]	[18]	[73]
[88]	[23]						
Section II-D: **Egocentric Summarization**							
[83]	[49]	[40]	[64]	[15]	[61]		
Section II-E: **Content-Based Search and Retrieval**							
[94]	[24]	[68]	[4]	[93]			

II. VISUAL LIFELOGGING ACQUISITION, SEGMENTATION, AND SUMMARIZATION

This section reviews the literature concerning acquiring, structuring, and summarizing visual lifelogging data, which is summarized in Table I.

A. Data Acquisition

The positioning of a wearable camera is of crucial importance for lifelogging data acquisition from the point of view of its later application. Mayol-Cuevas *et al.* [66] evaluated, partially through simulations on a 3-D facet model of the human body, four attributes of optical devices with respect to their position on the wearer's body: social acceptability, absolute field of view (FOV), resilience to body motion, and view of the handling space region. That study concluded that wearable cameras placed on the chest are the most socially acceptable and therefore offer the advantage of not interfering with social interactions. In addition, they are relatively resilient to the disturbances introduced by the wearer's own motion and are closely linked to the user's workspace, since they allow visualization of the manipulative space in front of the wearer's chest. However, the FOV is quite narrow and does not allow the focus of the wearer's attention to be modeled. In contrast, cameras worn on the head have a wider FOV and do allow this attention to be modeled, but they are the most sensitive to the wearer's motion and suffer from low social acceptability. A compromise between the size of the FOV, accessibility to the handling regions, sensitivity to ego-motion, and social acceptability is offered by wearable cameras placed on the shoulder. The authors also considered the possibility of wearing multiple devices on different parts of the body so that their FOVs would be complementary, with the joint FOV computed as the union of the individual FOVs.

Remarks: Since for long-term image acquisition, social acceptability is crucial, placement on the chest is usually considered the best choice. In addition, it has the advantage of offering access to the handling space, and the manipulation of objects can be focused.

B. Informative Image Detection

Once images have been acquired, before proceeding with any structuring, analysis, and summarization, proper cleaning of the images is necessary. This need stems from the fact that egocentric images are nonintentional images, that is, nobody decides when and of what to take a picture. As a result, a significant number of images can be blurred, can be dark, or can capture noninformative data (the sky, the ground, walls, etc.). In Xiong and Grauman [96], informative images are defined as "intentional" images, obtained once those with undesired artefacts, such as light saturation, blurred images, or useless information (the sky, walls, etc.) have been removed. Lidon *et al.* [61] define as informative any image that includes objects and/or people, and which is of reasonable quality, assuming that it does not include any undesired artefacts (e.g., blurring, darkness, or occlusions). With this definition, they trained a binary convolutional neural network (CNN) to make this distinction.

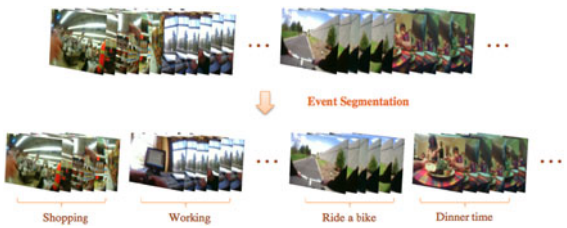

Fig. 4. Example of the desired event segmentation applied to lifelogging data. The goal is to group with respect to their main event, considering the activities, objects or people involved.

C. Temporal Segmentation

Lifelogging data typically consist of long unstructured videos or photo streams. Organizing and structuring them into homogeneous temporal segments, corresponding to different events and/or environments (see Fig. 4), are very important to facilitate browsing and analysis of the images. State-of-the-art methods for egocentric data segmentation can be classified into two broad classes depending on whether the homogeneous segments represent what the wearer sees or does.

The former class uses features that can capture the characteristics of the environment around the wearer as image representation. Early work aiming at segmenting the sequences into visually homogeneous segments was based on low-level features. Li *et al.* [60] have proven that it is possible to distinguish different events simply by treating SenseCam images as time-series data and calculating the eigenvalue peaks in consecutive windows of images. Doherty *et al.* [28], [30] used different descriptors for image representation and the metadata available from the camera sensors. Lin and Hauptmann [62] proposed a simple approach based on using color features in a time-constrained K-means clustering algorithm, capable of maintaining temporal coherence on the splitting of events. Spriggs *et al.* in [86] proposed a method for simultaneous temporal segmentation and recognition of activity related to cooking. They captured videos at the same time from a single wearable video camera and multiple other static cameras, sensors, microphones, etc., and used both sensor data and visual GIST descriptors to describe the frames. For the unsupervised scene segmentation, they applied a Gaussian mixture model. More recently, Talavera *et al.* [88] proposed the use of CNNs computed on the whole image using AlexNet as a fixed feature extractor for image represen-

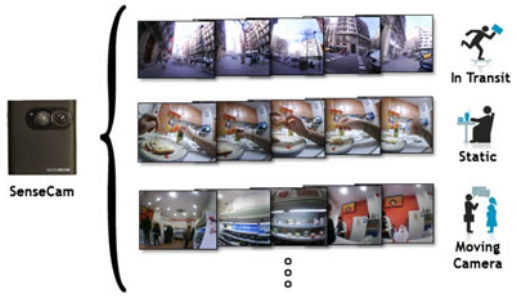

Fig. 5. Motion-based segmentation framework proposed in [18]. By including motion features to describe egocentric pictures, they can separate the events considering the dynamism of the activities performed.

tation. That work, designed for egocentric photo streams, uses a graph-cut algorithm to temporally segment the photo streams and includes an agglomerative clustering approach with concept drifting methodology, called ADWIN.

Methods focusing on what the camera wearer does mostly use motion information as image representation. Usually, optical flow is used to distinguish between static, moving the head/camera and in-transit frames [18], [64] (see Fig. 5). To focus on long-term ego-activities, Poleg *et al.* [73] proposed the use of the so-called integral motion, which is closely related to the wearer's activity. By integrating the instantaneous displacements at fixed image patches, the variations due to head rotation are eliminated, since their mean is practically zero, leaving only the consistent displacement caused by forward motion. A different approach, based on CNNs, is adopted by Castro *et al.* [23]. They gathered a large egocentric dataset from a single user and fine-tuned a CNN pretrained on ImageNet for activity classification. They proved that the network trained on the data of a single user can be retrained to generalize to new users. The main problem with this approach is that a new set (several thousands of images) must be labeled from scratch whenever it is necessary to predict the events affecting a new user with the model.

Remarks: The applicability of motion as a feature, though relevant when dealing with videos, has proven to be rather limited for photo streams. In the latter case, the use of richer representations, such as global CNN-based features, seems crucial to compensate this limitation. The use of time-dependent methods for egocentric segmentation is also a must considering the nature of the data. A promising approach to improve the results of the segmentation of egocentric sequences is the addition of semantic-level features (scenes, objects, social interaction, actions, etc.). This additional information would be an important step to bring machine segmentation closer to the way humans segment unconstrained streams of images.

D. Egocentric Summarization

Summarization is the process of generating a proper, compact, and meaningful representation [90] of a given sequence through a subset of representative frames or segments. This step is crucial to help manage and browse large volumes of lifelogging video content efficiently. Basically, there are two kinds of summaries that can be produced: a static video story board, which is composed of a set of salient images extracted or synthesized from the original sequence, and dynamic video skimming: a shorter version of the original video made up of several shots, comprised of a series of frames. To fully exploit the potential of visual lifelogs in a variety of applications, an egocentric summarization method should be designed to aid in the visualization, indexing, and browsing of autobiographical events, with the least possible semantic loss.

Story board summarization has been traditionally formulated as grouping images into coherent collections by relying on low-level spatiotemporal features and then selecting the most representative image (or set of images) from each collection [83]. Based on this classical approach, Chowdhury *et al.* [25] and Jinda-Apiraksa *et al.* [49] developed similar techniques for keyframe selection in egocentric sequences based on quality measures [25], [49] and both quality and diversity measures [25]. More complex features for grouping were used by Bolaños *et al.* [15]. Their methodology, adapted for photo cameras, uses the AlexNet CNN as a feature extractor to characterize each frame. Then, using those features, they apply event segmentation using a hierarchical clustering algorithm and a posterior single keyframe selection by applying the random walk algorithm to each of the segments.

While these methods rely solely on low-level features, some recent work has introduced a semantic level in the keyframe selection process. Ghosh *et al.* [40] suggested that video summarization should be driven by the presence of important people and objects. Following this idea, they proposed a method that reveals salient people and objects based on their interaction time with the camera wearer and then selected keyframes according to keyobject event occurrences. Lu and Grauman [64], following on from their previous work, suggested that video summarization should preserve the narrative character of a visual lifelog and proposed a shot selection consisting of three terms:

1) a term that models story coherence by favoring shots capable of following the inherent story;
2) a term that models importance, to choose only shots that show some important aspect of the day;
3) a term that models diversity and avoids repeating similar events.

Summarization that considers semantic topics was recently proposed by Schinasi *et al.* [81] and Varini *et al.* [91]. In [91], it is assumed that interesting scenes in a cultural experience, such as visiting a museum, are those associated with certain patterns of behavior of the camera wearer that are learned and used for classification. Taking into account the topic of interest of the user, different summaries can be generated from the same video. In [81], topics are revealed from a set of social media messages as highly connected messages in a graph, whose nodes encode messages and whose edges encode their similarities. Finally, the images that best represent the topic are selected based on their relevance and diversity. Lidon *et al.* [61], also working on photo sequences, proposed an event keyframe ranking method based on a tradeoff between image relevance and diversity after removing noninformative images (containing undesired artefacts, e.g., blurring, darkness or occlusion, or showing the sky, walls, or object parts) by using a new binary CNN-based filter. Their

relevance criteria took into consideration several semantic measurements, including whether faces and/or objects were present, as well as whether the images had a high saliency value.

Remarks: A semantic-oriented approach to egocentric summarization seems to be the most suitable for lifelogging data. Indeed, users would ideally search for complex autobiographical events that encompass simpler human actions and may not be directly correlated with their visual appearance. When dealing with photographic cameras, and due to the nature of their data, the only possible way to tackle the summarization problem is through the keyframe selection approach. Taking this into account, methods like [64] should be reformulated, either considering the video subshots as single frames, or developing a fine-grained segmentation procedure. This procedure should separate the data into a large number of events to have enough segments to apply the subshot selection correctly.

E. Content-Based Search and Retrieval

Retrieving images from a large personal database allows us to browse, search, and find images of previously seen objects or places and thereby has the potential to solve a broad range of problems in egocentric vision, such as:

1) searching for elements (Have I seen this before?);
2) navigating (How often do I visit this place?);
3) understanding the environment (Where am I right now?);
4) efficiently organizing huge amounts of data.

Following these premises, in [94], Wang *et al.* built a system for content-based searching and browsing that starts by splitting the stored data into segments and extracting three kinds of information:

1) time and other relevant attributes;
2) low visual features;
3) audio features.

Then, in the retrieval step, they applied time-based filtering by comparing the time attributes of the images in the database with the query introduced by the user. A clustering step then extracts a representative clip from each cluster, and finally, the user can provide one or more query images for the system to refine the search based on visual features and improve the query result. Still, several open issues remain: in many situations, it is difficult to recall the time and where the photo we are looking at was taken; visual features are too simple to capture real object shape and texture differences, and furthermore, audio features are not provided by all wearable devices. Aghazadeh *et al.* [4] proposed to retrieve novel scenes and actions with respect to a previously acquired egocentric dataset by using a set of "alignment" sequences and matching them with a new "query" sequence by using dynamic time warping.

Assuming that searching, browsing, or summarization in visual lifelogging would largely benefit from semantic concept representation, Wang and Smeaton [93] investigated the selection of the most appropriate combination of concepts for event representation. Their strategy basically consists of reasoning on semantic networks using a density-based approach. Min *et al.* [24], [68] represented millions of egocentric images on a sparse graph. They represented each image as a node in the graph and added an edge between two nodes, when they belonged to the same bag in a BoW representation. Relying on this representation, they showed that local density clustering is more suitable than global clustering methods, considering the high redundancy that lifelogging data inherently possess.

Remarks: Many issues remain regarding content-based retrieval techniques, for instance: How can we make use of the basic building blocks extracted from lifelogging (actions, people, and environments)? The usage of a multilevel and multimodal descriptions based on the recognition of actions, people, objects, and environments could provide a detailed image description close to text-level, which could allow high retrieval accuracy.

In methods such as [24] and [68], new challenges would arise when dealing with photo data, considering the higher variability of consecutive images compared to video sequences.

TABLE II
SUMMARY OF ALL THE VISUAL LIFELOGGING ANALYSIS-RELATED PAPERS REVIEWED

VISUAL LIFELOGGING ANALYSIS								
Section III-A: **Interacting? How?: Social Interactions**								
[31] [85]	[5]	[6]	[35]	[9]	[1]	[3]	[2]	[72]
Section III-B: **Where?: Scene Understanding**								
Concept Recognition				[21]				
Section III-B1: Object Recognition				[75]	[74]	[37]	[17]	[14]
Section III-B1: Object Discovery				[52]	[19]	[16]		
Section III-B2: Spatial Localization				[55]	[13]	[95]		
Section III-C: **When?: Time-Based Localization**								
[62]	[88]	[23]						
Section III-D: **What?: Action Recognition**								
Section III-D1: Body movements				[73]	[56]			
Section III-D2: Object-hand interaction				[34]	[87]	[10]	[71]	
				[59]	[58]	[77]	[76]	
Section III-D3: Attention				[36]	[65]	[79]		
Section III-D4: Other Approaches				[97]	[84]	[51]		

III. VISUAL LIFELOGGING ANALYSIS

We present an overview of the most important papers on visual lifelogging analysis and the problems they tackled, organized around four basic questions: Is the user interacting? How? Where is the user? When are the events occurring? and What is the user doing?. Table II lists the papers and related information.

A. Interacting? How?: Social Interactions

Following the definition by Rummel [78], social interactions are all acts, actions, or practices of two or more people mutually oriented towards each other. Given the powerful social nature of humans, the analysis of social interactions in lifelogging data is of fundamental importance to understanding human behavior. Furthermore, the presence of people and social interactions are consistently associated with event memorability [47], and therefore, their detection is also potentially useful for keyframe extraction or to estimate the importance of events

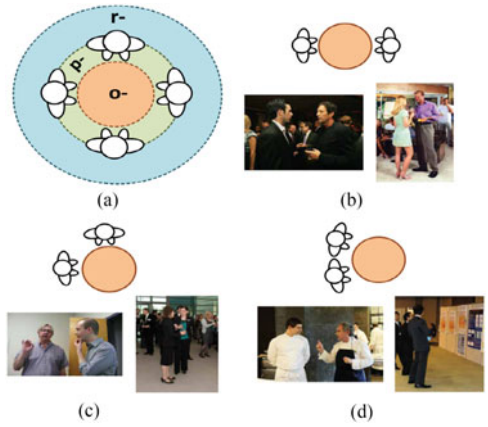

Fig. 6. Different arrangements of F-formations that are useful for social interaction analysis. (a) Circular arrangement. (b) Vis-a-vis arrangement. (c) L-arrangement. (d) Side-by-side arrangement. Image adapted from [82].

in a lifelog [31]. From the perspective of computer vision, social interactions can be characterized by patterns of attention between individuals. Analyzing attention patterns requires the detection, tracking, and locating of people in 3-D environments. Indeed, when interacting with others, we naturally tend to place ourselves in certain positions so as to stand close to those we interact with and avoid occlusions. F-formations [53] have been demonstrated to be a suitable formalism for modeling social interaction behavior. Following the original definition by Kendon [54]:

An F-formation arises whenever two or more people sustain a spatial and orientational relationship in which the space between them is one to which they have equal, direct, and exclusive access.

Examples of F-formations are given in Fig. 6. The F-formations theory has been successfully applied in social interaction analysis [46] using classical videos or still images, and more recently to egocentric videos [6]. Head estimation and 3-D location are crucial for the detection of F-formations. Indeed, a rough estimate of someone's head pose allows us to understand with a certain precision what the person is looking at, while it is important to estimate the distance people have from the camera wearer and other people if there is interaction.

In sequences captured through a wearable camera, pose estimation is a challenging task due to the continuous changes of aspect ratio, scale, and orientation. A common way to address this problem [5], [6], [9], [72] is to assume that where a group interacts in a discussion, the head of each person will be oriented for a while toward the person who is speaking, and to use a model to capture this behavior over time. Generally, in video sequences, this is achieved through a hidden Markov model or Markov random fields, where the latent variable corresponds to the head pose and the observed variables to the results of a multiple person tracker, applied to the input images. The only works devoted to the analysis of photo sequences are [1]–[3]. In this context, tracking people is very challenging due to the abrupt and very frequent changes of view. The proposed approach basically consists of computing backward and forward correspondences for each face detected in the sequence and of grouping similar tracklets into bags, which should correspond to different people (see Fig. 7). A combination of first-person and third-person views is considered by Park and Shi [85] to predict social saliency, considered as the likelihood of joint attention, in real-world scenes with multiple social groups. This is basically achieved by modeling social formation features that encode the geometric relation between the joint attention and spatial distribution of the members of a social group.

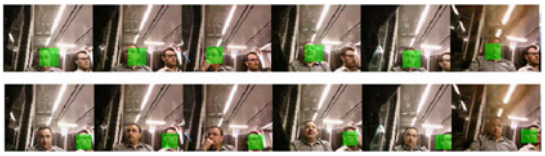

Fig. 7. Example of multiface tracking obtained by applying the method in [1] to track multiple faces in LTR sequences captured by a wearable camera. Each row represents the track of a different person.

Remarks: In general, there is common agreement about the need to track people, head orientation, and 3-D locations to detect F-formations that represent social groups in egocentric sequences; however, two fundamental problems arise. First, since in different social scenarios, distances and poses can assume different degrees of significance, clearly a need emerges for an algorithm to be able to adapt to different situations and learn how to treat distance and orientation features depending on the context. As a consequence, the choice of which data to use for training is crucial. Second, distances and poses strongly depend on where the camera is worn (eyeglasses, on the head, on the neck, etc.). Except [1], [3], all the methods mentioned above rely strongly on temporal coherence, since they were conceived for video sequences. Further advances in the analysis of social interactions through photographic cameras would require us to focus on features that are less sensitive to changes over time, such as people's body movements, which are consistently associated with emotional experiences [67] and could, therefore, be considered cues of social interactions.

B. *Where? Scene Understanding*

To answer the question "Where is the user?" we require a semantic understanding of the elements that surround the camera wearer, such as objects, people, and environments, since they represent the cues available to recognize his/her surroundings. In this section, we provide an overview of computer vision tasks related to scene understanding, such as object recognition, spatial localization, scene parsing, and scene recognition. All of them share the goal of determining what the most promising techniques are for understanding scenes in lifelogging data.

1) Object Recognition and Object Discovery: Scenes can also be characterized by a vocabulary of concepts that can be found in them. With this aim, we consider the following problems: object recognition, which intends to identify the category that a given object belongs to; and object discovery, which detects, recognizes, and reveals new objects in images that possibly have never been seen before by the algorithm in the previous images. Due to the free motion of the camera and to the passive acquisition of lifelogging data, objects are frequently occluded

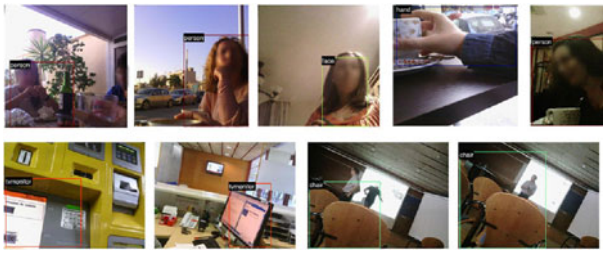

Fig. 8. Examples of objects revealed by the ego-object discovery methodology [16] for two different subjects (one per row). Better viewed in digital format.

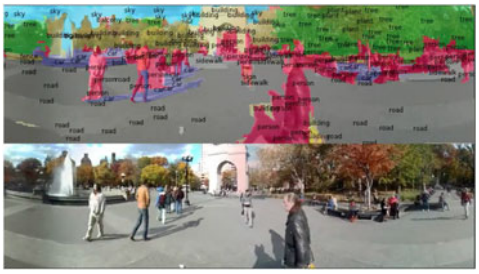

Fig. 9. Example of the result obtained (top) by applying a scene parsing algorithm to a conventional nonegocentric image (bottom). We can see the different segments found (separated by different colors) and the classes assigned to each of them. Picture adapted from [33].

and their appearance may vary broadly. Thus, the object recognition problem in egocentric data is becoming a challenging and active research field. The first work on object recognition in the domain of lifelogging is by Byrne *et al.* [21], who successfully validated supervised concept recognition, referring to relevant objects or scenes as concepts. Furthermore, using the output of the detector, they showed that the images that compose a lifelog collection tend to be temporally consistent in their visual properties, as well as in the concepts they contain. Because of this concept consistency, they suggested that an efficient automatic extraction and inference of higher level semantic concepts based on cooccurrences and known relationships would be feasible. Bolaños *et al.* [17] developed an active labeling method to generate a sufficiently large number of training examples to train an efficient supervised classifier. The method, based on a combination of hierarchical clustering trees, uses an unsupervised learning algorithm to organize the data, selecting the most informative part, asking the user for their labels, and using the feedback provided to improve the classification in a semisupervised way. Ren *et al.* in [74], [75] and Fathi *et al.* in [37] used head-mounted cameras and proposed methods that recognize objects held in the user's hand. They segmented the background from the foreground (hands and objects) using optical flow features and relying on the fact that foreground objects will usually move in a more dynamic way while the background is more static.

Focusing on the task of object discovery in lifelogging data (see example in Fig. 8), Kang *et al.* [52] proposed a method, starting from an initial segmentation, that clusters only samples with higher correlation that should belong to the same object type. To this end, starting from the initial segmentation, they provide a merging strategy for segments that closely cooccur in most images. In this way, they complete objects that might be composed of different, but clearly defined parts (e.g., a laptop composed by a screen and keyboard). With the same goal, Bolaños *et al.* in [16] and [19] proposed the use of a state-of-the-art objectness detector and a pretrained CNN specialized in object recognition to extract a set of rich features for each object candidate followed by clustering them. The clustering integrates a "Bag of Refill" strategy of previously discovered object instances as a knowledge reuse methodology.

2) Spatial Localization: Bettadapura *et al.* [13] proposed a method called FOV localization that combines localization techniques with egocentric images to localize the user(s) in the environment. To do so, they used a reference dataset, which can be images from Google Street View or prerecorded videos from fixed cameras, and matched them to the data acquired by the user's photographic or video camera to obtain his/her localization. They tested the system on multiple datasets captured indoors and outdoors. Additionally, they proposed a combined FOV localization system for simultaneous localization of multiple users of wearable devices. Wannous *et al.* [95] also proposed a methodology for localization and action-related event recognition. They used a shoulder-mounted video camera to acquire images of daily indoor living (e.g., kitchen, office, library, etc.) and built a 3-D model of the different scenes. In their work, they proved that their models were more powerful than simpler 2-D ones and were able to recover information from previously seen scenes with query images.

Remarks: Another interesting approach that egocentric vision could benefit from is scene parsing. This is based on image segmentation, that is, separating out all the regions in an image that belong to different objects or regions. Furthermore, these kinds of techniques classically consist of providing pixel-level segmentation of the whole image and at the same time assigning an object class to each of the pixels (see the example of scene parsing in Fig. 9). To do this, most of the methods use pixel-level classifiers to achieve an initial segmentation, and then, a graphical model is applied to smooth and correct the boundaries of the segments [33], [98]. A limited amount of work in this field can be found in the literature, but none of it was specifically designed or tested on egocentric and lifelogging datasets. Considering the differences we could find in an egocentric dataset (and more precisely in lifelogs) with respect to those typically used in scene parsing, we can enumerate some clear points to take into account when working on scene parsing.

1) Scene parsing datasets are usually composed of natural and urban scenes (in general, outdoors), and their corresponding class distributions have a high percentage of training samples related to those environments, that is, the egocentric lifelogging datasets for scene parsing would be very different considering the indoor and routine settings where people usually spend most of their time.
2) Also taking into account the fact that egocentric vision datasets are composed of routine and redundant scenes, scene parsing methods focusing on lifelogging images should provide some higher context and knowledge-reuse

mechanisms to take advantage of the previously parsed images in the egocentric sequence.

Related to scene parsing, it would also be useful to be able to recognize the scene the user is in. Although no work has been presented with this purpose using egocentric data, a good example with conventional images is the dataset Places205 [100]. This information could help when deciding, for instance, how we should segment the day into events or use this information to exploit the environment–object relationships.

Although good methodologies have been proposed for object recognition and object discovery using egocentric and lifelogging images, there is still a lot of work to do to semantically describe the camera wearer's environment at a high level. The development of object detection methods specifically designed for egocentric images could not only improve existent recognition and discovery methods, but also set a more robust basis for the future appearance of scene parsing of lifelogging images. To achieve these goals, new computer vision techniques able to cope with blurring, light saturation, and the occlusion of objects have to be developed. Hence, new techniques for gathering huge labeled datasets not only for object detection, but most importantly for scene parsing, must be developed. Furthermore, the addition of GPS or visual localization techniques to scene parsing could clearly improve understanding of the environment. The most promising technique applicable to scene parsing is using fully convolutional networks [63], which are able to infer the classes of each pixel treating the image as a whole instead of the current pixel-level centered classifications.

Finally, note that all the work on object recognition relies on the user-like focus and point of view that head-mounted cameras offer. This approach would not be feasible for real applications, where neck hanging cameras are usually used because they are considered less obtrusive and more user-friendly [43], despite not always being able to show what the user is doing. Moreover, these algorithms, which rely heavily on temporally close video frames and motion information, would not be applicable to LTR photographic cameras either.

C. When? Time-Based Localization

Time information is particularly important to determine the causal relations in human behavior. For instance, it could be useful in understanding which factors determine crises in people affected by bipolar disorder. The most common annotation tool used for keeping a record of the time in lifelogging data is the time stamp provided by cameras. By using this information, one can easily establish the temporal placement of the data in the long term, the order of the images, and their temporal distance for photographic cameras in the short term or daily. Some works have studied incorporating temporal information as a complementary feature indicator for achieving an indirect prediction. As an example, in [62] and [88], the authors have treated the data acquired as a time-series to properly segment the different events present in a day. In [23], both the day of the week and the time of the day have been used for training a classifier with the ability to categorize different events. Naaman *et al.* [69] studied the role of the time stamp as a memory cue in a psychological experiment on conventional images and concluded that people are unable to retrieve their memories when only given the time and date; consequently, additional information is needed for retrieval methods to be effective.

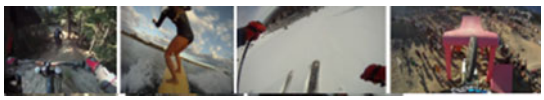

Fig. 10. Examples of first-person point of view images performing various sports. Image adapted from [56].

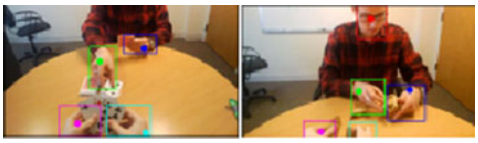

Fig. 11. Examples of first-person point of view images for recognizing activities involving hands. The algorithm is capable of detecting the left and right hands of the user, in pink and light blue, respectively, and the left and right hand of the person he/she is interacting with, in dark blue and green, respectively. Image adapted from [58], devoted to hand disambiguation.

D. What? Action Recognition

Inferring what the camera wearer is doing from a visual lifelog basically requires the categorization of everyday activities. The categories to focus on depend on the kind of application. For instance, in healthcare and well-being applications, occupational therapy research may guide the selection of the target activities and related concepts (see Fig. 10 as an example of sports category recognition). For diet monitoring applications, eating actions will be the focus, whereas in applications related to the diagnosis of dementia, the focus will be on daily life activities such as dressing, making coffee, and cooking. In quantified-self applications, activities like housework, watching TV, working/studying, eating/drinking, etc., are the most prevalent activities.

Traditional action recognition methods can be broadly classified depending on the kind of features they use to represent actions, with body movement analysis and the use of the objects involved in the action being the most common choices. Only very recently has the scene context been used to improve action recognition. Still, the choice of the representation strongly depends on the kind of actions to be classified.

1) Body-Movement-Based Methods: In an egocentric setting, general body movements such as running, walking, moving the head/camera, or staying still are usually estimated relying on motion features (when this is possible with the temporal resolution of the camera). Usually, based on such features, the ego-action classification can also be used for event segmentation. Typically, video cameras like GoPro, which capture around 30 frames/s, are used to gather data. Poleg *et al.* [73] proposed integrating instantaneous displacements of fixed image patches over a long period of time to remove the zero mean variations due to head rotation. By applying this process, they leave only the consistent displacement caused by forward motion. The cumulative displacement curves show different patterns for ego-motion activities so that activities become easy to classify. Instead of

TABLE III
SUMMARY OF CURRENTLY AVAILABLE PUBLIC EGOCENTRIC DATASETS

Name	Description	Type of Annotations	Camera
Egocentric Dataset of the University of Barcelona (EDUB) [16]	With 4912 images acquired by the wearable camera Narrative; divided into eight different days which capture daily life activities like shopping, eating, riding a bike, working, etc. It was acquired by four different subjects, two days each; and with 11 294 different object segmented instances from 21 different classes (TV, hand, person, car, sign, etc.).	Object labels and segmentations	Narrative
All I Have Seen (AIHS) [50]	Contains 19 days with a total of 45 612 images of 640 × 480 resolution, containing around 15 recurrent places/scenes appearing like home rooms, work office, work building, supermarkets, playgrounds, campus, biking trails, etc.	Not available	SenseCam
Intel Egocentric Object Dataset [75]	Has ten video sequences (100 000 frames) from two subjects manipulating 42 different types of everyday object instances.	Object labels and foreground and background segmentations	PointGrey
GeorgiaTech Egocentric Activities (GTEA) [37]	The videos captured by a cap-worn camera show seven types of daily activities, such as making a sandwich/coffee/tea, each performed by four different subjects. Each activity video is labeled with the list of objects involved; each frame has left hand, right hand, and background segmentation marks	Objects list and hands and background segmentations	GoPro
GTEA Gaze+ Dataset [36]	With video and audio recordings of seven meal-preparation activities such as making pizza/pasta/salad collected using eye-tracking glasses. Each activity was performed by five different subjects. Each frame has eye-gaze fixation data, and different activities such as opening fridge are annotated.	Gaze and actions performed	Tobii
First-Person Social Interactions Dataset [35]	Day-long videos of eight subjects spending their day at Disney World. The cameras are mounted on a cap worn by the subjects. Elan annotations containing the number of active participants in the scene, and the type of activity: walking, waiting, gathering, sitting, buying something, eating, etc.	Actions performed and social interactions at each time period	GoPro
Huji EgoSeg Dataset [73]	With 29 videos captured by an egocentric camera annotated in Elan format. The videos (some from YouTube and others recorded by Hebrew University of Jerusalem researchers) contain various daily activities.	Actions performed at each time period	GoPro
UT Ego Dataset [64]	Has four videos captured by a Looxcie wearable camera (head-mounted). Each video is about 3–5 h long, captured in a natural, uncontrolled setting. The videos capture a variety of daily activities.	Important regions annotation	Looxcie
Interactive Museum Dataset [8]	A gesture recognition dataset taken from an egocentric perspective in a virtual museum environment. It has five different users who performed seven hand gestures.	Hand gestures	No Information
VINST—Visual Diaries [4]	With 31 videos capturing the visual experience of a subject walking from a metro station to work. It consists of 7236 images in total. Each image is annotated with a location ID which covers nine unique labels in total. Temporal segments corresponding to novel ego-motions are annotated as well.	Location and "novel ego-motions" annotations per frame	No Information
UCI Activities of Daily Living Dataset (ADL) [71]	Has 1 million frames of dozens of people performing 18 daily indoor activities such as brushing their teeth, washing dishes, or watching television, each performed by 20 different subjects. It includes annotations of 42 object classes.	Activities, object bounding boxes and classes, hand positions and interaction events	GoPro
EGO-HPE [6]	A set of egocentric videos with different subjects for head pose estimation. Each video is annotated at the frame level for five yaw angle orientations (−75, −45, 0, 45, 75) with respect to the subject wearing the camera.	Face orientation	Vuzix Smart Glass
EGO-GROUP [6]	A social group detector dataset for egocentric vision, which consists of ten videos collected in different situations: a laboratory, a coffee break, a conference room and an outdoor scenario.	People group composition	Vuzix Smart Glass
JPL First-Person Interaction Dataset [79]	Human activity videos taken from a first-person viewpoint. The dataset specifically aims to provide first-person videos of interaction-level activities, recording how things look from the perspective of a person/robot participating in physical interactions.	Actions performed in each time period	GoPro
NUS First-person Interaction Dataset [70]	Dataset for interaction recognition with eight interactions in two perspectives (first-person and third-person) resulting in 16 classes in total. The dataset will be made publicly available at a later date. It contains two human–human interactions, two human–object–human interactions, and four human–object interaction classes. It contains 260 videos with at least 15 samples in each class.	Interaction type	GoPro
CMU Multi-Modal Activity Database (CMU-MMAC) [86]	Multimodal dataset of 18 subjects cooking five different recipes (brownies, pizza, etc.); also contains audio, body motion capture, and IMU data.	Frame-level action	No Information
CMU EDSH (hands under varying illuminations) [59]	Dataset of over 600 hand images taken under various illumination conditions and different backgrounds. Each image is segmented at the pixel level.	Hand segmentation	GoPro
EgoHands Dataset [7]	Contains 48 Google Glass videos of complex, first-person interactions between two people. The main intention of this dataset is to enable better, data-driven approaches to understand hands in first-person computer vision.	Hand segmentation	Google Glass
Unige-Hands Dataset [11]	Videos recorded in five different locations (office, street, bench, kitchen and coffee bar) intended for hand detection.	Hand/No Hand label per frame	GoPro
Yale Human Grasp Dataset [20]	Dataset with 27.7 h of tagged video recorded by two housekeepers and two machinists during their regular work activities. It includes the tagged grasp type with its time information, objects manipulated and parameters of the performed task.	Grasp tagging, and interval and object labels	RageCams

TABLE III (*Continued*)

Name	Description	Type of Annotations	Camera
UT Grasp Data Set [22]	Dataset under controlled environment performed by four different subjects. They were asked to grasp a set of objects placed on a desktop with specific types of grasps. The most common subset of 17 grasp types from Feix's Taxonomy [38] were selected to perform these everyday activities.	Hand grasp type and start/end frame number	GoPro
Life-logging EgoceNtric Activities (LENA) [84]	Egocentric video database containing 13 categories of activities relevant to lifelogging applications performed by ten different subjects. Each subject recorded two clips for one activity (20 clips per activity). Each clip has a duration of 30 s.	Activities performed.	Google Glass
COGNITO [10]	Nonperiodic manipulative tasks in an industrial context. All the video sequences were captured with on-body sensors consisting of IMUs, a backpack-mounted RGB-D camera for top-view and a chest-mounted fish-eye camera for the front view of the workbench.	Activity labels and objects and wrist tracklets	RGB-D and others
Michigan-Milan Indoor Dataset [39]	With ten video sequences collected with common smartphones in a variety of environments, including offices, corridors and large rooms, where the observer moves freely (6 DoF) around the scene.	Image segmentations with the labels "ceiling," "floor" or "wall"	Smartphone
Bristol Egocentric Object Interactions Dataset [26]	Dataset captured with wearable gaze tracker software containing various predefined actions of daily living in different indoor locations (kitchen, workspace, gym, laser printer, corridor, and weight-lifting machine). The videos in each sequence are recorded by three to five different users.	3-D maps and 3-D objects GT	ASL Mobile Eye XG
DogCentric Activity Dataset [48]	DogCentric Activity Dataset is composed of dog activity videos taken from a first-person animal viewpoint. The dataset contains ten different types of activities, including activities performed by the dog itself, interactions between people and the dog, and activities performed by people or cars. The videos are in 320 $\times$ 240 image resolution, 48 frames/s.	Activity performed	GoPro
UEC EgoAction Dataset [56]	A set of videos (acquired by the researchers or public from YouTube) recording different sports (skiing, mountain biking, etc.). Each video is several minutes long and contains a wide set of actions performed by the user.	Activities performed	GoPro

TABLE IV
LIST OF THE MOST RELEVANT PUBLIC SOFTWARE RELATED TO EGOCENTRIC VISION

Alireza Fathi's Egocentric Vision Toolbox [36], [37], [74]	Toolbox including functions for applying different data processing to egocentric videos, including motion estimation, image segmentation, object classification and action classification among others.
OpenCV and CUDA	http://ai.stanford.edu/~alireza/GTEA_Gaze_Website/Code/index.html
Ego-Object Discovery [16], [19]	Object Discovery Algorithm on Egocentric Images. Semisupervised algorithm that uses initial object proposal generation, a CNN-based feature representation, false positive filtering, and an interactive object discovery with Refill strategy.
Matlab and Caffe	https://github.com/MarcBS/Ego-Object_Discovery
Detecting Activities of Daily Living in First-person Camera Views [71]	Train and test code for the problem of detecting activities of daily living (ADL). It applies novel representations including temporal pyramids to approximate temporal correspondences, and composite object models that exploit the differences between the objects when being interacted with.
Matlab	http://people.csail.mit.edu/hpirsiav/codes/ADLdataset/adl.html
Temporal Pooling of CNN Vectors [80]	It includes the pooled time-series representation framework as well as basic per-frame descriptor extractions including a histogram of optical flows (HOF) and histogram of oriented gradients (HOG).
Java and OpenCV [exec. only]	https://github.com/mryoo/pooled_time_series/
Temporal Segmentation of Egocentric Videos [73]	Software for segmentation and event classification of egocentric HTR videos. It applies a hierarchical classification using cumulative displacement curves.
Matlab and C++	http://www.vision.huji.ac.il/egoseg/
Doherty Wearable Camera Browser [30]	Application for data segmentation annotation and browsing. It supports analysis of images from the following photographic cameras: Vicon Autographer, Revue, or SenseCam.
[exec. only]	http://sensecambrowser.codeplex.com/
R-Clustering for Event Segmentation [88]	Segmentation of events in egocentric lifelogging photo streams. It uses CNN features and an energy minimization (Graph-Cut) technique to segment photo sequences.
Matlab and Caffe	https://github.com/MarcBS/SR-Clustering
Motion-Based Egocentric Segmentation [18]	It applies a robust SIFT-Flow motion estimation suitable for photo sequences to perform photo stream segmentation in motion-related events.
Matlab	https://github.com/MarcBS/Motion_Video_Segmentation
Egocentric Vision Keyframe Summarization [15]	The code extracts a visual summary of a set of egocentric images captured by a photo camera. The result is a collage with one image summarizing every event in the image set. It uses a frame representation by means of a CNN followed by an event segmentation based on agglomerative clustering and keyframe selection based on Random Walk.
Matlab and Caffe	https://github.com/MarcBS/Egocentric-Visual-Keyframes-Summary
Egocentric Snap Points Detection [96]	Automatic prediction of snap points in unedited egocentric video that is, those frames that look as if they could be photos taken intentionally. It makes use of a generative model for snap points that rely on a photo prior to intentional (conventional) images together with domain-adapted features.
Matlab and C	https://github.com/bxiong1202/snap-points

focusing on the goal of building discriminative motion features, Kitani *et al.* [56] used several modifications of classical motion-based feature vectors and built a complex Bayesian model for clustering.

2) Object–Hand-Interaction-Based Methods: A first-person point of view offers an ideal perspective from which to analyze hand–object manipulation or hand–eye coordination (see Fig. 11). The main idea, introduced by Fathi *et al.* [34] and further improved by the authors of [10], [71], and [87], is that objects are correlated with actions (e.g., dish and nibbling) and actions with activities, and these correlations can be exploited to build robust object models. However, the challenges come from additional occlusions (from manipulated objects, or self-occlusions of fingers by the palm) and the fact that hands interact with the environment and often leave the camera FOV.

Others have focused on different problems related to hand–object manipulation such as capturing the variability of hand appearance over a diverse set of imaging conditions and hand poses [59], disambiguating and tracking the observers hands and those of social partners [58], improving robustness against camera motion [74], [76], [77], or capturing the appearance of visual composites of humans and objects in interaction [71].

3) Attention-Based Methods: The use of manipulation-based approaches is restricted to scenes and objects where the user's hands present significant information. Attention-based approaches aim to identify objects to which the user pays particular attention, even in the absence of manipulation, since they could be key factors in self-behavior recognition. In general, these methods are applicable to data acquired by head, eyeglass, or ear-mounted cameras only. Attention can be used to find salient objects as in [65], or to capture the relationship between action and gaze, as in [36].

4) Other Approaches: To detect activities that cannot be fully characterized by body movement, object–hand manipulation, or object–gaze relationships, motion has been the most commonly used feature. Instead of trying to compute ego-motion, these approaches describe the frames that compose the actions; they use a set of motion and visual word features in a local (on a single frame) and global (on a set of consecutive frames) manner and create a specific structure for obtaining a temporally and spatially consistent representation of the action. Song *et al.* [84] obtained an accuracy rate of activity recognition of about 80% using the dataset they published (LENa dataset), by adopting the dense trajectory approach. In [79], the authors used a wearable video camera to capture and recognize a diverse set of actions (e.g., throwing, hand shaking, hugging, or waving) which, in this case, is made by other people toward the camera user. Recently, a newer approach for action recognition was proposed by the same authors in [80]. On this occasion, they used CNN features to describe the frames of an HTR video. To obtain a rich and motion-like representation, they then proposed the use of a temporal pooling operator. An interesting alternative to motion was proposed by Yan *et al.* [97], who exploited the fact that typically people tend to perform the same actions in the same environment (e.g., people at work typically have a coffee break), and their results show the advantage of sharing information between tasks. Kanade and Hebert [51] explored the problem of activity recognition from a deeper perspective. They proposed several methods for activity recognition, some based on object and scene understanding, which are specifically adapted to their eye-glass-mounted wearable device.

Remarks: In essence, the most common cues on which activity recognition in egocentric videos relies on are body movement, object–hand interaction, and patterns of attention. Body-movement-based methods rely on motion estimation and, therefore, are not directly applicable to data acquired by photographic cameras. Object–hand interaction and patterns of attention are feasible for data acquired by wearable cameras attached to the head or somewhere near the person's eyes that could follow his/her gaze. However, when the camera is worn as a necklace or attached to the clothes, attention-based methods fail, making it impossible to see what the user is manipulating and making it very difficult to estimate the centre of attention. Similarly, object–hand methods can be very difficult to apply considering the free motion of the camera and the difficulty in regularly showing the hands of the user. To the best of our knowledge, there is no published work on recognition of egocentric activities recorded by freely worn cameras. In this context, it would be a requirement for robust activity recognition to take into account information concerning whether the camera wearer is stationary or moving.

IV. Availability of Datasets and Software

A. Egocentric Vision Datasets

As egocentric vision is a relatively new research field, the creation of standardized and rich enough datasets and annotations to test and compare the new algorithms is crucial to boost the development of the field. In Table III, we provide a summary of currently available public egocentric datasets, specifying, for each of them, the following information: the name and the reference paper where the datasets were presented or were used for the first time (where data can be found); a short description; the kind of annotated data they contain; and the camera used to acquire the data.

Only two of the publicly available egocentric datasets, EDUB [16] and AIHS [50], use photographic cameras and, thus, are useful to test and compare algorithms for visual lifelogging. Most of them are acquired using video (HTR cameras), making the analysis of long periods of time difficult. Although nearly all of them show scenes of daily living and some of them record many continuous hours of video [64], [71], [73], there is a strong need to create rich datasets with detailed annotations to ensure the robustness, applicability, and usability of the algorithms for visual storytelling construction.

Following, we enumerate the available datasets (referenced by their main citation) for each of the relevant tasks applicable for analyzing the main building blocks of lifelogging data:

1) social interaction analysis: [6], [35], [70];
2) object recognition/detection/discovery: [10], [16], [20], [26], [37], [71], [75];
3) gaze prediction: [36];

4) hand detection/segmentation: [7], [8], [11], [37], [59], [71];
5) gesture recognition: [8], [20], [22];
6) activity recognition: [10], [35], [36], [48], [56], [71], [73], [79], [84], [86];
7) novelty or informative region detection: [4], [64].

This analysis reveals the lack of well-established and widely accepted datasets.

B. Egocentric Vision Software

The publication of the source code is crucial to guarantee the reproducibility of research results and to allow quantitative comparisons on different datasets. To divulge available egocentric vision-related software, we present a list of the most relevant repositories, including source code for object recognition, object discovery, activity recognition, event segmentation, keyframe-based summarization, and informative image detection in Table IV.

V. Conclusion and Future Directions

This review summarized the state of the art of visual lifelogging analysis from a storytelling perspective, focusing on the progresses made so far in this context in the field of computer vision. In the first part of this survey, we reviewed several techniques for acquiring, organizing, summarizing, and browsing large collections of unstructured data. In the second part, we organized the available literature around the central questions necessary to address the storytelling problem: Was the user interacting with somebody? How?, Where is he/she?, When did the event occur? and What is the person wearing the camera doing?. For each research question, we highlighted the weaknesses and strengths of available methods with respect to their applicability to the LTR domain. Additionally, we reviewed all the available datasets and source code.

Generally, from this review, we can draw some conclusions regarding the crucial points that must be followed in short-term research into egocentric vision. First, there is a need to develop more algorithms suited to data acquired through photo cameras, in particular for social interaction detection and analysis, as well as for activity and context recognition. Second, in view of the large number of datasets made publicly available in the last few years, it would be useful to foster cooperation within the lifelogging scientific community to elaborate richer lifelogging datasets. By doing this, researchers could validate their algorithms and promote competition. Third, considering that visual storytelling has to preserve semantics, a promising direction is to continue leveraging semantic information for both egocentric data analysis and summarization. Given the wide variety of settings in which lifelogging cameras are being deployed, visual recognition could largely benefit from the use of ontologies. Moreover, this paper showed that the interest in analysis from the computer vision community over the last few years has increased considerably. In parallel, we witnessed a burst in the study and applicability of CNNs, suggesting that expectations for making progress in the coming years are growing fast. This progress should be accompanied by the creation of larger and more consolidated datasets that will compensate the enormous data demand of CNNs. In particular, research efforts should focus on the problems of 1) developing more sophisticated transfer learning strategies able to reduce the need of large annotated datasets and 2) exploiting temporal coherence of concepts that characterize visual lifelogs. However, given the current limitations of CNNs in terms of computational cost and resources, the analysis would be limited to postprocessing. Finally, a promising area of research that has not been explored for storytelling via ego-vision yet, is text description generation from images. This problem, tackled for instance in [92] and [99], consists of rendering a visual to text translation of what is happening in the images. The development of these new kinds of multimodal techniques could open up a new area, full of potential for egocentric storytelling, in which we could provide a human-like description of what happened in a precise scene or event. The application of these algorithms to the medical field, and more precisely to people with dementia, could help provide patients with a richer context to understand better what happened to them in a given situation.

References

[1] M. Aghaei, M. Dimiccoli, and P. Radeva, "Multi-face tracking by extended bag-of-tracklets in egocentric photo-streams," *Comput. Vision Image Understanding*, vol. 149, pp. 146–156, 2015.
[2] M. Aghaei, M. Dimiccoli, and P. Radeva, "Towards social interaction detection in egocentric photo streams," in *Proc. Int. Conf. Mach. Vision*, 2015, Art. no. 987514.
[3] M. Aghaei and P. Radeva, "Bag-of-tracklets for person tracking in lifelogging data," in *Artificial Intelligence Research and Development: Recent Advances and Applications*, vol. 269. Amsterdam, The Netherlands: IOS Press, 2014.
[4] O. Aghazadeh, J. Sullivan, and S. Carlsson, "Novelty detection from an ego-centric perspective," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2011, pp. 3297–3304.
[5] A. Alletto, G. Serra, S. Calderara, and R. Cucchiara, "Head pose estimation in first-person camera views," in *Proc. 22nd Int. Conf. Pattern Recognit.*, 2014, pp. 4188–4193.
[6] S. Alletto, G. Serra, S. Calderara, F. Solera, and R. Cucchiara, "From ego to Nos-vision: Detecting social relationships in first-person views," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 594–599.
[7] S. Bambach, S. Lee, D. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 1949–1957.
[8] L. Baraldi, F. Paci, G. Serra, L. Benini, and R. Cucchiara, "Gesture recognition in ego-centric videos using dense trajectories and hand segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 702–707.
[9] L. Bazzani *et al.*, "Social interactions by visual focus of attention in a three-dimensional environment," *Expert Syst.*, vol. 30, no. 2, pp. 115–127, 2013.
[10] A. Behera, D. C. Hogg, and A. G. Cohn, "Egocentric activity monitoring and recovery," in *Proc. 11th Asian Conf. Comput. Vision*, 2013, pp. 519–532.
[11] A. Betancourt, P. Morerio, E. I Barakova, L. Marcenaro, M. Rauterberg, and C. S. Regazzoni, "A dynamic approach and a new dataset for hand-detection in first person vision," in *Computer Analysis of Images and Patterns*. Berlin, Germany: Springer, 2015, pp. 274–287.
[12] A. Betancourt, P. Morerio, C. S. Regazzoni, and M. Rauterberg, "The evolution of first person vision methods: A survey," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 5, pp. 744–760, May 2015.
[13] V. Bettadapura, I. Essa, and C. Pantofaru, "Egocentric field-of-view localization using first-person point-of-view devices," in *Proc. IEEE Winter Conf. Appl. Comput. Vision*, 2015, pp. 626–633.
[14] V. Bettadapura, E. Thomaz, A. Parnami, G. D. Abowd, and I. Essa, "Leveraging context to support automated food recognition in restaurants," in *Proc. IEEE Winter Conf. Appl. Comput. Vision*, 2015, pp. 580–587.

[15] M. Bolaños, R. Mestre, E. Talavera, X. Giró-i Nieto, and P. Radeva, "Visual summary of egocentric photostreams by representative keyframes," in *Proc. IEEE Int. Conf. Multimedia Expo. Workshops*, 2015, pp. 1–6.

[16] M. Bolaños and P. Radeva, "Ego-object discovery," *arXiv preprint arXiv:1504.01639*, 2015.

[17] M. Bolaños, M. Garolera, and P. Radeva, "Active labeling application applied to food-related object recognition," in *Proc. ACM Int. Workshop Multimedia Cooking Eating Activities*, 2013, pp. 45–50.

[18] M. Bolaños, M. Garolera, and P. Radeva, "Video segmentation of lifelogging videos," in *Articulated Motion and Deformable Objects*. Berlin, Germany: Springer, 2014, pp. 1–9.

[19] M. Bolaños, M. Garolera, and P. Radeva, "Object discovery using CNN features in egocentric videos," in *Pattern Recognition and Image Analysis*. Berlin, Germany: Springer, 2015, pp. 67–74.

[20] I. M. Bullock, T. Feix, and A. M. Dollar, "The yale human grasping dataset: Grasp, object, and task data in household and machine shop environments," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 251–255, 2015.

[21] D. Byrne, A. R. Doherty, C. G. M. Snoek, G. J. F. Jones, and A. F. Smeaton, "Everyday concept detection in visual lifelogs: Validation, relationships and trends," *Multimedia Tools Appl.*, vol. 49, no. 1, pp. 119–144, 2010.

[22] M. Cai, K. M. Kitani, and Y. Sato, "A scalable approach for understanding the visual structures of hand grasps," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 1360–1366.

[23] D. Castro *et al.*, "Predicting daily activities from egocentric images using deep learning," in *Proc. ACM Int. Symp. Wearable Comput.*, 2015, pp. 75–82.

[24] V. Chandrasekhar, C. Tan, W. Min, L. Liyuan, L. Xiaoli, and L. J. Hwee, "Incremental graph clustering for efficient retrieval from streaming egocentric video data," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2014, pp. 2631–2636.

[25] S. Chowdhury, P. J. McParlane, S. Ferdous, and J. Jose, "My day in review: Visually summarising noisy lifelog data," in *Proc. ACM Int. Conf. Multimedia Inf. Retrieval*, 2015, pp. 607–610.

[26] D. Damen, O. Haines, T. Leelasawassk, A. Calway, and W. Mayol-Cuevas, "Multi-user egocentric online system for unsupervised assistance on object usage," in *Proc. Eur. Conf. Comput. Vision Workshops*, 2014, pp. 481–492.

[27] M. Dimiccoli and P. Radeva, "Visual lifelogging in the era of outstanding digitization," *Digit. Presentation Preservation Cultural Sci. Heritage*, vol. V, pp. 59–64, 2015.

[28] A. R. Doherty, C. Ó Conaire, M. Blighe, A. F. Smeaton, and N. E. O'Connor, "Combining image descriptors to effectively retrieve events from visual lifelogs," in *Proc. ACM Int. Conf. Multimedia Inf. Retrieval*, 2008, pp. 10–17.

[29] A. R. Doherty *et al.*, "Experiences of aiding autobiographical memory using the SenseCam," *Human–Comput. Interact.*, vol. 27, nos. 1/2, pp. 151–174, 2012.

[30] A. R. Doherty and A. F. Smeaton, "Automatically segmenting lifelog data into events," in *Proc. Int. Workshop Image Audio Anal. Multimedia Interactive Serv.*, 2008, pp. 20–23.

[31] A. R. Doherty and A. F. Smeaton, "Combining face detection and novelty to identify important events in a visual lifelog," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. Workshops*, 2008, pp. 348–353.

[32] A. R. Doherty *et al.*, "Wearable cameras in health: The state of the art and future possibilities," *Amer. J. Preventive Med.*, vol. 44, no. 3, pp. 320–323, 2013.

[33] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[34] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *IEEE Int. Conf. Proc. Comput. Vision*, 2011, pp. 407–414.

[35] A. Fathi, J. K. Hodgins, and J. M. Rehg, "Social interactions: A first-person perspective," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 1226–1233.

[36] A. Fathi, Y. Li, and J. M. Rehg, "Learning to recognize daily actions using gaze," in *Proc. Eur. Conf. Comput. Vision*, 2012, pp. 314–327.

[37] A. Fathi, X. Ren, and J. M. Rehg, "Learning to recognize objects in egocentric activities," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2011, pp. 3281–3288.

[38] T. Feix, R. Pawlik, H.-B. Schmiedmayer, J. Romero, and D. Kragic, "A comprehensive grasp taxonomy," in *Proc. Robot. Sci. Syst., Workshop Understanding Human Hand Adv. Robot. Manipulation*, 2009, pp. 2–3.

[39] A. Furlan, S. Miller, D. G Sorrenti, L. Fei-Fei, and S. Savarese, "Free your camera: 3d indoor scene understanding from arbitrary camera motion," in *Proc. Brit. Mach. Vision Conf.*, 2013, pp. 24.1–24.12.

[40] J. Ghosh, Y. J. Lee, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 1346–1353.

[41] C. Gurrin, A. F. Smeaton, and A. R. Doherty, "Lifelogging: Personal big data," *Found. Trends Inf. Retrieval*, vol. 8, no. 1, pp. 1–125, 2014.

[42] M. Harvey, M. Langheinrich, and G. Ward, "Remembering through lifelogging: A survey of human memory augmentation," *Pervasive Mobile Comput.*, vol. 27, pp. 14–26, 2016.

[43] D. S. Hayden *et al.*, "The accuracy-obtrusiveness tradeoff for wearable vision platforms," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshop Egocentric Vision*, 2012.

[44] S. Hodges *et al.*, "SenseCam: A retrospective memory aid," in *Proc. 8th Int. Conf. Ubiquitous Comput.*, 2006, pp. 177–193.

[45] F. Hopfgartner, Y. Yang, L. M. Zhou, and C. Gurrin, "User interaction templates for the design of lifelogging systems," in *Proc. Semantic Models Adaptive Interactive Syst.*, 2013, pp. 187–204.

[46] H. Hung and B. Kröse, "Detecting f-formations as dominant sets," in *Proc. ACM Int. Conf. Multimodal Interfaces*, 2011, pp. 231–238.

[47] P. Isola, J. Xiao, A. Torralba, and A. Oliva, "What makes an image memorable?" in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2011, pp. 145–152.

[48] Y. Iwashita, A. Takamine, R. Kurazume, and M. S. Ryoo, "First-person animal activity recognition from egocentric videos," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2014, pp. 4310–4315.

[49] A. Jinda-Apiraksa, J. Machajdik, and R. Sablatnig, "A keyframe selection of lifelog image sequences," Erasmus Mundus M.Sc. in Visions and Robotics thesis, Vienna Univ. Technol., Vienna, Austria, 2012.

[50] N. Jojic, A. Perina, and V. Murino, "Structural epitome: A way to summarize ones visual experience," in *Proc. Adv. Neural Inf. Process. Syst. Conf.*, 2010, pp. 1027–1035.

[51] T. Kanade and M. Hebert, "First-person vision," *Proc. IEEE*, vol. 100, no. 8, pp. 2442–2453, Aug. 2012.

[52] H. Kang, M. Hebert, and T. Kanade, "Discovering object instances from scenes of daily living," in *Proc. IEEE Int. Conf. Comput. Vision*, 2011, pp. 762–769.

[53] A. Kendon, *Studies in the Behavior of Social Interaction*, vol. 6. Atlantic Highlands, NJ, USA: Humanities Press Int., 1977.

[54] A. Kendon, "*Conducting Interaction: Patterns of Behavior in Focused Encounters*, vol. 7. Cambridge, U.K.: Cambridge Univ. Press, 1990.

[55] B. Kikhia, A.y Boytsov, J. Hallberg, H. Jonsson, and K. Synnes, "Structuring and presenting lifelogs based on location data," in *Pervasive Computing Paradigms for Mental Health*. Berlin, Germany: Springer, 2014, pp. 133–144.

[56] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, "Fast unsupervised ego-action learning for first-person sports videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2011, pp. 3241–3248.

[57] M. L. Lee and A. K. Dey, "Lifelogging memory appliance for people with episodic memory impairment," in *Proc. 10th Int. Conf. Ubiquitous Comput.*, 2008, pp. 44–53.

[58] S. Lee, S. Bambach, D. J Crandall, J. M. Franchak, and C. Yu, "This hand is my hand: A probabilistic approach to hand disambiguation in egocentric video," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 557–564.

[59] C. Li and K. M. Kitani, "Pixel-level hand detection in ego-centric videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 3570–3577.

[60] N. Li, M. Crane, and H. J. Ruskin, "Automatically detecting "significant events" on SenseCam," *Int. J. Wavelets, Multiresolution Inf. Process.*, vol. 11, no. 6, 2013, Art. no. 1350050.

[61] A. Lidon, M. Bolaños, M. Dimiccoli, P. Radeva, M. Garolera, and X. Giró-i Nieto, "Semantic summarization of egocentric photo stream events," *arXiv preprint arXiv:1511.00438*, 2015.

[62] W.-H. Lin and A. Hauptmann, "Structuring continuous video recordings of everyday life using time-constrained clustering," *Proc. SPIE*, vol. 6073, 2006, Art. no. 60730D.

[63] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 3431–3440.

[64] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 2714–2721.

[65] K. Matsuo, K. Yamada, S. Ueno, and S. Naito, "An attention-based activity recognition for egocentric video," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 565–570.
[66] W. W. Mayol-Cuevas, B. J. Tordoff, and D. W. Murray, "On the choice and placement of wearable vision sensors," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 39, no. 2, pp. 414–425, Mar. 2009.
[67] A. Mehrabian, "Significance of posture and position in the communication of attitude and status relationships," *Psychol. Bull.*, vol. 71, no. 5, pp. 359–372, 1969.
[68] W. Min, X. Li, C. Tan, B. Mandal, L. Li, and J.-H. Lim, "Efficient retrieval from large-scale egocentric visual data using a sparse graph representation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 541–548.
[69] M. Naaman, S. Harada, Q. Wang, H. Garcia-Molina, and A. Paepcke, "Context data in geo-referenced digital photo collections," in *Proc. ACM Int. Conf. Multimedia*, 2004, pp. 196–203.
[70] S. Narayan, M. S. Kankanhalli, and K. R. Ramakrishnan, "Action and interaction recognition in first-person videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 526–532.
[71] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 2847–2854.
[72] F. Poiesi and A. Cavallaro, "Predicting and recognizing human interactions in public spaces," *J. Real-Time Image Process.*, vol. 10, pp. 785–803, 2014.
[73] Y. Poleg, C. Arora, and S. Peleg, "Temporal segmentation of egocentric videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014, pp. 2537–2544.
[74] X. Ren and C. Gu, "Figure-ground segmentation improves handled object recognition in egocentric video," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2010, pp. 3137–3144.
[75] X. Ren and M. Philipose, "Egocentric recognition of handled objects: Benchmark and analysis," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2009, pp. 1–8.
[76] G. Rogez, M. Khademi, J. S. Supančič, J.-M. M. Montiel, and D. Ramanan, "3D hand pose detection in egocentric RGB-D images," in *Proc. Eur. Conf. Comput. Vision Workshops*, 2014, pp. 356–371.
[77] G. Rogez, J. S. Supancic III, and D. Ramanan, "Egocentric pose recognition in four lines of code," *arXiv preprint arXiv:1412.0060*, 2014.
[78] R. J. Rummel, "Social behavior and interaction," in *Understanding Conflict and War—The Conflict.* New York, NY, USA: Wiley, 1976, ch. 9.
[79] M. S. Ryoo and L. Matthies, "First-person activity recognition: What are they doing to me?" in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 2730–2737.
[80] M. S. Ryoo, B. Rothrock, and L. Matthies, "Pooled motion features for first-person videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 896–904.
[81] M. Schinas, S. Papadopoulos, Y. Kompatsiaris, and P. A. Mitkas, "Visual event summarization on social media using topic modelling and graph-based ranking algorithms," in *Proc. 5th ACM Int. Conf. Multimedia Inf. Retrieval*, 2015, pp. 203–210.
[82] F. Setti, C. Russell, C. Bassetti, and M. Cristani, "F-formation detection: Individuating free-standing conversational groups in images," *PloS One*, vol. 10, no. 5, 2015, Art. no. e0123783.
[83] A. F. Smeaton, P. Over, and A. R. Doherty, "Video shot boundary detection: Seven years of TRECVid activity," *Comput. Vision Image Understanding*, vol. 114, no. 4, pp. 411–418, 2010.
[84] S. Song, V. Chandrasekhar, N.-M. Cheung, S. Narayan, L. Li, and J.-H. Lim, "Activity recognition in egocentric life-logging videos," in *Proc. Comput. Vision, ACCV Workshops*, 2014, pp. 445–458.
[85] H. Soo Park and J. Shi, "Social saliency prediction," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 4777–4785.
[86] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. Workshops*, 2009, pp. 17–24.
[87] S. Sundaram and W. W. Mayol-Cuevas, "Egocentric visual event classification with location-based priors," in *Proc. 6th Int. Conf. Adv. Visual Comput.*, 2010, pp. 596–605.
[88] E. Talavera, M. Dimiccoli, M. Bolaños, M. Aghaei, and P. Radeva, "R-clustering for egocentric video segmentation," in *Pattern Recognition and Image Analysis*. Berlin, Germany: Springer, 2015, pp. 327–336.
[89] C. Tan, H. Goh, V. Chandrasekhar, L. Li, and J.-H. Lim, "Understanding the nature of first-person videos: Characterization and classification using low-level features," in *Proc. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 549–556.
[90] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 3, no. 1, 2007, Art. no. 3.
[91] P. Varini and R. Serra, G. and Cucchiara, "Personalized egocentric video summarization for cultural experience," in *Proc. ACM Int. Conf. Multimedia Inf. Retrieval*, 2015, pp. 539–542.
[92] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence-video to text," in *Proc. IEEE Conf. Comput. Vision*, 2015, pp. 4534–4542.
[93] P. Wang and A. F. Smeaton, "Semantics-based selection of everyday concepts in visual lifelogging," *Int. J. Multimedia Inf. Retrieval*, vol. 1, no. 2, pp. 87–101, 2012.
[94] Z. Wang, M. D. Hoffman, P. R. Cook, and K. Li, "Vferret: Content-based similarity search tool for continuous archived video," in *Proc. ACM Workshop Continuous Archival Retrieval Pers. Experiences*, 2006, pp. 19–26.
[95] H. Wannous, V. Dovgalecs, R. Mégret, and M. Daoudi, *Place Recognition Via 3D Modeling for Personal Activity Lifelog Using Wearable Camera*. New York, NY, USA: Springer, 2012.
[96] B. Xiong and K. Grauman, "Detecting snap points in egocentric video with a web photo prior," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 282–298.
[97] Y. Yan, E. Ricci, G. Liu, and N. Sebe, "Recognizing daily activities from first-person videos with multi-task clustering," in *Proc. 12th Asian Conf. Comput. Vision*, 2014, pp. 522–537.
[98] J. Yang, B. Price, S. Cohen, and M.-H. Yang, "Context driven scene parsing with attention to rare classes," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014, pp. 3294–3301.
[99] L. Yao *et al.*, "Describing videos by exploiting temporal structure," *Stat*, vol. 1050, p. 25, 2015.
[100] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Proc. Adv. Neural Inf. Process. Syst. Conf.*, 2014, pp. 487–495.

Marc Bolaños received the B.Sc. degree in computer science from the Universitat de Barcelona (UB), Barcelona, Spain, in 2013, and the M.Sc. degree in artificial intelligence from the Universitat Politècnica de Catalunya, Barcelona, in 2015. He is currently working toward the Ph.D. degree at the UB.

His research interests include deep neural networks and egocentric vision for health improvement.

Mariella Dimiccoli received the degree in computer engineering (*cum laude*) from the Polytechnic University of Bari, Bari, Italy, in 2004, and the Ph.D. degree in image processing (*cum laude* and European Hons.) from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2009.

She is a Beatriu de Pinós Fellow (Marie-Curie COFUND action) with the Computer Vision Center, Barcelona, and an Associate Professor at the Universitat de Barcelona, Barcelona. Her current research interests include automatic analysis and organization of lifelogging data acquired by a wearable camera and their use in health applications.

Petia Radeva received the undergraduate degree from the University of Sofia, Sofia, Bulgaria, in 1989, and the Ph.D. degree in computer vision applied to medical imaging from the Universitat Autònoma de Barcelona, Barcelona, Spain, in 1996.

She is currently a tenured Associate Professor with the Universitat de Barcelona (UB), Barcelona, Spain. She is the Head of the Consolidated Research Group "Computer Vision" at the UB and the Head of MiLab at the Computer Vision Center, Barcelona. Her research interests include the development of learning-based approaches, in particular, in lifelogging applied to health. She has published more than 200 international SCI journal papers and conference contributions.

Dr. Radeva is an Associate Editor of *Pattern Recognition* and the *Journal of Visual Communication and Image Representation*. She received the ICREA Academia 2015 Award from Catalonia given to the best 30 researchers of the year.

Video Segmentation of Life-Logging Videos

Marc Bolaños[1], Maite Garolera[2], and Petia Radeva[1,3]

[1] University of Barcelona, Barcelona, Spain
[2] Hospital de Terrassa-Consorci Sanitari de Terrassa, Terrassa, Spain
[3] Computer Vision Center of Barcelona, Bellaterra (Barcelona), Spain
mark.bs.1991@gmail.com, MGarolera@cst.cat, petia.ivanova@ub.edu

Abstract. Life-logging devices are characterized by easily collecting huge amount of images. One of the challenges of lifelogging is how to organize the big amount of image data acquired in semantically meaningful segments. In this paper, we propose an energy-based approach for motion-based event segmentation of life-logging sequences of low temporal resolution. The segmentation is reached integrating different kind of image features and classifiers into a graph-cut framework to assure consistent sequence treatment. The results show that the proposed method is promising to create summaries of everyday person's life.

Keywords: Life-logging, video segmentation.

1 Introduction

Recently, with the appearance of different LifeLogging (LL) devices (SenseCam [1], Looxcie [2], Narrative (previously called Memoto) [3], Autobiographer), people wearing them are getting eager for capturing details about their daily life. Capturing images along the whole day leads to a huge amount of data that should be organized and summarized in order to be able to store them and review later, being able to focus just on the most important aspects. On the other hand, LL data appear very promising to design new therapies for treating different diseases. LL data have been used to retain and recover memory abilities for patients with Alzheimer's disease [1] as well as to capture and display the healthy habits like nutrition, physical activities, emotions or social interaction. In [4] they are used as an aid for recording the everyday life in order to be able to detect and recognize elements that can measure persons' quality of life and, thus, to improve it [5,6].

LL devices being worn by a person the whole day, have the property to capture images for long periods of time. Depending on where the device is positioned (head-mounted, on glasses, camera with a pin, hung camera, ear-mounted, etc.) determines the field of view and the camera motion (usually, glass camera would be more stable and would give information, where the person is looking at, meanwhile camera hung on the person's neck moves more and lacks information on where the person is looking at). On the other hand, a hung up camera has the advantage that is considered more unobtrusive and thus, causes less repeal

F.J. Perales and J. Santos-Victor (Eds.): AMDO 2014, LNCS 8563, pp. 1–9, 2014.

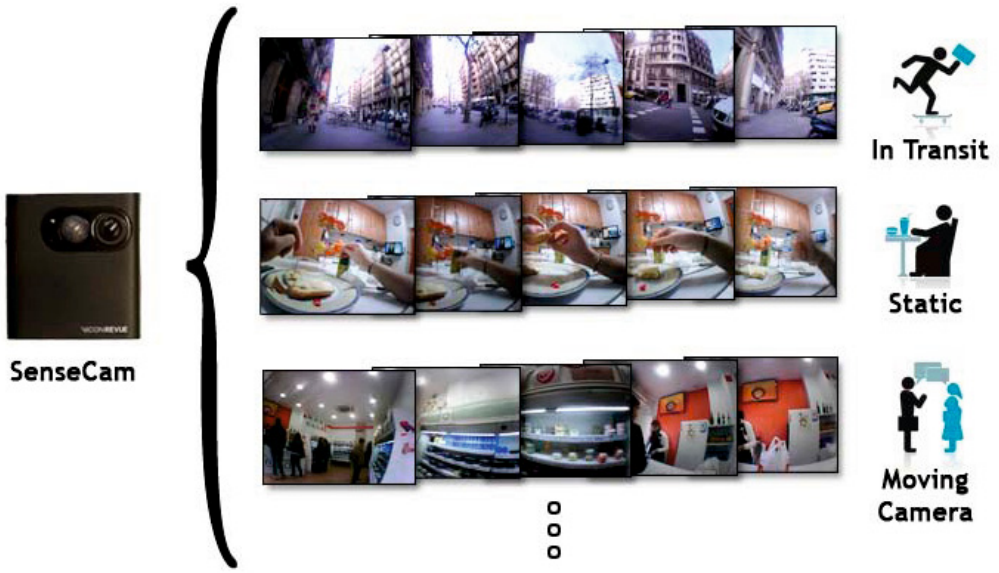

Fig. 1. Illustration of the three person movement-related events to be detected

from the persons around recorded by the camera [7]. Another important characteristic is the temporal resolution of the device. Meanwhile Looxcie has high temporal resolution and, thus, provides smooth continuous videos, many other LL devices like SenseCam has low temporal resolution (2-4 frames per minute) making difficult to consider consecutive frames as videos. Moreover, objects in consecutive images can appear in very different positions. On the other hand, low temporal resolution cameras have the advantages to acquire a reasonable amount of images in order to capture the whole day of the person and allow to process images covering long periods of time (weeks, months). Due to this reason, in this article, we focus on sequence segmentation with a SenseCam that is able to acquire and store images during the whole day activities of the person wearing the camera. Moreover, being hung on the neck, SenseCam is less obtrusive that head-mounted devices, but has low temporal resolution and significant free camera motion. Usually, a day captured by a SenseCam used to contain around 4000 images with no smooth transition between consecutive frames; in a month more than 100.000 images are generated.

Developing tools to reduce redundancy, organize data in events and ease LL review is of high interest. In [8], the authors proposed a method for segmenting and summarizing LL videos based on the detection of "important" objects [9]. Doherty et.al. proposed different methods like selecting specific keyframes [10], combining image descriptors [11] and using a dissimilarity score based on CombMIN [12] to segment and summarize also low-resolution LL data. The work in [13] reviews different techniques for extracting information from egocentric videos, like object recognition, activity detection, sports activities or video summarization.

Event segmentation in LL data is characterized by the action (movement) of the person wearing the device. The relation between scene and event depends

on the person's action that is not always visible in the images; thus, standard event detection techniques in video are not useful. We group consecutive frames in three general event classes according to the human movement (see Figure 1): "Static" (person and camera are maintaining static), "In Transit" (person is moving or running) and "Moving Camera" (person is not changing his/her surroundings, but the camera is moving - e.g. person is interacting with another person, manipulating an object, etc.). Similar event classification has been proposed and addressed by video techniques in [8],where high-temporal resolution LL data are processed. Taking into account that in the case of low-temporal resolution data, video analysis techniques are not usable, we study a novel set of image features and integrate them in an energy-minimization approach for video segmentation[1]. In contrast to [8], we show that an optimal approach is achieved by combining a set of features (bluriness[14], colour, SIFT flow[15], HoG[16]) and classifiers integrated in a Graph Cut (GC) formulation for spatially coherent treatment of LL consecutive frames.

The paper is organized as follows: in section 2, we explain the design and implementation of our event extraction method. In Section 3, we discuss the results obtained and finish the paper with Conclusions.

2 Methodology

To address the event segmentation problem, our approach is based on two main steps: first, we extract motion, color and blurriness information from the images and apply a classifier to obtain a rough approximation of the class labels in single frames (Figure 2). Second, we apply an energy-minimization technique based on GC to achieve spatial coherence of labels assigned by the classifier and separate the sequences of consecutive images in events.

2.1 Feature Extraction of Life-Logging Data

Given that the three classes are basically distinguished by the motion of the camera or the person, as well as the big difference between frames, robust event segmentation needs motion features that do not assume smooth image transition. Hence, we propose to extract the folowing feature types:

SIFT flow data[15,17]: calculated as 8 components, which describe the motion on each cardinal direction scaled by its magnitude.

Blurriness [14]: calculated as 9 components representing the blurrines in each cell dividing the image in 3x3 equal rectangles.

Color difference: color histogram difference between the current image and the five previous ones. With the use of the SIFT flow features between each pair of consecutive images, we expect to find differences between sequences of images with label "Static", which should have a low magnitude and little resilient direction of the descriptors in a significant part of the images. Labels "Moving

[1] Although the low temporal resolution, we still speak about videos of data, refering to the consecutive image collection acquired during a day.

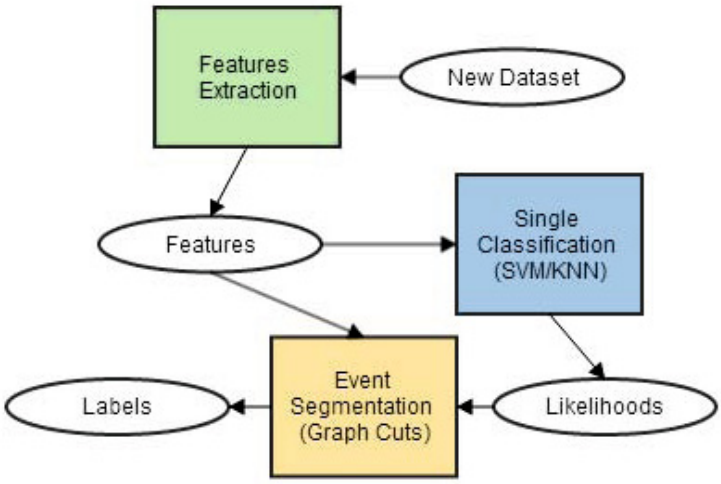

Fig. 2. Diagram of the main steps followed by our method

Camera" and "In Transit" should have a more clear movement. At the same time, the last two classes should be differentiated having vectors of flow with undefined and constantly changing direction (in the "Moving Camera" class) vs. those pointing from the center to the external part of the image due to the movement, when walking for the "In Transit" class. The advantage of SIFT flow is that it is able to find the correspondence of points almost independently of their difference in the image position. About the second descriptor, blurriness, we also expect different behaviour for distinguishing the "Static" from the other labels, which should have a more marked blur effect. Color differences is expected to be informative specially for the "Moving Camera" and "In Transit" classes.

2.2 GC-Based Event Segmentation of LL Data

Events are supposed to be sequences of frames with the same class label. In order to obtain such sequences, we apply a GC-based [18,19] energy-minimization technique to get a reliable event segmentation. GCs are based on the minimization of the energy resulting from the sum of two different terms:

$$E(f) = \sum_{i \in S} U_i(f_i) + W \sum_{\{i,n \in N_i\}} \frac{1}{N_i} P_{i,n}(f_i, f_n)$$

where f_i are the set of features used for the energy minimization, U_i is the unary term, $P_{i,n}$ is the pairwise term, which relates any image i in the sequence with each of its neighbours $n \in N_i$, and W is the weighting term for balancing the trade-off between the unary and the pairwise term. The **unary term,** U_i in our case, is set to $1 - L_H$, being L_H the result from a classifier output that represents the likelihood for each image to belong to one of the three defined classes. The **pairwise term** $P_{i,n}$ is a similarity measure for each sample on each cliqué (all neighbours of a given sample) with respect to the chosen image features that determines the likelihood for each neighbouring pair of images (with a neighbourhood length of 11 in our case) to have the same label. The GC

Fig. 3. Fraction of the total summary of events resulting from a dataset using KNN-based GC segmentation. Each row represents one of the 3 classes, with the total number of images and label belonging to each of them at the right.

algorithm [18,19] using a graph structure finds the optimal cut that minimizes the sum of energies $E(f)$ assigning a class label to each sample as a result of the energy minimization.

Taking into account that the pairwise term should "catch" the features relation between consecutive frames, it uses different features from the classifier ones, namely:

- **Color:** RGB color histograms with 9 bins (3 per color).
- **HoG [16]:** Histogram of oriented gradients with 81 components per image to capture changes in the image structures. The GC algorithm assigns all the consecutive images with the same label to the same class, and thus determines the final event division. Figure 3 illustrates different samples of the extracted events from the three classes. The length of each event is given on the right. For visualization purpose, each event is uniformly subsampled. Note that the "T" events represent image sequences with significant change of the scene (rows 4 and 7). "S" events are representing a static person although the images can differ due to hand manipulation (rows 2, 6 and 9), and "M" events suggest moving person's body (rows 1, 3, 5, 8, and 10).

3 Results

In this section, we review the datasets used in our experiments and the most relevant performed validation tests.

3.1 Datasets Description

Given that there is no public SenseCam dataset with event labels, for the validation we used the dataset from [6] that contains 31749 labeled images from 10 different days taken with a SenseCam. For the purpose of the article, 553 events were manually annotated with 57.41 images per event, on average.

3.2 Parameter Optimization

Regarding the GC unary term, we performed different tests using the output of two of the most popular classifiers in the bibliography: Support Vector Machines (SVM) [20] and K-Nearest Neighbour (KNN). Nevertheless, the method allows to use any classifier that provides a score or likelihood to be used in the graph-cut scheme. In pursuance of obtaining the most generalized result possible, when applying the Radial Basis Function SVM and the KNN, we designed a nested fold cross-validation for obtaining the best regularizing (λ) and deviation (σ) parameters for the first, and the best K value for the second classifier. We used a 10-fold cross validation selecting randomly the balanced training samples. The optimal parameters obtained were: $\lambda = 3$ and $\sigma = 3$, $K = 11$ on KNN with Euclidean distance metric and $K = 21$ on KNN with cosine distance metric.

With these tests, our purpose was to test the weighting GC parameter and to prove the importance of using the GC scheme compared to the frame classification obtained by the SVM/KNN classifiers. Regarding the weight value W, we used a range from 0 to 3.75 in intervals of 0.15 points. We can see in Figure 4 the difference in accuracy between the KNN and the GC for different W values. Note that for $W = 1.75$, the classification of frames improved from 0.72 to 0.86. It resulted that in this case, we obtained 108 events compared to the 56 events in the groundtruth of test set 10. Note that in this case the accuracy is 0.86 representing 15% of improvement regarding the baseline classification result, although the automatic approach tends to oversegment the events.

3.3 GC Performance for Event Segmentation

A summary of the average improvement of using frame classifier (the SVM/KNN) versus integrating it in the GC scheme can be seen in Figure 4. Here, KNNe stands for KNN using Euclidean metrics and KNNc stands for KNN with Cosine metrics. Analysing the results, we can observe that the KNN obtains higher accuracy than the SVM, and that adding the GC "label smoothing" after it, the results are widely improved. The only aspect to take into account, specially, when using the KNN with Euclidean distance is that the performance on all the classes is far from the balanced one (the accuracy of class "S" is much higher than that of the other classes). In this case, a KNN with cosine metrics is a good compromise of overall accuracy as well as accuracy of each class, separately. Regarding the result using SVM, GC has not been able to improve the results of the SVM (on average). However, it still has two advantages: 1) obtaining an average number of events more suitable and realistic with respect to each dataset and

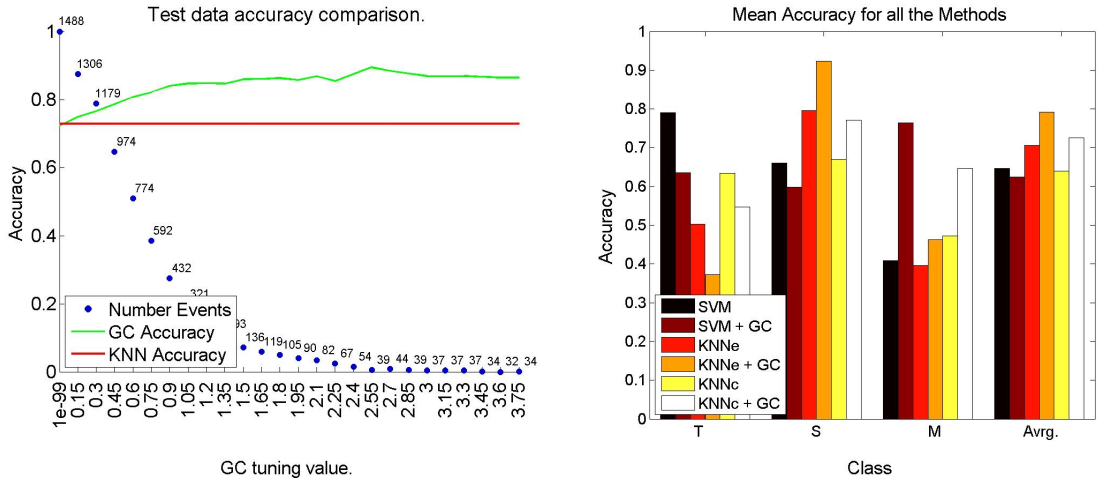

Fig. 4. Improvement in accuracy using different weights for the GC with respect to the KNN with cosine metrics; tests on the 10th dataset (left). Accuracy for each class (T,S,M) and average accuracy for the classifiers (SVM,KNN) and the GC (right).

2) having more similar average of accuracy for each class (without any negative peack of performance like class "M" in case of SVM).

In order to seek redundant image features, we applied a Feature Selection (FS) based on the Student's t-test. We tested the gain obtained by the FS method and the best p-value for it not using the less relevant features neither for the classifier (SVM/KNN) nor for the GC. Comparing the accuracy results, we obtained no statistical difference in performance of the method with and without feature selection. Very similar results were obtained by the Sequential Forward Floating Search method [21].

Once we have applied the GC video segmentation, we have the final sequence divided into events and classified with the respective labels. Events with a very low number of images, would correspond to too short events i.e. with less than 8 images (less than 2 minutes in real time). Since such sequences will not be enough to extract information in the future, neither for obtaining a summary nor for detecting actions of interest of the user, they are deleted.

The limitations of the method are related to the ambiguity between the "T" and "M" labels, due to their motion similarity, that make difficult to classify. Moreover, the "free" motion of the camera is difficult to differentiate (for any of the classifiers used), and this, added to the fact that we use the HOGs without any previous image orientation (that might be a problem when the camera is rotated), are some aspects that might be improved in future work.

4 Conclusions

In this work, we proposed a new method for motion-based segmentation of sequences produced by LL devices with low temporal resolution. The most remarkable results are represented by integrating a wide set of image features and

a KNN classifier with cosine metrics into the GC energy-minimization. The proposed algorithm achieved the most balanced accuracy for the 3 different classes.

Our method proposes tools to detect motion-related events that can be used for higher-level semantic analysis of LL data. The method could ease the recognition of person's action and the elements involved (objects around, manipulated objects, persons). The events can be used as a base to create information "capsules" for memory enhancement of Alzheimer patients. Moreover, the method can relate the "In Transit" label to exercising action of the person, or the abundance and length of "Static" events evidencing sedentary habits [22,23]. Following works on high-temporal resolution LL data [9], important people and objects can be detected and related to the most useful and summarized stories found in the LL events [24]. Our next steps are directed towards LL summarization and detection of interesting events, people and objects in low-resolution temporal LL for either improving the memory of the user or visualizing summarized lifestyle data to ease the management of the user's healthy habits (sedentary lifestyles [22], nutritional activity of obese people, etc.).

Acknowledgments. This work was partially founded by the projects TIN2012-38187-C03-01, Fundació "Jaume Casademont" - Girona and SGR 1219.

References

1. Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K.: Sensecam: A retrospective memory aid. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 177–193. Springer, Heidelberg (2006)
2. Eisenberg, A.: When a Camcorder becomes a life partner, vol. 6. New York Times (2010)
3. Bowers, D.: Lifelogging: Both advancing and hindering personal information management (2013)
4. Sellen, A.J., Whittaker, S.: Beyond total capture: a constructive critique of lifelogging. Communications of the ACM 53(5), 70–77 (2010)
5. Hoashi, H., Joutou, T., Yanai, K.: Image recognition of 85 food categories by feature fusion. In: 2010 IEEE International Symposium on Multimedia (ISM), pp. 296–301. IEEE (2010)
6. Bolaños, M., Garolera, M., Radeva, P.: Active labeling application applied to food-related object recognition. In: Proceedings of the 5th International Workshop on Multimedia for Cooking & Eating Activities, pp. 45–50. ACM (2013)
7. Vondrick, C., Hayden, D.S., Landa, Y., Jia, S.X., Torralba, A., Miller, R.C., Teller, S.: The accuracy-obtrusiveness tradeoff for wearable vision platforms. In: Second IEEE Workshop on Egocentric Vision, CVPR (2012)
8. Lu, Z., Grauman, K.: Story-driven summarization for egocentric video. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2714–2721. IEEE (2013)
9. Lee, Y.J., Ghosh, J., Grauman, K.: Discovering important people and objects for egocentric video summarization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012), pp. 1346–1353. IEEE (2012)

10. Doherty, A.R., Byrne, D., Smeaton, A.F., Jones, G.J.F., Hughes, M.: Investigating keyframe selection methods in the novel domain of passively captured visual lifelogs. In: Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval, pp. 259–268. ACM (2008)
11. Doherty, A.R., Ó Conaire, C., Blighe, M., Smeaton, A.F., O'Connor, N.E.: Combining image descriptors to effectively retrieve events from visual lifelogs. In: Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, pp. 10–17. ACM (2008)
12. Doherty, A.R., Smeaton, A.F.: Automatically segmenting lifelog data into events. In: Image Analysis for Multimedia Interactive Services, WIAMIS 2008, pp. 20–23. IEEE (2008)
13. Bambach, S.: A survey on recent advances of computer vision algorithms for egocentric video (2013)
14. Crete, F., Dolmiere, T., Ladret, P., Nicolas, M.: The blur effect: Perception and estimation with a new no-reference perceptual blur metric. Human Vision and Electronic Imaging XII 6492, 64920 (2007)
15. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.T.: SIFT flow: Dense correspondence across different scenes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 28–42. Springer, Heidelberg (2008)
16. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 1, pp. 886–893 (2005)
17. Liu, C.: Beyond pixels: exploring new representations and applications for motion analysis, Ph.D. thesis, Massachusetts Institute of Technology (2009)
18. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(11), 1222–1239 (2001)
19. Delong, A., Osokin, A., Isack, H.N., Boykov, Y.: Fast approximate energy minimization with label costs. International Journal of Computer Vision 96(1), 1–27 (2012)
20. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20(3), 273–297 (1995)
21. Pudil, P., Novovičová, J., Kittler, J.: Floating search methods in feature selection. Pattern Recognition Letters 15(11), 1119–1125 (1994)
22. Kelly, P., Doherty, A., Berry, E., Hodges, S., Batterham, A.M., Foster, C.: Can we use digital life-log images to investigate active and sedentary travel behaviour? results from a pilot study. International Journal on Behavioral Nutrition and Physical Activities 8(44), 44 (2011)
23. Kerr, J., Marshall, S.J., Godbole, S., Chen, J., Legge, A., Doherty, A.R., Kelly, P., Oliver, M., Badland, H.M., Foster, C.: Using the sensecam to improve classifications of sedentary behavior in free-living settings. American Journal of Preventive Medicine 44(3), 290–296 (2013)
24. Shahaf, D., Guestrin, C.: Connecting the dots between news articles. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 623–632. ACM (2010)

R-Clustering for Egocentric Video Segmentation

Estefania Talavera[1,2( )], Mariella Dimiccoli[1,3], Marc Bolaños[1], Maedeh Aghaei[1], and Petia Radeva[1,3]

[1] Universitat de Barcelona, Barcelona, Spain
{etalavera,marc.bolanos,maghaeigavari,petia.ivanova}@ub.edu
[2] University of Groningen, Groningen, The Netherlands
[3] Computer Vision Center, Barcelona, Bellaterra, Spain
mariella.dimiccoli@cvc.uab.es

Abstract. In this paper, we present a new method for egocentric video temporal segmentation based on integrating a statistical mean change detector and agglomerative clustering(AC) within an energy-minimization framework. Given the tendency of most AC methods to oversegment video sequences when clustering their frames, we combine the clustering with a concept drift detection technique (ADWIN) that has rigorous guarantee of performances. ADWIN serves as a statistical upper bound for the clustering-based video segmentation. We integrate both techniques in an energy-minimization framework that serves to disambiguate the decision of both techniques and to complete the segmentation taking into account the temporal continuity of video frames descriptors. We present experiments over egocentric sets of more than 13.000 images acquired with different wearable cameras, showing that our method outperforms state-of-the-art clustering methods.

Keywords: Temporal video segmentation · Egocentric videos · Clustering

1 Introduction

Lifestyle behaviour is closely related with health outcomes, in particular, to non-communicable diseases such as obesity and depression, that represent a major burden in developed countries. A promising way towards studying one's lifestyle is through the use of wearable cameras, able to digitally capture a person's everyday activities into the so called lifelogging. However, the automatic recognition of daily routines using wearable devices is very challenging due to the huge amount of collected data (up to 3.000 images per day). Moreover, daily routines are typically composed of many complex events, with a large variability depending on factors such as time, location and individual. This work proposes an algorithm for grouping similar temporally adjacent images into segments, providing a structure to egocentric videos, that is important for further analysis such as video summarization and analysis. Considering the environment as a strong characteristic of an event, these segments are supposed to characterize different environments in which the camera wearer acts (Fig. 1).

R. Paredes et al. (Eds.): IbPRIA 2015, LNCS 9117, pp. 327–336, 2015.
DOI: 10.1007/978-3-319-19390-8_37

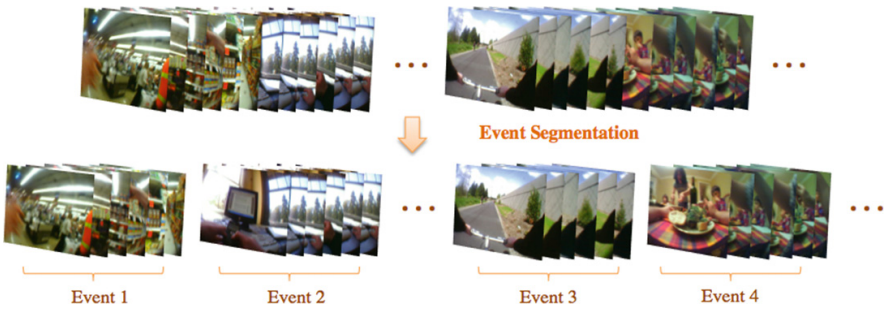

Fig. 1. Example of temporal segmentation of a SenseCam sequence.

Previous methods for egocentric temporal segmentation can be classified into two broad classes, depending on whether they rely on low-level or high-level features. Basically, the former class has as focus what the *wearer sees* and uses as image representation features that are able to capture the characteristics of the environment around the wearer, such as color and texture; the latter class focuses on what the *wearer does* and thus uses as image representation high-level concepts such as objects and activities.

Early works based on low-level features include the one of Doherty et al. [5], which is based on the use of MPEG-7 descriptors for image representation that are available from the sensor, and the one of [16], that uses a time-constrained k-means algorithm based on color descriptors. Recent methods focus on motion-based features. Usually, optical flow is used to distinguish between static, moving the head/camera and *in-transit* frames [3,17]. This classification offers a segmentation that focuses in the activities and movements performed by the user, but is prone to fail when the environment changes while performing the same activity (e.g. the user is in transit but, first gets out of their workplace, then is walking on the street and finally enters to the underground). To focus on long-term activities, Poleg et al. [21] proposed the use of integral motion, which is closely related to wearers' activity. By integrating the instantaneous displacements at fixed image patches, the variations due to head rotation are eliminated, since their mean is practically zero, leaving only the consistent displacement caused by forward motion. Methods based on motion analysis assume high temporal resolution, but the temporal resolution of many lifelogging devices, such those considered in this paper, is very low.

Works based on high-level features are generally more recent. In [14], first important people and objects are discovered by measuring their interaction with the camera wearer and then the frames which reflect the key objects happening are selected. In [20], the authors propose a summarization tool based on analysis of video structures and video highlights. By emphasizing on both the content balance and the perceptual quality of the summary, the authors employ a normalized cut algorithm to globally and optimally partition a video into clusters. Furthermore, in [13], the authors present a video segmentation approach based on the study of spatio-temporal activities within the video, that leads to a visual

activity estimation by measuring the number of interest points, jointly obtained in the spatial and temporal domains.

In this paper, we rely on low-level features. Our approach is a *Graph-Cut (GC) extension* technique [3,4] that takes advantage of two methods having complementary properties: ADWIN [2] - a concept drift technique for mean change detection that is highly precise, but usually leads to temporal under-segmentation; and agglomerative clustering (AC), which usually has a high recall, but leads to temporal over-segmentation. Our approach, that we call **R-Clustering**, *regularizes* the over-segmentation of the AC through the upper bound provided by ADWIN. Based on the excellent accuracy achieved recently for classification in a variety of computer vision tasks [8,12], we use Convolutional Neural Network (CNN) vector activation over the entire image as a global image feature descriptor. CNN features are able to focus just in the environment appearance and do not need to rely on a motion information that, would be unfeasible to estimate reliably taking into account the very low temporal resolution of the wearable devices we considered (up to 3fpm). As an example of application, we illustrate the utility of the proposed method for the detection of social events. In the next section, we detail the proposed approach. In Sect. 3, we discuss experimental results and, finally, in Sect. 4 we draw some conclusions.

2 The R-Clustering Approach for Temporal Video Segmentation

Due to the low-temporal resolution of egocentric videos, as well as to the camera wearer's motion, temporally adjacent egocentric images may be very dissimilar between them. Hence, we need robust techniques to group them and extract meaningful video segments. In the following, we detail each step of our approach that relies on an AC regularized by a robust change detector within a GC framework.

Clustering Methods. The AC method follows a general bottom-up clustering procedure, where the criterion for choosing the pair of clusters to be merged in each step is based on the distances among the image features. The inconsistency between clusters is defined through the *cut* parameter. In each iteration, the most similar pair of clusters are merged and the similarity matrix is updated until no more consistent clustering are possible. We chose the Cosine Similarity to measure the distance between frames features, since it is a widely used measure of cohesion within clusters, specially in high-dimensional positive spaces [23]. However, due to the lack of incidence for determining the clustering parameters, the final result is usually over-segmented.

Statistical Bound for the Clustering. To bound the over-segmentation produced by AC, we propose to model the video as a multivariate data stream and detect changes in the mean distribution through an online learning method

called Adaptative Windowing (**ADWIN**) [2]. ADWIN works by analyzing the content of a sliding window, whose size is adaptively recomputed according to its rate of change: when the data is stationary the window increases, whereas when the data is statistically changing, the window shrinks. According to ADWIN, whenever two large enough temporally adjacent (sub)windows of the data, say W_1 and W_2, exhibit distinct enough means, the algorithm concludes that the expected values within those windows are different, and the older (sub)window is dropped. *Large enough* and *distinct enough* are defined by the Hoeffding's inequality [9], testing if the difference between the averages on W_1 and W_2 is larger than a threshold, which only depends on a pre-determined confidence parameter δ. The Hoeffding's inequality guarantees rigorously the performance of the algorithm in terms of false positive rate.

This method has been recently generalized in [6] to handle $k-$dimensional data streams by using the mean of the norms. In this case, the bound has been shown to be:

$$\epsilon_{cut} = k^{1/p}\sqrt{\frac{1}{2m}\ln\frac{4}{k\delta'}}$$

where p indicates the $p-$norm, $|W| = |W0|+|W1|$ is the length of $W = W_1 \cup W_2$, $\delta' = \frac{\delta}{|W|}$, and m is the harmonic mean of $|W0|$ and $|W1|$. Given a confidence value δ, the higher the dimension k is, the more samples $|W|$ the bound needs to reach assuming the same value of ϵ_{cut}. The higher the norm is used, the less important is the dimensionality k. Since we model the video as a high dimensional multivariate data stream, ADWIN is unable to predict changes involving a small number of samples, which often characterizes life-logging data, leading to under-segmentation. Moreover, since it considers only the mean change, it is enable to detect changes due to other statistics such as the variance. The ADWIN under-segmentation represents a statistical bound for the AC (see Fig. 2 (right)). We use GC as a framework to integrate both approaches and to regularize the over-segmentation of AC by the statistical bound provided by ADWIN.

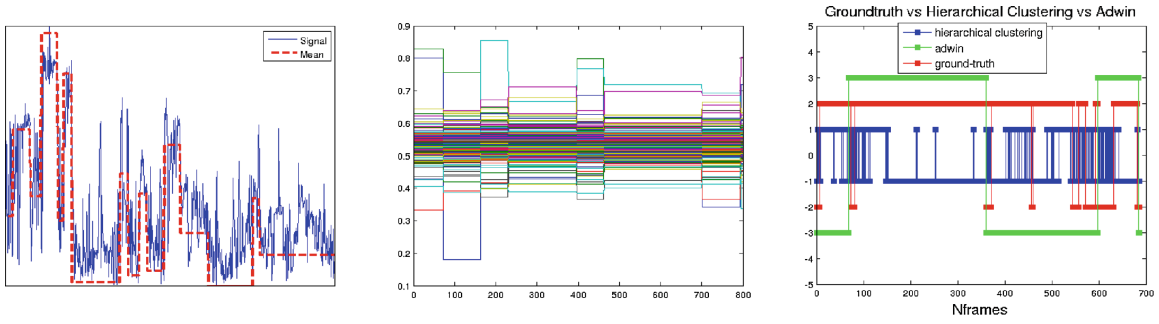

Fig. 2. Left: change detection by ADWIN on a $1-D$ data stream, where the red line represents the estimated mean of the signal by ADWIN; Center: change detection by ADWIN on a 500-D data stream, where, in each stationary interva, the mean is depicted with a different color in each dimension; Right: results of the temporal segmentation by ADWIN (green) vs AC over-segmentation (blue) vs ground-truth shots (red) along the temporal axis (the abscissa)(Color figure online).

Graph-Cut Regularization of Egocentric Videos. GC is an energy-minimization technique that minimizes the energy resulting from a weighted sum of two terms: the *unary energy* $U(\_)$, that describes the relationship of the variables to a possible class and the *binary energy* $V(\_,\_)$, that describes the relationship between two neighbouring samples (temporally close video frames) according to their feature similarity. GC has the goal to smooth boundaries between similar frames, while attempts to keep the cluster membership of each video frame according to its likelihood. We define the unary energy as a sum of 2 parts ($U_{ac}(f_i)$ and $U_{adw}(f_i)$) according to the likelihood of a frame to belong to segments coming from the AC and ADWIN. The GC energy to minimize is as follows:

$$E(f) = \sum_i ((1-\omega_1)U_{ac}(f_i) + \omega_1 U_{adw}(f_i)) + \omega_2 \sum_{i,n\in N_i} \frac{1}{N_i} V_{i,n}(f_i, f_n)$$

where $f_i, i = \{1, ..., m\}$ are the set of image features, N_i are the temporal frame neighbours of image i, ω_1 and ω_2 ($\omega_1, \omega 2 \in [0, 1]$) are the unary and the binary weighting terms respectively. Defining how much weight do we give to the likelihood of each unary term (AC and Adwin, always combining the events split of both methods), and balancing the trade-off between the unary and the pairwise energies, respectively. The minimization is achieved through the max-cut algorithm, leading to a temporal video segmentation with similar frames having as large likelihood as possible to belong to the same event, while maintaining video segment boundaries in neighbouring frames with high feature dissimilarity.

Features. As image representation for both segmentation techniques, we used the CNN features [10]. The CNN features trained on ImageNet [12] have demonstrated to be successfully transferred to other visual recognition tasks such as scene classification and retrieval. In this work, we extracted the 4096-D CNN vectors by using the Caffe [10] implementation trained on ImageNet. Since each CNN feature has a large variation distribution in its value, and this could be problematic when computing distances between vectors, we used a signed root normalization to produce more uniformly distributed data [24]. First, we apply the function $f(x) = sign(x)|x|^\alpha$ on each dimension and then we l_2-normalize the feature vector. In all the experiments, we take $\alpha = 0.5$. Following we apply a PCA dimensionality reduction keeping 95 % of the data variance. Only in the GC pair-wise term we use a different feature pre-processing, where we simply apply a 0–1 data normalization.

3 Results and Validation

In this section, we discuss the datasets, the statistical validation measurements, tests and comparison to other methods as well as a possible application of the R-Clustering.

Data. To evaluate the performance of our method, we used 2 datasets (one public [11] and one made by us), composed of 10 days with a total of 13324 images, acquired by two different wearable devices: SenseCam [22] and Narrative (http://getnarrative.com/). The main differences between the two kind of devices are the frame rate (3 fpm vs 2 fpm) and the lens (fisheye vs normal). The data adquired by the SenseCam contain a larger number of frames per day with a larger field of view and significant deformation and blurring. Both datasets include 5 days each, containing a mix of indoor and outdoor scenes with numerous foreground and background objects. All data has been manually annotated to provide ground-truth segmentation.

Statistical Measurements. As evaluation criterion (following [15]), we used the F-Measure (FM): $FM = 2(RP)/(R+P)$, where P is the precision ($P = TP/(TP+FP)$, R is the recall ($R = TP/(TP+FN)$ and TP, FP and FN are the number of true positives, false positives and false negatives.

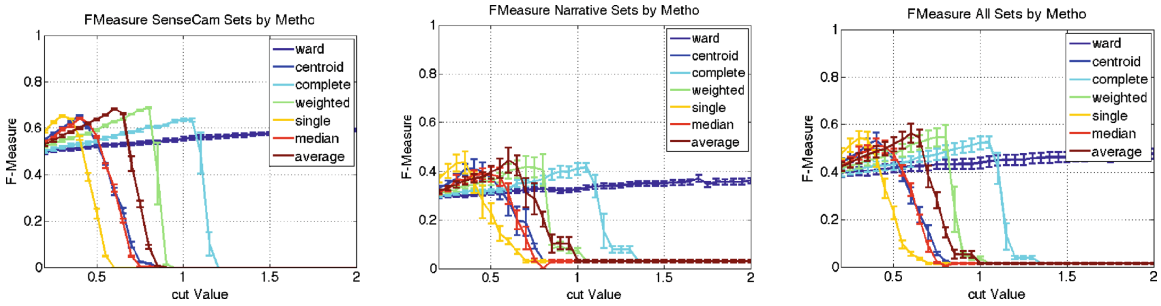

Fig. 3. F-Measure evolution for the two kind of datasets, applying different clustering methods and cut value. The abscissa (X) defines the cut value and the ordinate (Y) - the F-Measure.

Tests on Different Agglomerative Clustering methods. We performed several tests on different AC, namely: single, centroid, average, weighted, complete, ward, and median, that basically vary in the way the distance between cluster elements is estimated [19]. Figure 3 (left) represents the F-Measure of the different clusterings on the SenseCam data, Fig. 3 (center) - on the Narrative data and Fig. 3 (right) on all data. We can observe that the clustering follows the same behaviour for the two types of data sets, although for the SenseCam the methods are achieving better results than for the Narrative sets. That is reasonable due to the significant difference in image appearance. Despite for the SenseCam sets the complete is achieving the same results as the average methods, the third figure shows how for the whole data the average method is achieving the best results (FM=0.56), followed by the complete (FM=0.55) and the single (FM=0.54). The cut value seems to be very influential for the results since there is a point from which, for each method, clusters all data in one single cluster, leading to FM=0.

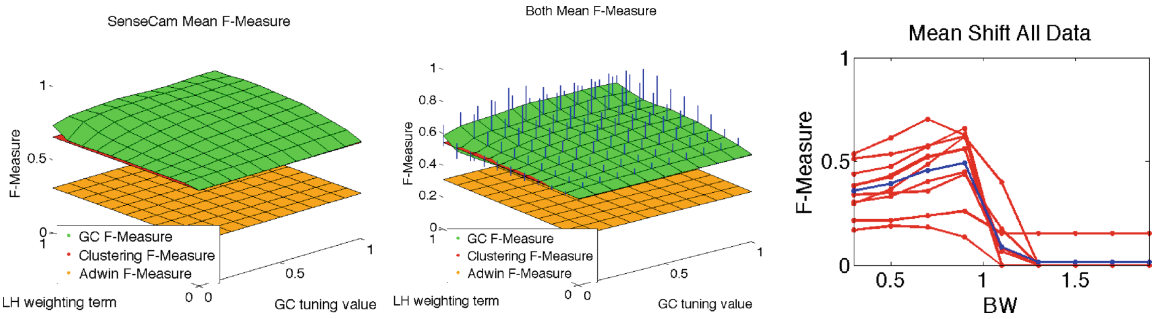

Fig. 4. Average F-Measure of R-Clustering with the best parameters for the SenseCam data (left), and on all datasets (center). The abscissa (X) defines the pair-wise term, the ordinate (Y) the ADWIN vs. AC trade-off and the applicate (Z) shows the corresponding FM. The red surface represents the F-measure of AC and the orange one of ADWIN. MeanShift performance for video segmentation (right). The abscissa (X) defines the bandwith. The blue line represents the average FM, whereas the red lines are the FM per each dataset (Color figure online).

Tests on R-Clustering Using Graph-Cuts. We tested the R-Clustering performances according to the parameters ω_1 and ω_2. Figure 4 (left) shows the average measure on the Sensecam data as a function of both parameters. The optimal F-Measure on all datasets is achieved when $\omega_1 = 1$ and $\omega_2 = 0.5$ (Fig. 4 (center)). Despite the average AC achieves the best performance on our data sets (FM=0.56), the R-Clustering based on this AC method just achieves a FM=0.63. Whereas when it is based on the single clustering, the one that was achieving in AC the second best results (FM=0.54), it achieves the highest FM=0.66 with R-Clustering. Table 1 shows the optimal F-measure for AC, ADWIN and R-Clustering, where the application of R-Clustering method clearly outperforms the F-Measure obtained by the AC and ADWIN technique. Thus, by having $\omega_1 = 1$ as unary energy parameter proves that the combination of ADWIN (by using its resulting likelihoods and labels initialization) and AC (by using its clusters split in the GC labels initialization) helps to obtain better results by the R-Clustering. In Fig. 4 (center), the lines depict the standard deviation on each combination of parameters, hence the standard deviation of the best peak results is very low (std=0.17, short line) compared to the higher deviation (longer lines) in the center, meaning that our method is robust and stable. Final video segments can be seen in Fig. 5 that shows three segments corresponding to metro, office and street environments, extracted from a Narrative set.

Tests on Other Clustering Methods. We compare R-Clustering to K-Means [18] and MeanShift (MS) [7] (see Fig. 4 (right) and Table 1) that achieved FM=0.52 and FM=0.49, respectively. The worse performance can be explained by several facts. The k-Means algorithm requires the number of clusters to be specified and it is not a robust method due to its local minima problem. Considering MeanShift (MS), it is based on density estimation which can deal with arbitrarily shaped data distributions, but its problem is that it is very sensitive

to the bandwith (BW) parameter: a large BW can slow down the convergence and a small BW can make it quickly converge leading to over-segmentation.

Table 1. Average F-Measure result for each of the tested methods on our egocentric datasets.

Datasets	K-Means	Mean-Shift	ADWIN	AC	R-Clustering
Narrative	0.32	0.38	0.32	0.45	**0.55**
SenseCam	0.65	0.60	0.31	0.68	**0.79**
All	0.52	0.49	0.31	0.56	**0.66**

Fig. 5. Illustration of our R-Clustering segmentation results for 3 events from a Narrative set.

Fig. 6. Illustration of detecting social events in temporally segmented videos.

Application to Human Tracking for Social Events characterization. Temporal segmentation is very useful to detect social events, which are characterized by the presence of people with whom the camera's wearer communicates. Since the presence of people in a specific event likely last from the beginning of the event to its end, social events can be extracted by relying on temporal segmentation. As outlined in [1], due to the substantial difference in frame rate between videos captured by a SenseCam and classical videos, state-of-the-art tracking methods are not directly applicable to lifelogging videos. In

[1], the authors introduced a novel approach, called bag-of-tracklets, that allows to extract robust tracklet prototypes from video segments containing trackable people. While in this work temporal segments are defined manually, we use our detected segments as a pre-processing step for extracting tracklets of people in egocentric videos (Fig. 6).

4 Conclusions

In this work, we proposed a novel methodology for automatic egocentric video segmentation that is able to segment low temporal resolution data by global low-level processing. R-Clustering is a robust segmentation approach based on a GC extension technique, that integrates a statistical bound by the concept drift method ADWIN and AC, two methods with complementary properties for temporal video segmentation. We evaluated the performance of R-Clustering on different clustering techniques and on 10 datasets acquired through different wearable devices, and we showed the improvement of the proposed method with respect to the state-of-the-art.

Acknowledgments. This work was partially founded by TIN2012-38187-C03-01 and SGR 1219.

References

1. Aghaei, M., Radeva, P.: Bag-of-tracklets for person tracking in life-logging data. In: CCIA 2014, pp. 35–44, Barcelona, Spain, October 2014
2. Bifet, A., Gavalda, R.: Learning from time-changing data with adaptive windowing. In: SDM, vol. 7. SIAM (2007)
3. Bolaños, M., Garolera, M., Radeva, P.: Video segmentation of life-logging videos. In: Perales, F.J., Santos-Victor, J. (eds.) AMDO 2014. LNCS, vol. 8563, pp. 1–9. Springer, Heidelberg (2014)
4. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)
5. Doherty, A.R., Smeaton, A.F.: Automatically segmenting lifelog data into events. In: Proceedings of WIAMIS 2008, pp. 20–23. IEEE Computer Society, Washington, DC (2008)
6. Drozdzal, M., Vitria, J., Segui, S., Malagelada, C., Azpiroz, F., Radeva, P.: Intestinal event segmentation for endoluminal video analysis. In: ICIP (2014)
7. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inf. Theor. **21**(1), 32–40 (2006)
8. Goodfellow, I.J., Ibarz, J., Bulatov, Y., Arnoud, S., Shet, V.: Multi-digit Number Recognition from Street View Imagery Using Deep Convolutional Neural Networks. Google Inc., Mountain View (2014)
9. Hoeffding, W.: Probability inequalities for sums of bounded random variables. J. Am. Stat. Assoc. **58**(301), 13–30 (1963)
10. Jia, Y. : Caffe: An open source convolutional architecture for fast feature embedding (2013). http://caffe.berkeleyvision.org/

11. Jojic, N., Perina, A., Murino, V.: Structural epitome: a way to summarize one's visual experience. In: NIPS, pp. 1027–1035 (2010)
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C., Bottou, L., Weinberger, K. (eds.) NIPS 25, pp. 1097–1105. Curran Associates Inc., Red Hook (2012)
13. Laganire, R., Bacco, R., Hocevar, A., Lambert, P., Pas, G., Ionescu, B.: Video summarization from spatio-temporal features. In: TVS, pp. 144–148. ACM (2008)
14. Lee, Y.J., Ghosh, J., Grauman, K.: Discovering important people and objects for egocentric video summarization. In: CVPR, pp. 1346–1353. IEEE (2012)
15. Li, Z., Wei, Z., Jia, W., Sun, M.: Daily life event segmentation for lifestyle evaluation based on multi-sensor data recorded by a wearable device. In: EMBC 2013, pp. 2858–2861. IEEE (2013)
16. Lin, W.-H., Hauptmann, A.: Structuring continuous video recording of everyday life using time-constrained clustering. Computer Science Department 959 (2006)
17. Lu, Z., Grauman, K.: Story-driven summarization for egocentric video. In: CVPR, pp. 2714–2721. IEEE (2013)
18. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Le Cam, L.M., Neyman, J. (eds.) Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297 (1967)
19. Murtagh, F., Contreras, P.: Methods of hierarchical clustering. CoRR, abs/1105.0121 (2011)
20. Ngo, C.-W., Ma, Y.-F., Zhang, H..: Automatic video summarization by graph modeling. pages 104–109. IEEE Computer Society (2003)
21. Poleg, Y., Arora, C., Peleg, S.: Temporal segmentation of egocentric videos. In: IEEE Conference On Computer Vision and Pattern Recognition (CVPR) (2014)
22. SenseCam. Sensecam overview (2013)
23. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining, 1st edn. Addison-Wesley Longman Publishing Co., Boston (2005)
24. Zheng, L., Wang, S., He, F., Tian, Q.: Seeing the big picture: Deep embedding with contextual evidences. CoRR, abs/1406.0132 (2014)

Computer Vision and Image Understanding 155 (2017) 55–69

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

SR-clustering: Semantic regularized clustering for egocentric photo streams segmentation

Mariella Dimiccoli[a,c,1,*], Marc Bolaños[a,1,*], Estefania Talavera[a,b], Maedeh Aghaei[a], Stavri G. Nikolov[d], Petia Radeva[a,c,*]

[a] Universitat de Barcelona, Barcelona, Spain
[b] University of Groningen, Groningen, Netherlands
[c] Computer Vision Center, Barcelona, Bellaterra, Spain
[d] Imagga Technologies Ltd and Digital Spaces Living Lab, Sofia, Bulgaria

ARTICLE INFO

Article history:
Received 14 January 2016
Revised 11 October 2016
Accepted 16 October 2016
Available online 19 October 2016

Keywords:
Temporal segmentation
Egocentric vision
Photo streams clustering

ABSTRACT

While wearable cameras are becoming increasingly popular, locating relevant information in large unstructured collections of egocentric images is still a tedious and time consuming process. This paper addresses the problem of organizing egocentric photo streams acquired by a wearable camera into semantically meaningful segments, hence making an important step towards the goal of automatically annotating these photos for browsing and retrieval. In the proposed method, first, contextual and semantic information is extracted for each image by employing a Convolutional Neural Networks approach. Later, a vocabulary of concepts is defined in a semantic space by relying on linguistic information. Finally, by exploiting the temporal coherence of concepts in photo streams, images which share contextual and semantic attributes are grouped together. The resulting temporal segmentation is particularly suited for further analysis, ranging from event recognition to semantic indexing and summarization. Experimental results over egocentric set of nearly 31,000 images, show the prominence of the proposed approach over state-of-the-art methods.

1. Introduction

Among the advances in wearable technology during the last few years, wearable cameras specifically have gained more popularity (Bolaños et al., 2016). These small light-weight devices allow to capture high quality images in a hands free fashion from the first-person point of view. Wearable video cameras such as GoPro and Looxcie, by having a relatively high frame rate ranging from 25 to 60 fps, are mostly used for recording the user activities for a few hours. Instead, wearable photo cameras, such as the Narrative Clip and SenseCam, capture only 2 or 3 fpm and are therefore mostly used for image acquisition during longer periods of time (e.g. a whole day). The images collected by continuously recording the user's life, can be used for understanding the user's lifestyle and hence they are potentially beneficial for prevention of non-communicative diseases associated with unhealthy trends and risky profiles (such as obesity, depression, etc.). In addition, these images can be used as an important tool for prevention or hindrance of cognitive and functional decline in elderly people (Doherty et al., 2013). However, egocentric photo streams generally appear in the form of long unstructured sequences of images, often with high degree of redundancy and abrupt appearance changes even in temporally adjacent frames, that harden the extraction of semantically meaningful content. Temporal segmentation, the process of organizing unstructured data into homogeneous chapters, provides a large potential for extracting semantic information. Indeed, once the photo stream has been divided into a set of homogeneous and manageable segments, each segment can be represented by a small number of key-frames and indexed by semantic features, providing a basis for understanding the semantic structure of the event.

State-of-the-art methods for temporal segmentation can be broadly classified into works with focus on what-the-camera-wearer-sees (Castro et al., 2015; Doherty and Smeaton, 2008; Talavera et al., 2015) and on what-the-camera-wearer-does (Poleg et al., 2014; 2015). As an example, from the what-camera-wearer-does perspective, the camera wearer spending time in a bar while

* Corresponding authors.
E-mail addresses: mariella.dimiccoli@cvc.uab.es (M. Dimiccoli), marc.bolanos@ub.edu (M. Bolaños), etalavera@ub.edu (E. Talavera), maghaeigavari@ub.edu (M. Aghaei), stavri.nikolov@imagga.com (S.G. Nikolov), petia.ivanova@ub.edu (P. Radeva).
[1] The first two authors contributed equally to this work.

http://dx.doi.org/10.1016/j.cviu.2016.10.005
1077-3142/

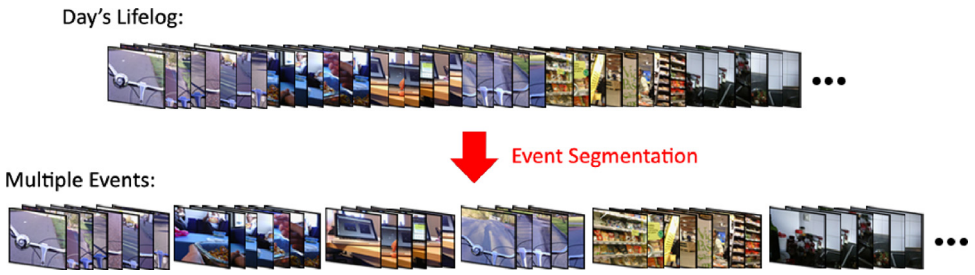

Fig. 1. Example of temporal segmentation of an egocentric sequence based on what the camera wearer sees. In addition to the segmentation, our method provides a set of semantic attributes that characterize each segment.

sit, will be considered as a unique event (sitting). From the what-the-camera-wearer-sees perspective, the same situation will be considered as several separated events (waiting for the food, eating, and drinking beer with a friend who joins later). The distinction between the aforementioned points of view is crucial as it leads to different definitions of an event. In this respect, our proposed method fits in the what-the-camera-wearer-sees category. Early works on egocentric temporal segmentation (Doherty and Smeaton, 2008; Lin and Hauptmann, 2006) focused on what the *camera wearer sees* (e.g. people, objects, foods, etc.). For this purpose, the authors used as image representation, low-level features to capture the basic characteristics of the environment around the user, such as color, texture or information acquired through different camera sensors. More recently, the works in Bolaños et al. (2015) and Talavera et al. (2015) have used Convolutional Neural Network (CNN) features extracted by using the AlexNet model (Krizhevsky et al., 2012) trained on ImageNet as a fixed feature extractor for image representation. Some other recent methods infer from the images what the *camera wearer does* (e.g. sitting, walking, running, etc.). Castro et al. (2015) used CNN features together with metadata and color histogram (Castro et al., 2015).

Most of these methods use as image representation ego-motion (Bolaños et al., 2014; Lu and Grauman, 2013; Poleg et al., 2014; 2015), which is closely related to the user motion-based activity but cannot be reliably estimated in photo streams. The authors combined a CNN trained on egocentric data with a posterior Random Decision Forest in a late-fusion ensemble, obtaining promising results for a single user. However, this approach lack of generalization, since it requires to re-train the model for any new user, implying to manually annotate large amount of images. To the best of our knowledge, except the work of Castro et al. (2015); Doherty and Smeaton (2008) and Talavera et al. (2015), all other state-of-the-art methods have been designed for and tested on videos. In our previous work (Talavera et al., 2015), we proposed an unsupervised method, called *R-Clustering*, aiming to segment photo streams from the what-the-camera-wearer-see perspective. The proposed methods relies on the combination of Agglomerative Clustering (AC), that usually has a high recall, but leads to temporal over-segmentation, with a statistically founded change detector, called ADWIN (Bifet and Gavalda, 2007), which despite its high precision, usually leads to temporal under-segmentation. Both approaches are integrated in a *Graph-Cut (GC)* (Boykov et al., 2001) framework to obtain a trade-off between AC and ADWIN, which have complementary properties. The graph-cut relies on CNN-based features extracted using AlexNet, trained on ImageNet, as a fixed feature extractor in order to detect the segment boundaries.

In this paper, we extend our previous work by adding a semantic level to the image representation. Due to the free motion of the camera and its low frame rate, abrupt changes are visible even among temporally adjacent images (see Figs. 1 and 7). Under these conditions motion and low-level features such as color or image layout are prone to fail for event representation, hence urges the need to incorporate higher-level semantic information. Instead of representing images simply by their contextual CNN features, which capture the basic environment appearance, we detect segments as a set of temporally adjacent images with the same contextual representation in terms of semantic visual concepts. Nonetheless, not all the semantic concepts in an image are equally discriminant for environment classification: objects like trees and buildings can be more discriminant than objects like dogs or mobile phones, since the former characterizes a specific environment such as forest or street, whereas the latter can be found in many different environments. In this paper, we propose a method called Semantic Regularized Clustering (SR-Clustering), which takes into account semantic concepts in the image together with the global image context for event representation.

To the best of your knowledge, this is the first time that semantic concepts are used for image representation in egocentric videos and images. With respect to our previous work published in Talavera et al. (2015), we introduce the following contributions:

- Methodology for egocentric photo streams description based on semantic information.
- Set of evaluation metrics applied to ground truth consistency estimation.
- Evaluation on an extended number of datasets, including our own, which will be published with this work.
- Exhaustive evaluation on a broader number of methods to compare with.

This manuscript is organized as follows: Section 2 provides a description of the proposed photo stream segmentation approach discussing the semantic and contextual features, the clustering and the graph-cut model. Section 3 presents experimental results and, finally, Section 4 summarizes the important outcomes of the proposed method providing some concluding remarks.

2. SR-clustering for temporal photo stream segmentation

A visual overview of the proposed method is given in Fig. 2. The input is a day long photo stream from which contextual and semantic features are extracted. An initial clustering is performed by AC and ADWIN. Later, GC is applied to look for a trade-off between the AC (represented by the bottom colored circles) and ADWIN (represented by the top colored circles) approaches. The binary term of the GC imposes smoothness and similarity of consecutive frames in terms of the CNN image features. The output of the proposed method is the segmented photo stream. In this section, we introduce the semantic and contextual features of SR-clustering and provide a detailed description of the segmentation approach.

2.1. Features

We assume that two consecutive images belong to the same segment if they can be described by similar image features. When we refer to the features of an image, we usually consider low-level

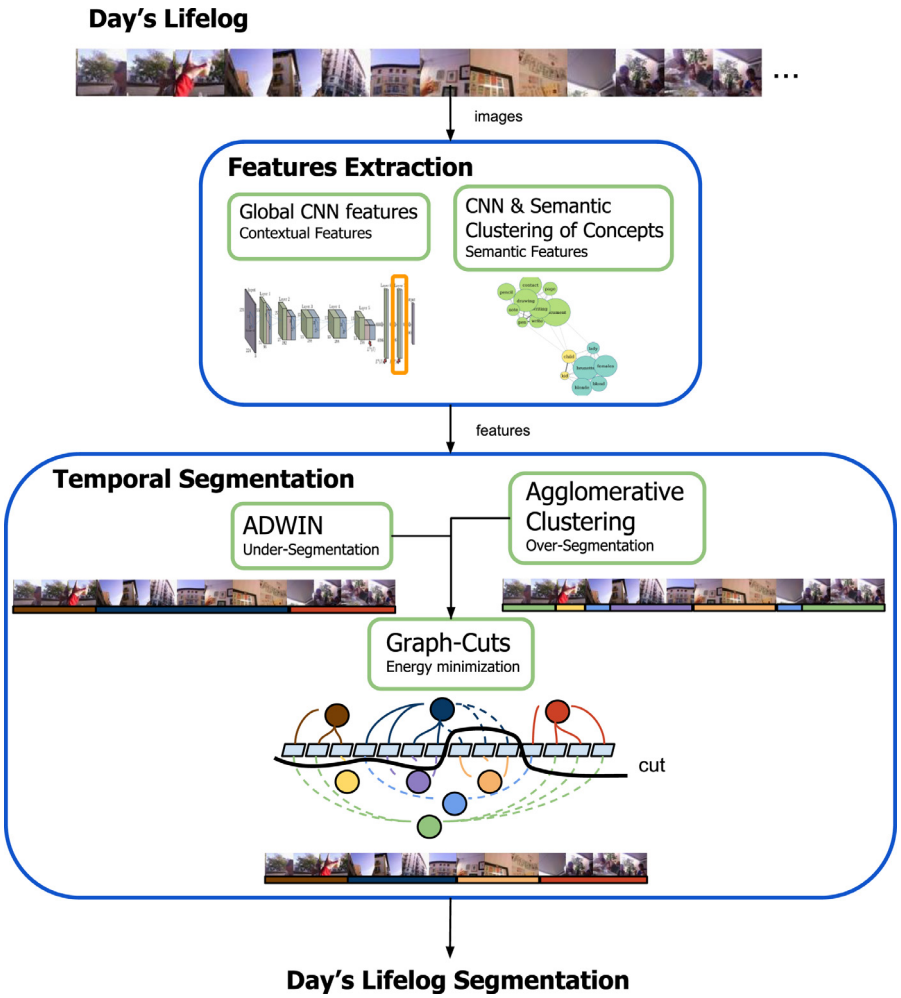

Fig. 2. General scheme of the Semantic Regularized Clustering (SR-Clustering) method.

image features (e.g. color, texture, etc.) or a global representation of the environment (e.g. CNN features). However, the objects or concepts that semantically represent an event are also of high importance for the photo stream segmentation. Below, we detail the features that semantically describe the egocentric images.

2.1.1. Semantic features

Given an image I, let us consider a tagging algorithm that returns a set of objects/tags/concepts detected in the images with their associated confidence value. The confidence values of each concept form a semantic feature vector to be used for the photo streams segmentation. Usually, the number of concepts detected for each sequence of images is large (often, some dozens). Additionally, redundancies in the detected concepts are quite often due to the presence of synonyms or semantically related words. To manage the semantic redundancy, we will rely on WordNet (Miller, 1995), which is a lexical database that groups English words into sets of synonyms, providing additionally short definitions and word relations.

Given a day's lifelog, let us cluster the concepts by relying on their synset ID in WordNet to compute their similarity in meaning, and following, apply clustering (e.g. Spectral clustering) to obtain 100 clusters. As a result, we can semantically describe each image in terms of 100 concepts and their associated confidence scores. Formally, we first construct a semantic similarity graph $\mathcal{G} = \{V, E, W\}$, where each vertex or node $v_i \in V$ is a concept, each edge $e_{ij} \in E$ represents a semantic relationship between two concepts, v_i and v_j and each weight $w_{ij} \in W$ represents the strength of the semantic relationship, e_{ij}. We compute each w_{ij} by relying on the meanings and the associated similarity given by WordNet, between each appearing pair. To do so, we use the max-similarity between all the possible meanings m_i^k and m_j^r in M_i and M_j of the given pair of concepts v_i and v_j:

$$w_{ij} = \max_{m_i^k \in M_i, m_j^r \in M_j} sim(m_i^k, m_j^r).$$

To compute the Semantic Clustering, we use their similarity relationships in the spectral clustering algorithm to obtain 100 semantic concepts, $|C| = 100$. In Fig. 3, a simplified example of the result obtained after the clustering procedure is shown. For instance, in the purple cluster, similar concepts like 'writing', 'document', 'drawing', 'write', etc. are grouped in the same cluster, and 'writing' is chosen as the most representative term. For each cluster, we choose as its representative concept, the one with the highest sum of similarities with the rest of elements in the cluster.

The semantic feature vector $f^s \in \mathbb{R}^{|C|}$ for image I is a 100-dimensional array, such that each component $f^s(I)_j$ of the vector represents the confidence with which the jth concept is detected in the image. The confidence value for the concept j, representing the cluster C_j, is obtained as the sum of the confidences r_I of all the concepts included in C_j that have also been detected on image I:

$$f^s(I)_j = \sum_{c_k \in \{C_j\}} r_I(c_k)$$

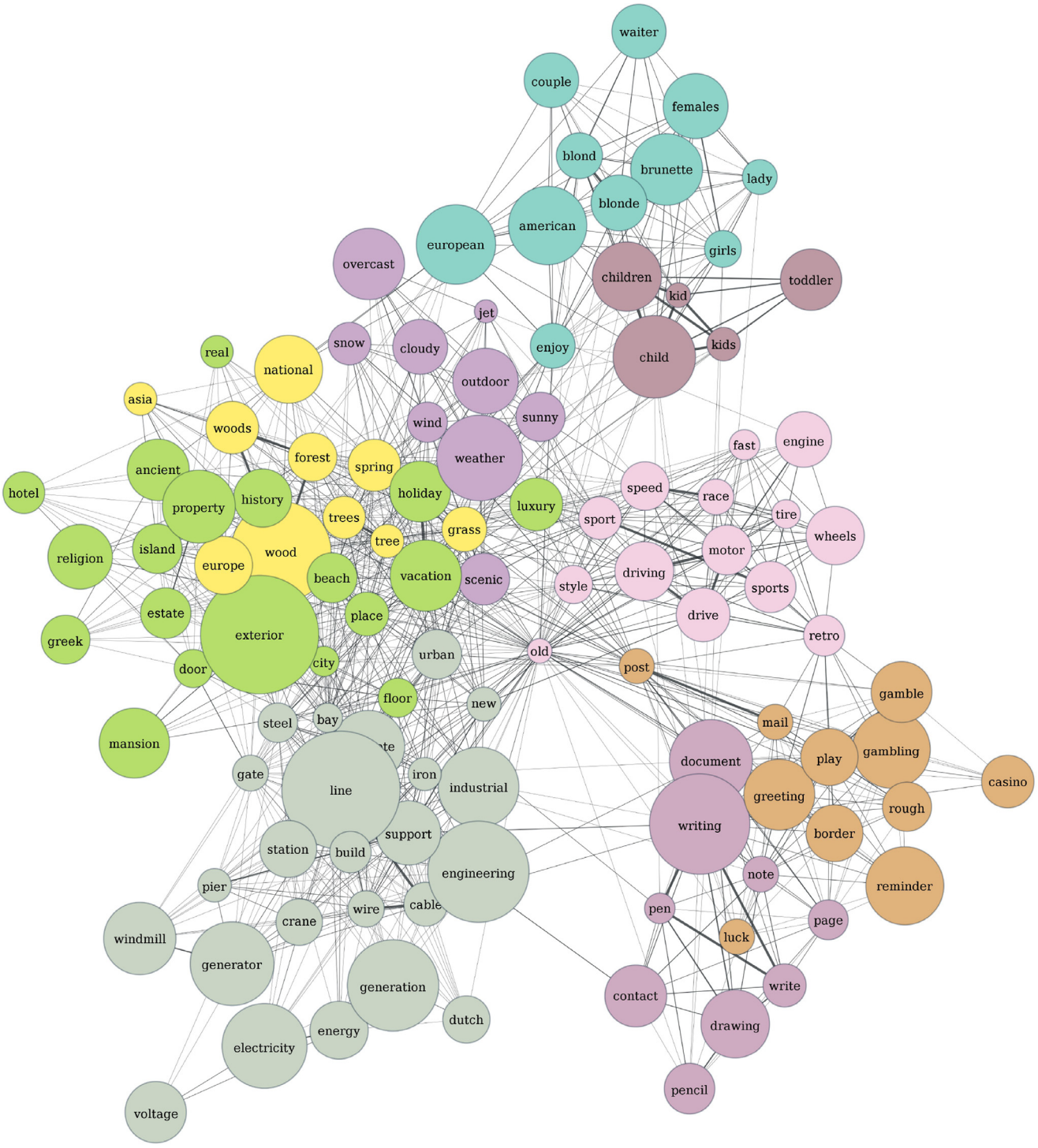

Fig. 3. Simplified graph obtained after calculating similarities of the concepts of a day's lifelog and clustering them. Each color corresponds to a different cluster, the edge width represents the magnitude of the similarity between concepts, and the nodes size represents the number of connections they have (the biggest node in each cluster is the representative one). We only showed a small subset of the 100 clusters. This graph was drawn using graph-tool (http://graph-tool.skewed.de).

where C_I is the set of concepts detected on image I, C_j is the set of concepts in cluster j, and $r_I(c_k)$ is the confidence associated to concept c_k on image I. The final confidence values are normalized so that they are in the interval [0, 1].

Taking into account that the camera wearer can be continuously moving, even if in a single environment, the objects that can be appearing in temporally adjacent images may be different. To this end, we apply a Parzen Window Density Estimation method (Parzen, 1962) to the matrix obtained by concatenating the semantic feature vectors along the sequence to obtain a smoothed and temporally coherent set of confidence values. Additionally, we discard the concepts with a low variability of confidence values along the sequence which correspond to non-discriminative concepts that can appear on any environment. The low variability of confidence value of a concept may correspond to constantly having high or low confidence value in most environments.

In Fig. 4, the matrix of concepts (semantic features) associated to an egocentric sequence is shown, displaying only the top

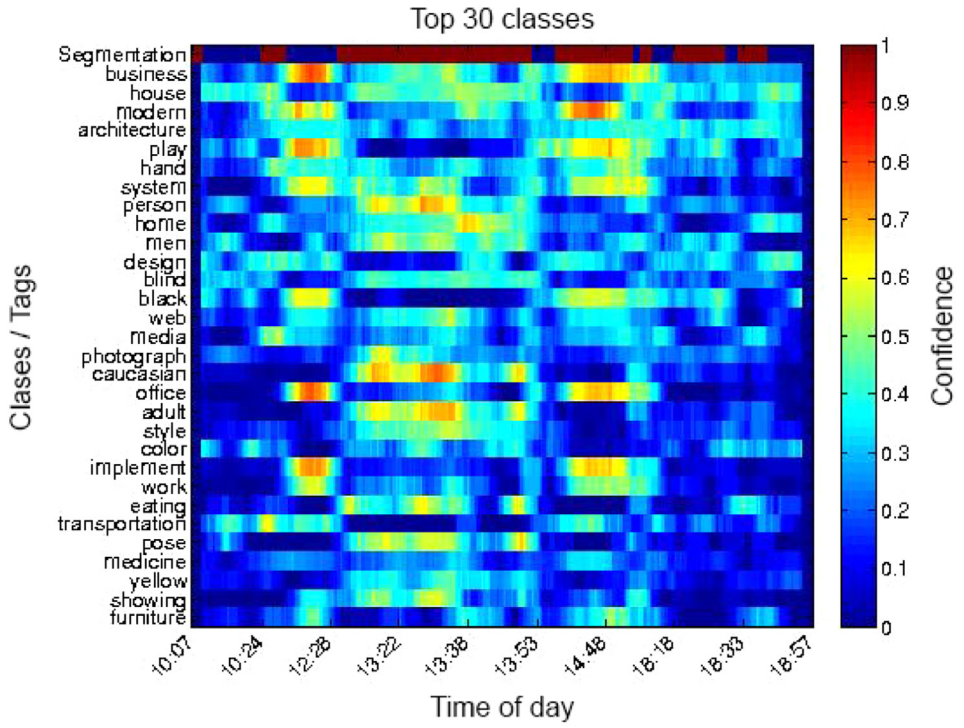

Fig. 4. Example of the final semantic feature matrix obtained for an egocentric sequence. The top 30 concepts (rows) are shown for all the images in the sequence (columns). Additionally, the top row of the matrix shows the ground truth (GT) segmentation of the dataset.

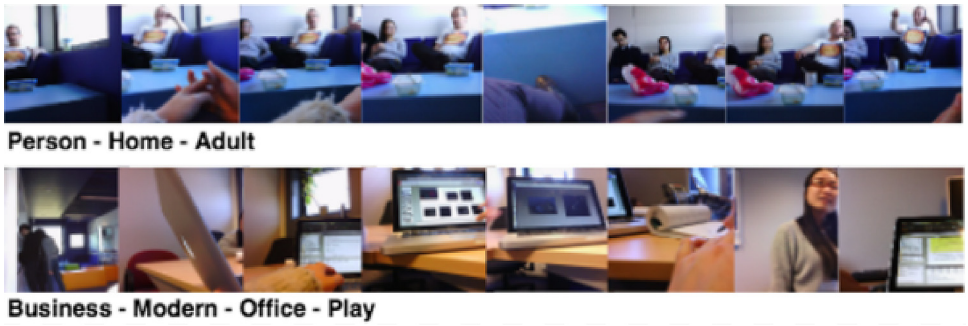

Fig. 5. Example of extracted tags on different segments. The first one corresponds to the period from 13.22 to 13.38 where the user is having lunch with colleagues, and the second, from 14.48 to 18.18, where he/she is working in the office with the laptop.

30 classes. Each column of the matrix corresponds to a frame and each row indicates the confidence with which the concept is detected in each frame. In the first row, the ground truth of the temporal segmentation is shown for comparison purposes. With this representation, repeated patterns along a set of continuous images correspond to the set of concepts that characterizes an event. For instance, the first frames of the sequence represent an indoor scene, characterized by the presence of people (see examples Fig. 5). The whole process is summarized in Fig. 6.

In order to consider the semantics of temporal segments, we used a concept detector based on the auto-tagging service developed by Imagga Technologies Ltd. Imagga's auto-tagging technology[2] uses a combination of image recognition based on deep learning and CNNs using very large collections of human annotated photos. The advantage of Imagga's Auto Tagging API is that it can directly recognize over 2700 different objects and in addition return more than 20,000 abstract concepts related to the analyzed images.

2.1.2. Contextual features

In addition to the semantic features, we represent images with a feature vector extracted from a pre-trained CNN. The CNN model that we use for computing the images representation is the AlexNet, which is detailed in Krizhevsky et al. (2012). The features are computed by removing the last layer corresponding to the classifier from the network. We used the deep learning framework Caffe (Jia, 2013) in order to run the CNN. Due to the fact that the weights have been trained on the ImageNet database (Deng et al., 2009), which is made of images containing single objects, we expect that the features extracted from images containing multiple objects will be representative of the environment. It is worth to remark that we did not use the weights obtained using a pre-trained CNN on the scenes from Places 205 database (Zhou et al., 2014), since the Narrative camera's field of view is narrow, which means that mostly its field-of-view is very restricted to characterize the whole scene. Instead, we usually only see objects on the foreground. As detailed in Talavera et al. (2015), to reduce the large variation distribution of the CNN features, which results problematic when computing distances between vectors, we used

[2] http://www.imagga.com/solutions/auto-tagging.html

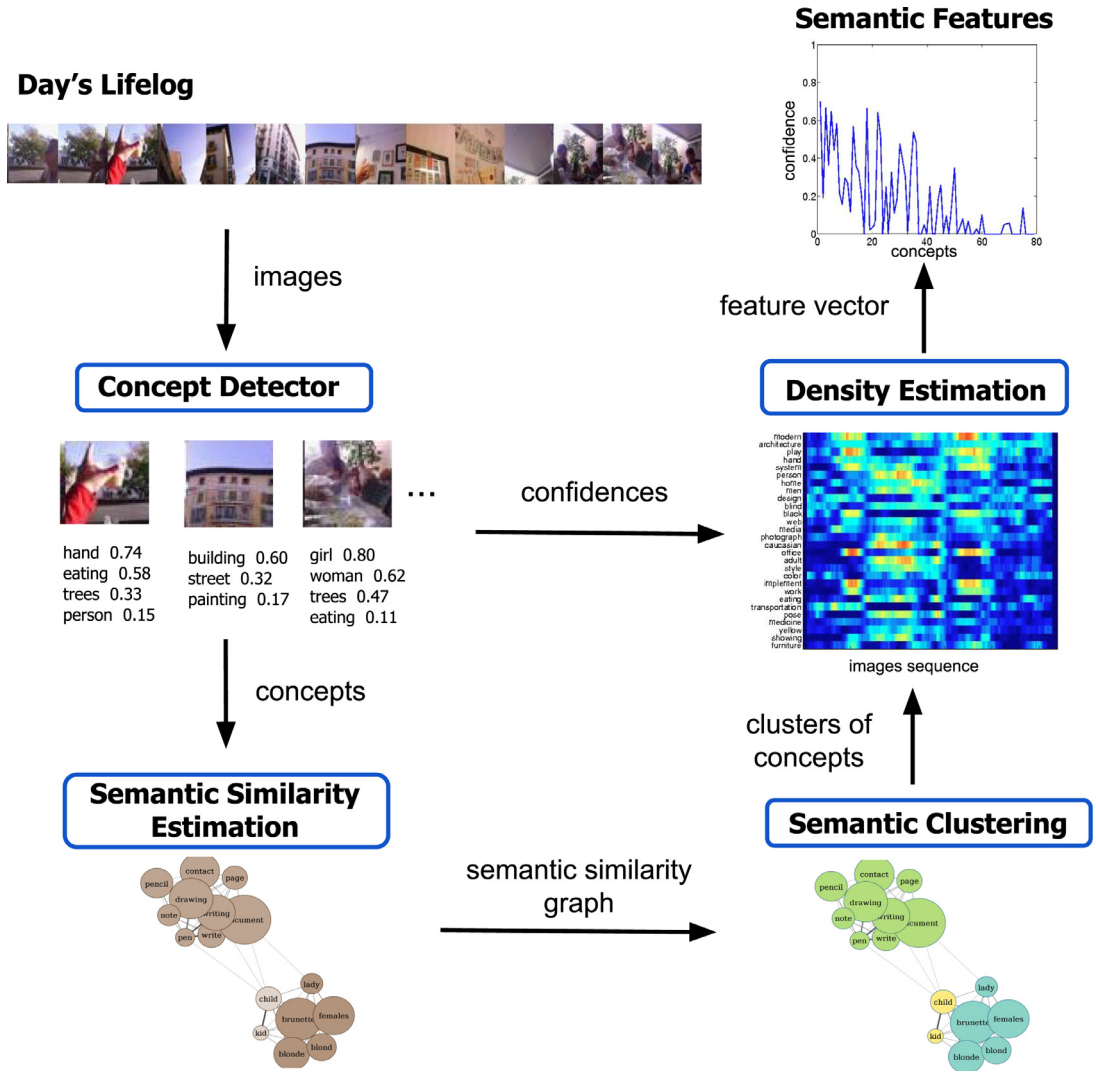

Fig. 6. General scheme of the semantic feature extraction methodology.

a signed root normalization to produce more uniformly distributed data (Zheng et al., 2014).

2.2. Temporal segmentation

The SR-clustering for temporal segmentation is based on fusing the semantic and contextual features with the R-Clustering method described in Talavera et al. (2015).

2.2.1. Agglomerative clustering

After the concatenation of semantic and contextual features, the hierarchical Agglomerative Clustering (AC) method is applied following a bottom-up clustering procedure. In each iteration, the method merges the most similar pair of clusters based on the distances among the image features, updating the elements similarity matrix. This is done until exhausting all possible consistent combinations. The *cutoff* global parameter defines the consistency of the merged clusters. We use the Cosine Similarity between samples, which is suited for high-dimensional positive spaces (Tan et al., 2005). The shortcoming of this method is that it tends to over-segment the photo streams.

2.2.2. ADWIN

To compensate the over-segmentation produced by AC, we proposed to model the egocentric sequence as a multi-dimensional data stream and to detect changes in the mean distribution through an adaptive learning method called ADWIN (Bifet and Gavalda, 2007), which provides a rigorous statistical guarantee of performance in terms of false positive rate. The method, based on the Hoeffding's inequality (Hoeffding, 1963), tests recursively if the difference between the averages of two temporally adjacent (sub)windows of the data, say W_1 and W_2, is larger than a threshold. The value of the threshold takes into account if both subwindows are *large enough* and *distinct enough* for a $k-$dimensional signal (Drozdzal et al., 2014), computed as:

$$\epsilon_{cut} = k^{1/p}\sqrt{\frac{1}{2m}\ln\frac{4}{k\delta'}}$$

where p indicates the p-norm, $\delta \in (0, 1)$ is a user defined confidence parameter, and m is the harmonic mean between the lengths of W_1 and W_2. In other words, given a predetermined confidence, ADWIN statistically guarantees that it will find any major change in the data means. Given a confidence value δ, the higher the dimension k is, the more samples n the bound needs to reach assuming the same value of ϵ_{cut}. The higher the norm used is, the less important the dimensionality k is. Since we model the sequence as a high dimensional data stream, ADWIN is unable to predict changes involving a relatively small number of samples, which often characterizes Low Temporal Resolution (LTR) egocen-

tric data, leading to under-segmentation. Moreover, since it considers only the mean change, it is able to detect changes due to other statistics such as the variance.

2.2.3. Graph-Cuts regularization

We use Graph-Cuts (GC) as a framework to integrate both of the previously described approaches, AC and ADWIN, to find a compromise between them that naturally leads to a temporally consistent result. GC is an energy-minimization technique that works by finding the minimum of an energy function usually composed of two terms: the *unary term U*, also called data term, that describes the relationship of the variables to a possible class and the *binary term V*, also called pairwise or regularization term, that describes the relationship between two neighboring samples (temporally close images) according to their feature similarity. The binary term smooths boundaries between similar frames, while the unary term keeps the cluster membership of each sequence frame according to its likelihood. In our problem, we defined the unary term as the sum of 2 parts ($U_{ac}(f_i)$ and $U_{adw}(f_i)$). Each of them expresses the likelihood of an image I_i represented by the set of features f_i to belong to segments coming from the corresponding previously applied segmentation methods. The energy function to be minimized is the following:

$$E(f) = \sum_{i}^{n}\left[(1-\omega_1)U_{ac}(f_i) + \omega_1 U_{adw}(f_i)\right] + \omega_2 \sum_{i}^{n}\left[\frac{1}{|N_i|}\sum_{j\in N_i} V_{i,j}(f_i, f_j)\right]$$

where $f_i = [f^c(I_i), f^s(I_i)], i = \{1, \ldots, n\}$ are the set of contextual f^c and semantic image features f^s for the ith image, N_i is a set of temporal neighbors centered at i, and ω_1 and ω_2 ($\omega_1, \omega_2 \in [0, 1]$) are the unary and the binary weighting terms, respectively. We can improve the segmentation outcome of GC by defining how much weight do we give to the likelihood of each unary term and balancing the trade-off between the unary and the pairwise energies, respectively. The minimization is achieved through the max-cut algorithm, leading to a temporal segmentation with similar frames having as large likelihood as possible to belong to the same segment, while maintaining segment boundaries in temporally neighboring images with high feature dissimilarity.

More precisely, the unary energy is composed of two terms representing, each of them, the likelihoods of each sample to belong to each of the clusters (or decisions) obtained either applying ADWIN (T_{adw}) or AC (T_{ac}) respectively:

$$U_{ac}(f_i) = P_{ac}(f_i \in T_{ac}), \quad U_{adw}(f_i) = P_{adw}(f_i \in T_{adw})$$

The pair-wise energy is defined as:

$$V_{i,j}(f_i, f_n) = e^{-\mathrm{dist}(f_i, f_j)}$$

An illustration of this process is shown in Fig. 2.

3. Experiments and validation

In this section, we discuss the datasets and the statistical evaluation measurements used to validate the proposed model and to compare it with the state-of-the-art methods. To sum up, we apply the following methodology for validation:

1. Three different datasets acquired by 3 different wearable cameras are used for validation.
2. The F-Measure is used as a statistical measure to compare the performance of different methods.
3. Two consistency measures to compare different manual segmentations is applied.
4. Comparison results of SR-Clustering with 3 state-of-the-art techniques is provided.
5. Robustness of the final proposal is proven by validating the different components of SR-Clustering.

3.1. Data

To evaluate the performance of our method, we used 3 public datasets (EDUB-Seg, AIHS and Huji EgoSeg's sub dataset) acquired by three different wearable cameras (see Table 1).

EDUB-Seg: is a dataset acquired by people from our lab with the Narrative Clip, which takes a picture every 30 seconds. Our Narrative dataset, named EDUB-Seg (Egocentric Dataset of the University of Barcelona - Segmentation), contains a total of 18,735 images captured by 7 different users during overall 20 days. To ensure diversity, all users were wearing the camera in different contexts: while attending a conference, on holiday, during the weekend, and during the week. The EDUB-Seg dataset is an extension of the dataset used in our previous work (Talavera et al., 2015), that we call EDUB-Seg (Set1) to distinguish it from the newly added in this paper EDUB-Seg (Set2). The camera wearers, as well as all the researchers involved on this work, were required to sign an informed written consent containing set of moral principles (Kelly et al., 2013; Wiles et al., 2008). Moreover, all researchers of the team have signed to do not publish any image identifying a person in a photo stream without his/her explicit permission, except unknown third parties.

AIHS subset: is a subset of the daily images from the database called *All I Have Seen* (AIHS) (Jojic et al., 2010), recorded by the SenseCam camera that takes a picture every 20 seconds.The original AIHS dataset [3] has no timestamp metadata. We manually divided the dataset in five days guided by the pictures the authors show in the website of their project and based on the daylight changes observed in the photo streams. The 5 days sum up a total of 11,887 images. Comparing both cameras (Narrative and SenseCam), we can remark their difference with respect to the cameras' lens (fish eye vs normal), and the quality of the images they record. Moreover, SenseCam acquires images with a larger field of view and significant deformation and blurring. We manually defined the GT for this dataset following the same criteria we used for the EDUB-Seg photo streams.

Huji EgoSeg: due to the lack of other publicly available LTR datasets for event segmentation, we also test our temporal segmentation method to the ones provided in the dataset Huji EgoSeg (Poleg et al., 2014). This dataset was acquired by the GoPro camera, which captures videos with a temporal resolution of 30 fps. Considering the very significant difference in frame rate of this camera compared to Narrative (2 fpm) and SenseCam (3 fpm), we applied a sub-sampling of the data by just keeping 2 images per minute, to make it comparable to the other datasets. In this dataset, several short videos recorded by two different users are provided. Consequently, after sub-sampling all the videos, we merged the resulting images from all the short videos to construct a dataset per each user, which consists of a total number of 700 images. The images were merged following the numbering order that was provided by the authors to their videos. We also manually defined the GT for this dataset following the same used criteria for the EDUB-Seg dataset.

In summary, we evaluate the algorithms on 27 days with a total of 31,322 images recorded by 10 different users. All datasets contain a mixture of highly variable indoor and outdoor scenes with a large variety of objects. We make public the EDUB-Seg dataset[4],

[3] http://research.microsoft.com/en-us/um/people/jojic/aihs/
[4] http://www.ub.edu/cvub/dataset/

Table 1
Table summarizing the main characteristics of the datasets used in this work: frame rate (FR), spatial resolution (SR), number of users (#Us), number of days (#Days), number of images (#Img). The Huji EgoSeg dataset has been subsampled to 2 fpm as detailed in the main text.

Dataset	Camera	FR	SR	#Us	#Days	#Img
EDUB	Narrative	2 fpm	2592x1944	7	20	18,735
AIHS-subset	SenseCam	3 fpm	640x480	1	5	11,887
Huji EgoSeg	GoPro Hero3+	30 fps*	1280x720	2	2	700

together with our GT segmentations of the datasets Huji EgoSeg and AIHS subset. Additionally, we release the SR-Clustering ready-to-use complete code[5].

3.2. Experimental setup

Following Li et al. (2013), we measured the performances of our method by using the F-Measure (FM) defined as follows:

$$FM = 2\frac{RP}{R+P},$$

where P is the precision defined as ($P = \frac{TP}{TP+FP}$) and R is the recall, defined as ($R = \frac{TP}{TP+FN}$). *TP, FP* and *FN* are the number of true positives, false positives and false negatives of the detected segment boundaries of the photo stream. We define the FM, where we consider TPs the images that the model detects as boundaries of an event and that were close to the boundary image defined in the GT by the annotator (given a tolerance of 5 images in both sides). The FPs are the images detected as events delimiters, but that were not defined in the GT, and the FNs the lost boundaries by the model that are indicated in the GT. Lower FM values represent a wrong boundary detection while higher values indicate a good segmentation. Having the ideal maximum value of 1, where the segmentation correlates completely with the one defined by the user.

The annotation of temporal segmentations of photo streams is a very subjective task. The fact that different users usually do not perform the same when annotating, may lead to bias in the evaluation performance. The problem of the subjectivity when defining the ground truth was previously addressed in the context of image segmentation (Martin et al., 2001). In Martin et al. (2001), the authors proposed two measures to compare different segmentations of the same image. These measures are used to validate if the performed segmentations by different users are consistent and thus, can be served as an objective benchmark for the evaluation of the segmentation performances. In Fig. 7, we report a visual example that illustrates the urge of employing this measure for temporal segmentation of egocentric photo streams. For instance, the first segment in Fig. 7 is split in different segments when analyzed by different subjects although there is a degree of consistency among all segments. Inspired by this work, we re-define the local refinement error, between two temporal segments, as follows:

$$E(S_A, S_B, I_i) = \frac{|R(S_A, I_i) \setminus R(S_B, I_i)|}{|R(S_A, I_i)|},$$

where $\setminus$ denotes the set difference and, S_A and S_B are the two segmentations to be compared. $R(S_X, I_i)$ is the set of images corresponding to the segment that contains the image I_i, when obtaining the segmentation boundaries S_X.

If one temporal segment is a proper subset of the other, then the images lie in one interval of refinement, which results in the local error of zero. However, if there is no subset relationship, the two regions overlap in an inconsistent manner that results in a non-zero local error. Based on the definition of local refinement we provided above, two error measures are defined by combining the values of the local refinement error for the entire sequence. The first error measure is called Global Consistency Error (GCE) that forces all local refinements to be in the same direction (segments of segmentation A can be only local refinements of segments of segmentation B). The second error measure is the Local Consistency Error (LCE), which allows refinements in different directions in different parts of the sequence (some segments of segmentation A can be of local refinements of segments of segmentation B and vice verse). The two measures are defined as follows:

$$GCE(S_A, S_B) = \frac{1}{n} min\left\{\sum_i^n E(S_A, S_B, I_i), \sum_i^n E(S_B, S_A, I_i)\right\}$$

$$LCE(S_A, S_B) = \frac{1}{n}\sum_i^n min\{E(S_A, S_B, I_i), E(S_B, S_A, I_i)\}$$

where n is the number of images of the sequence, S_A and S_B are the two different temporal segmentations and I_i indicates the i-th image of the sequence. The GCE and the LCE measures produce output values in the range [0, 1] where 0 means no error.

To verify that there is consistency among different people for the task of temporal segmentation, we asked three different subjects to segment each of the 20 sets of the EDUB-Seg dataset into events. The subjects were instructed to consider an *event* as a semantically perceptual unit that can be inferred by visual features, without any prior knowledge of what the camera wearer is actually doing. No instructions were given to the subjects about the number of segments they should annotate. This process gave rise to 60 different segmentations. The number of all possible pairs of segmentations is 1800, 60 of which are pairs of segmentations of the same set. For each pair of segmentations, we computed GCE and LCE. First, we considered only pairs of segmentations of the same sequence and then, considered the rest of possible pairs of segmentations in the dataset. The first two graphics in Fig. 8 (first row) show the GCE (left) and LCE (right) when comparing each set segmentations with the segmentations applied on the rest of the sets. The two graphics in the second row show the distribution of the GCE (left) and LCE (right) error when analyzing different segments describing the same video. As expected, the distributions that compare the segmentations over the same photo stream have the center of mass to the left of the graph, which means that the mean error between the segmentations belonging to the same set is lower than the mean error between segmentations describing different sets. In Fig. 9 we compare, for each pair of segmentations, the measures produced by different datasets segmentations (left) and the measures produced by segmentations of the same dataset (right). In both cases, we plot LCE vs. GCE. As expected, the average error between segmentations of the same photo stream (right) is lower than the average error between segmentations of different photo streams (left). Moreover, as indicated by the shape of the distributions on the second row of Fig. 9 (right), the peak of the LCE is very close to zero. Therefore, we conclude that given the task of segmenting an egocentric photo stream into events, different people tend to produce consistent and valid segmentation. Figs. 10 and 11 show segmentation com-

[5] https://github.com/MarcBS/SR-Clustering

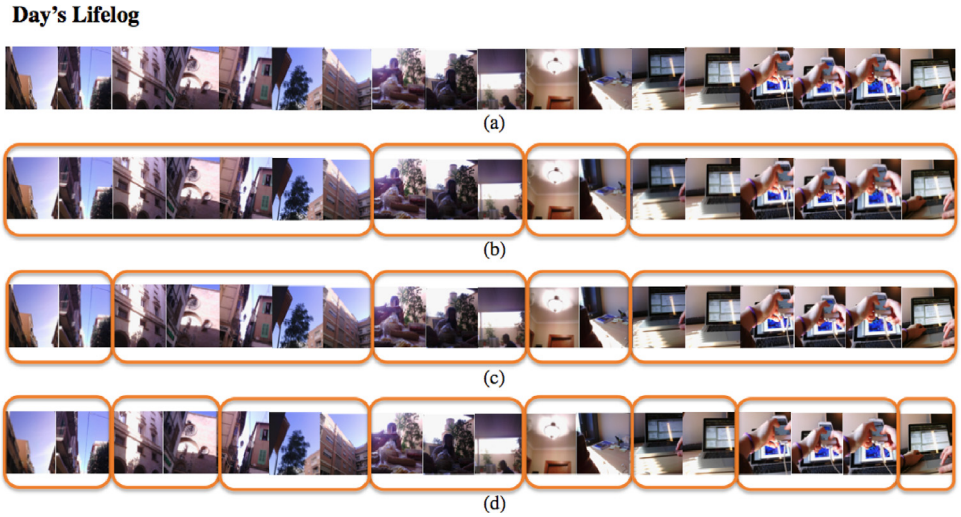

Fig. 7. Different segmentation results obtained by different subjects. (a) Shows a part of a day. (b), (c) and (d) are examples of the segmentation performed by three different persons. (c) and (d) are refinements of the segmentation performed by (b). All three results can be considered as being correct, due to the subjective intrinsic of the task. As a consequence, a segmentation consistency metric should not penalize different, yet consistent results of the segmentation.

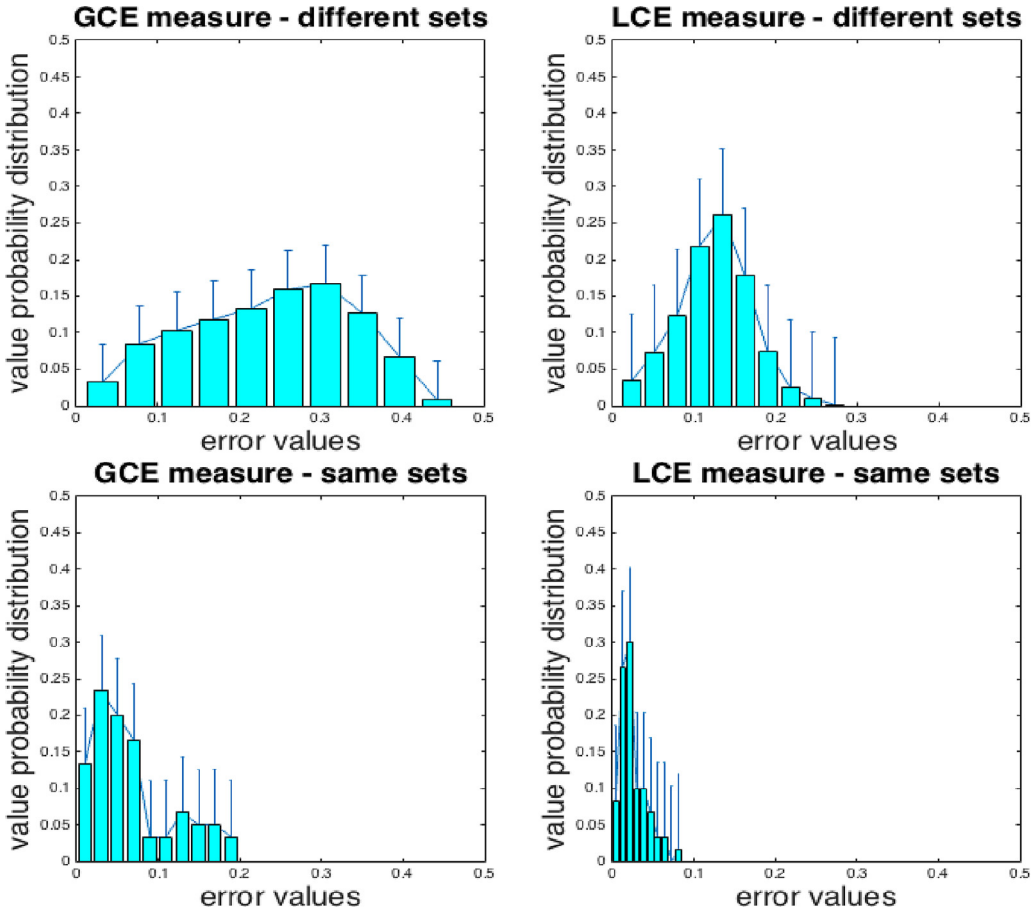

Fig. 8. GCE (left) and LCE (right) normalized histograms with the error values distributions, showing their mean and variance. The first row graphs represent the distribution of errors comparing segmentations of different sequences while the second row graphs show the distribution of error when comparing segmentations of the same set, including the segmentation of the camera wearer.

parisons of three different persons (not being the camera wearer) that were asked to temporally segment a photo stream and confirm our statement that different people tend to produce consistent segmentations.

Since our interpretation of events is biased by our personal experience, the segmentation done by the camera wearer could be very different by the segmentations done by third persons. To quantify this difference, in Figs. 8 and 9 we evaluated the LCE and the GCE including also the segmentation performed by the camera wearer. From this comparison, we can observe that the error mean does not vary but that the degree of local and global consistency is higher when the set of annotators does not include

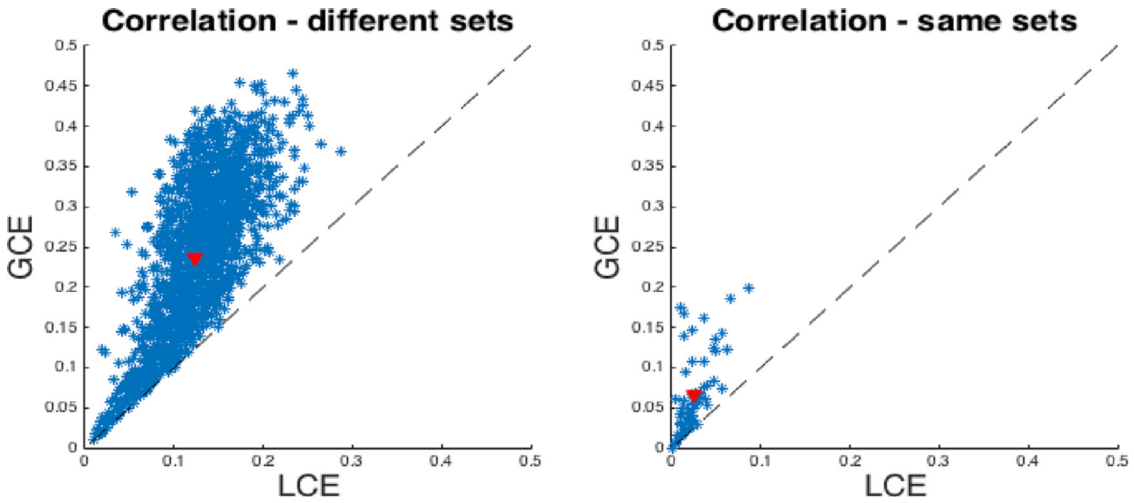

Fig. 9. LCE vs GCE for pairs of segmentations of different sequences (left) and for pairs of segmentations of the same sequence (right). The differences w.r.t. the dashed line x = y show how GCE is a stricter measure than LCE. The red dot represents the mean of all the cloud of values, including the segmentation of the camera wearer. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

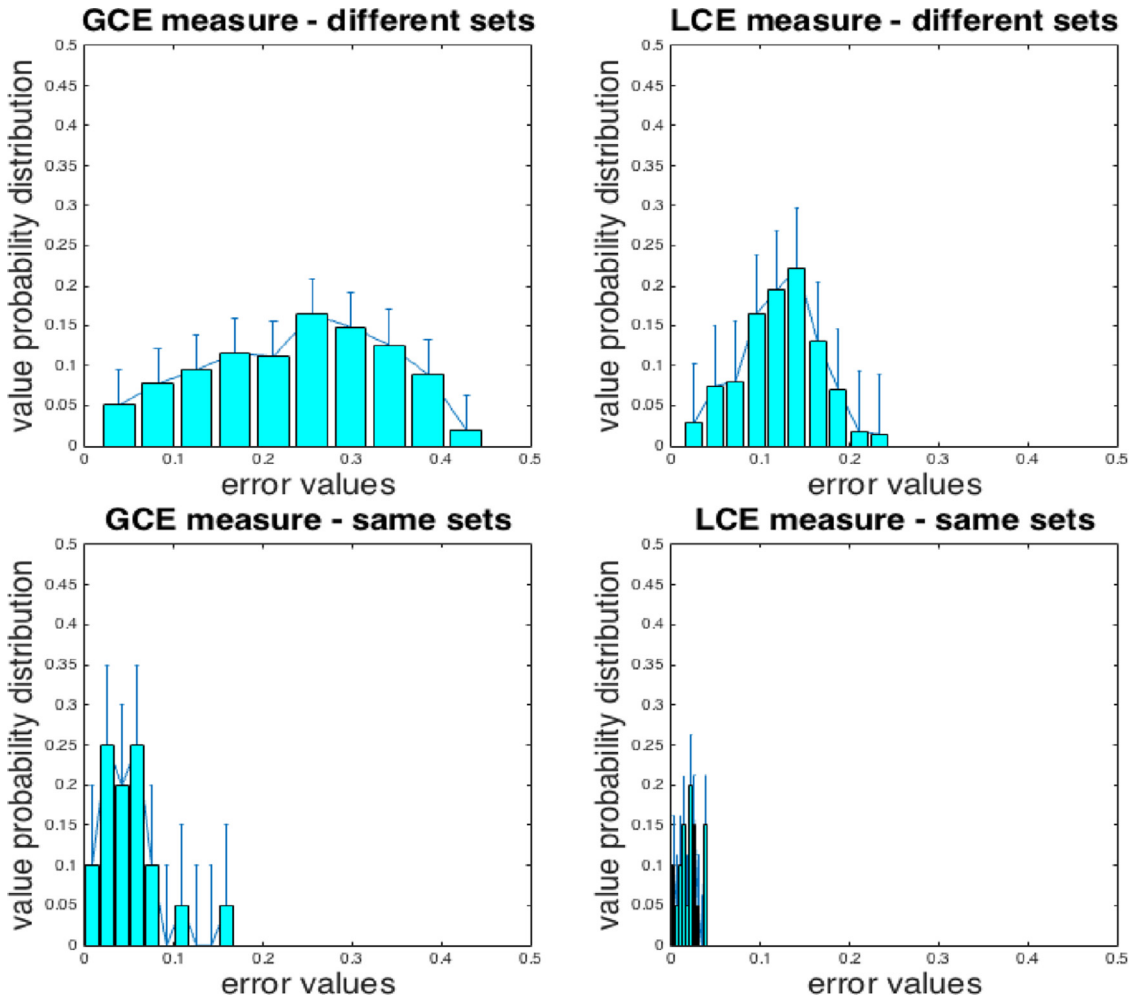

Fig. 10. GCE (left) and LCE (right) normalized histograms with the error values distributions, showing their mean and variance. The first row graphs represent the distribution of the errors comparing segmentations of different sequences while the second row graphs show the distribution of the errors when comparing segmentations of the same set, excluding the segmentation of the camera wearer.

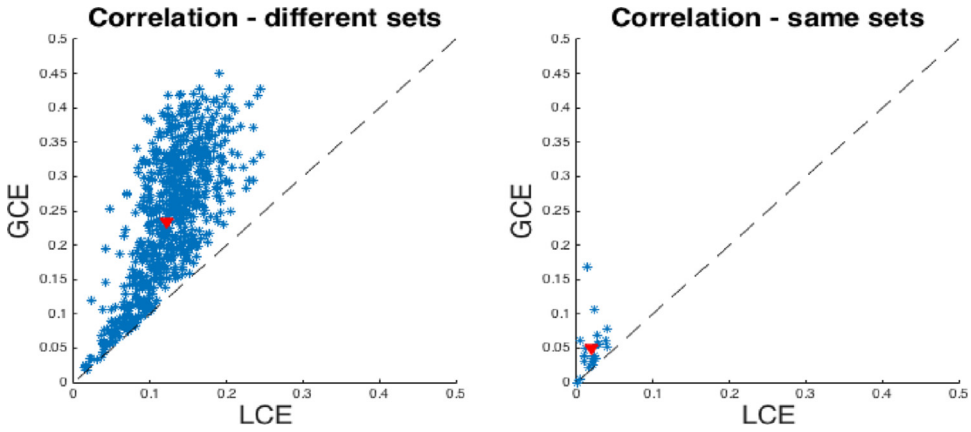

Fig. 11. LCE vs GCE for pairs of segmentations of different sequences (left) and for pairs of segmentations of the same sequence (right). The differences w.r.t. the dashed line x = y show how GCE is a stricter measure than LCE. The red dot represents the mean of all the cloud of values, excluding the segmentation of the camera wearer. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Average FM results of the state-of-the-art works on the egocentric datasets (first part of the table); for each of the components of our method (second part); and for each of the variations of our method (third part). The last line shows the results of our complete method. AC stands for Agglomerative Clustering, ADW for ADWIN and ImaggaD is our proposal for semantic features, where D stands for Density Estimation.

	AIHS (Jojic et al., 2010)	EDUB-Seg Set1	EDUB-Seg Set2	EDUB-Seg
Motion (Bolaños et al., 2014)	0.66	0.34		
AC-Color (Lee and Grauman, 2015)	0.60	0.37	0.54	0.50
R-Clustering (Talavera et al., 2015)	**0.79**	0.55		
ADW	0.31	0.32		
ADW-ImaggaD	0.35	0.55	0.29	0.36
AC	0.68	0.45		
AC-ImaggaD	0.72	0.53	0.64	0.61
SR-Clustering-LSDA	0.78	0.60	0.64	0.61
SR-Clustering-NoD	0.77	0.66	0.63	0.60
SR-Clustering	0.78	**0.69**	**0.69**	**0.66**

the camera wearer as it can be appreciated by the fact that the distributions are slightly shifted to the left and thinner. However, since this variation is of the order of 0.05%, we can conclude that event segmentation of egocentric photo streams can be objectively evaluated.

When comparing the different segmentation methods w.r.t. the obtained FM (see Section 3.3), we applied a grid-search for choosing the best combination of hyper-parameters. The set of hyper-parameters tested are the following:

- AC linkage methods $\in$ {*ward, centroid, complete, weighted, single, median, average,*}
- AC cutoff $\in \{0.2, 0.4, \ldots, 1.2\}$,
- GraphCut unary weight ω_1 and binary weight $\omega_2 \in \{0, 0.1, 0.2, \ldots, 1\}$,
- AC-Color $t \in \{10, 25, 40, 50, 60, 80, 90, 100\}$.

3.3. Experimental results

In Table 2, we show the FM results obtained by different segmentation methods over different datasets. The first two columns correspond to the datasets used in Talavera et al. (2015): AIHS-subset and EDUB-Seg (Set1). The third column corresponds to the EDUB-Seg (Set2) introduced in this paper. Finally, the fourth column corresponds to the results on the whole EDUB-Seg. The first part of the table (first three rows) presents comparisons to state-of-the-art methods. The second part of the table (next 4 rows), shows comparisons to different components of our proposed clustering method with and without semantic features. Finally, the third part of the table shows the results obtained using different variations of our method.

In the first part of Table 2, we compare to state-of-the-art methods. The first method is the Motion-Based segmentation algorithm proposed by Bolaños et al. (2014). As can be seen, the average results obtained are far below SR-Clustering. This can be explained by the type of features used by the method, which are more suited for applying a motion-based segmentation. This kind of segmentation is more oriented to recognize activities and thus, is not always fully aligned with the event segmentation labeling we consider (i.e. in an event where the user goes outside of a building, and then enters to the underground tunnels can be considered "in transit" by the Motion-Based segmentation, but be considered as three different events in our event segmentation). Furthermore, the obtained FM score on the Narrative datasets is lower than the SenseCam's for several reasons: Narrative has lower frame rate compared to Sensecam (AIHS dataset), which is a handicap when computing motion information, and a narrower field of view, which decreases the semantic information present in the image. We also evaluated the proposal of Lee and Grauman (2015) (best with $t = 25$), where they apply an Agglomerative Clustering segmentation using LAB color histograms. In this case, we see that the algorithm is even far below the obtained results by AC, where the Agglomerative Clustering algorithm is used over contextual CNN features instead of colour histograms. The main reason for this performance difference comes from the high difference in features expressiveness, that supports the necessity of using a rich set of features for correctly segmenting highly variable egocentric data. The last row of the first section of the table shows the results obtained by our previously published method (Talavera et al., 2015), where we were able to outperform the state-of-the-art of egocentric segmentation using contextual CNN features both on

Table 3
Average FM score on each of the tested methods using our proposal of semantic features on the dataset presented in Poleg et al. (2014).

	Huji EgoSeg (Poleg et al., 2014) LTR
ADW-ImaggaD	0.59
AC-ImaggaD	**0.88**
SR-Clustering	**0.88**

AIHS-subset and on EDUB-Seg Set1. Another possible method to compare with would be the one from Castro et al. (2015), although the authors do not provide their trained model for applying this comparison.

In the second part of Table 2, we compare the results obtained using only ADWIN or only AC with (ADW-ImaggaD, AC-ImaggaD) and without (ADW, AC) semantic features. One can see that the proposed semantic features, leads to an improved performance, indicating that these features are rich enough to provide improvements on egocentric photo stream segmentation.

Finally, on the third part of Table 2, we compared our segmentation methodology using different definitions for the semantic features. In the SR-Clustering-LSDA case, we used a simpler semantic features description, formed by using the weakly supervised concept extraction method proposed in Hoffman et al. (2014), namely LSDA. In the last two lines, we tested the model using our proposed semantic methodology (Imagga's tags) either without Density Estimation, SR-Clustering-NoD or with the final Density Estimation (SR-Clustering), respectively.

Comparing the results of SR-Clustering and R-Clustering on the first two datasets (AIHS-subset and EDUB-Seg Set1), we can see that our new method is able to outperform the results adding 14 points of improvement to the FM score, while keeping nearly the same FM value on the SenseCam dataset. The improvement achieved using semantic information can be also corroborated, when comparing the FM scores obtained on the second half of EDUB-Seg dataset (Set2 on the 3rd column) and on the complete version of this data (see the last column of the Table).

In Table 3 we report the FM score obtained by applying our proposed method on the sub-sampled Huji EgoSeg dataset to be comparable to LTR cameras. Our proposed method achieves a high performance, being 0.88 of FM for both AC and SR-Clustering when using the proposed semantic features. The improvement of the results when using the GoPro camera with respect to Narrative or SenseCam can be explained by two key factors: 1) the difference in the field of view captured by GoPro (up to 170°) compared to SenseCam (135°) and Narrative (70°), 2) the better image quality achieved by the head mounted camera.

In addition to the FM score, we could not consider the GCE and LCE measures to compare the consistency of the automatic segmentations to the ground truth, since both methods lead to a number of segments much larger than the number of segments in the ground truth and therefore these measures would not descriptive enough. This is due to the fact that any segmentation is a refinement of one segment for the entire sequence, and one image per segment is a refinement of any segmentation. Consequently, these two trivial segmentations, one segment for the entire sequence and one image per segment, achieve error zero for LCE and GCE. However, we observed that on average, the number of segments obtained by the method of Lee and Grauman (2015) is about 4 times bigger than the number of segments we obtained for the SenseCam dataset and about 2 times bigger than for the Narrative datasets. Indeed, we achieve an higher FM score with respect to the method of Lee and Grauman (2015), since it produces a considerable over-segmentation.

3.4. Discussion

The experimental results detailed in Section 3.3 have shown the advantages of using semantic features for the temporal segmentation of egocentric photo streams. Despite the common agreement about the inability of low-level features in providing understanding of the semantic structure present in complex events (Habibian and Snoek, 2014), and the need of semantic indexing and browsing systems, the use of high level features in the context of egocentric temporal segmentation and summarization has been very limited. This is mainly due to the difficulty of dealing with the huge variability of object appearance and illumination conditions in egocentric images. In the works of Doherty and Smeaton (2008) and Lee and Grauman (2015), temporal segmentation is still based on low level features. In addition to the difficulty of reliably recognizing objects, the temporal segmentation of egocentric photo streams has to cope with the lack of temporal coherence, which in practice means that motion features cannot reliably be estimated. The work of Castro et al. (2015) relies on the visual appearance of single images to predict the activity class of an image and on meta-data such as the day of the week and hour of the day to regularize over time. However, due to the huge variability in appearance and timing of daily activities, this approach cannot be easily generalized to different users, implying that for each new user re-training of the model and thus, labeling of thousand of images is required.

The method proposed in this paper offers the advantage of being needless of a cumbersome learning stage and offers a better generalization. The employed concept detector, has been proved to offer a rich vocabulary to describe the environment surrounding the user. In Fig. 12 are shown some segments, with the top eight associated concepts, obtained by using the proposed approach. This rich characterization is not only useful for better segmentation of sequences into meaningful and distinguishable events, but also serves as a basis for event classification or activity recognition among others. For example, Aghaei et al. (2015, 2016a,b) employed the temporal segmentation method in Talavera et al. (2015) to extract and select segments with trackable people to be processed. However, incorporating the semantic temporal segmentation proposed in this paper, would allow, for example, to classify events into social or non-social events. Moreover, using additional existing semantic features in a scene may be used to differentiate between different types of a social event ranging from a official meeting (including semantics such as laptop, paper, pen, etc.) to a friendly coffee break (coffee cup, cookies, etc.). Moreover, the semantic temporal segmentation proposed in this paper is useful for indexing and browsing.

4. Conclusions and future work

This paper proposed an unsupervised approach for the temporal segmentation of egocentric photo streams that is able to partition a day's lifelog in segments sharing semantic attributes, hence providing a basis for semantic indexing and event recognition. The proposed approach first detects concepts for each image separately by employing a CNN approach and later, clusters the detected concepts in a semantic space, hence defining the vocabulary of concepts of a day. Semantic features are combined with global image features capturing more generic contextual information to increase their discriminative power. By relying on these semantic features, a GC technique is used to integrate a statistical bound produced by the concept drift method, ADWIN and the AC, two methods with complementary properties for temporal segmentation. We evaluated the performance of the proposed approach on different segmentation techniques and on 17 day sets acquired by three different wearable devices, and we showed the improvement of the proposed method with respect to the state-of-the-art. Additionally, we

Fig. 12. Illustration of our SR-Clustering segmentation results from a subset of pictures from a Narrative set. Each line represents a different segment. Below each segment we show the top 8 found concepts (from left to right). Only a few pictures from each segment are shown.

introduced two consistency measures to validate the consistency of the ground truth. Furthermore, we made publicly available our dataset EDUB-Seg, together with the ground truth annotation and the code. We demonstrated that the use of semantic information on egocentric data is crucial for the development of a high performance method.

Further research will be devoted to exploit the semantic information that characterizes the segments for event recognition, where social events are of special interest. Additionally, we are interested in using semantic attributes to describe the camera wearer context. Hence, opening new opportunities for development of systems that can take benefit from contextual awareness, including systems for stress monitoring and daily routine analysis.

Acknowledgments

This work was partially founded by TIN2012-38187-C03-01, SGR 1219 and grant to research project 20141510 to Maite Garolera (from Fundació Marató TV3). The funders had no role in the study design, data collection and analysis, decision to publish, or preparation of the manuscript. M. Dimiccoli is supported by a *Beatriu de Pinós* grant (Marie-Curie COFUND action). P. Radeva is partly supported by an *ICREA Academia'2014* grant.

References

Aghaei, M., Dimiccoli, M., Radeva, P., 2015. Towards social interaction detection in egocentric photo-streams. In: Eighth International Conference on Machine Vision. International Society for Optics and Photonics. 987514–987514

Aghaei, M., Dimiccoli, M., Radeva, P., 2016. Multi-face tracking by extended bag-of–tracklets in egocentric videos. Comput. Vis. Image Underst. 149, 146–156. Special Issue on Assistive Computer Vision and Robotics

Aghaei, M., Dimiccoli, M., Radeva, P., 2016. With whom do I interact? Detecting social interactions in egocentric photo-streams. In: Proceedings of the International Conference on Pattern Recognition.

Bifet, A., Gavalda, R., 2007. Learning from time-changing data with adaptive windowing. In: Proceedings of SIAM International Conference on Data Mining.

Bolaños, M., Dimiccoli, M., Radeva, P., 2016. Towards storytelling from visual lifelogging: an overview. IEEE Trans. Human-Mach. Syst.. To appear on

Bolaños, M., Garolera, M., Radeva, P., 2014. Video segmentation of life-logging videos. In: Articulated Motion and Deformable Objects. Springer-Verlag, pp. 1–9.

Bolaños, M., Mestre, R., Talavera, E., Giró-i-Nieto, X., Radeva, P., 2015. Visual summary of egocentric photostreams by representative keyframes. Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on, 1–6.

Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. 23 (11), 1222–1239.

Castro, D., Hickson, S., Bettadapura, V., Thomaz, E., Abowd, G., Christensen, H., Essa, I., 2015. Predicting daily activities from egocentric images using deep learning. In: proceedings of the 2015 ACM International symposium on Wearable Computers. ACM, pp. 75–82.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: a large-scale hierarchical image database. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255.

Doherty, A.R., Hodges, S.E., King, A.C., et al., 2013. Wearable cameras in health: the state of the art and future possibilities. Am. J. Prev. Med. 44, 320–323. Springer

Doherty, A.R., Smeaton, A.F., 2008. Automatically segmenting lifelog data into events. In: Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services, pp. 20–23.

Drozdzal, M., Vitria, J., Segui, S., Malagelada, C., Azpiroz, F., Radeva, P., 2014. Intestinal event segmentation for endoluminal video analysis. In: Proceedings of International Conference on Image Processing.

Habibian, A., Snoek, C., 2014. Recommendations for recognizing video events by concept vocabularies. Comput. Vis. Image Underst. 124, 110–122.

Hoeffding, W., 1963. Probability inequalities for sums of bounded random variables. J. Am. Stat. Assoc. 58 (301), pp.13–30.
Hoffman, J., Sergio, S., Tzeng, E.S., Hu, R., J. Donahue, R.G., Darrell, T., Saenko, K., 2014. LSDA: large scale detection through adaptation. In: Advances in Neural Information Processing Systems, pp. 3536–3544.
Jia, Y., 2013. Caffe: an open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/.
Jojic, N., Perina, A., Murino, V., 2010. Structural epitome: a way to summarize one's visual experience, Advances in neural information processing systems, 1027–1035.
Kelly, P., Marshall, S., Badland, H., Kerr, J., Oliver, M., Doherty, A., Foster, C., 2013. An ethical framework for automated, wearable cameras in health behavior research. Am. J. Prev. Med. 44 (3), 314–319.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.
Lee, Y., Grauman, K., 2015. Predicting important objects for egocentric video summarization. Int. J. Comput. Vis. 114 (1), 38–55. doi:10.1007/s11263-014-0794-5.
Li, Z., Wei, Z., Jia, W., Sun, M., 2013. Daily life event segmentation for lifestyle evaluation based on multi-sensor data recorded by a wearable device. In: Proceedings of Engineering in Medicine and Biology Society. IEEE, pp. 2858–2861.
Lin, W.-H., Hauptmann, A., 2006. Structuring continuous video recording of everyday life using time-constrained clustering. Proceedings of SPIE, Multimedia Content Analysis, Management, and Retrieval 959.
Lu, Z., Grauman, K., 2013. Story-driven summarization for egocentric video.. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 2714–2721.
Martin, D., Fowlkes, C., Tal, D., Malik, J., 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of 8th International Conference on Computer Vision, pp. 416–423.
Miller, G.A., 1995. Wordnet: a lexical database for english. Commun. ACM 38 (11), 39–41.
Parzen, E., 1962. On estimation of a probability density function and mode. Ann. Math. Stat. 1065–1076.
Poleg, Y., Arora, C., Peleg, S., 2014. Temporal segmentation of egocentric videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2537–2544.
Poleg, Y., Ephrat, A., Peleg, S., Arora, C., 2015. Compact CNN for indexing egocentric videos. CoRR. 1504.07469.
Talavera, E., Dimiccoli, M., Bolanos, M., Aghaei, M., Radeva, P., 2015. R-clustering for egocentric video segmentation. In: Iberian Conference on Pattern Recognition and Image Analysis. Springer, pp. 327–336.
Tan, P.N., Steinbach, M., Kumar, V., 2005. Introduction to Data Mining, (first ed.) Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
Wiles, R., Prosser, J., Bagnoli, A., Clark, A., Davies, K., Holland, S., Renold, E., 2008. Visual ethics: Ethical issues in visual research. National Centre for Research Methods.
Zheng, L., Wang, S., He, F., Tian, Q., 2014. Seeing the big picture: dseep embedding with contextual evidences. CoRR abs/1406.0132.
Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A., 2014. Learning deep features for scene recognition using places database. In: Advances in Neural Information Processing Systems, pp. 487–495.

Dr. Mariella Dimiccoliis a Beatriu de Pinòs fellow (Marie-Curie COFUND action) at the Computer Vision Center. She received a computer engineering degree from the Technical University of Bari and a PhD from the Universitat Politècnica de Catalunya in 2004 and 2009 respectively. Her research interests are in the area of machine learning and computer vision with focus on first-person vision.

Marc Bolaños received his BSc degree in computer science from Universitat de Barcelona in 2013 and his MSc degree in artificial intelligence from Universitat Politècnica de Catalunya in 2015. He is a PhD candidate at Universitat de Barcelona. His research interests are deep learning and ego-vision for health applications.

Estefanía Talavera received her BSc degree in electronic engineering from Balearic Islands University in 2012 and her MSc degree in biomedical engineering from Polytechnic University of Catalonia in 2014. She is currently a PhD student at the University of Barcelona and University of Groningen. Her research interests are lifelogging and health applications.

Maedeh Aghaei received her M.Sc. degree in artificial intelligence in 2013 from Polytechnic University of Catalunya and is currently a Ph.D. Candidate in the Department of applied mathematics and analysis at the University of Barcelona. Her research interest include event recognition with emphasis on social signal processing.

Dr. Stavri Nikolov is co-founder and Research Director of Imagga Technologies Ltd. and Founding Director of the Digital Spaces Living Lab. Imagga Technologies Ltd is leader in the Image-Analysis-as-a-Service field. DSLL is a Living Lab focused on digital media technologies, location-based and smart city services, lifelogging and Cultural Heritage applications.

Dr. Petia Radeva is a senior researcher and associate professor at the University of Barcelona. She is Head of Computer Vision at University of Barcelona group and the MiLab of Computer Vision Center. Her present research interests are on development of learning-based approaches for computer vision, egocentric vision and medical imaging.

Object Discovery Using CNN Features in Egocentric Videos

Marc Bolaños[1( )], Maite Garolera[2], and Petia Radeva[1,3]

[1] Universitat de Barcelona, Barcelona, Spain
{marc.bolanos,petia.ivanova}@ub.edu
[2] Hospital de Terrassa-Consorci Sanitari de Terrassa, Terrassa, Spain
mgarolera@cst.cat
[3] Computer Vision Center of Barcelona, Bellaterra, Spain

Abstract. Lifelogging devices based on photo/video are spreading faster everyday. This growth can represent great benefits to develop methods for extraction of meaningful information about the user wearing the device and his/her environment. In this paper, we propose a semi-supervised strategy for easily discovering objects relevant to the person wearing a first-person camera. The egocentric video sequence acquired by the camera, uses both the appearance extracted by means of a deep convolutional neural network and an object refill methodology that allow to discover objects even in case of small amount of object appearance in the collection of images. We validate our method on a sequence of 1000 egocentric daily images and obtain results with an F-measure of 0.5, 0.17 better than the state of the art approach.

Keywords: Object discovery · Egocentric videos · Lifelogging · CNN

1 Introduction

Ubiquitous computing is more present everyday in our lifes, and with it: lifelogging devices [1,2] are increasing their popularity and spread. By wearing lifelogging cameras, we can build applications that convert huge amounts of data into meaningful information about the persons and their environment. Wearable cameras offer an easy manner to acquire information about our daily life tasks, and extract information about our typical activities and habits.

In this paper, we address the problem of discovering which are the usual objects that form the environment of a person wearing the camera from a lifelogging sequence by means of an Object Discovery (OD) method. Using this technique, as we can see in Fig. 1, we want to find the objects and environments that are able to distinguish the users of the wearable camera. Several works have been previously done in the OD field, some using segmentation techniques [3,4], others extracting objects relying on visual words [4–6], and combining clustering techniques with context information [7,8].

On the other hand, recently the use of Deep Neural Networks, and more precisely, Convolutional Neural Networks (CNN) is proving its huge potential to

R. Paredes et al. (Eds.): IbPRIA 2015, LNCS 9117, pp. 67–74, 2015.
DOI: 10.1007/978-3-319-19390-8_8

Fig. 1. Subset of frames from 3 different lifelogging sets. The first and third belonging to the same person, and the second to another one.

address different problems in the field of computer vision ([9–11], just to mention a few). Lately, a new method for egocentric activity recognition [12] using CNN data has been proposed for activity recognition. However, no methods on object discovery using these features exist yet.

Our proposal mainly relies on combining an object discovery method inspired by the work of Lee and Grauman [13]. However, we use both an appearance mode based on a feature extraction provided by a CNN pre-trained on ImageNet [14,15], and a refill methodology on already discovered instances. This strategy allows to construct classes of categories even with a low number of instances and also to discover in an iterative and semi-supervised way the important objects present in lifelogging videos (Fig. 2) according to their importance and frequency of appearance.

Fig. 2. Images acquired by a lifelogging device, where objects of interest appear like: mobile phone, person, or TV monitor.

2 The Object Discovery Approach

Our algorithm is based on several steps: it extracts image regions representing object candidates from each image, similar to [13] separates a part of them to the initial pool of discovered samples (40 %), assesses the "easiness" for the remaining, and applies an iterative process by clustering, labeling the best cluster and applying a supervised expansion to find harder instances of the discovered object (see Fig. 3). It uses both appearance and local context features about each object. Appearance are extracted with a CNN [15], and context is provided by the refill procedure, very suitable for lifelogging.

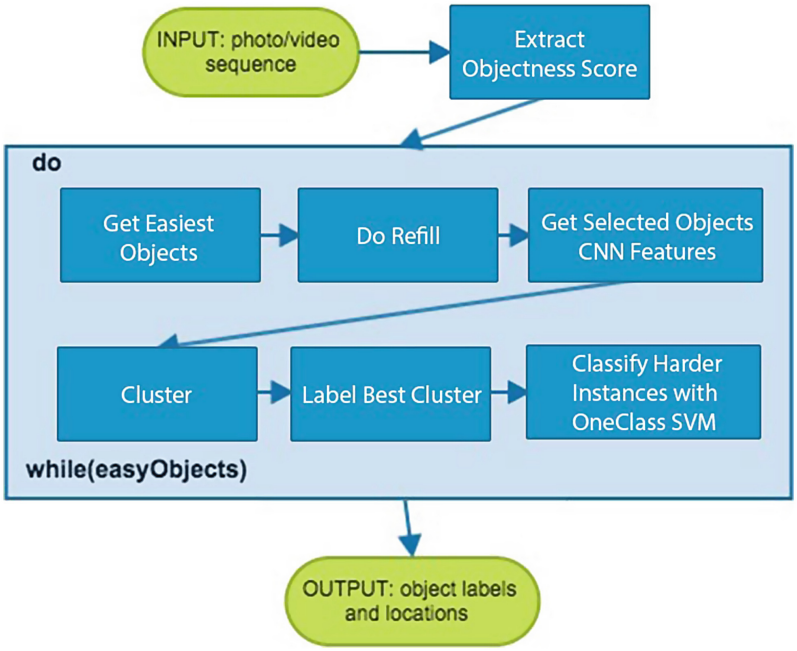

Fig. 3. Object discovery algorithm scheme.

2.1 Objectness and Easiness

The object sampling and candidates extraction, we used, rely on the objectness score proposed by Ferrari et al. [16], which combines 3 different methods and obtains an $objScore \in [0, 1]$ proportional to the probability of a window being an object. The $easyScore$ measure is defined as:

$$easyScore(\omega) = objScore(\omega). \tag{1}$$

A subset of samples is selected at each iteration filtering by their easyScore so that:

$$easyScore(\omega) > \mu + 1.25\sigma - 0.1t, \tag{2}$$

where μ and σ are respectively, the mean and the standard deviation of all scores. Hence, the number of easy samples increases at each iteration.

2.2 Refill Strategy

The objectness measure seems a promising method for obtaining object candidates, in general. However, this technique does not obtain the same results in lifelogging datasets due to the fact that images are not captured by a person looking on objects of the world, but are acquired while person is wearing the camera. Due to the inherent low frequency of appearance of different objects of the real world, to the limited image quality of the wearable lifelogging devices and to the constant moving of the user, wide part of the photos are unclear, dark, blurry, or deformed by the fisheye camera (see Fig. 1). All this causes lower precision of object candidates extraction leading to a very high number of image

regions obtained by the objectness method [16] with "No Object" instances (see the 7th image in Fig. 2).

In order to solve this problem, we define a "refill" methodology as follows: at each iteration, the selected easiest samples are complemented with a certain percentage of already labeled samples distributed on all the object classes (except the "No Object" class).

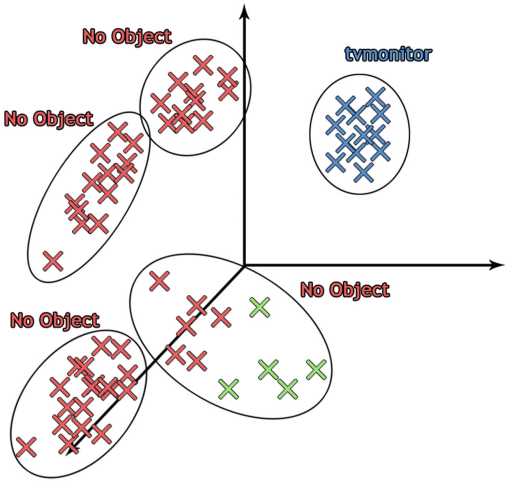

Fig. 4. Example of clusters formed only using the easiest samples

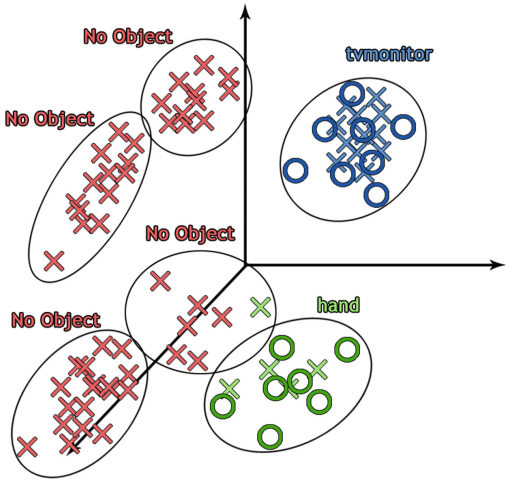

Fig. 5. Example of clusters formed adding the refill samples

In this way, we aid to address two problems: (1) difficulty to form a cluster from a very small set of class instances, and (2) difficulty to link samples of the same class that were blurry and unclear[1] (see Figs. 4 and 5).

2.3 Features for Object Discovery

As features, we used a pre-trained CNN, which captures information about millions of images in a succession of convolutional and pooling layers [14]. We deleted the last layer, which offers a supervised classification in 1,000 ImageNet classes, and used the output of the penultimate layer as our features (4096 variables). Note that our approach is different to the one of [13] that used: LAB histograms for extracting colour information, Pyramid HOG for extracting shape information, and Spatial Pyramid Matching [17] for extracting texture information.

2.4 Clustering and Hard Instancess Classification

After the features are extracted for the easiest and the refilled instances on each iteration, we apply an agglomerative Ward clustering. We use as cutoff criterion (similarly to the easiness filtering) 2 times the standard deviation plus

[1] Refilling the space with more samples of the same class can form a more compact and clear cluster.

the mean of the distances between clusters in the resulting hierarchy. Moreover, once the clusters are formed, we get the Silhouette Coefficient [18] on each of the clusters for selecting the most reliable one and label it with a majority voting strategy w.r.t the ground truth. This coefficient is only calculated on the unlabeled samples, never using the refilled ones. At the end of each iteration, an OneClass-SVM (with $\nu = 0.1$) is built with the new cluster and the rest of the easy samples are classified[2] with it for searching for harder instances.

3 Results

In this section, we discuss the lifelogging dataset, we used, and expose the different types of tests applied to illustrate the performance of the method proposed.

3.1 Lifelogging Dataset and Test Settings

Our dataset is a subset of the one used in [19,20], consisting of 1.000 images from a person's work day, from which 50.000 object candidates were extracted. To validate our method, we used the labels of the most frequent objects appearing, that are: "tvmonitor", "mobilephone", "hand" and "person" (Fig. 6).

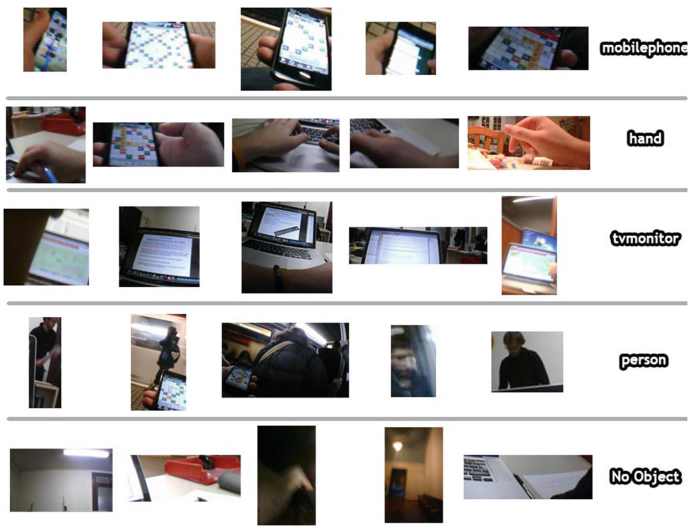

Fig. 6. Image samples of each object class

Using the objectness measure provided by [16], and after previously labeling all the objects present in the image, we assigned the corresponding true labels to each object candidate, considering only valid matches when their Overlapping Score (OS) was greater or equal than 0.4:

$$OS = \frac{|GT \cap \omega|}{|GT \cup \omega|}, \tag{3}$$

[2] On any case, the refilled samples, which were already labeled, can only get their labels changed if they did not belong to the initial selection set (40 %).

where GT stands for ground-truth and ω for the window detected by the method. Due to the challenging images (Subsect. 2.2 Refill Method) presented to the objectness measure, a very high percentage of samples (more than 76 %) could not be considered objects, and were labeled as "No Objects".

We performed three different test settings to evaluate our proposal:

1. CNN Features.
2. CNN Features with Refill.
3. Features of [13].

3.2 Tests Comparison

To evaluate our approach, we first calculated the general accuracy of the clustering methodology, giving the same weight to any sample from any class. The associated purity is defined as exactly the same as the accuracy, but without taking into account the "No Object" samples. The average per-class precision and recall defined by Sokolova et al. [21], allows to obtain the average F-measure. All measures were averaged by 5 executions per setting. Using these different measures, we compared all settings at the end of the easiest samples discovery (Fig. 7) and the F-measure for all settings on each iteration (Fig. 8).

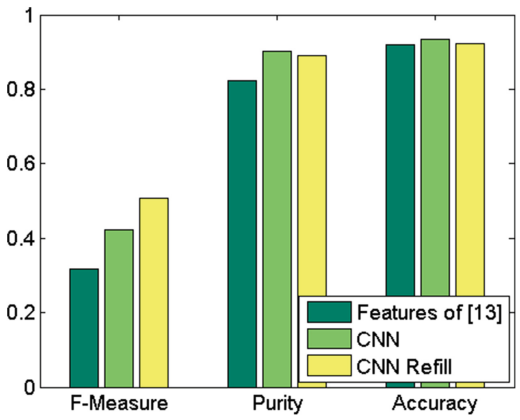

Fig. 7. Final F-measure, purity and accuracy for each setting

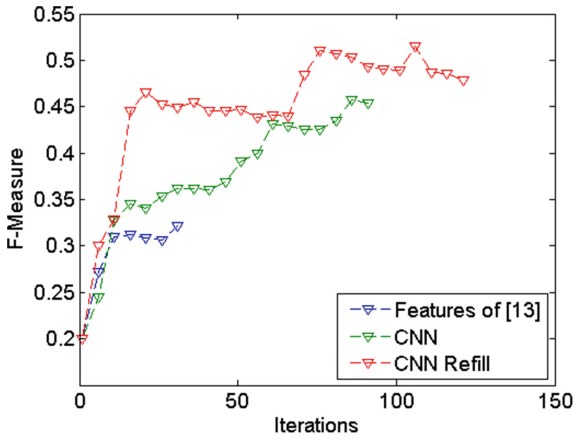

Fig. 8. F-measure evolution for each different setting

Looking at the first figure (Fig. 7), we can see that there is no change for any method on the accuracy, which is clearly due to the high number of "No Object" samples available, which are very easy to find everywhere and make the greater percentage of it. About the purity, we can observe changes on any setting using the CNN features, indicating that they can form more pure clusters for their best representation. Although there is a subtle purity decrease when using Refill, it is caused by the random initialization of the pool of discovered samples which, furthermore, is not statistically different. And finally, comparing the F-measure obtained, we can see more clearly that using CNN outperforms the features of [13], and that our complete method of CNN and Refill outperforms simply using the CNN features.

Comparing the evolution of the F-measure through the iterations (Fig. 8), first we have to consider that although in all the settings the amount of discovered samples at the end of the iterative process is similar, using CNN features (compared to those in [13]) produces smaller clusters because they are more distinguishable from the rest, and this makes the discovery process longer. And a similar phenomen happens when using the refill strategy, but in this case it is due to the fact that we add more samples at each iteration appart from the easiest ones.

Hence, using the CNN features combined with the refill strategy, the results clearly improved. This is caused by the discovery of different classes of samples. While when using the features of [13], we are only able to discover the two classes with more samples ("No Object" and "tvmonitor"), getting about 0.3 of F-measure, with CNN and the refill strategy, we can discover instances of each of the classes, getting about 0.5 of F-measure. On average, the approximate amount of image samples discovered using the best method (CNN Refill) are: "No Object" - 5000, "tvmonitor" - 500, "hand" - 50, "mobilephone" - 50 and "person" - 20. The total number of clusters labeled on average were about 120.

4 Conclusions

In this paper, we proposed a new object discovery algorithm that relies on features extracted from a pre-trained CNN, adapted for lifelogging photo/video sequences, and using a refill strategy for finding easily the classes with less samples. We proved that both the CNN features and the refill strategy can produce much better F-measure results and can discover a greater number of unfrequent classes than the state of the art approach. Furthermore, it has been proved that this combined strategy also works better than the previous ones for very noisy, blurry images and those with no objects.

As a future work, we plan to improve the objectness measure by training it on lifelogging images, to extend our object discovery including a context-awareness term similar to [13], and to use the discovered objects to characterize the environment of the persons wearing the camera.

References

1. Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K.: SenseCam: a retrospective memory aid. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 177–193. Springer, Heidelberg (2006)
2. Michael, K.: Wearable computers challenge human rights. ABC Science (2013)
3. Schulter, S., Leistner, C., Roth, P., Bischof, H.: Unsupervised object discovery and segmentation in videos. In: Proceedings of the British Machine Vision Conference, pp. 53.1–53.12. BMVA Press (2013)
4. Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1605–1614. IEEE (2006)

5. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their location in images In: Tenth International Conference on Computer Vision, ICCV, vol. 1, pp. 370–377. IEEE (2005)
6. Liu, D., Chen, T.: Unsupervised image categorization and object localization using topic models and correspondences between images. In: 11th International Conference on Computer Vision, ICCV, pp. 1–7. IEEE (2007)
7. Lee, Y.J., Ghosh, J., Grauman, K.: Discovering important people and objects for egocentric video summarization. In: Conference on CVPR, pp. 1346–1353. IEEE (2012)
8. Lee, Y.J., Grauman, K.: Object-graphs for context-aware visual category discovery. IEEE Trans. Pattern Anal. Mach. Intell. **34**(2), 346–358 (2012)
9. Honglak, L., Roger, G., Rajesh, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. Computer Science Department, Stanford University, Stanford (2009)
10. Honglak, L., Yan, L., Rajesh, R., Peter, P., Ng, A.Y.: Unsupervised feature learning for audio classification using convolutional deep belief networks. Computer Science Department, Stanford University, Stanford (2009)
11. Goodfellow, I.J., Bulatov, Y., Ibarz, J., Arnoud, S.: Vinay Shet: Multi-digit Number Recognition from Street View Imagery Using Deep Convolutional Neural Networks. Google Inc., Mountain View (2014)
12. Moghimi, M., Azagra, P., Montesano, L., Murillo, A.C., Belongie, S.: Experiments on an RGB-D wearable vision system for egocentric activity recognition. In: 3rd Workshop on Egocentric (First-person) Vision, CVPR (2014)
13. Lee, Y.J., Grauman, K.: Learning the easy things first: self-paced visual category discovery. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1721–1728. IEEE (2011)
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
15. Jia, Y.: Caffe: an open source convolutional architecture for fast feature embedding (2013). http://caffe.berkeleyvision.org/
16. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 73–80. IEEE (2010)
17. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178. IEEE (2006)
18. Tu, Z.: Auto-context and its application to high-level vision tasks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 1–8. IEEE (2008)
19. Bolaños, M., Garolera, M., Radeva, P.: Active labeling application applied to food-related object recognition. In: Proceedings of the 5th International Workshop on Multimedia for Cooking & Eating Activities, ACM Multiedia International Conference, pp. 45–50 (2013)
20. Bolaños, M., Garolera, M., Radeva, P.: Video segmentation of life-logging videos. In: Perales, F.J., Santos-Victor, J. (eds.) AMDO 2014. LNCS, vol. 8563, pp. 1–9. Springer, Heidelberg (2014)
21. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manage. **45**(4), 427–437 (2009)

Image and Vision Computing
journal homepage: www.elsevier.com

Ego-Object Discovery

Marc Bolaños[a], Petia Radeva[a,b]

[a] *Universitat de Barcelona, Gran Via de les Corts Catalanes, 585, Barcelona 08007, Spain*
[b] *Computer Vision Center, Building O Campus UAB, Bellaterra (Barcelona) 08193, Spain*

ABSTRACT

Lifelogging devices are spreading faster everyday. This growth can represent great benefits to develop methods for extraction of meaningful information about the user wearing the device and his/her environment. In this paper, we propose a semi-supervised strategy for easily discovering objects relevant to the person wearing a first-person camera. Given an egocentric video/images sequence acquired by the camera, our algorithm uses both the appearance extracted by means of a convolutional neural network and an object refill methodology that allows to discover objects even in case of small amount of object appearance in the collection of images. An SVM filtering strategy is applied to deal with the great part of the False Positive object candidates found by most of the state of the art object detectors. We validate our method on a new egocentric dataset of 4912 daily images acquired by 4 persons as well as on both PASCAL 2012 and MSRC datasets. We obtain for all of them results that largely outperform the state of the art approach. We make public both the EDUB dataset[1] and the algorithm code[2].

1. Introduction

Ubiquitous computing is more present everyday in our lives, and with it lifelogging devices (Hodges et al., 2006; Michael, 2013) are increasing their popularity and spread. By using wearable cameras, we can acquire continuous data about the life of persons, and build applications that convert this huge amount of data into meaningful information about their lifestyle. Hence, wearable cameras offer an easy manner to acquire information about our daily life tasks, and extract information about our typical activities and habits (Betancourt et al.) from an egocentric (or first-person) point of view. For example, Fig. 1 shows datasets acquired in three days by 3 different users. We can observe that different persons have different environments. Probably, the most remarkable reason for being able to detect visually the differences in the users' datasets is usually due to the distribution and aspect of scenes, objects and people that appear. Following these premises, in this paper, we address the problem of automatically discovering which are the usual objects that form the environment of a person wearing the camera by means of a novel Object Discovery (OD) method. We must note the difference between *Object Recognition*, where the goal is to discriminate objects according to their classes by a classifier previously trained with a set of training samples;

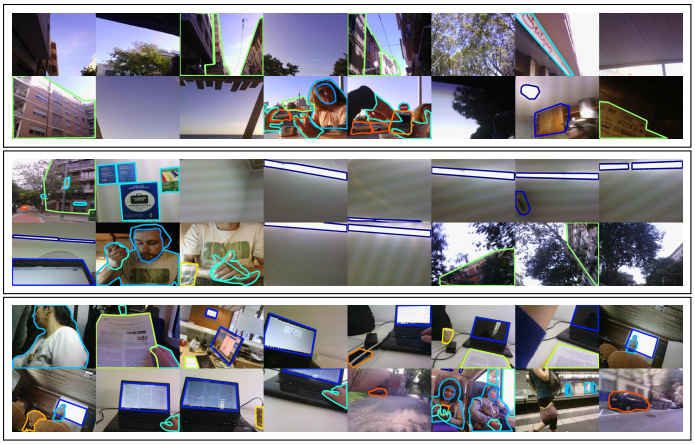

Fig. 1. Lifelogging sets from 3 users (each 2 row correspond to a different user). Note how objects help to discriminate different environments. The annotated objects are to be discovered by the object discovery algorithm.

** Marc Bolaños: Tel.: +34-669-648-301
e-mail: marc.bolanos@ub.edu (Marc Bolaños)

[1] https://www.dropbox.com/s/py8xhalqxz15co3/EDUB%202015.zip?dl=0
[2] https://github.com/MarcBS/Ego-Object_Discovery/releases

Object Detection, where we should detect the subregion in the image where an object appears; and *Object Discovery*, where we have to both detect new object instances or concepts, and assign them a label even without having training examples from all possible classes of objects.

1.1. Previous Work

Several works have been previously done in the OD field, some using segmentation techniques (Schulter et al., 2013; Russell et al., 2006), others extracting objects relying on visual words (Russell et al., 2006; Sivic et al., 2005; Liu and Chen, 2007). In (Chatzilari et al., 2011), a semi-supervised method for segmentation-level labeling is presented and in (Tuytelaars et al., 2010) a comparison of unsupervised OD methods is shown. One of the best performing OD methods is the one Lee's et.al. published in (Lee and Grauman, 2011), where the authors propose a semi-supervised OD approach for object discovery. It starts by selecting the easiest objects by an objectness detector and keeps an iterative discovery procedure by clustering object candidates, selecting the best one as the one corresponding to the newly discovered object and applying an One-Class SVM to discover harder instances of it. The authors use a set of low-level image appearance (texture, colour and shape) and context features. One of its main drawbacks is that the features that it used are not rich enough to capture the characteristics of any existent real world object. More recently, in (Kading et al., 2015), a method for object discovery relying in active learning was presented. The authors base their work in the assumption that when dealing with an active learning problem, the oracle does not always know all the classes in advance and that, furthermore, not all the classes are always interesting for the problem at hand. With this in mind, they propose an Expected Model Output Change (EMOC) criterion for selecting the most relevant and useful images to label for the problem they are addressing, and at the same time trying to avoid no valid objects by using a local density measure. Cho et al. in (Cho et al., 2015) worked on a part-based object discovery by proposing a new probabilistic matching strategy (Probabilistic Hough Matching) based on HOG descriptors for finding similar objects in different images. Additionally, they propose an associated confidence for finding the most outstanding object in each image.

In egocentric data, object discovery has been studied in much less extent. There, the OD brings new challenges considering the non-intentionallity of the images, that is, compared to usual intentional images, the objects and people (if any) usually do not appear in centered positions, and partial occlusions produced by other objects or the image border are quite frequent. In (Kang et al., 2011), the authors define a method for finding new objects that a person can encounter in their daily living. They start by applying a segmentation of the images at different levels, extracting colour, texture and shape information from each segment and applying a series of grouping and refinement steps to find consistent clusters that can represent new concepts. The authors in (Fathi et al., 2011) develop an object recognition method that uses segmentation techniques for extracting objects on egocentric visual data. In this case, the data acquired is captured using head-mounted cameras with high-temporal resolution (about 30 fps), what makes impossible to record the whole day of the person (due to memory and battery constraints). In order to solve this problem, we use cameras with low-temporal resolution (2-3 frames per minute) that are worn on chest level for maximizing the user comfort. As a result, we obtain a collection of images instead of a video, where objects are captured non-intentionally, and frequently appear blurred and non-centred. The main additional challenges these cameras cause are: 1) having frames so much temporally spaced disable the possibility to directly infer information from sequential frames and 2) extracted motion information is not reliable enough.

The main handicaps of existent OD methods are: 1) they lack a way to capture and reuse the knowledge acquired when analyzing the previous data, which is very important considering the redundancy of the data acquired in lifelogging (Min et al., 2014), and 2) many OD methods rely on using as a first step an object detection algorithm like (Alexe et al., 2010; Cheng et al., 2014; Arbeláez et al., 2014; Uijlings et al., 2013) for having an initial set of object candidates. As we prove in section 3.2, these methods usually produce a very high number of False Positives (FP) that should be dealt with.

1.2. Contributions

In this paper, we propose a new OD method for egocentric data (based on our previous work presented in (Bolaños et al., 2015)), that we call Ego-Object Discovery (EOD). Our contributions start by using a set of powerful features extracted by means of a Convolutional Neural Network (CNN). These networks are proving their huge potential to address different problems in the field of Computer Vision ((Honglak et al., 2009a,b; Goodfellow et al., 2014), just to mention a few). Lately, a new method (Moghimi et al., 2014) using CNN data has been proposed for egocentric activity recognition. However, no methods on OD using these features exist yet. To overcome the problem present in previous works of nonexistent knowledge reuse we use a new Refill methodology, which allows to discover new samples from the categories, even having a low number of instances, which are quite present in egocentric sequences. As additional contributions w.r.t. our previous work, we here present a strategy for solving the high number of FPs (or 'No Object' candidates) produced by the object detection methods: a SVM filtering strategy. Also introduce the first egocentric object discovery dataset (EDUB) of lifelogging data with ground truth (GT) object segmentations, apply a comparison with the state of the art object detection algorithms, and analyze the results of our method also on two public datasets of intentional images (PASCAL and MSRC).

The article is organized as follows: in section 2, we define the EOD algorithm. In section 3, we present the datasets used to validate our method, the tests of EOD on all datasets, comparison of state of the art object detectors and discussions on the obtained results. We finish with some conclusions and future work.

2. The Ego-Object Discovery Approach

Given the problem of OD in low-temporal resolution egocentric data, our algorithm is formulated as an iterative procedure. At the beginning, it should be provided with a seed of initial objects information to expand, defined as a small bag of labeled objects, represented by their regions, and called a **bag of refill**. The EOD algorithm passes through several steps (see Fig. 2): a) it detects image regions representing object candidates and their corresponding objectness scores from each new set of images, b) extracts object candidates features by using a pre-trained CNN, c) filters false object ('No Object') instances and d) proceeds with a clustering-based iterative procedure as follows: 1) on the *easiest* objects, it applies a *refill strategy* by using the bag of refill, 2) clusters them by using an agglomerative clustering approach and labels the best cluster that represents the newly discovered object and 3) applies a supervised expansion to find harder instances of it. After a fixed number of t iterations or until no easy sample remains, it outputs the set of found object coordinates and labels.

To describe and cluster the candidates, EOD uses both appearance and local context features. Appearance are extracted by a CNN (Jia, 2013), and context is provided by both the inherent description of the object background that also extracts the CNN, and indirectly the refill procedure, that will introduce instances of the same classes but with different backgrounds. Being very suitable for lifelogging images considering the redundancy of the objects we routinely see. In the following subsections, we give details about each step of the EOD procedure.

2.1. Object Candidates Preparation

Object Candidates Generation: The first step needed to characterize the environment of the user through object discovery is extracting a set of object candidates for each image. To do so, we used the Objectness detector provided by Ferrari et al. in (Alexe et al., 2010), which additionally to the bounding box for each candidate, outputs a score associated to the probability of being a true object (objectness score). This score is produced by three visual cues: *Multi-scale Saliency* (finds blob-like structures at multiple scales that could indicate the presence of an object); *Color Contrast* (finds high colour differences between the analyzed bounding box and its surroundings); and *Superpixels Straddling* (penalizes the bounding boxes that do not respect the boundaries of the superpixels in the image).

Object Candidates Characterization: As features to cluster the object candidates, we used a pre-trained CNN (Krizhevsky et al., 2012), which was trained on millions of images and is composed as a succession of convolutional and pooling layers. We deleted the last layer, which offers a supervised classification of 1.000 ImageNet classes, and used the output of the penultimate layer as our features (4096 variables). Note that our approach is different to the one of (Lee and Grauman, 2011) that used: LAB histograms for extracting colour information, Pyramid HOG for extracting shape information, and Spatial Pyramid Matching (Lazebnik et al., 2006) for extracting texture information.

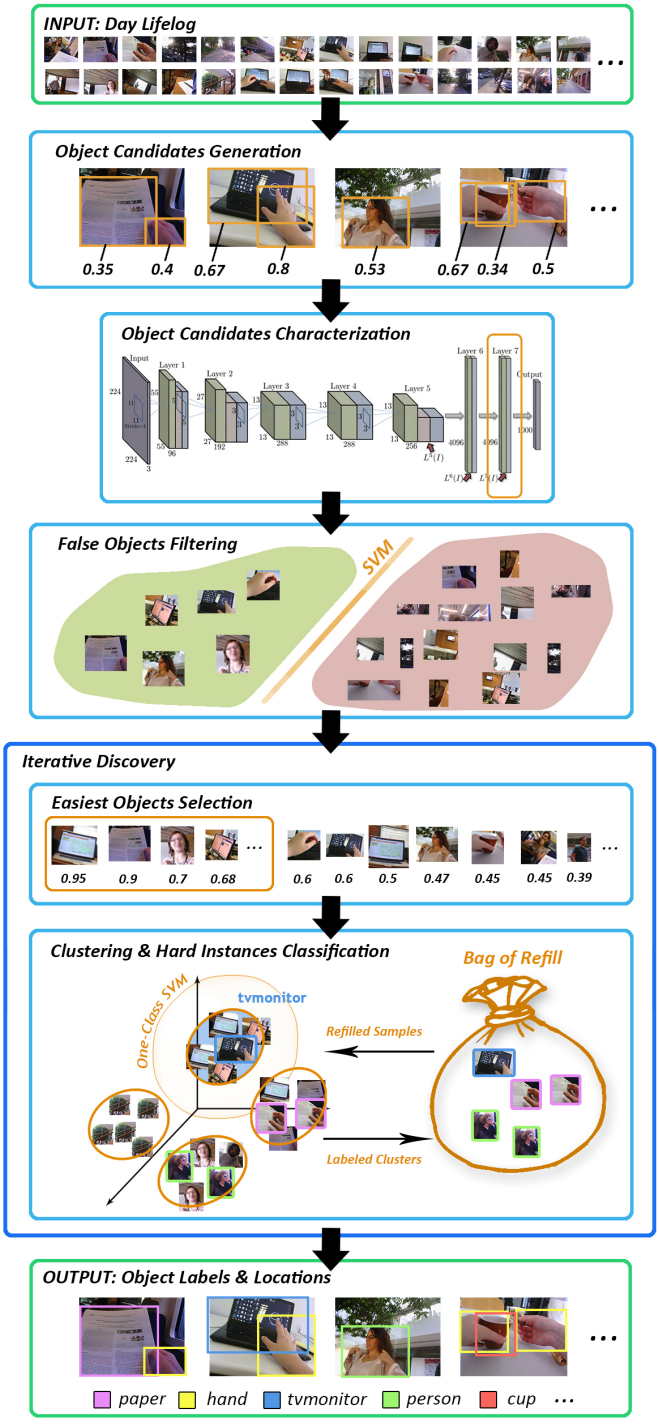

Fig. 2. Ego-Object Discovery methodology scheme. The different algorithms applied in each part of the methodology are represented in orange.

False Objects Filtering: The main drawback of most object detection methods is the huge number of FPs, they produce. Given that it is not enough to rely on the objectness score for discarding the 'No Object' instances, we filter the object candidates by an RBF-SVM classifier trained on CNN features to distinguish 'Object' vs. 'No Object' instances.

2.2. Iterative Discovery

Easiest Objects Selection: In order to achieve an iterative easy-first discovery, we used their associated objectness score to decide if a candidate ω is considered in the current iteration:

$$objectnessScore(\omega) > \mu + \omega_1\sigma - \omega_2 t, \quad (1)$$

where μ and σ are respectively, the mean and the standard deviation of all scores, t is the current iteration, and ω_1 and ω_2 are weights. This easiness measure seems a promising method for obtaining object candidates in general. However, this technique does not obtain the same results in egocentric datasets than in intentional images due to the fact that images are not captured by a person looking at objects of the world, but are acquired non-intentionally while a person is loosely wearing the camera. As a result of the inherent low frequency of appearance of different objects of the real world, to the limited image quality of the wearable egocentric devices and to the constant moving of the user, a great part of the photos are unclear, dark or blurry (see Fig. 1). All this causes lower precision, when clustering the obtained object candidates.

Refill Strategy: In order to solve these problems, we define a "refill" methodology as follows: at each iteration, the set of selected easiest samples is completed with a certain percentage (w.r.t. the number of easy samples retrieved) of samples from the Bag of Refill, which are randomly chosen labeled samples distributed on the already discovered object classes. In this way, we address two problems: 1) difficulty to form a cluster from a very small set of class instances, and 2) difficulty to link samples of the same class that were blurry and unclear. So, refilling the space with more samples of the same class of objects, we can obtain more compact clusters (see Fig. 3 and Fig. 4).

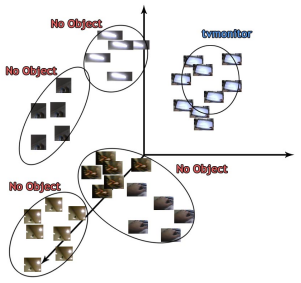

Fig. 3. Clusters formed by the easiest samples.

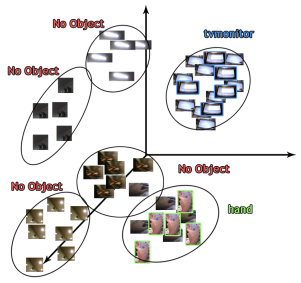

Fig. 4. Clusters formed by the refilled and easiest samples.

Clustering and Hard Instances Classification: In this step we apply an Agglomerative Ward clustering on the object candidates. Moreover, once the clusters are formed, we get the Silhouette Coefficient (Tan and Steinbach, 2011) on each cluster and select the best for the user to assign it a label. This coefficient is only calculated on the unlabeled samples, never using the refilled ones for selecting the most reliable cluster. At the end of each iteration, a OneClass-SVM for searching for harder instances is built with the new cluster and the rest of the easy samples are classified.

3. Results

In this section, we discuss the three datasets we used (summarizing their characteristics in Table 1), and expose the different tests applied to illustrate the EOD performance.

3.1. Datasets

Due to the low number of publicly available egocentric datasets and the complete lack of egocentric object-labeled datasets, we considered very important to construct one and make it public in order to serve as a base for algorithms comparison for the egocentric community.

The **Egocentric Dataset of the University of Barcelona (EDUB)** (see Fig. 5) is a dataset composed of 4912 images acquired by 4 people using the Narrative wearable camera (www.getnarrative.com). It is divided in 8 different days, 2 days per person. The objects appearing in the images were segmented using the online tool LabelMe (Russell et al., 2008) (although here we only use their bounding box) and their annotation files are similar to the ones provided by PASCAL. EDUB includes the following classes (number of samples per class are given in parenthesis): 'lamp' (2299), 'tvmonitor' (1274), 'hand' (1232), 'person' (1175), 'glass' (831), 'building' (732), 'face' (565), 'aircon' (530), 'sign' (506), 'cupboard' (392), 'paper' (377), 'car' (315), 'bottle' (260), 'door' (199), 'chair' (179), 'mobilephone' (145), 'window' (138), 'dish' (65), 'motorbike' (64), 'bicycle' (12), and 'train' (4). Note that in our tests, we did not use the classes with few instances (i.e. smaller than 100), considering that it would not be possible to discover them with a clustering strategy.

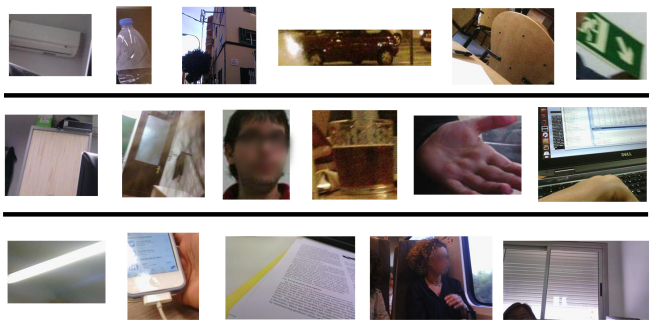

Fig. 5. Object candidates obtained by the Ferrari's objectness detector on the EDUB dataset. From left to right and top to bottom: aircon, bottle, building, car, chair, sign, cupboard, door, face, glass, hand, tvmonitor, lamp, mobilephone, paper, person, window.

The second of the datasets we considered is the **PASCAL VOC 2012** (Everingham et al., 2012), being one of the most widely used in object detection/recognition research , with very difficult and challenging images. We used the 'trainval' (for having more samples) set of images for our tests, but previously deleted the images that had in common with its 2007 version. We applied this pre-processing to avoid any bias in the results, since some of the used object detection methods were trained using PASCAL VOC 2007.

Table 1. Image/object characteristics for each of the used datasets.

	images	*object candidates*	*GT objects*	*classes*
MSRC	3,427	171,350	4,217	16
PASCAL	16,369	818,450	38,144	20
EDUB	4,912	245,600	11,149	17

The last of the datasets, we chose is the **Microsoft Research Cambridge (MSRC)** (Lee and Grauman, 2005), which was also used in (Lee and Grauman, 2011) for object discovery, and therefore will ease the comparison of the results. Considering that MSRC dataset is labeled at pixel level, we had to extract the bounding boxes corresponding to each of the objects making some assumptions: 1) the bounding box for an object is the minimal closing box around all the connected pixels that belong to the same class; 2) given the dataset is split in folders, we only considered valid the objects with the same class as the folder's name; 3) the minimal area for an object to be valid was set to 50x50 image pixels (about 0.81% of the whole image); and 4) we excluded the labels 'grass', 'sky', 'mountain', 'water' and 'road', because they are not objects, but rather environments.

Fig. 1 and 6 show some image samples from the 3 datasets. MSRC dataset, compared to the other two should obtain better results due to the position of the objects (central to the image) and their clear appearance. Even though in general PASCAL has some object instances very difficult to find, the hardest one is the EDUB (also considering the high rate of objects occlusions, blurriness and lower image quality).

Fig. 6. MSRC image samples (top) and PASCAL 12 samples (bottom).

3.2. Object Detection Methods

Given that the first step of the algorithm is to obtain object candidates from the images, we tested and compared four different state of the art object detection methods on the three datasets (see Table 2). We chose Objectness (Alexe et al., 2010), BING (Cheng et al., 2014), Multiscale Combinatorial Grouping (MCG) (Arbeláez et al., 2014) and Selective Search (Uijlings et al., 2013) methods considering their good performances. For MCG, we applied its quickest, but less exhaustive version.

Due to the dramatic increase of space needed to store all the samples[1], we extracted the top $W = 50$ object candidates per image sorted by their objectness score.

Analyzing the percentage of NO (see overlapping score in section 3.3) and DR of each method, we can see that the DR is not as high as desired and the % of NO is remarkably high. Meaning that using any of the best state of the art approaches for object detection makes us lose a lot of information, so we have to consider that our final results will be inevitably biased and worsen for this reason.

[1]Considering the PASCAL 12 dataset, we needed nearly 30GB of data to store all the images and features for the tests

Table 2. Percentage of 'No Objects' (NO) (or of False Positives) and Detection Rate (DR) comparison of the four object detection methods on our three datasets.

		Objectness	*BING*	*MCG*	*Sel.Search*
MSRC	NO	91.69	96.68	**48.42**	61.95
	DR	**88.83**	64.15	79.61	70.98
PASCAL	NO	92.14	92.93	**65.16**	71.30
	DR	**60.47**	56.93	49.36	36.71
EDUB	NO	92.75	95.43	**79.17**	84.27
	DR	**60.45**	50.00	49.57	29.09

Comparing the different datasets, as one could immediately expect looking at the images, it is clearly easier for any objectness measure to get good results on the MSRC dataset, meanwhile it is quite more difficult on PASCAL and EDUB, having an extra difficulty for the second one due to the non-intentional acquisition and less clear images of the wearable cameras.

Given our final goal of being able to discover the true distribution of object classes and as many individual GT objects as possible, we considered that the objectness measure that obtained better results for EOD was the one proposed by (Alexe et al., 2010), because we are interested in getting most of the GT objects in the dataset, even if we have to deal with a lot of NO (i.e. noisy or FP) instances.

3.3. Experimental Setup

In order to perform the methodology validation, we first **leave a 50% of the object classes in the unlabeled pool** as a test set. Note that we need to test if the algorithm is able to discover unseen object classes. From the remaining part of classes, similar to (Lee and Grauman, 2011), we separated a 40% of the total object candidates to represent the initial knowledge located into the bag of refill and used the remaining 60% for testing, too.

In order to say that a candidate matches a GT object bounding box, we followed the PASCAL VOC challenge criterion, that uses the Overlapping Score (OS). Given a window region ω produced by the object detector, is considered a hit on a GT label, iff:

$$OS = \frac{|GT \cap \omega|}{|GT \cup \omega|} > 0.5 \qquad (2)$$

Due to the challenging images presented to the object detector, a very high percentage of samples (more than 92% using Ferrari's objectness) could not be considered objects, and were labeled as NO.

In order to tune the parameters for the SVM filter strategy for each of the datasets, we applied a nested 5-fold cross-validation with 5 test divisions with a grid of parameters of $\sigma \in \{0.1, 0.5, 3, 10, 100, 1000\}$ and $C \in \{0.1, 0.5, 3, 10, 100, 1000\}$. All the tests were performed for each dataset separately and on a

randomly selected fraction of its samples to save computational time. With these tests, we finally found that the best parameters for filtering as many NO instances and at the same time keeping as many 'Object' instances as possible (high sensitivity and high specificity) for both the PASCAL and the MSRC classifiers were $\sigma = 100$ and $C = 3$. In the labeling step, for simulation purposes, we labeled the best cluster with a majority voting strategy w.r.t the GT, although this labeling is intended to be made by the camera user his-/herself.

We designed different test settings to evaluate our proposal:

S1: Features of (Lee and Grauman, 2011).

S2: CNN object features.

S3: CNN object features with Refill strategy.

S4: CNN object concatenated with CNN scene features and Refill strategy.

S5: CNN object features with Refill and SVM filter.

S6: CNN object features with Refill, SVM filter and PCA.

With the first pair of settings, we intend to compare the generalization capabilities of the appearance features from (Lee and Grauman, 2011) against the extracted CNN features. In setting S4, we tested adding a context about the scene, and in setting S6, we applied a PCA feature dimensionality reduction and transformation in case there is redundancy in the extracted CNN features.

3.4. Silhouette Coefficient Comparison

In order to check if the clusters formed by using CNN features are more robust than the ones formed by using the features from (Lee and Grauman, 2011), we can analyse the mean silhouette coefficient values obtained in several iterations. In Fig. 7 we plot the difference on the silhouette coefficient values obtained by using the two kind of features. The comparison is applied for the top 15 clusters on the first 50 iterations of the algorithm.

We can immediately realise that the average compactness of the clusters and their difference to the other clusters (which is what Silhouette Coefficient measures) is always higher when using CNN features, and will lead to get purer clusters and a better labeling.

3.5. F-Measure Comparison on EDUB

To evaluate our approach, we used the F-Measure, because it objectively penalizes the FP and FN objects in each class, that is, represents a trade-off between the Precision and Recall of the method. At the same time, we want to give the same importance to all classes, and are interested in finding as many different classes as possible, but always leaving the NO instances aside, without considering them into the quality measures. Hence, we applied the average per-class precision and recall defined in

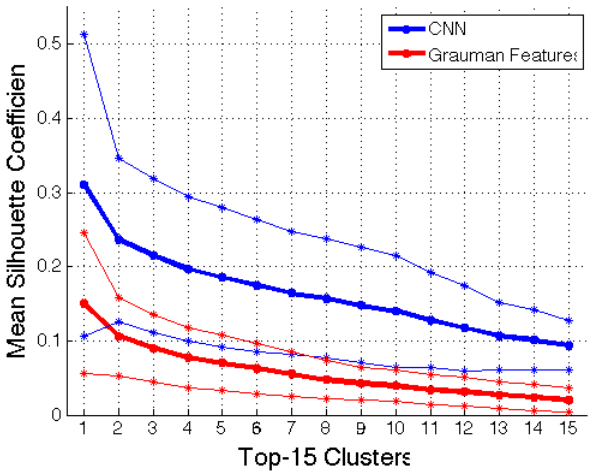

Fig. 7. Comparison of mean silhouette coefficient (thick lines) and standard deviation (thin lines) for the top 15 clusters on 50 algorithm iterations (high values are better).

(Sokolova and Lapalme, 2009) in order to obtain the average F-Measure:

$$\text{F-Measure} = 2\frac{Precision_M * Recall_M}{Precision_M + Recall_M}, \tag{3}$$

where $Precision_M$ and $Recall_M$ are the mean precision and recall of all classes, giving the same weight to all of them.

All measures were averaged by at least 5 executions per setting and for a maximum of 100 algorithm iterations. Using these tests, we compared all settings at the end of the easiest samples discovery (Fig.8) and on each iteration (Fig.9).

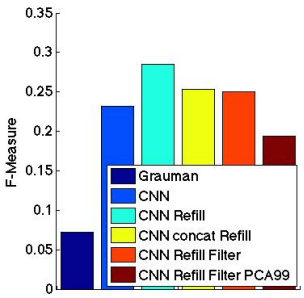

Fig. 8. Final F-Measure for each setting.

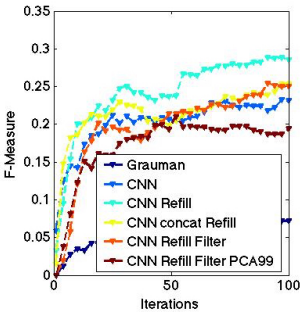

Fig. 9. F-Measure evolution for each different setting.

Looking at Fig.8, we can clearly see that using CNN outperforms the features of (Lee and Grauman, 2011), indicating that they can form purer clusters and find a wider variety of classes thanks to their best representation. Then, adding the Refill technique, the EOD method outperforms the one using the CNN features only. The rest of the methods can not reach the same results as CNN + Refill. Moreover, using the additional CNN features of the whole image adds just noise to the set of features. That is, simply by using the CNN with the bounding box of the object candidate already captures the closest and most relevant object context. Considering the high dimensionality of CNN features, it seems that including a PCA dimensionality

reduction to the data does not provide any benefit to the object discovery.

Comparing the evolution of the F-Measure through the iterations (Fig. 9), we see that any of the settings using CNN features experiments a much higher increase in the F-Measure value just in the first 5-10 iterations, meaning that they can find clusters of true objects quicker than using the setting S1.

Also, using the CNN features combined with the refill strategy, the results clearly improved from 0.072 to 0.285. This is caused by the discovery of different classes of samples. While when using the features of (Lee and Grauman, 2011), we are only able to discover 3 or 4 classes at most, achieving an average of 0.072 F-Measure; with the setting S3, we can discover instances of more than half of the classes, getting nearly 0.29 of F-Measure. Although on the EDUB using the setting S5 (CNN + Refill + SVM Filtering) does not seem to get as good F-Measure results as on the other settings, in other datasets, as we will be able to see, it outperforms or nearly reaches the results of setting S3. Furthermore, it gets a wider variety of object classes.

3.6. F-Measure Comparison on All Datasets

After having found the best combination of methods and parameters to use, we tested and compared how good the new method was contrasting it with the state of the art method (Lee and Grauman, 2011) for any of the datasets (EDUB, PASCAL 2012 and MSRC). In table 3, we can see a summary of the F-Measure results obtained for each of the datasets and each of the best test settings (average on at least 5 tests per setting).

Table 3. F-Measure comparison for the three datasets, the state of the art (Lee and Grauman, 2011) and our best test settings (CNN + Refill and CNN + Refill + Filter).

F-Measure	S1	S3 (ours)	S5 (ours)
MSRC	0.121	**0.431**	0.410
PASCAL	0.002	0.145	**0.179**
EDUB	0.072	**0.285**	0.250
Average	0.065	0.287	0.280

As we can see, using any of our best methods (either setting S3 or setting S5) clearly outperforms the state of the art features, having from a 350% to a 9000% of improvement depending on the dataset and the settings, and a 453% of average improvement with the best setting.

Even though the average F-Measure result obtained using the SVM filtering (setting S5) is worse than without it (setting S3), we must consider that these classifiers have been built with samples from different datasets than the ones on test (1/2 of the PASCAL samples for MSRC tests and all MSRC samples for both PASCAL and EDUB tests), meaning that the generalization will be poorer than if we built a general classifier with images from any of the datasets.

Another important consideration we must take into account, is that for the MSRC tests, although the final (after 100 iterations) F-Measure results are better without the filtering, in fact they were better with the filtering from the 1st to the 75th iteration, meaning that in some cases, it can offer better results if we want to stop early the discovery method.

3.7. Object Discovery Results

In this section, we analyze the object discovery results in more general terms. In Fig. 10, we can see the absolute number of object instances found by each of the methods compared to the GT and the ones found by the Objectness measure ((Alexe et al., 2010), in this case without counting repeated instances of the same object).

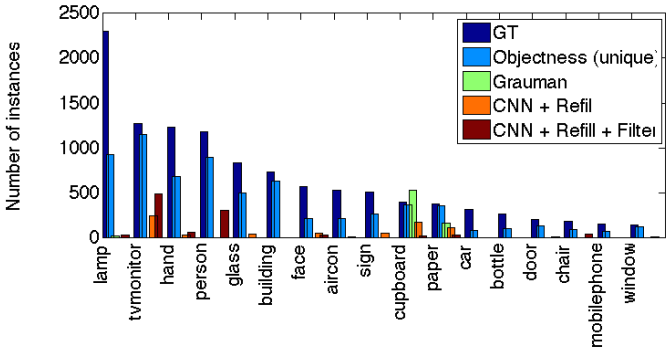

Fig. 10. Objects found by each method compared to the GT and the ones found by the Objectness measure (Alexe et al., 2010).

As we can see, using the parameters of setting S1 (Lee and Grauman, 2011), we are only able to find instances from 3 different classes, which causes the previously seen very low F-Measure results. On the other hand, using either CNN + Refill (setting S3) or CNN + Refill + Filter (setting S5), we can clearly discover objects from a wider variety of classes, which also causes the higher resulting F-Measure. Moreover, we get a wider variety of classes with setting S5 (10 different classes) than with setting S3 (8 different classes).

If we check the discovery order of the classes in each of the methods (see Fig. 12), we can see that some classes are more easily discovered and repeated over the following iterations than others. This is caused not only by the number of class instances appearing in the dataset, but also by the previously acquired knowledge (refill), the general method used, and/or the intra-class variability.

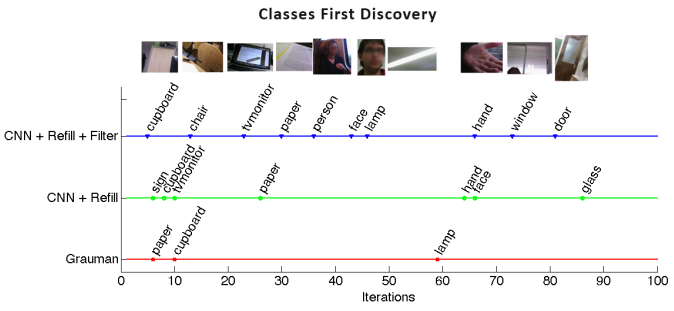

Fig. 12. First discovery of the object classes as a function of iterations.

Table 4. Number of clusters found for each class using any of the settings S1, S3 or S5.

Test	No Object	hand	lamp	cupboard	car	glass	chair	face	door	window	tvmonitor	building	paper	person	mobilephone	sign
S1	96	0	1	2	0	0	0	0	0	0	0	0	1	0	0	0
S3	71	1	0	3	0	1	0	6	0	0	8	0	4	0	0	3
S5	49	2	3	6	0	0	4	5	1	1	23	0	1	5	0	0

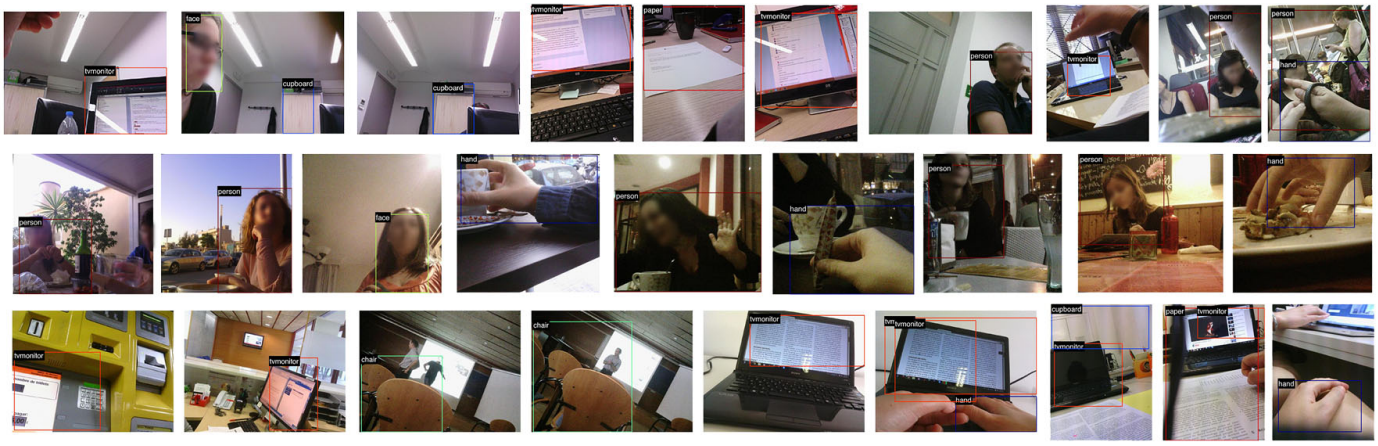

Fig. 11. Examples of discovered objects for three different subjects (one row each). Better viewed in digital format.

If we analyse the clusters number, where we find each class (see Table 4), we can see that even though having the same percentage of NO candidates (92.75%), using Grauman's features (setting S1), we get 96% of the clusters labeled as NO, but only 71% of them using CNN + Refill (setting S3). Then, comparing it when adding the SVM filtering (setting S5), we can see that it gets reduced to a 49% of the clusters thanks to the dramatic reduction of NO instances in the pool of unlabeled samples.

In Fig. 13, we can see the evolution of GT unique instances discovered by each of the methods on the accumulated iterations (each data point corresponds to an algorithm iteration) w.r.t. the F-Measure obtained by the method.

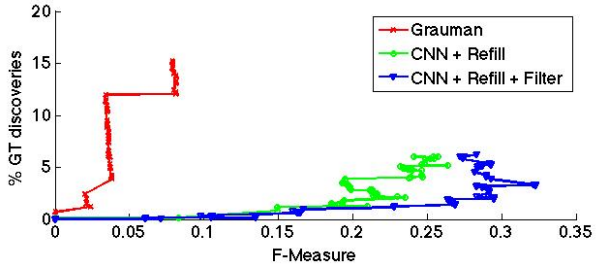

Fig. 13. Percentage of GT object discoveries accumulated on each iteration w.r.t. the F-Measure obtained.

We can see that using Grauman's features seems to cover a wider variety of object samples than either with settings S3 or S5 (about 16% against about 6-7% of the GT samples). This result is probably directly related to the lower F-Measure obtained. Due to the lower generalization and representation capabilities of the set of features used (compared to CNN), the labeled clusters contain a wider variety of samples and objects, causing to label more unique object instances, but at the same time having a worse average result.

In Fig. 11 there are some examples of objects discovered by our methodology. We can see that it is able to discover instances of the same classes even having a high intra-class variability (person or hand). Note that some samples are not yet discovered due to the limited number of iterations applied (100).

Regarding the complexity of EOD, it is easy to see that (independently to the length of our feature vectors):

- The objectness score extraction is of complexity $O(N)$, being N the number of images in the dataset;
- The SVM filtering has complexity $O(N)$;
- The sorting of easiest objects is $O(N*Wlog(N*W))$, being W the number of candidates extracted for each image;
- The refill strategy is $O(1)$;
- The CNN features extraction is $O(M)$, being M the easy objects number in the current iteration;
- The clustering of easy objects is $O(M^2)$;
- The best cluster labeling is $O(1)$;
- The one-class SVM cost is $O(M)$.

Leading in total a cost of $O(N*Wlog(N*W)+M^2)$, for each iteration.

4. Conclusions

In this paper, we proposed a novel semi-supervised object discovery algorithm for egocentric data that relies on features extracted from a pre-trained CNN and uses a refill strategy for finding easily the classes with less samples. Moreover, we added a SVM filtering strategy for discarding a great part of the high amount of 'No Object' classes produced by any of the

objectness measures. We compared 4 of the state of the art objectness measures in terms of 'No Object' instances produced and the Detection Rate obtained when extracting a low number of object candidates (W=50). We proved that the CNN features, the refill strategy (and the SVM filtering) can produce much better F-Measure results and can discover a larger number of infrequent classes than the state of the art approach on three datasets (MSRC, PASCAL 12 and EDUB), either being from general easy images, to egocentric and very difficult ones. Furthermore, we proved that this combined strategy also works better than the previous ones for very noisy and blurry images.

5. Future Work

Our future work involves the following tasks:

1. Define an algorithm to discover objects, scenes and people to characterize the environment of the persons wearing the camera,
2. Propose an iterative and combined scene and object discovery to take profit of the samples discovered from the complementary categories, and
3. Make the method discriminative i.e. to detect which are the objects and scenes that characterize the environment of a person and distinguish them with respect to those of the other people.

Acknowledgments

This work was partially founded by the projects TIN2012-38187-C03-01 and SGR 1219.

References

Alexe, B., Deselaers, T., Ferrari, V., 2010. What is an object?, in: CVPR, Conference on, IEEE. pp. 73–80.

Arbeláez, P., Pont-Tuset, J., Barron, J.T., Marques, F., Malik, J., 2014. Multiscale combinatorial grouping, CVPR.

Betancourt, A., Morerio, P., Regazzoni, C.S., Rauterberg, M., . The evolution of first person vision methods: A survey .

Bolaños, M., Garolera, M., Radeva, P., 2015. Object discovery using cnn features in egocentric videos, in: Iberian Conference on Pattern Recognition and Image Analysis (in press). Springer.

Chatzilari, E., Nikolopoulos, S., Papadopoulos, S., Zigkolis, C., Kompatsiaris, Y., 2011. Semi-supervised object recognition using flickr images, in: CBMI, 2011 9th International Workshop on, IEEE. pp. 229–234.

Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P., 2014. Bing: Binarized normed gradients for objectness estimation at 300fps, in: IEEE CVPR.

Cho, M., Kwak, S., Schmid, C., Ponce, J., 2015. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. arXiv preprint arXiv:1501.06170 .

Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A., 2012. The pascal visual object classes challenge 2012 (VOC2012) results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

Fathi, A., Ren, X., Rehg, J.M., 2011. Learning to recognize objects in egocentric activities, in: CVPR, 2011 IEEE Conference On, IEEE. pp. 3281–3288.

Goodfellow, I.J., Bulatov, Y., Ibarz, J., Arnoud, S., Shet, V., 2014. Multi-digit number recognition from street view imagery using deep convolutional neural networks. Google Inc., Mountain View, CA .

Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K., 2006. Sensecam: A retrospective memory aid, in: UbiComp 2006: Ubiquitous Computing. Springer, pp. 177–193.

Honglak, L., Roger, G., Rajesh, R., Andrew Y., N., 2009a. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. Computer Science Department, Stanford University, Stanford .

Honglak, L., Yan, L., Rajesh, R., Peter, P., Andrew Y., N., 2009b. Unsupervised feature learning for audio classification using convolutional deep belief networks. Computer Science Department, Stanford University, Stanford .

Jia, Y., 2013. Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/.

Kading, C., Freytag, A., Rodner, E., Bodesheim, P., Denzler, J., 2015. Active learning and discovery of object categories in the presence of unnameable instances, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4343–4352.

Kang, H., Hebert, M., Kanade, T., 2011. Discovering object instances from scenes of daily living, in: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE. pp. 762–769.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: NIPS, pp. 1097–1105.

Lazebnik, S., Schmid, C., Ponce, J., 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: CVPR, Computer Society Conference on, IEEE. pp. 2169–2178.

Lee, Y.J., Grauman, K., 2005. Microsoft research cambridge object recognition image database. http://research.microsoft.com/en-us/downloads/b94de342-60dc-45d0-830b-9f6eff91b301/default.aspx.

Lee, Y.J., Grauman, K., 2011. Learning the easy things first: Self-paced visual category discovery, in: CVPR, Conference on, IEEE. pp. 1721–1728.

Liu, D., Chen, T., 2007. Unsupervised image categorization and object localization using topic models and correspondences between images, in: ICCV, 11th International Conference on, IEEE. pp. 1–7.

Michael, K., 2013. Wearable computers challenge human rights. ABC Science .

Min, W., Li, X., Tan, C., Mandal, B., Li, L., Lim, J.H., 2014. Efficient retrieval from large-scale egocentric visual data using a sparse graph representation, in: CVPRW, 2014 Conference on, IEEE. pp. 541–548.

Moghimi, M., Azagra, P., Montesano, L., Murillo, A.C., Belongie, S., 2014. Experiments on an RGB-D wearable vision system for egocentric activity recognition. 3rd Workshop on Egocentric (First-person) Vision, CVPR .

Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J., Zisserman, A., 2006. Using multiple segmentations to discover objects and their extent in image collections, in: CVPR, Computer Society Conference on, IEEE. pp. 1605–1614.

Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T., 2008. Labelme: a database and web-based tool for image annotation. International journal of computer vision 77, 157–173.

Schulter, S., Leistner, C., Roth, P., Bischof, H., 2013. Unsupervised object discovery and segmentation in videos, in: Proceedings on BMVA, BMVA Press. pp. 53.1–53.12.

Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T., 2005. Discovering objects and their location in images, in: ICCV, Tenth International Conference on, IEEE. pp. 370–377.

Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Information Processing & Management 45, 427–437.

Tan, P., Steinbach, M.K., 2011. Introduction to data mining.

Tuytelaars, T., Lampert, C.H., Blaschko, M.B., Buntine, W., 2010. Unsupervised object discovery: A comparison. IJCV 88, 284–302.

Uijlings, J.R., van de Sande, K.E., Gevers, T., Smeulders, A.W., 2013. Selective search for object recognition. IJCV 104, 154–171.

VISUAL SUMMARY OF EGOCENTRIC PHOTOSTREAMS BY REPRESENTATIVE KEYFRAMES

Marc Bolaños[1], *Ricard Mestre*[2], *Estefanía Talavera*[1], *Xavier Giró-i-Nieto*[2*], *Petia Radeva*[1,3†]

[1]Universitat de Barcelona, Barcelona, Catalonia/Spain
{marc.bolanos,etalavera,petia.ivanova}@ub.edu
[2]Universitat Politécnica de Catalunya, Barcelona, Catalonia/Spain
xavier.giro@upc.edu
[3]Computer Vision Center, Bellaterra, Catalonia/Spain

ABSTRACT

Building a visual summary from an egocentric photostream captured by a lifelogging wearable camera is of high interest for different applications (e.g. memory reinforcement). In this paper, we propose a new summarization method based on keyframes selection that uses visual features extracted by means of a convolutional neural network. Our method applies an unsupervised clustering for dividing the photostreams into events, and finally extracts the most relevant keyframe for each event. We assess the results by applying a blind-taste test on a group of 20 people who assessed the quality of the summaries.

Index Terms— egocentric, lifelogging, summarization, keyframes

1. INTRODUCTION

Lifelogging devices offer the possibility to record a rich set of data about the daily life of a person. A good example of this are wearable cameras, that are able to capture images from an egocentric point of view, continuously and during long periods of time. The acquired set of images comes in two formats depending on the device used: 1) high-temporal resolution videos, which usually produce more than 30fps and capture a lot of dynamical information, but they are only capable of storing some hours of data, or 2) low-temporal resolution photostreams, which usually produce only 1 or 2 fpm, but are able to capture events that happen during a whole day (having around 16 hours of autonomy).

Being able to automatically analyze and understand the large amount of visual information provided by these devices would be very useful for developing a wide range of applications. Some examples could be building a nutrition diary based on what, where and in which conditions the user eats for keeping track of any possible unhealthy habit, or providing an automatic summary of the whole day of the user for offering a memory aid to mild cognitive impairment (MCI) patients by reactivating their memory capabilities [1].

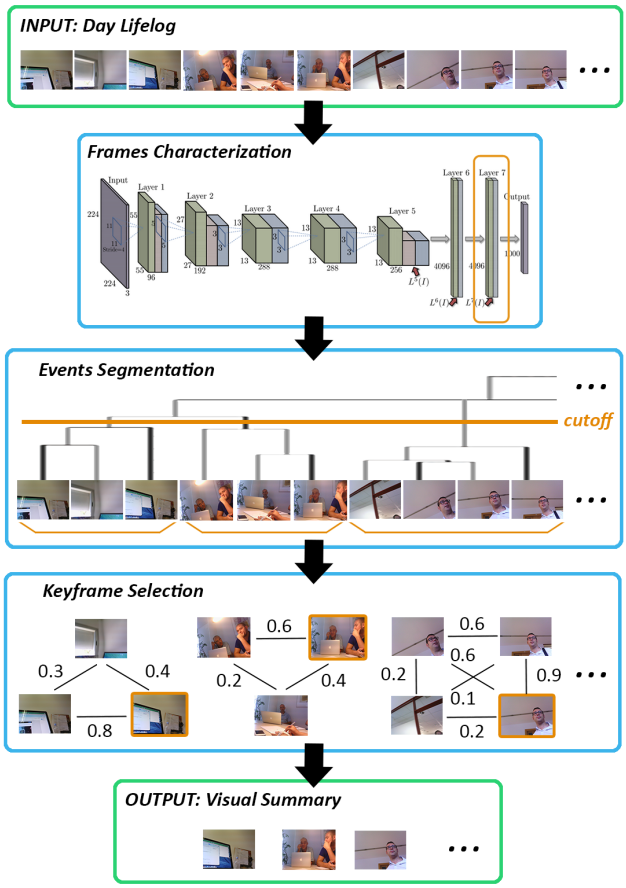

Fig. 1. Scheme of the proposed visual summarization.

*This work has been developed in the framework of the project BigGraph TEC2013-43935-R, funded by the Spanish Ministerio de Economa y Competitividad and the European Regional Development Fund (ERDF). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GeoForce GTX Titan Z used in this work.

†This work has been partially founded by the projects TIN2012-38187-C03-01 and SGR1219. We acknowledge the support and collaboration of Maite Garolera.

In order to take into account our ultimate goal, we propose an approach that starts by extracting a set of features for frames characterization by means of a convolutional neural network. These visual descriptors are used to segment events by running an agglomerative clustering, which is post-processed to guarantee a temporal coherency (similar to [2]). Finally, a representative keyframe for each event is selected using the Random Walk [3] or Minimum Distance [4] algorithms. The overall scheme is depicted in Figure 1.

This paper is structured as follows. Section 2 overviews previous work for event segmentation and summarization in the field of egocentric video. Our approach is described in Section 3 and its quantitative and qualitative evaluation in Section 4. Finally, Section 5 draws the final conclusions and outlines our future work.

2. RELATED WORK

The two main problems addressed in this paper, event segmentation and summarization, have been addressed in related egocentric data works, as presented in this section.

2.1. Egocentric event segmentation

Most existing techniques agree that the first step for a summary construction is a shot- or event-based segmentation of the photostream or video. Lu and Grauman in [5] and Bolaños et al. in [6] both propose event segmentation that relies on motion information, colour and blurriness, integrated in an energy-minimization technique. The result is final event segmentation that is able to capture the different motion-related events that the user experiences. In the former approach [5], the authors use high-temporal videos and an optical flow descriptor for characterizing the neighbouring frames. In the latter one [6], instead of working with low-temporal data, a SIFT-flow descriptor is used, as it is more robust for capturing long-term motion relationships. Poleg et al. [7] also propose motion-based segmentation, but they use a new method of Cumulative Displacement Curves for describing the motion between neighbouring video frames. The proposed solution is able to focus on the forward user movement and removes the noise of the head motion produced by head-mounted wearable cameras. Other methods have been proposed using low-level sensor features like the work in [8] that splits low-temporal resolution lifelogs in events. Lin and Hauptmann [2] also propose a simple approach based on using colour features in a Time-Constrained K-Means clustering algorithm for keeping temporal coherence. In [9], Talavera et al. design a segmentation framework also based on an energy minimization framework. In this case, the authors offer the possibility to integrate different clustering and segmentation methods, offering more robust results. Considering the main goal of our approach is providing a good keyframe selection following also the timeline of the events, we followed the results provided by Talavera et al. using agglomerative clustering at the same time that we applied a method for having time-constrained results with comparable results to Lin et al.

2.2. Egocentric summarization

Focusing on the summarization of lifelog data after event segmentation, there are two basic research directions, both of them aiming at removing those data, which are redundant or low-informative. In the case of video recordings (high-temporal resolution), it is a common practice to select a subset of video segments/subshots to create a video summary. On the other hand, when working like in our case with devices that take single pictures at a low frame rate, the problem is usually tackled by selecting the most representative keyframes due to temporal resolution constraints. The most relevant work in the literature following the video approach is from Grauman et al. in [5, 10], where a summary methodology for egocentric video sequences is proposed. The authors rely on an initial event segmentation, followed by the detection of salient objects and people, create a graph linking events and the important objects/people, and finish with a selection of a subset of the events of interest. This final selection is based on combining three different measures: 1) *Story* (choosing a set of shots that are able to follow the inherent story in the dataset), 2) *Importance* (aimed at choosing only shots that show some important aspect of the day) and 3) *Diversity* (adding a way to avoid repeating similar actions or events in the summary). When considering the keyframe selection approach, one of the most relevant works is by Doherty et al. [4], where the authors study various selection methods like: 1) getting the frame in the middle of each segment, 2) getting the frame that is the most similar w.r.t. the rest of the frames in the event, or 3) selecting the closest frame to the event average. In our approach, considering the kind of data used, we are closer to Doherty's et al. work, but at the same time we considered of a great importance the evaluation method applied by Grauman et al. (see section 4.3).

3. METHODOLOGY

This section presents our methodology for keyframe-based summarization of egocentric photostreams, depicted in Figure 1. We start by characterizing each of the lifelog frames with a global scale visual descriptor. These features are used to create a visual-based event segmentation, which incorporates a post-processing step to guarantee time consistency. Finally, the most visually repetitive frame is selected as the most representative of the event.

3.1. Frames characterization

Convolutional Neural Networks (convnets or CNNs) have recently outperformed hand-crafted features in several computer vision tasks [11, 12]. These networks have the ability to learn sets of features optimised for a pattern recognition problem described by a large amount of visual data.

The last layer of these convnets is typically a soft-max classifier, which in some works is ignored, and the penultimate fully connected layers are directly used as feature vectors. These visual features have been successfully used as any other traditional hand-crafted features for purposes such as image retrieval [13] or classification [14].

In the field of egocentric video segmentation, convnets have also been proved as suitable for clustering purposes [9]. For this reason, we used a set of features extracted by means of the pre-trained *CaffeNet* convnet included in the Caffe library [15]. This convnet was inspired by [12] and trained on ImageNet [16]. In our case, we used as features the output of the penultimate layer, a fully connected layer of 4,096 components, discarding this way the final soft-max layer, which was intended to classify 1,000 different semantic classes from ImageNet.

3.2. Events segmentation

The egocentric photostream is segmented with an unsupervised hierarchical Agglomerative Clustering (AC) [17] based on the convnet visual features. As proved in [9], this clustering methodology reaches a reasonable accuracy for this task. In this way, we can define sets of images, each of them representing a different event. AC algorithms can be applied with different similarity measures. Different configurations were tested (see details in Section 4.2) and the best approach was obtained with the *average* linkage method with Euclidean distance. This option determines the two most similar clusters to be fused in each iteration using the following distance:

$$arg\,min_{C_i,C_j \in \mathbf{C_t}}\ D(C_i, C_j),\ \text{where} \tag{1}$$

$$D(C_i, C_j) = \frac{1}{|C_i| \times |C_j|} \sum_{s_{k,i} \in C_i, s_{l,j} \in C_j} \sqrt{f(s_{k,i})^2 - f(s_{l,j})^2}, \tag{2}$$

where $\mathbf{C_t}$ is the set of clusters at iteration t, $s_{k,i}$ and $s_{l,j}$ are the samples in cluster C_i and C_j, respectively, and $f(s)$ are the visual features extracted by means of the convnet.

However, creating the clusters based only on visual features often generates non-consistent solutions from a temporal perspective. Typically, images captured in the same scenario will be visually clustered as a single event despite corresponding to separate moments. For example, frames from the beginning of the day, (e.g. when the user takes the train for commuting to work) may be visually indistinguishable with other frames from the end of the day (e.g. when the user is going back home by train too). Additionally, another usual problem when relying only on visual features is that sometimes very small clusters can be generated, a result which should be avoided because an event is typically required to have a certain span in time (e.g. 3 minutes, in our work).

In order to solve these problems, we introduce two post-processing steps for refining the resulting clusters: *Division* and *Fusion*. The *Division* step splits in different events those images in the same cluster which are temporally interrupted by events defined in other clusters. For example, the event in orange from Figure 2 a) is divided in two events (orange and yellow) in Figure 2 c) due to a *Corridor scene* event (in green) interrupting the original *Office scene*. On the other hand, the second post-processing step, *Fusion*, will merge all those events shorter than a threshold with the closest neighboring event in time.

3.3. Keyframe selection

Once the photostream is split into the events, the next step is to carefully select a good subset of keyframes. To do so, we explored two different methods: *random walk* and a *minimum distance* approach. Both approaches are based on the assumption that the best photo to represent the event is the one, which is the most visually similar with the rest of the photos in the same cluster. As a result, each event can be automatically represented by a single image and, when all images combined, they will provide a visual summary of the user's day.

3.3.1. Random Walk

We propose to use the Random Walk algorithm [3] in each of the events, separately. As a result, the algorithm will select the photo, which is more visually similar to the rest of the photos in the event. After applying the same procedure for all the events, we can have a good general representation of the main events that happened in the user's daily life.

The Random Walk algorithm works as follows: 1) the visual similarity for each pair of photos in the event is computed; 2) a graph described by a transitional probabilities matrix is built using the extracted similarities as weights on each of the edges; 3) the matrix eigenvectors are obtained, and 4) the image associated to the largest value in the first eigenvector is considered as the keyframe of the event.

3.3.2. Minimum distance

The second considered option selects the individual frame with the minimal accumulated distance with respect to all the other images in the same event. That is, let us consider the adjacency matrix $A = \{a_{i,j}\} = \{d_{s_i,s_j}\}$, where d_{s_i,s_j} is the Euclidean distance between the descriptors of images s_i and s_j extracted by the convnet, $i = 1, ...N$, $j = 1, ...N$, where N is the number of frames of the event. Let us consider the vector $v = (\sum_j a_{i,j})$ of accumulated distances. One can easily see that the index of the minimal component of vector v i.e. $k = argmin_i\{v_i\},\ i = 1, ...N$ determines the closest frame to the rest of frames in the corresponding event with respect to the L_1 norm [4].

4. RESULTS

This section presents the quantitative and qualitative experiments run on a home-made egocentric dataset to assess the performance of the presented technique.

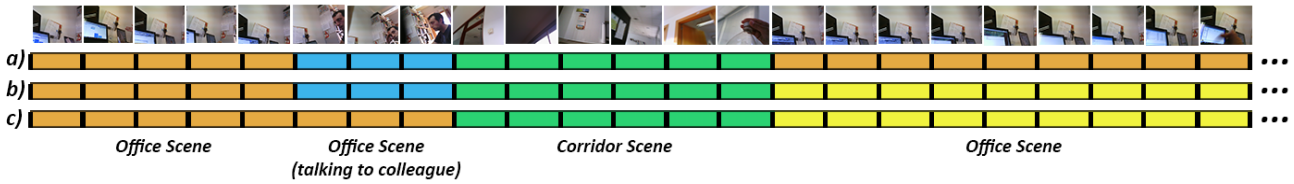

Fig. 2. Example of the events labeling produced by a) simply using the AC algorithm, b) applying the division strategy and c) additionally applying the fusion strategy. Each color represents a different event.

4.1. Dataset

Our experiments were performed on a home-made dataset of images acquired with a Narrative (`www.getnarrative.com`) wearable camera. This device is typically clipped on the users' clothes under the neck or around the chest area. The dataset, we used, is a subset of the one used by the authors in [9] (not using the SenseCam sets). It is composed of 5 day lifelogs of 3 different persons and has a total of 4,005 images. Furthermore, it includes the ground truth (GT) events segmentation for assessing the clustering results.

4.2. Quantitative evaluation of event segmentation

The first test assessed the quality of the photostream segmentation into events. In order to make this evaluation, we used the Jaccard Index, which is intended to measure the overlap of each of the resulting events and the GT the following way:

$$J(E, GT) = \sum_{e_i \in E} \sum_{g_j \in GT} M_{ij} \frac{e_i \cap g_j}{e_i \cup g_j}, \quad (3)$$

where E is the resulting set of events, GT is the ground truth, e_i and g_j are a single event and a single GT segment respectively, and M_{ij} is an indicator matrix with values 1, iff e_i has the highest match with g_j.

We compared different cluster distance methods with respect to the chosen cut-off parameter (which determines how many clusters are formed considering their distance value) for the AC (see Figure 3). We choose the "average" with cutoff $= 1.154$ as the best option and, with this configuration, we measured the gain of introducing the Division-Fusion strategy, illustrated in Figure 4.

4.3. Qualitative evaluation with blind-test taste

The assessment of visual summaries of a day, like the one shown in Figure 5, is a challenging problem, because there is not a single solution for it. Different summaries of the same day may be considered equally satisfactory due to near duplicate images and subjectivity in the judgments. Therefore, we followed an evaluation procedure similar to the one adopted by Lu and Grauman [5]. We designed a blind-taste test and asked to a group of 20 people to rate the output of different

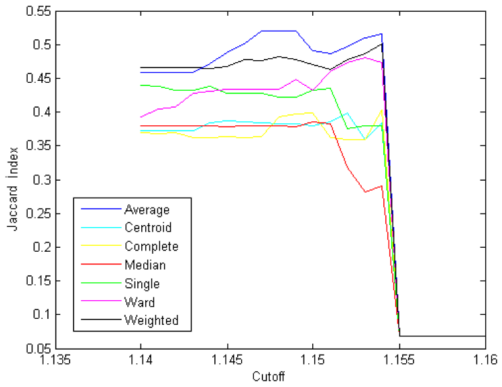

Fig. 3. Average Jaccard index value obtained for the 5 sets. We compare each of the methods after applying the division-fusion strategy with respect to the best cut-off AC values.

solutions, without knowing which of them corresponded to each configuration.

4.3.1. Keyframe selection

The first qualitative evaluation focused on the keyframe selection strategy, comparing both presented algorithms (*Random Walk* and *Minimum Distance*) with a third one, *Random Baseline*. In this first part, the three selection strategies were applied on each of the events defined by the GT annotation.

On the first part, we showed to the user a complete event according to the GT labels and, afterwards, the three keyframes selected by the three methods under comparison in a random sorting[1]. Then, the user had to answer if each of the candidates was representative of the current event (results in Figure 6), and also choose which of them was the best one (results in Figure 7). This procedure was applied on each of the events of the day and results averaged per day.

Scoring results presented in Figures 6 and 7 indicate how both proposed solutions consistently outperform the random baseline for each day. The difference is more remarkable, when we asked the user to choose between only one of the possibilities (Figure 7). We must note that usually the result

[1]If any of the results for the different methods was repeated, only one image was shown and the results were counted for both methods.

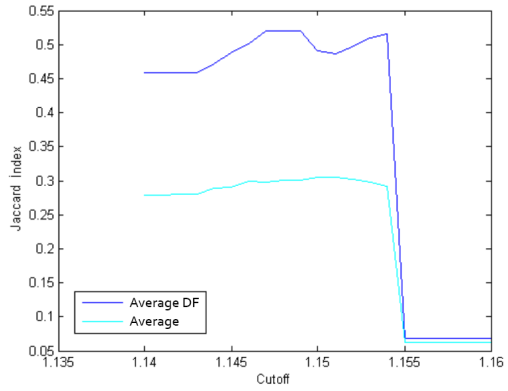

Fig. 4. Effect when using (dark blue) the division-fusion (DF) strategy and when not using it (light blue) in the average Jaccard index result for all the sets.

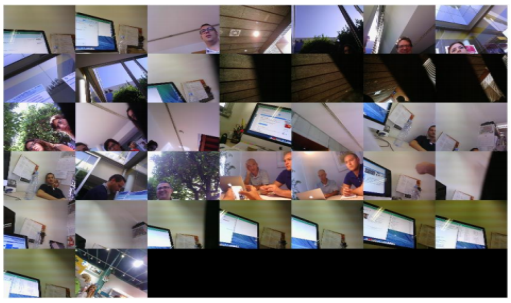

Fig. 5. Example of one of the summaries obtained by applying our approach on a dataset captured with Narrative camera.

was very similar either for the Random Walk and the Minimum Distance, since in most of the cases both algorithms selected the same keyframe.

4.3.2. Visual daily summary

In the second part of our qualitative study, we assessed the whole daily summary, built with the automatic event segmentation and the different solutions for keyframe selection. In this experiment, we added a fourth configuration that built a visual summary with a temporal *Uniform Sampling* of the day photostream, in such a way that the total amount of frames was the same as the amount of events detected through AC.

This time the user was shown the four summaries of the day generated by the four configurations. Figure 5 provides an example built with the *Random Walk* solution. For each summary, the user was firstly asked whether the set could represent the day (results in Figure 8), and also which of the four was the one that better described the day (results in Figure 9).

Focusing on the average results in Figure 5, we can state that, either applying *Random Walk* (88%) or *Minimal Distance* (86%), most of the generated summaries were positively assessed by the graders. Moreover, when it comes to

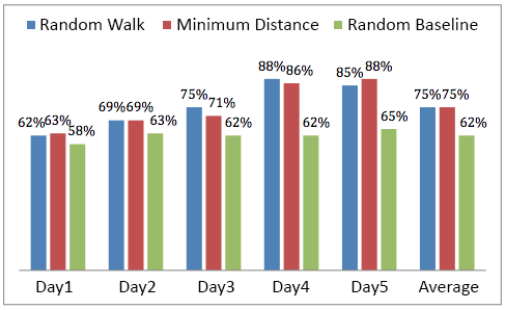

Fig. 6. Results answering "yes" to the question "Is this image representative for the current event?"

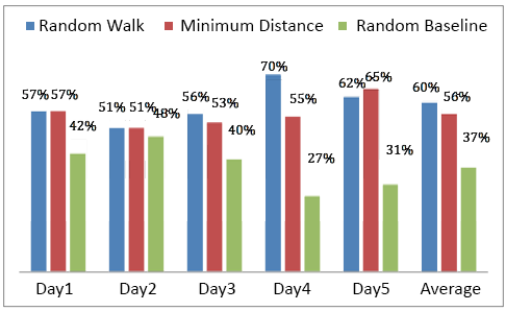

Fig. 7. Results to the question "Which of the previous frames is the most representative for the event?"

choose only the best summary, our method gathered 58% of the total votes if we consider that the voting is exclusive and that the summaries produced by the Random Walk and the Minima Distance methods are very similar. As a result, we obtained 34% and 41% of improvement respectively w.r.t the Random and the Uniform baselines.

5. CONCLUSIONS

In this work, we presented a new methodology to extract a keyframe-based summary from egocentric photostreams. After the qualitative validation made by 20 different users, we can state that our method achieves very good and representative summary results from the final user point of view.

Additionally, and always considering that the ultimate goal of this project is to reactivate the memory pathways of MCI patients, it offers satisfactory results in terms of capturing the main events of the daily life of the wearable camera users. A public-domain code developed for our visual summary methodology, is published in[2]

6. REFERENCES

[1] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin

[2] https://imatge.upc.edu/web/publications/visual-summary-egocentric-photostreams-representative-keyframes-0.

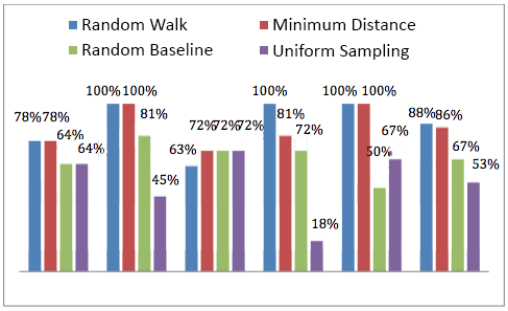

Fig. 8. Results answering "yes" to the question "Can this set of images represent the day?"

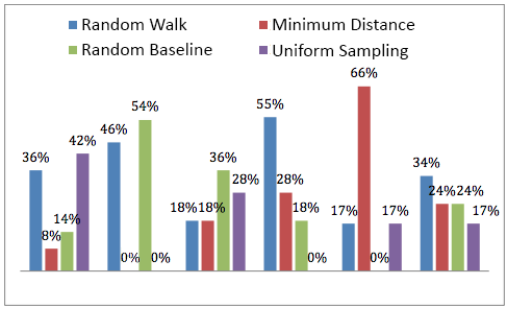

Fig. 9. Results to the question "Which of the previous summaries does better describe the day?"

Smyth, Narinder Kapur, and Ken Wood, "Sensecam: A retrospective memory aid," in *UbiComp 2006: Ubiquitous Computing*, pp. 177–193. Springer, 2006.

[2] Wei-Hao Lin and Alexander Hauptmann, "Structuring continuous video recordings of everyday life using time-constrained clustering," in *Electronic Imaging 2006*. International Society for Optics and Photonics, 2006, pp. 60730D–60730D.

[3] Karl Pearson, "The problem of the random walk," *Nature*, vol. 72, no. 1865, pp. 294, 1905.

[4] Aiden R Doherty, Daragh Byrne, Alan F Smeaton, Gareth JF Jones, and Mark Hughes, "Investigating keyframe selection methods in the novel domain of passively captured visual lifelogs," in *Proceedings of the 2008 international conference on Content-based image and video retrieval*. ACM, 2008, pp. 259–268.

[5] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video.," in *CVPR*. 2013, pp. 2714–2721, IEEE.

[6] M. Bolaños, M. Garolera, and P. Radeva, "Video segmentation of life-logging videos," in *Articulated Motion and Deformable Objects*, pp. 1–9. Springer-Verlag, 2014.

[7] Y. Poleg, Ch. Arora, and Shm. Peleg, "Temporal segmentation of egocentric videos," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference On*, 2014.

[8] A. R. Doherty and A. F. Smeaton, "Automatically segmenting lifelog data into events," in *Proceedings*, Washington, USA, 2008, WIAMIS '08, pp. 20–23, IEEE Comp. Society.

[9] E. Talavera, M. Dimiccoli, M. Bolaños, M. Aghaei, and P. Radeva, "R-clustering for egocentric video segmentation," in *Iberian Conference on Pattern Recognition and Image Analysis (in press)*. Springer, 2015.

[10] Y. J. Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization.," in *CVPR*. 2012, pp. 1346–1353, IEEE.

[11] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

[13] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky, "Neural codes for image retrieval," in *Computer Vision–ECCV 2014*, pp. 584–599. Springer, 2014.

[14] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *British Machine Vision Conference*, 2014.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[17] William HE Day and Herbert Edelsbrunner, "Efficient algorithms for agglomerative hierarchical clustering methods," *Journal of classification*, vol. 1, no. 1, pp. 7–24, 1984.

Semantic Summarization of Egocentric Photo Stream Events

Aniol Lidon
Universitat Politecnica de Catalunya
Barcelona, Catalonia/Spain

Marc Bolaños
Universitat de Barcelona
Barcelona, Spain
marc.bolanos@ub.edu

Mariella Dimiccoli
Universitat de Barcelona
Computer Vision Center
Barcelona, Spain
mariella.dimiccolig@cvc.uab.es

Petia Radeva
Universitat de Barcelona
Barcelona, Spain
petia.radeva@ub.edu

Maite Garolera
Consorci Sanitari de Terrassa
Terrassa, Spain
maite.garolera@cst.cat

Xavier Giro-i-Nieto
Universitat Politecnica de Catalunya
Barcelona, Catalonia/Spain
xavier.giro@upc.edu

ABSTRACT

With the rapid increase of users of wearable cameras in recent years and of the amount of data they produce, there is a strong need for automatic retrieval and summarization techniques. This work addresses the problem of automatically summarizing egocentric photo streams captured through a wearable camera by taking an image retrieval perspective. After removing non-informative images by a new CNN-based filter, images are ranked by relevance to ensure semantic diversity and finally re-ranked by a novelty criterion to reduce redundancy. To assess the results, a new evaluation metric is proposed which takes into account the non-uniqueness of the solution. Experimental results applied on a database of 7,110 images from 6 different subjects and evaluated by experts gave 95.74% of experts satisfaction and a Mean Opinion Score of 4.57 out of 5.0. Source code to reproduce this work is available at https://github.com/imatge-upc/egocentric-2017-lta.

ACM Reference format:
Aniol Lidon, Marc Bolaños, Mariella Dimiccoli, Petia Radeva, Maite Garolera, and Xavier Giro-i-Nieto. 2017. Semantic Summarization of Egocentric Photo Stream Events. In *Proceedings of LTA'17, Mountain View, CA, USA, October 23, 2017,* 9 pages.
https://doi.org/10.1145/3133202.3133204

1 INTRODUCTION

From smartphones to wearable devices, digital cameras are becoming ubiquitous. This process is being accompanied by the progressive reduction of digital storage cost, making it possible to collect large amounts of high-quality pictures in a easy and affordable way. This situation arises a number of natural questions: how to manage this large amount of pictures? Do we really need to store all of them? Summarization, the process of generating a proper, compact and meaningful representation of a given image collection through a subset of representative images, is crucial to help managing and browsing efficiently large volumes of video content. Although summarization is not a new research topic in Computer Vision [13, 29], it is still a largely open problem.

Figure 1: Example of temporal neighbouring images acquired by a wearable photo camera.

Our goal in this paper is to address the summarization problem focusing on a particular scenario: the one of analyzing photo streams acquired through a wearable camera. Motivation for this work is given by the explosion of the number of wearable camera users in recent years and, consequently, by the huge amount of data they produce. To record daily experiences from an egocentric, first-person perspective is a trend that has been growing progressively since 1998, when Steve Mann proposed the WearCam [23]. In 2000, Mayol et al. proposed a necklace-like lifelogging device [24] and, in 2006, Microsoft Research started to commercialize the first egocentric lifelogging portable camera, the SenseCam, for research purposes [14].

Several authors [21, 27, 28, 31] have studied the benefits of lifelogging cues such as egocentric images to help people with dementia to enhance their memory or to help them remember about their forgotten past. Sellen et al. [31] showed that episodic details from a visual 'lifelog' can be presented to users as memory cues to assist them in remembering the details of their original experience. To support people with dementia, Piasek et al. [27] introduced the "SenseCam Therapy" as a therapeutic approach similar to the well established "Cognitive Stimulation Theraphy" [33]. Participants were asked to wear SenseCam in order to collect images of events from their everyday lives, then images were reviewed with a trained therapist.

However, lifelogging technologies produce huge amounts of data (thousands of images per day) that should be reviewed by both patients and caregivers. To be efficient, lifelogging systems need to summarize the most relevant information in the images. On the other hand, in order to make possible the recording of images from the whole day, it is necessary to use wearable cameras with low temporal resolution (2 fpm). The peculiarity of these image collections is that, due to the low-temporal resolution of the camera

LTA'17, October 23, 2017, Mountain View, CA, USA

ACM ISBN 978-1-4503-5503-2/17/10…$15.00
https://doi.org/10.1145/3133202.3133204

and to the free motion of its wearer, temporally adjacent images may be very different in appearance even if they belong to the same event and should therefore be grouped together. For instance, during a meeting, the people the camera wearer is interacting with and the objects around them may change their position and frequently appear occluded (see Fig. 1).

As a consequence of this, taking the semantics into account is crucial to summarize egocentric sequences acquired by a low temporal resolution camera. This paper proposes a method to summarize each of the events present in a daily egocentric sequence, aiming at preserving semantic information and diversity, while reducing the total number of images. The method consists of three major steps: first, non-informative images are removed; second, they are ranked by semantic relevance; and finally, a new re-rank is applied by enforcing diversity among the chosen subset of pictures. Our contributions can be summarized as follows:

(1) Propose a CNN-based informativeness estimator for egocentric images.
(2) Define a set of semantic relevance criteria for egocentric images.
(3) Formulate the summarization task as a retrieval problem by combining informativeness, relevance and novelty criteria.
(4) Define a soft metric to assess the novelty from partially annotated image datasets.
(5) Our results have been validated by medical experts with the aim of being used in a cognitive training framework to reinforce the memory of patients with mild cognitive impairment.

The rest of the paper is organized as follows: the next section reviews the related work, section 3 details the proposed method, section 4 describes our experimental setup, section 5 presents the experimental results and finally, section 6 ends the paper with some concluding remarks. Source code to reproduce this work is available at https://github.com/imatge-upc/egocentric-2017-lta.

2 RELATED WORK

2.1 Summarization by key-frame selection in Lifelogging

Egocentric photo stream summarization has been traditionally formulated as the problem of grouping lifelog images into coherent collections (or events) by: first, extracting low-level spatio-temporal features and then, selecting the most representative image from each event. In this spirit, many authors proposed different strategies for temporal segmentation and key-frame selection. Doherty et al. [10] proposed a key-frame selection technique, which seeks to select the image with the highest 'quality' as key-frame by relying on five types of features: contrast, color variance, global sharpness, noise, saliency and external sensors data (accelerometers and light). In addition to image quality, Blighe et al. [2] considered image similarity to select a key-frame in an event. Basically, after applying an image quality filter that removes all poor quality images, they selected the image which has the highest average similarity to all other images in the event as the key-frame. In that case, similarity was measured relying on the distance of SIFT descriptors. In [18], the key-frame is selected by a nearest neighbour ratio strategy that favors high quality images and, if the difference in quality is not large enough, favors images closer to the middle of the temporal segment. More recently, the auhors in [3] proposed to use a Random Walk for selecting a single and most representative image for each temporal segment.

While these methods rely solely on low-level or mid-level features for the temporal segmentation and key-frame detection, a few recent works have introduced a higher semantic level in the selection process for video cameras. Although, in this cases, due to the higher temporal resolution of the camera (about 30fps), aiming at selecting subshots (short video sub-sequences) instead of unique key-frames. Lu and Grauman [22] and lately Ghosh et al. [11] suggested that video summarization should preserve the narrative character of a visual lifelog and, therefore, it should ideally be made of a coherent chain of video subshots in which each subshot influences the next through some subset of key visual objects. Following this idea, in [11, 22], first important people and objects are discovered based on their interaction time with the camera wearer and then, a subshot selection driven by key-object event occurrences is applied. Subshot selection is performed by incorporating into an objective function a term corresponding to the influence between subshots as well as image diversity. To model diversity, the authors relied on GIST descriptors and color histograms to model scenes and proposed a measure of diversity that is high when the scenes in sequential subshots are dissimilar to ensure visual uniqueness. Visual diversity in lifelog summaries is modeled in [1] through the concept of novelty, which the authors heuristically defined as the deviation from some standard background. According to this definition, novelty is detected based on the absence of a good registration, in terms of ego-motion and the environment, between a new sequence and stored reference sequences. More recently, Gong et al. [12] proposed the so called Sequential Determinantal Point Process (seqDPP) approach for video summarization, a probabilistic model with the ability to teach the system how to select informative and diverse subsets from human-created summaries, so as to best fit human-perception quality based on evaluation metrics. This work is an adaptation to sequences of [20], which is able to capture the strong dependency structures between items in sequences. It is worth to mention that all these works have been conceived to deal with video data, where, assuming that the temporal segmentation is good, temporal coherence and frame redundancy make the key-frame selection process easier.

2.2 Diversity and novelty in information retrieval

One of the first works that tried to approach the problem of obtaining a diverse set of elements was presented in 1998 by Carbonell & Goldstain [4]. Their proposal, which was applied to the context of text retrieval and summarization, aimed at obtaining results highly relevant for the query, but presenting a low redundancy. They define the marginal relevance as the linear combination of relevance and novelty, measured independently. They aimed at maximizing it iteratively, defining this way what they call the Maximal Marginal Relevance (MMR).

A similar formulation to MMR of the diversification problem was given more recently by Deselaers et al. [8] in the problem of image

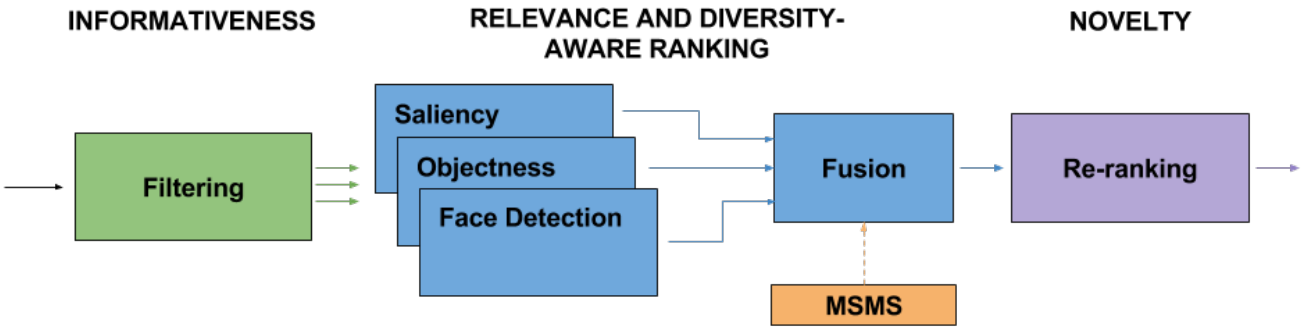

Figure 2: General scheme of our event summarization methodology.

retrieval. Likewise, they jointly optimized the relevance and the diversity of the query, although reformulating the general solution by incorporating dynamic programming techniques to the initial proposal.

Diversity in social image retrieval was one of the focus of the MediaEval 2013, 2014 and 2015 benchmarks and attracted the interest of many groups working in this area. Most participants developed diversification approaches that combined clustering with a key-frame selection strategy to extract representative images for each cluster. Spyromitros et al. proposed an MMR-based approach [34] that jointly considers relevance and diversity, but using a supervised classification model to obtain the relevance scores learned from the user feedback. The contribution from Dang-Nguyen et al. [6] was to filter out non-relevant images at the beginning of the process before applying diversity, hence simplifying it.

In [32], the authors presented a method for detecting and resolving the ambiguity of a query based on the textual features of the image collection. If a query has an ambiguous nature, this ambiguity should be reflected in the diversity of the result to increase user satisfaction. Leuken et al. [37] proposed to reduce the ambiguity of results provided by image search engines relying on textual descriptions by seeking for the visual diversification of image search results. This is achieved by clustering the retrieved images based on their visual similarity and by selecting a representative image for each cluster. In these works 'diversity' is aimed at addressing the 'ambiguity' of the textual descriptions (tags) in queries rather than avoiding redundancy in search results. The same terminology is used by [5], where the term 'novelty' is meant to address redundancy in the retrieved documents. In this work we will use the terms novelty and diversity as defined in [5].

3 METHODOLOGY

Inspired by the image retrieval work, we question which would be the most suitable diversity and relevance criteria applied on egocentric images that kept the narrative character of the visual lifelog. Taking into account that the images are acquired non-intentionally, it is important to disregard the non-informative images (see Section 3.1) and keep the minimal set that represents the visual event. In this section, we explain the four main steps applied to construct the final resulting summary (see Fig. 2):

(1) Informativeness Filtering (Section 3.1): egocentric images are acquired non-intentionally and therefore, many of them may be non-informative, capturing neither (or partially) objects nor people, or being blurred or dark. By means of a CNN-based informativeness filtering method, we discard most of the non-informative images from the egocentric event.

(2) Relevance and Diversity-aware Ranking (Section 3.2): an initial relevance image ranking is computed taking into account different criteria such as Saliency and Objectness for dealing with the ambiguity or under-specification of queries like *What is the user doing?*, or *Where is the user?*, and Face Detection for answering the question *With whom is the user most likely interacting?*.

(3) Novelty-based Re-ranking (Section 3.3): a final re-ranking based on a novelty-maximization procedure is applied on the images already ranked by relevance. This step is crucial to select those images that represent the most varied set of concepts appearing avoiding redundancy without semantic loss.

(4) Estimation of Fusion Weights with Mean Sum of Maximal Similarities (Section 3.4): in this step, we define a novel soft metric, which we call Mean Sum of Maximal Similarities (MSMS), in order to define the priorities of the different relevance terms and construct the final summary.

3.1 Informativeness Filtering

Considering both the free motion of egocentric cameras and the non-intentionality of the pictures they take, several problems are inherent to them as over- or under-light exposure, blurriness, pictures of the sky or ground, pictures where possible objects of interest are badly centered in the image or even non-existent, etc. Such images are considered as *non-informative pictures*. A way to avoid a good amount of the undesired images in the final summary and also to boost the performance of the next steps in our methodology, would be being able to discriminate the *informative pictures*. Fig. 3 illustrates informative vs. non-informative photos taken by the wearable camera.

In order to learn a model of informative pictures, and taking into account the complexity of distinguishing visually whether an image is informative enough or not, we propose training a CNN for a binary problem. Thus, all the images with an undesired artifact (empty image, blurred image, image with small amount of information, image where something occludes most of the image

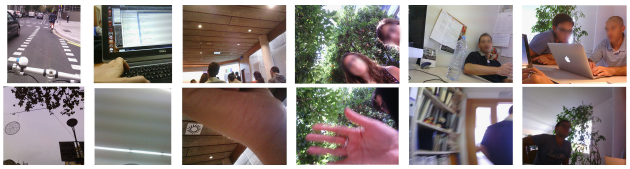

Figure 3: Examples of informative images (top) and non-informative images (bottom) belonging to the same events. Faces appearing in the images have been manually blurred for privacy concerns.

Figure 4: Images acquired by a lifelogging device, where objects of interest appear like: computer, mobile, coffee, hand, bicycle, person, face, flower, apple, etc.

region, sky, ground, ceiling, room wall, etc.) will be considered non-informative (label 0), and the rest (images with semantic content) will be considered informative (label 1). With this procedure, we will be able to extract an *informativeness score* for each image and filter the unuseful ones. In order to remove as many images as possible, but with the ultimate goal of having a very high recall (always try to keep any informative image for the next steps), we only discard the images with $informativenessScore < 0.025$, which correspond to the ones that are considered non-informative for certain by the CNN (see experimental results in section 5.1).

The network training for the binary class distinction is performed by fine-tuning the CaffeNet [19] pre-trained on the ImageNet [7] dataset, provided in the software Caffe [17].

3.2 Relevance and Diversity-aware Ranking

Once non-informative images are discarded, we proceed to rank the remaining images by considering a relevance criteria. Our solution formulates the summarization problem in similar terms as in information retrieval. A ranked list of event frames is generated in such a way that a summary of T images directly corresponds to a truncation of the ranked list of N elements at its T-th position. Classic retrieval problems build their ranked lists as a response to a user query, in the summarization of events, this query would correspond to: *Select T images to describe the event depicted into these N frames.* This query is highly ambiguous as an event can be described from different perspectives. For example, an egocentric visual summary of an event may respond to multiple intentions, such as: *Where* is the user? *What* activity is the user performing? *With whom* is the user interacting? We hypothesize that the relevance of each image with respect to these questions can be estimated with computer vision tools for saliency prediction, detection of objects and detection of faces.

3.2.1 Saliency Prediction for image relevance. We assume that images with more salient content are relevant and should have a higher probability to be included in the summary. Visual saliency can be triggered by a broad range of reasons, such as objects, people or characteristic features appearing in the picture. Many computer vision algorithms try to estimate the fixation points of the human eyes in a scene by means of *saliency maps*. These are heat maps of the same size of the image, whose higher values correspond to the image locations with a higher probability of capturing the human visual attention.

In this work, we compute the saliency maps with SalNet [26], an end-to-end convolutional network for saliency prediction. We adopt the overall sum of the values in the saliency map as a quantitative estimator of the image relevance.

3.2.2 Object Detection for image relevance. We assume that those images containing objects are relevant since these objects likely correspond to the ones the user is interacting with. These objects would address the question of *What is the user doing?* (see Fig. 4). In addition, introducing a semantic interpretation of the scene, targets the summary from a higher abstraction level than saliency maps.

In our work, we used the off-the-shelf tool *Large Scale Detection through Adaptation (LSDA)* presented in [15]. This object detector is based on a CNN fine-tuned for local scale and provides a semantic label and a confidence score in the localization of the objects in the scene.

This tool allows to estimate the relevance of each frame by summing the detection scores of all objects in the picture, so that the frames with higher confidence detection will be considered as more relevant.

3.2.3 Face Detection for image relevance. We assume that images containing people are also relevant, since these people likely correspond to the ones the user is interacting with, and they would be useful to answer the question *With whom is the user interacting?*. Therefore, a face detector complements the object detector into providing a cue for the user social interactions during the event.

In our solution, we adopted the off-the-shelf face detector by Zhu et al. [39], which provides a confidence for each of the detected faces. The relevance of each frame in terms of social interaction was estimated by summing the confidence scores of the face detectors. In the particular implementation of [39], detection scores may also be described with negative values, so we actually used these scores in an exponential sum, which conveniently deals with the negative scores as well as encourages the selection of frames with multiple detected faces.

3.2.4 Diversity-aware Ranking. The three criteria used for the relevance detailed above allow to cope with the ambiguity of the query, since they estimate the relevance from three different perspectives. The relevance scores computed for each case are then used to build three ranked lists that will be combined into a single one. This combination is based on generating a set of normalized scores based simply on the position of the frames. Normalized scores $r_k(x)$ are linearly distributed from the top (1 for most relevant) to the bottom (0 for non relevant) of the ranked list as follows:

$$r_k(x) = \frac{M - R_k(x)}{M - 1},$$

where $R_k(x) = 1, \ldots, M$ is the ranking position associated with image x according to each relevance criterion $k \in \{1, 2, 3\}$ and M

is the number of informative frames in the event, being $M < N$ and N the total number of images in the event. The standard score normalization [25, 30] with the min and max scores was also tested giving similar results, so the rank-based normalization was adopted to save computational resources.

3.3 Novelty-based Re-ranking

The relevance and diversity-aware ranked list sorts the informative images combining different criteria for relevance, but does not explicitly cope with the redundancy in the information. Further processing is necessary to maximize the *novelty* provided by each image with respect to the rest in the summary. In information retrieval, *novelty* is defined as a quality of a system that avoids redundancy [5]. In our approach for building a summary by truncating a ranked list of images, introducing novelty implies that each image in the list should differ as much as possible from its predecessors.

Our approach adopts the *greedy* selection algorithm presented by Deselaers et. al [9] to re-rank the fused list based on novelty. The goal of our summarization algorithm is to analyze the input set of informative and ranked images $\mathcal{X} = \{x_1, ...x_M\}$ to iteratively build another set with minimal redundancy, say $\mathcal{Y}^T = \{x_{y_1}, ...x_{y_T}\}$, where $T \leq M$. Our approach starts by selecting the top ranked image x_1 in the diversity-aware ranked list as the first element of $\mathcal{Y}$, that is $\mathcal{Y}^1 = \{x_{y_1}\}$. The *novelty* of each candidate image, x^* to be added to the summary at iteration $t = 2, ..., T$ is defined as:

$$n(x^*, \mathcal{Y}^t) = 1 - s(x^*, \mathcal{Y}^t) = 1 - \max_{x_{y_j} \in \mathcal{Y}^t} s(x^*, x_{y_j}) \quad (1)$$

$s(x^*, x_{y_j})$ is a normalized similarity measure between the candidate x^* and image x_{y_j}. In this way, the more different a new image is with respect to the ones in $\mathcal{Y}^t$, the higher its novelty is.

In our work, the similarity $s(x^*, x_{y_j})$ is based on visual appearance. Each image is described by a feature vector corresponding to the seventh fully-connected layer of the CaffeNet convolutional neural network [17], which was also used as initialization for the informativeness (see section 3.1), and whose architecture was inspired by AlexNet [19] and trained with the ImageNet dataset [7]. The similarity score is computed with the Euclidean distance of the feature vectors.

The summary of the event is built by combining the relevance and the novelty of the candidate frames, x^*. A greedy selection algorithm iteratively chooses the next image in the summary as:

$$\begin{aligned} x_{y_{t+1}} &= \arg\max_{x^* \in \mathcal{X} \setminus \mathcal{Y}^t} (r(x^*) + n(x^*, \mathcal{Y}^t)), \\ \mathcal{Y}^{t+1} &= \mathcal{Y}^t \cup \{x_{y_{t+1}}\} \end{aligned} \quad (2)$$

that is, the image with the highest sum of relevance and novelty. We detail the computation of $r(x^*)$ in Equation (4).

3.4 Estimation of Fusion Weights with Mean Sum of Maximal Similarities

Different relevance criteria would have different importance for the final ranking. To this aim, we propose a novel approach to estimate the priorities of each criterion and fuse them, based on comparing the similarity of the images from a validation set of

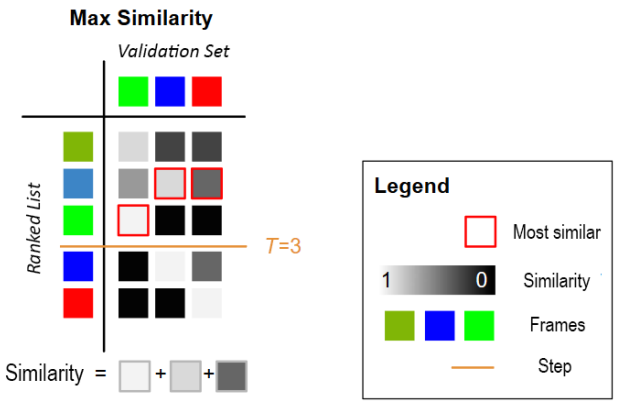

Figure 5: Visual representation of the Sum of Max Similarities when comparing the first $T = 3$ images from the ranked list with the $P = 3$ red, green and blue images from the validation set. The similarity between images from the validation set and images from the ranked list is represented as a $T \times P$ matrix of gray level squares, where each square (i, j) of the matrix represents the similarity between the image i from the ranked list and the image j from the validation set. High intensity gray level values indicate high similarity.

P elements $\mathcal{V} = \{x_{v_1}, ..., x_{v_P}\}$ with a summary of T elements $\mathcal{Y}^T = \{x_{y_1}, ...x_{y_T}\}$.

The *Sum of Maximal Similarities (SMS)* of $\mathcal{V}$ with respect to $\mathcal{Y}^T$ is defined as:

$$SMS(\mathcal{V}, \mathcal{Y}^T) = \frac{1}{P} \sum_{i=1}^{P} s(x_{v_i}, \mathcal{Y}^T) \quad (3)$$

where $s(x_{v_i}, \mathcal{Y}^T)$ is the similarity of image x_{v_i} from the validation set with respect to $\mathcal{Y}^T$. Following this metric, the more similar is the validation set $\mathcal{V}$ to the selected images $\mathcal{Y}^T$, the highest the average similarity.

The main advantage of our soft metric presented in Equation (3) is that automatic summaries, which are very similar to the validation images, although do not coincide, still will be assessed as a valid solution. This feature of our soft metric is specially important when working with sequences of egocentric photo streams, that usually contain a high amount of redundancy.

Our summaries are built as a truncation of a ranked list, so the value of *SMS* depends on T. Fig. 5 shows a schematic example, where a validation set composed of $P = 3$ images (represented by three colors: green, blue and red) is compared with an event of $M = 5$ images, which have been previously filtered and ranked based on our diversity and novelty-aware criterion. In this example, a summary with $T = 3$ images is considered. Let us imagine that the green image from $\mathcal{V}$ is matched with the third one in the ranked list, while the blue and red images are matched with the second one.

Notice that, as in our set up the validation set $\mathcal{V}$ is always a subset of the input set $\mathcal{X}$, hence, the final average similarity (i.e. considering the whole sequence, $T = M$ images) for $\mathcal{Y}^M$ will always

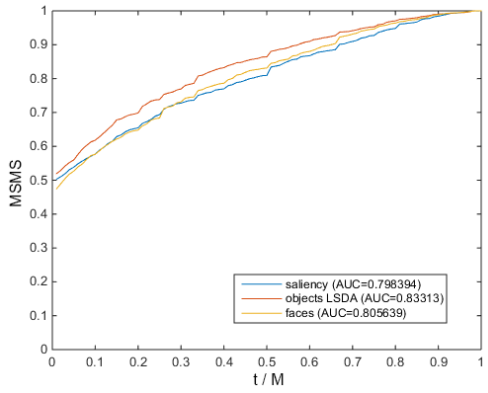

Figure 6: MSMS curves and AUC for each relevance criterion used separately.

correspond to one. Actually, in the example of Fig. 5, SMS must reach a value of one when all the ranked list is considered, that is, when $T = M$.

Let us consider the evolution of the $\overline{SMS}$ of summary $\mathcal{Y}$ as the parameter t grows ($t = 1, 2, \ldots, M$) defined as:

$$\overline{SMS}(\mathcal{V}, \mathcal{Y}) = \{SMS(\mathcal{V}, \mathcal{Y}^1), \ldots, SMS(\mathcal{V}, \mathcal{Y}^M)\}$$

We construct the $\overline{SMS}$ curves for all the validation set, interpolate them, normalize them with respect to the length M of each sequence and get the average of the curves, one curve for each event in the validation set. The resulting curve represents the *Mean Sum of Maximal Similarities (MSMS)*.

Fig. 6 shows three examples of MSMS curves. The curve illustrates the evolution of the MSMS as a function of the percentage of images covered by the visual summary. This curve is defined over the X-axis representing the proportion of event images represented in the top t items in the list, that is, plotting the *MSMS* over t/M, where M is the amount of already filtered images in the event. In this way, the curves of events of different lengths M can be compared on the same plot. The best MSMS curves are those which reach higher values with the minimal amount of images in the summary.

Note that the MSMS can be computed for the three relevance criteria that can give as a quality of performance of each of them. Hence, we introduce a final refinement in the fusion of the ranked lists by estimating the confidence of each of the three relevance criteria and using it to weight the corresponding relevance terms. Thus, we leverage the contribution of each of the three relevance criteria based on their stand-alone performance (see Fig. 6). This weight corresponds to the normalized Area Under the Curve (AUC) of the MSMS measure.

As a result, the fusion weight $w(k)$ for the relevance criterion k is estimated as:

$$w(k) = \frac{AUC(k)}{\sum_{i=1}^{3} AUC(i)}.$$

Finally, the three normalized relevance scores associated to each frame are aggregated with a weighted sum to obtain their fused score $r(x)$, as described by the following equation:

$$r(x) = \sum_{k=1}^{3} w(k)\, r_k(x), \tag{4}$$

and the final set of frames is obtained according to the updated relevance and novelty criteria according to Equation (2).

4 EXPERIMENTAL SETUP

4.1 Dataset

The dataset used for validation of our method was acquired with the wearable camera Narrative Clip (www.getnarrative.com), which takes a picture every 30 seconds (2 fpm). It is composed of 10 day lifelogs from 5 different subjects, with a total of 7,110 images[1]. Each day has been segmented in between 10 to 25 events manually, although any automatic segmentation method can be used (e.g. [36]). The event segmentation separated the pictures in a set of ordered and relevant semantic events or segments. In this paper, we use this segmentation as starting point for the semantic summarization.

In order to apply a quantitative evaluation of our method, each day lifelog was annotated at two levels by psychologists in the following way:

(1) GT Level 1 - Informativeness: positive or negative label depending on whether the image is considered informative or non-informative (see section 3.1). The proportion of images labeled as informative is 61.22% of the complete dataset.

GT Level 2 - Grouping of similar images: all highly similar informative images that belong to the same event are grouped together (see section 3.2.4). This distinction, resembling the one used in *MediaEval 2014 Retrieving diverse social images challenge* [16], intends to provide a way of measuring how many different clusters/groups from the ground truth are represented among the results in the final diversity selection.

4.2 Semantic Assessment of Summaries

The assessment of visual summaries is a challenging task due to the rich semantic content of the images and the ambiguity in the evaluation criteria. In addition, human annotation is an expensive resource, which becomes dramatically scarce, when dealing with tasks that require expert annotations in the domain. In our case, the ground truth annotations have been performed by psychologists who defined the event's summaries from an human-centered point of view.

We addressed the challenge of summary assessment as follows: starting from the daily events defined by the psychologists, the automatic summaries were built for each of the events separately by using the best configuration of our system. Later, the results were presented in a blind taste test to the expert annotators.

4.2.1 Validation based on MSMS. The classical methods to evaluate summaries cannot be applied in our setup, because they require the annotation of the full dataset. In a traditional case, each document in the ground truth is labeled as relevant or non-relevant for the summary and, in some cases, also clustered in groups of redundant items that cover the same sub-topic. Such annotations allow the definition of metrics like *Precision* [4] for relevance and

[1]The link to the dataset and its ground truth are prepared to be made public domain when the article is published.

Cluster Recall (or subtopic-recall) [38] for diversity and/or novelty. However, in other cases (like ours), only a small portion of the dataset is annotated.

Given this limitation, following the framework of Subsection 3.3, we propose an evaluation approach based on the similarity of the images in the ground truth set $\mathcal{G} = \{x_{g_1}, ..., x_{g_P}\}$, when compared to the automatic summary of T elements, $\mathcal{Y}^T = \{x_{y_1}, ...x_{y_T}\}$. In this case, we apply the SMS, $SMS(\mathcal{G}, \mathcal{Y}^T)$ of ground truth $\mathcal{G}$ with respect to the extracted summary, $\mathcal{Y}^T$. As before, we obtain the MSMS as the average value, when considering all events in the dataset of the ground truth, which is finally plotted in the Y-axis of our evaluation scores. Finally, the AUC is computed to obtain a quantitative measure of each configuration for the summary.

4.2.2 Blind taste test. The resulting summaries are evaluated with a *blind taste test* by the team of experts who generated the ground truth summaries. This methodology, previously used in [3, 22], shows different summaries from the same event to the experts, so they can rate them from a comparative perspective. By randomly changing the position of the different techniques at each event, graders are unaware of what technique was used to generate each of the summaries, guaranteeing this way that their judgments are not biased towards any of the algorithm components.

The amount of images included in each summary corresponds to the number of frames selected by the experts, when building the ground truth summaries, that is, $P = T$. In this way, the summary shown to the graders is a truncation of the final ranked list of images. In our case, the experts decided the length of the automatic summary to be a percentage of the length of the whole event. Note that developing an automatic system to establish the length of the summary is out of the scope of this paper.

Additionally, the selected images are sorted in the summary according to their temporal time stamps. This final sorting helps the user to reconstruct the story between the images to have a better understanding of the event.

An online platform was developed for the experts to evaluate our results. In a first round of questionnaires, we wanted to compare the result of our diversity method using ImageNet or Places as a similarity criterion (see Equation (1)). For each event, the questionnaire first shows all the images belonging to the event, and then the user has to answer the questions *Is the summary representative of the event?* and *Which summary do you prefer?* for each similarity method (using ImageNet or Places).

On the second questionnaire the expert had to give a grade to each of the presented ranked summaries. The three summaries correspond respectively to: our approach, a uniform sampling of images as a lower-bound example, and the ground truth summaries constructed by the same experts two months earlier as an upper-bound of the performance score. The order in which the different summaries are shown is also randomly chosen for each event to avoid again any bias due to the sorting.

In this case, a comparison between the three types of summaries is obtained by asking the experts to grade each solution from 1 (worst) to 5 (best). This data collected allowed the computation of the *Mean Opinion Score (MOS)* for each configuration (see section 5.3).

5 RESULTS

Two types of experiments validate the presented results: one for the informativeness filtering and a second one using online questionnaires for the relevance, diversity and novelty on the final visual summaries. In both cases, the evaluation was based on the feedback provided by the experts.

5.1 Informativeness Filtering

To validate how successful is our algorithm to filter non-informative frames, we applied a 10-fold (a day set out) cross-validation. For all the experiments, we used the following parameters: $base\_lr = 10^{-8}$, $lr\_policy =$ "$step$", $gamma = 0.1$, $stepsize = 3000$, $blobs\_lr_{fc6} = blobs\_lr_{conv5} \times 5$, $blobs\_lr_{fc7} = blobs\_lr_{conv5} \times 8$ and $blobs\_lr_{fc8} = blobs\_lr_{conv5} \times 10$. In Table 1, we present the validation accuracy obtained on each of the sets and the number of iterations applied to achieve the best result. After having the networks trained for each cross-validation, we evaluated the general average performance in terms of Accuracy, Precision, Recall and F-Measure for different informativeness score threshold values[2]. As we can see, we are able to obtain the highest F-Measure value, when we filter images with an $informativenessScore < 0.025$. In Fig. 7, we show the Precision-Recall curves for all thresholds, obtaining the best results with respect to the F-Measure of: $FMeasure = 0.881$, $Accuracy = 0.846$, $Precision = 0.84$ and $Recall = 0.926$ with $threshold = 0.05$.

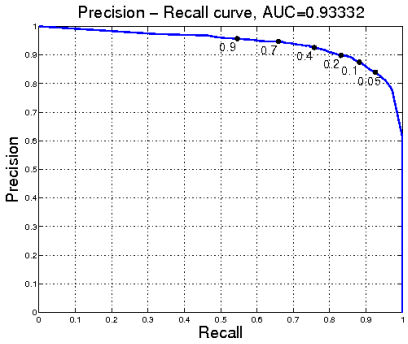

Figure 7: Precision-Recall curve for different informativeness score threshold values (a small subset of them are shown in black dots) for all the sets.

5.2 Diversity

Fig. 8 illustrates qualitatively the differences obtained when introducing diversity to the ranked list. As we can see, when we introduce the novelty re-ranking step, we are able not only to obtain a more visually acceptable set of images, but also to describe with pictures everything that is happening during the whole event. Thus, we are able to avoid focusing on a single set of high relevant images that picture the same concept, activity or background.

[2]If the informativeness score of a sample is below the threshold, it will be considered non-informative.

Table 1: Best validation accuracy on each set and the respective number of training iterations performed to achieve the results. *SubjX* represents the anonymized sets for a given Subject.

	$SubjA_1$	$SubjA_2$	$SubjB_1$	$SubjB_2$	$SubjB_3$	$SubjC_1$	$SubjD_1$	$SubjE_1$	$SubjF_1$	$SubjF_2$
Accuracy	0.759	0.841	0.805	0.795	0.799	0.837	0.805	0.867	0.795	0.897
Iteration #	3,600	1,000	3,600	3,400	3,800	2,600	2,200	3,600	2,400	18,000

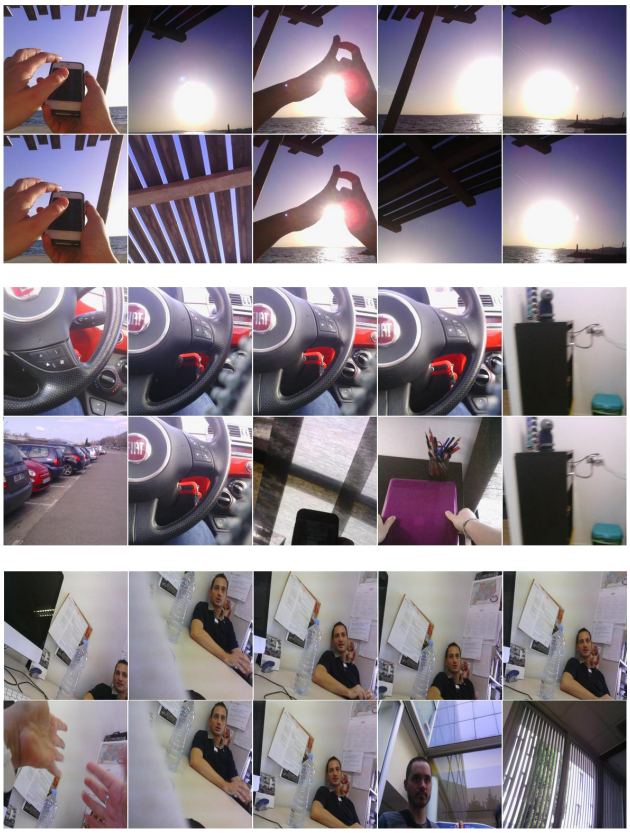

Figure 8: Three examples of the top 5 images obtained before introducing diversity (uneven rows) and after introducing it (even rows).

5.3 Ranking Summary Quality

The first round of blind taste tests posed two questions for each event. Firstly, experts decided whether each presented summary was representative, and secondly, they chose the preferred one Given the question *Is the summary representative of the event?*, in 95.74% the experts agreed with our solution, using the ImageNet features to compute the similarity in the novelty-based re-ranking. Using the other alternative, features from Places CNN, the obtained result was 94.33%. We conclude that our approach generates representative summavery high portion of events.

The second question asked was: *Which summary do you prefer?* and experts had to choose their preferred summary, allowing just one answer. The solution based on Imagenet features was chosen in 59.57% of teh cases, while the one based on Places was selected 53.19% of the times. The total adds more than 100% because some summaries were identical for both configurations.

The second round of evaluations aimed to compare our solution based on ImageNet features with a baseline of uniform sampling and and upper-bound defined by the summaries in the ground truth. Experts were asked to *grade each visual summary from 1 (worse) to 5 (best)*, so we could compute the Mean Opinion Score (MOS) [35] of each solution. We adopted MOS as a metric given the highly subjective and complex nature of the task.

Table 2: Mean Opinion Score for ImageNet, ground-truth and uniform sampling summaries.

Our solution	Ground-truth	Uniform Sampling
4,57	**4,94**	**3,99**

The performance obtained by uniform sampling (3.99/5) is truly commendable, since this score can be interpreted as a *good* solution. The results obtained with our solution are also satisfactory, because they are closer to the ground-truth than to the uniform sampling. Experts have shown coherence with their ground-truth giving a 4.94 of Mean Opinion Score.

6 CONCLUSIONS

In this paper, we presented a novel approach for semantic summarization of egocentric photo stream events, constructed as a ranked list of general semantic criteria. Our method is based on two main criteria: relevance to optimize semantic diversity in the summary, and novelty to avoid redundancy in the final result. After applying a pre-processing step to filter non-informative images through a new CNN-based method, relevance diversity-aware ranking is obtained by integrating state of the art techniques for saliency detection, object recognition and face detection. This list is re-ranked to reduce redundancy so that each image of the truncated list differs as much as possible from its predecessors. We proposed a new soft metric to rank the informative frames and construct the final summary that does not penalize summaries equivalent (very similar, but not coinciding exactly) to the ground truth. Experimental results indicate high acceptance and satisfaction of psychologists achieving mean opinion score of 4.57 out of 5.0.

ACKNOWLEDGMENTS

This research was supported by contracts SGR1219 and SGR1421 by the Catalan AGAUR office, and TIN2012-38187-C03-01 and TEC2013-43935-R by the Spanish Ministerio de Economia y Competitividad and the European Regional Development Fund (ERDF). We acknowledge the support of NVIDIA Corporation for the donation of GPUs.

REFERENCES

[1] Omid Aghazadeh, Josephine Sullivan, and Stefan Carlsson. 2011. Novelty detection from an ego-centric perspective. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 3297–3304.

[2] Michael Blighe, Aiden Doherty, Alan F. Smeaton, and Noel E. O'Connor. 2008. Keyframe Detection in Visual Lifelogs. In *Proceedings of the 1st International Conference on PErvasive Technologies Related to Assistive Environments (PETRA '08)*. ACM, New York, NY, USA, Article 55, 2 pages. https://doi.org/10.1145/1389586.1389652

[3] Marc Bolaños, Ricard Mestre, Estefanía Talavera, Xavier Giró-i Nieto, and Petia Radeva. 2015. Visual Summary of Egocentric Photostreams by Representative Keyframes. *arXiv preprint arXiv:1505.01130* (2015).

[4] Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '98)*. ACM, New York, NY, USA, 335–336. https://doi.org/10.1145/290941.291025

[5] Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 659–666.

[6] Duc-Tien Dang-Nguyen, Luca Piras, Giorgio Giacinto, Giulia Boato, and FGB De Natale. 2014. Retrieval of Diverse Images by Pre-filtering and Hierarchical Clustering. *Working Notes of MediaEval* (2014).

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 248–255.

[8] Thomas Deselaers, Tobias Gass, Philippe Dreuw, and Hermann Ney. 2009. Jointly Optimising Relevance and Diversity in Image Retrieval. In *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR '09)*. ACM, New York, NY, USA, Article 39, 8 pages. https://doi.org/10.1145/1646396.1646443

[9] Thomas Deselaers, Tobias Gass, Philippe Dreuw, and Hermann Ney. 2009. Jointly optimising relevance and diversity in image retrieval. In *Proceedings of the ACM international conference on image and video retrieval*. ACM, 39.

[10] Aiden R. Doherty, Daragh Byrne, Alan F. Smeaton, Gareth J.F. Jones, and Mark Hughes. 2008. Investigating Keyframe Selection Methods in the Novel Domain of Passively Captured Visual Lifelogs. In *Proceedings of the 2008 International Conference on Content-based Image and Video Retrieval (CIVR '08)*. ACM, New York, NY, USA, 259–268. https://doi.org/10.1145/1386352.1386389

[11] Joydeep Ghosh, Yong Jae Lee, and Kristen Grauman. 2012. Discovering important people and objects for egocentric video summarization. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1346–1353.

[12] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. 2014. Diverse Sequential Subset Selection for Supervised Video Summarization. In *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger (Eds.). Curran Associates, Inc., 2069–2077. http://papers.nips.cc/paper/5413-diverse-sequential-subset-selection-for-supervised-video-summarization.pdf

[13] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. 2014. Creating summaries from user videos. In *European conference on computer vision*. Springer, 505–520.

[14] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. 2006. SenseCam: A Retrospective Memory Aid. In *Proceedings of the 8th International Conference on Ubiquitous Computing (UbiComp'06)*. Springer-Verlag, Berlin, Heidelberg, 177–193. https://doi.org/10.1007/11853565_11

[15] Judy Hoffman, Sergio Guadarrama, Eric Tzeng, Jeff Donahue, Ross B. Girshick, Trevor Darrell, and Kate Saenko. 2014. LSDA: Large Scale Detection Through Adaptation. *CoRR* abs/1407.5035 (2014). http://arxiv.org/abs/1407.5035

[16] Bogdan Ionescu, Adrian Popescu, Mihai Lupu, Alexandru L Gınsca, and Henning Müller. 2014. Retrieving diverse social images at mediaeval 2014: Challenge, dataset and evaluation. In *MediaEval 2014 Workshop, Barcelona, Spain*.

[17] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 675–678.

[18] Amorncheld Jinda-Apiraksa, Jana Machajdik, and Robert Sablatnig. 2012. A keyframe selection of lifelog image sequences. *Erasmus Mundus M. Sc. in Visions and Robotics thesis, Vienna University of Technology (TU Wien)* (2012).

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[20] Alex Kulesza and Ben Taskar. 2012. Determinantal point processes for machine learning. *arXiv preprint arXiv:1207.6083* (2012).

[21] Matthew L. Lee and Anind K. Dey. 2008. Lifelogging Memory Appliance for People with Episodic Memory Impairment. In *Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp '08)*. ACM, New York, NY, USA, 44–53. https://doi.org/10.1145/1409635.1409643

[22] Zheng Lu and Kristen Grauman. 2013. Story-driven summarization for egocentric video. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2714–2721.

[23] S. Mann. 1998. 'WearCam' (The wearable camera): personal imaging systems for long-term use in wearable tetherless computer-mediated reality and personal photo/videographic memory prosthesis. In *Wearable Computers, 1998. Digest of Papers. Second International Symposium on*. 124–131. https://doi.org/10.1109/ISWC.1998.729538

[24] W. W. Mayol, B. J. Tordoff, and D. W. Murray. 2002. Wearable Visual Robots. *Personal Ubiquitous Comput.* 6, 1 (Jan. 2002), 37–48. https://doi.org/10.1007/s007790200004

[25] Mark Montague and Javed A Aslam. 2001. Relevance score normalization for metasearch. In *Proceedings of the tenth international conference on Information and knowledge management*. ACM, 427–433.

[26] Junting Pan, Kevin McGuinness, and Xavier Giró-i Nieto. 2016. End-to-end Convolutional Network for Saliency Prediction. In *Submitted to CVPR*.

[27] P. Piasek, K. Irving, and A.F. Smeaton. 2011. SenseCam intervention based on Cognitive Stimulation Therapy framework for early-stage dementia. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on*. 522–525.

[28] Paulina Piasek, Alan F Smeaton, et al. 2014. Using lifelogging to help construct the identity of people with dementia. (2014).

[29] Sachan Priyamvada Rajendra and N Keshaveni. 2014. A survey of automatic video summarization techniques. *International Journal of Electronics, Electrical and Computational System* 2 (2014).

[30] M Elena Renda and Umberto Straccia. 2003. Web metasearch: rank vs. score based rank aggregation methods. In *Proceedings of the 2003 ACM symposium on Applied computing*. ACM, 841–846.

[31] Abigail J. Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, and Ken Wood. 2007. Do Life-logging Technologies Support Memory for the Past?: An Experimental Study Using Sensecam. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 81–90. https://doi.org/10.1145/1240624.1240636

[32] Kai Song, Yonghong Tian, Wen Gao, and Tiejun Huang. 2006. Diversifying the Image Retrieval Results. In *Proceedings of the 14th Annual ACM International Conference on Multimedia (MULTIMEDIA '06)*. ACM, New York, NY, USA, 707–710. https://doi.org/10.1145/1180639.1180789

[33] Aimee Spector, Lene Thorgrimsen, Bob Woods, Lindsay Royan, Steve Davies, Margaret Butterworth (deceased), and Martin Orrell. 2003. Efficacy of an evidence-based cognitive stimulation therapy programme for people with dementia. *The British Journal of Psychiatry* 183, 3 (2003), 248–254. https://doi.org/10.1192/bjp.183.3.248

[34] Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, Yiannis Kompatsiaris, and Ioannis Vlahavas. 2014. SocialSensor: Finding Diverse Images at MediaEval 2014. (2014).

[35] Robert C Streijl, Stefan Winkler, and David S Hands. 2016. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. *Multimedia Systems* 22, 2 (2016), 213–227.

[36] Estefania Talavera, Mariella Dimiccoli, Marc Bolaños, Maedeh Aghaei, and Petia Radeva. 2015. R-clustering for egocentric video segmentation. In *Pattern Recognition and Image Analysis*. Springer, 327–336.

[37] Reinier H. van Leuken, Lluis Garcia, Ximena Olivares, and Roelof van Zwol. 2009. Visual Diversification of Image Search Results. In *Proceedings of the 18th International Conference on World Wide Web (WWW '09)*. ACM, New York, NY, USA, 341–350. https://doi.org/10.1145/1526709.1526756

[38] Cheng Xiang Zhai, William W Cohen, and John Lafferty. 2003. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM, 10–17.

[39] Xiangxin Zhu and Deva Ramanan. 2012. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2879–2886.

Serious Games Application for Memory Training Using Egocentric Images

Gabriel Oliveira-Barra[1], Marc Bolaños[1(✉)], Estefania Talavera[1,2], Adrián Dueñas[1], Olga Gelonch[3], and Maite Garolera[3]

[1] Universitat de Barcelona, Barcelona, Spain
marc.bolanos@ub.edu
[2] University of Groningen, Groningen, The Netherlands
[3] Consorci Sanitari de Terrassa, Terrassa, Spain

Abstract. Mild cognitive impairment is the early stage of several neurodegenerative diseases, such as Alzheimer's. In this work, we address the use of lifelogging as a tool to obtain pictures from a patient's daily life from an egocentric point of view. We propose to use them in combination with serious games as a way to provide a non-pharmacological treatment to improve their quality of life. To do so, we introduce a novel computer vision technique that classifies rich and non rich egocentric images and uses them in serious games. We present results over a dataset composed by 10,997 images, recorded by 7 different users, achieving 79% of F1-score. Our model presents the first method used for automatic egocentric images selection applicable to serious games.

Keywords: Lifelogging · Serious games · Egocentric vision
Mild cognitive impairment · Machine learning · Computer vision

1 Introduction

Dementia can result from different causes, the most common being Alzheimers disease (AD) [10], and it is often preceded by a pre-dementia stage, known as Mild Cognitive Impairment (MCI), characterized by a cognitive decline greater than expected by an individual's age, but which does not interfere notably with their daily life activities [11,19]. Currently, medical specialists design and apply special activities that could serve as a treatment tool for cognitive capabilities enhancement. Even though, these activities are not specially designed for the patients, which limits their engagement in some cases (Fig. 1).

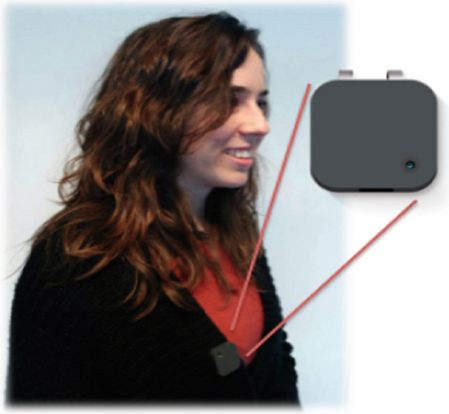

Fig. 1. Person using the Narrative Clip camera.

S. Battiato et al. (Eds.): ICIAP 2017 International Workshops, LNCS 10590, pp. 120–130, 2017.
https://doi.org/10.1007/978-3-319-70742-6_11

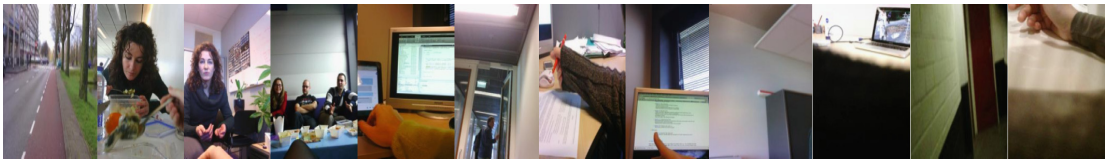

Fig. 2. Examples of egocentric images recorded by the Narrative Clip camera.

A possible alternative to the application of generic exercises would be the use of personalized images of the daily life of the patients acquired by lifelogging devices. Lifelogging consists of a user continuously recording their everyday experiences, typically via wearable sensors including accelerometers and cameras, among others. When the visual signal is the only one recorded, typically by a wearable camera, it is referred to as visual lifelogging [4]. This is a trend that is rapidly increasing thanks to advances in wearable technologies over recent years. Nowadays, wearable cameras are very small devices that can be worn all-day long and automatically record the everyday activities of the wearer in a passive fashion, from a first-person point of view. As an example, Fig. 2 shows pictures taken by a person wearing such a camera.

Recent studies have described wearable cameras or lifelogging technologies as useful devices for memory support for people with episodic memory impairment, such as the one present in MCI [8,15]. The design of new technologies to be applied on this field requires to take into account people capabilities, limitations, needs and the acceptance of the wearable devices, since it can directly affect the treatment. So far, some studies have deeply focus into the factors associated to the use of these devices [13,24].

Lifelogging and privacy: In terms of privacy, in 2011, the European Union agency ENISA evaluated the risks, threats and vulnerabilities of lifelogging applications with respect to central topics as privacy and trust issues. In their final report, they highlighted that lifelogging itself is still in its infancy but nevertheless will play an important role in the near future [3]. Therefore, they recommended further and extensive research in order to influence its evolution to be better prepared to mitigate the risks and maximize the benefits of these technologies. In addition, other researchers have also evaluated the possible ethical risks involved on using lifelogging devices on medical studies [7].

Serious games for MCI: Serious games (also known as games with a purpose) are digital applications specialized for purposes other than simply entertaining, such as informing, educating or enhancing physical and cognitive functions. Nowadays they are widely recognized as promising non-pharmacological tools to help assess and evaluate functional impairments of patients, as well as to aid with their treatment, stimulation, and rehabilitation [21]. Boosted by the publication of a Nature letter showing that video game training can enhance cognitive control in older adults [2], there is now a growing interest in developing serious

games specifically adapted to people with AD and related disorders. Preliminary evidence shows that serious games can successfully be employed to train physical and cognitive abilities in people with AD, MCI, and related disorders [17]. [18] performed a literature review of the experimental studies conducted to date on the use of serious games in neurodegenerative disorders and [21] studied recommendations for the use of serious games in people with AD and related disorders, reporting positive effects on several health-related capabilities of MCI patients such as voluntary motor control, cognitive functions like attention and memory or social and emotional functions. For instance they can improve their mood and increase their sociability, as well as reduce their depression.

Our contribution: Different studies have proven the benefits of directly stimulating the working memory. Our contribution in this paper consists in using as stimuli the autobiographical images of the MCI patients that was acquired by the wearable cameras. By doing this, we intend to accomplish the goal of enhancing their motivation and at the same time treat them in a more functional and multimodal manner [1,9,16]. The application, which will allow the user to exercise either at the sanitary center or at home, will be composed by serious games where the patient has to observe a series of images and interact with them.

Although the stimuli provided by egocentric images can be of greater importance than non-personal images, it is important to note both, that egocentric images are captured in an uncontrolled environment, and that wearable cameras usually have free motion that might cause most images to be blurry, dark or empty of semantic content. Considering this important limitations together with the limited capabilities of MCI patients, we propose the development of an egocentric rich images detection system intended to select only images with semantic and relevant content. Our hypothesis is that, by using personal daily life rich images, the motivation of the patient will increase, and as a consequence, the health-related benefits provided by the treatment.

This paper is organized as follows. We describe the proposed serious game and model for rich images selection in Sect. 2 and Sect. 3, respectively. In Sect. 4, we describes the experimental setup and show quantitative and qualitative evaluation. Finally, Sect. 5 draws conclusions and outlines future works.

2 Proposed Serious Game:"Position Recall"

MCI patients experiment problems in their working memory [23], therefore, it is of high importance to do exercises for stimulating it. All this under the neuroplasticity paradigm, which has proven that it is possible to modify the brain capabilities and the hypothesis of "use it or lose it", which are the basis of the studies related to the cognitive stimulation of elderly people [22]. Thus, in this work, we introduce a serious game that we name as "Position Recall", which was designed by neuropsychologist of Consorci Sanitari de Terrassa for improving the working memory. The mechanics of this game follow this scheme:

The first screen explains to the patient the instructions of the game and in the second the patient is informed that, before starting the game, there will be some practice examples that will serve to understand its logic. To start, the patient must select his preferred level of difficulty (Level 1, 2 or 3).

- **Level 1** shows 3 images of the patients' day during 8 s and they are asked to remember their positions. Immediately after they disappear, a single "target" image is shown and they are asked to select in what position it was placed. After some trials the number of images displayed are increased to 4 and then to 5.
- **Level 2** follows the same procedure as the 1st level, but the timespan between the moment where the images disappear and the target image is shown is increased. During this timespan, called latency time, a black screen is shown.
- **Level 3** follows the same procedure as the 2nd level, but now a distractor image is shown instead of a black screen during the latency time. The distractor image is also an image belonging to the patients' day.

The reward system of the game are points that are given after each level, and are calculated as $100x$*number of correct answers*. There are 10 trials per level translating into a maximum of 1000 points per level and maximum of 3000 points per game. Figure 3a and b show the mechanics of the developed game.

(a) (b)

Fig. 3. (a) A predefined number of pictures of the patient is shown to him during few seconds at random positions in the screen. (b) After a certain time passed, the patient is asked to recall in what position one of the pictures, picked up randomly, was placed before.

The images to be shown during the serious games should be significant for the patient. We propose to use images that represent past moments of the user's life, i.e. from the egocentric photostreams recorded by the patient. On the following section, we describe the proposed model for rich images selection.

3 What Did I See? Rich Images Detection

The main factor for providing a meaningful image selection algorithm is the fact that the proposed serious games intend to work on cognitive and sentiment enhancement. Considering the free-motion and non-intentionality of the pictures taken by wearable cameras [4], it is very important to provide a robust method for images selection.

Two of the most important and basic factors that determine the memorability of an image [5,14] can be described as (1) the appearance of human faces, and (2) the appearance of characteristic and recognizable objects. In this paper, we focus on satisfying the second criterion by proposing an algorithm based on computer vision. Our proposal consists in a rich images detection algorithm, which intends to detect images with a high number of objects and variability and at the same time avoids images with low semantical content, understanding as rich any image that is neither blur, nor dark and that contains clearly visible non-occluded objects. In Fig. 4 we show the general pipeline of our proposal.

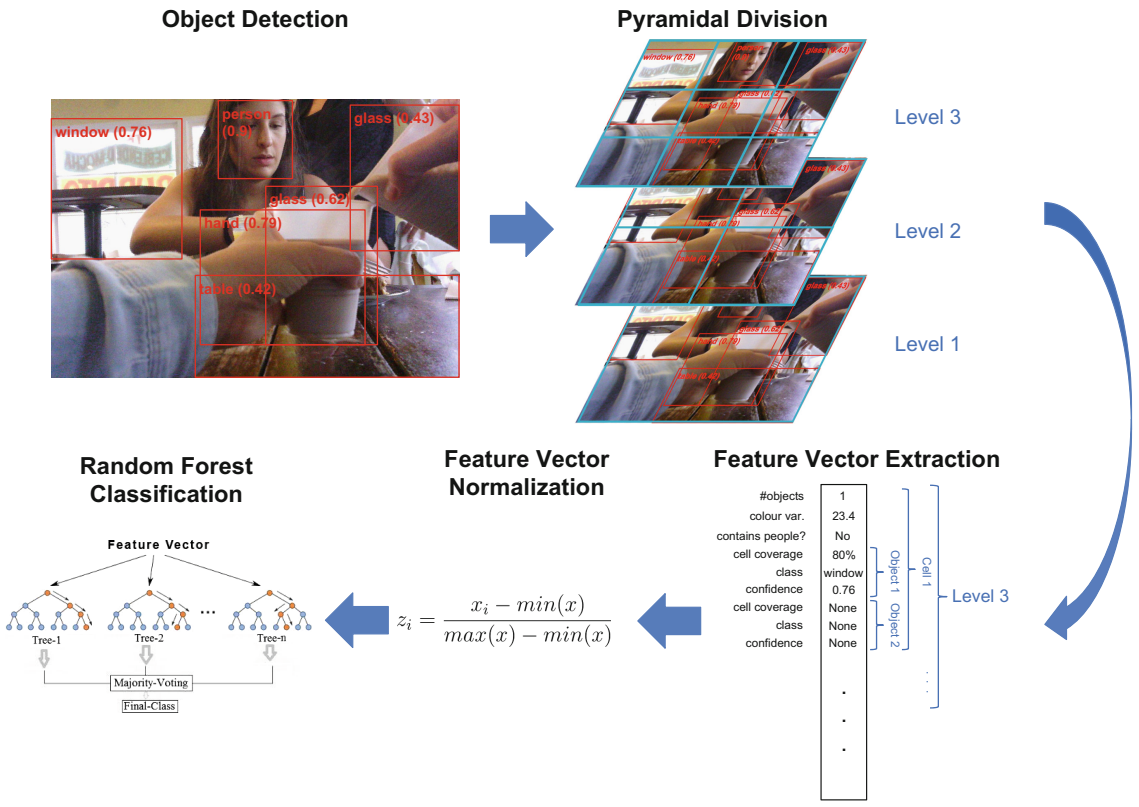

Fig. 4. Scheme of the proposed rich images detection model. (color figure online)

Our algorithm for rich images detection (consists in 1) objects detection: where the neural network named YOLO9000 [20] is applied in order to detect any existent object in the images and their associated confidences c_i. (2) the image is divided in a pyramidal structure of cells, (3) a set of richness-related features are

extracted, (4) the extracted features are normalized and (5) a Random Forest Classifier (RFC) [6] is trained to distinguish the differences between rich or non-rich images. When extracting features, the image is divided in a pyramidal structure of cells with different sizes at each level. The set of extracted features are:

- **Numbers of objects the cell contains.**
- **Variance of color in the cell.**
- **Does the cell contain people?**
- **Object Scale.** Real number between 0 and 1.
- **Object Class.** Class identifier that varies between 1 and 9418.
- **Object Confidence** c_i.

where all features are repeated for each cell and the last three kinds of features are repeated for each object appearing in the cells. The image cell divisions applied are $1x1$, $2x2$ and $3x3$, the maximum of objects selected per cell are 5, 3 and 2, respectively and all objects are sorted by their confidence c_i before selection. If the number of objects is less than the maximum number are found, the feature value in that specific position is set to 0.

The pyramidal division of the images helps us consider smaller objects at higher levels (more cells) and bigger objects at lower levels (less cells). Thus, both small and big objects will be considered for the final prediction.

In order to define the feature "Does the cell contain people?" We manually selected a set of person-related objects detected by the employed object detection method. The concepts representing people that we selected are "person", "worker", "workman", "employee", "consumer", "groom" and "bride".

4 Results

This section describes the results obtained in a quantitative and qualitative form. We compare the results obtained by variations of the proposed method on a self-made dataset of rich images.

Dataset: The dataset used for evaluating our model was acquired by the wearable camera Narrative Clip 2[1], which takes a picture every thirty seconds automatically. The camera was worn during 15 days by 7 different people. Considering that on average the camera takes 1,500 images per day, our dataset consists of 10,997 photographs.

The resulting data was labeled by neuropsychologist experts on MCI cognition following the criteria that any rich image has to be (1) properly illuminated, (2) not blurry and (3) contain one or more objects that are not occluded. After this manual selection the acquired images where split in 6,399 rich images and 4,598 non-rich images.

[1] www.getnarrative.com.

In Fig. 5a we can see some examples of egocentric rich images and in Fig. 5b non-rich images. We observe that rich images show people or recognizable places. However, non-rich images are meaningless or dark images (that can hardly be seen), including pictures of the sky, ceilings or floor.

(a) (b)

Fig. 5. (a) Rich images and (b) Non-rich images

The resulting data was divided in training, validation, and test. Considering the pictures taken during the same day can be very similar, we proceeded to randomly separate the different days into the three different sets. First, the training set consists of 60% of the days, in this case 9. Second, 20% of the days, in this case 3, were defined as the validation set. Finally, the remaining 20% was used for the test set.

Evaluation Metrics: In order to evaluate the different results and compare them to get the best one, we make use of the F1-score (or F-measure) metric:

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}} = 2 * \frac{precision * recall}{precision + recall}$$

where *precision* is the quotient between the number of True Positives objects and the number of predicted positive elements; and *recall* is the quotient between the number of True Positives objects and the number of real positive elements.

Quantitative Results: Currently, there are no previous works addressing the challenge we introduce in this work. Thus, in order to compare the performance of our proposed model, we have defined and compared several variations to our main pipeline (see results in Table 1).

As an alternative to our proposed approach (1), we tested an alternative feature vector representation by means of using the (2) Word2Vec word embedding [12]. This word characterization is a 300-dimensional vector representation created by Google that represents words in space depending on their semantic meaning (i.e. words with similar definitions will be represented close in space).

The Word2Vec representation was used in two ways. On the one hand was used for defining the set of concepts related to "person" in the feature described as "Does the cell contain people?". Thus, we computed the similarity between the word "person" and any other concept detected in the image by the object detection and the maximum similarity achieved was used as an alternative to a 0/1 representation. On the other hand, the feature described as "Object Class" was replaced by the 300-dimensions Word2Vec representation.

In the test setting (3) we additionally applied a PCA dimensionality reduction to the Word2Vec representation. Finally, in (4) we used a Support Vector Machine (SVM) classifier instead of a Random Forest Classifier. We applied a Grid Search on the variables C and *gamma* for parameter selection over the validation set.

Table 1. Comparison of the results

		Precision	Recall	F1-score
(1)	RFC	0.79	0.79	0.79
(2)	RFC + Word2Vec	0.78	0.78	0.78
(3)	RFC + Word2Vec + PCA	0.74	0.75	0.75
(4)	SVM	0.68	0.67	0.68

In conclusion we can see that using an RFC classifier (1) obtains better results than SVM (4) and at the same time none of the Word2Vec representations (2) and (3) helped improving the base results.

Fig. 6. Example of rich (left) images selection, vs non-rich images rejection. From an egocentric photostream composed by 972 images, 221 were considered rich.

Qualitative Results: Examples of the selected images by the proposed algorithm are shown in Fig. 6. On one hand, we can observe that rich images (left) are clearer, without shadows and with people or focused objects, which allows

the user to infer what is happening in the scene. On the other hand, non-rich images (right) are discarded since they are not illustrative and make difficult the scene interpretation.

Images selected by the proposed model are rich in information and memory trigger. We can foresee that the proposed model cannot only be used for serious games images selection, but also as a tool for images selection for autobiographical memories creation.

5 Conclusions

In this work, we have introduced a novel type of wearable computing application, aiming to provide non pharmacological treatment for MCI patients and to improve their life quality. We discussed lifelogging pictures obtained from wearable cameras combined with serious games as a channel for personalized treatments. We also introduced and tested a novel computer vision technique to classify rich and non rich images obtained from first-person point of view. We obtain 79% F1-score, promising results that will be further studied.

As future work, we will implement more serious games to be included in the application tool. Specialists will use it for MCI patients, aiming to prove the memory reinforcement hypothesis introduced in this work, as well as the motivation experienced by the subjects increase when using personalized rich images and serious games. Furthermore, in [25], positiveness from egocentric images was addressed. Moreover, we will go deeper on the analysis of users acceptance over the proposed technology, their willingness to use it, and the factors that determine their acceptance toward it. Further improvements of the methodology will be developed in order to obtain more accurate results.

Acknowledgements. This work was partially founded by Ministerio de Ciencia e Innovación of the Gobierno de España, through the research project TIN2015-66951-C2. SGR 1219, CERCA, *ICREA Academia 2014*, Grant 20141510 (Marató TV3) and Grant FPU15/01347. The funders had no role in the study design, data collection, analysis, and preparation of the manuscript. The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

1. Alves, J., Alves-Costa, F., Magalhães, R., Gonçalves, Ó.F., Sampaio, A.: Cognitive stimulation for Portuguese older adults with cognitive impairment: a randomized controlled trial of efficacy, comparative duration, feasibility, and experiential relevance. Am. J. Alzheimer's Dis. Other Dement.® **29**(6), 503–512 (2014)
2. Anguera, J.A., Boccanfuso, J., Rintoul, J.L., Al-Hashimi, O., Faraji, F., Janowich, J., Kong, E., Larraburo, Y., Rolle, C., Johnston, E., et al.: Video game training enhances cognitive control in older adults. Nature **501**(7465), 97–101 (2013)
3. Askoxylakis, I., Brown, I., Dickman, P., Friedewald, M., Irion, K., Kosta, E., Langheinrich, M., McCarthy, P., Osimo, D., Papiotis, S., et al.: To log or not to log?-Risks and benefits of emerging life-logging applications (2011)

4. Bolaños, M., Dimiccoli, M., Radeva, P.: Toward storytelling from visual lifelogging: an overview. IEEE Trans. Hum.-Mach. Syst. **47**(1), 77–90 (2017)
5. Carné, M., Giro-i-Nieto, X., Radeva, P., Gurrin, C.: Egomemnet: visual memorability adaptation to egocentric images
6. Criminisi, A., Shotton, J., Konukoglu, E., et al.: Decision forests: a unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. Found. Trends® Comput. Graph. Vis. **7**(2–3), 81–227 (2012)
7. Doherty, A.R., Hodges, S.E., King, A.C., Smeaton, A.F., Berry, E., Moulin, C.J., Lindley, S., Kelly, P., Foster, C.: Wearable cameras in health. Am. J. Prev. Med. **44**(3), 320–323 (2013)
8. Doherty, A.R., Pauly-Takacs, K., Caprani, N., Gurrin, C., Moulin, C.J., O'Connor, N.E., Smeaton, A.F.: Experiences of aiding autobiographical memory using the sensecam. Hum.-Comput. Interact. **27**(1–2), 151–174 (2012)
9. Flak, M.M., Hernes, S.S., Skranes, J., Løhaugen, G.C.: The memory aid study: protocol for a randomized controlled clinical trial evaluating the effect of computer-based working memory training in elderly patients with mild cognitive impairment (MCI). Trials **15**(1), 156 (2014)
10. Fratiglioni, L., Grut, M., Forsell, Y., Viitanen, M., Grafström, M., Holmen, K., Ericsson, K., Bäckman, L., Ahlbom, A., Winblad, B.: Prevalence of Alzheimer's disease and other dementias in an elderly urban population relationship with age, sex, and education. Neurology **41**(12), 1886–1886 (1991)
11. Gauthier, S., Reisberg, B., Zaudig, M., Petersen, R.C., Ritchie, K., Broich, K., Belleville, S., Brodaty, H., Bennett, D., Chertkow, H., et al.: Mild cognitive impairment. Lancet **367**(9518), 1262–1270 (2006)
12. Goldberg, Y., Levy, O.: word2vec explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722 (2014)
13. Gurrin, C., Smeaton, A.F., Doherty, A.R., et al.: Lifelogging: personal big data. Found. Trends® Inf. Retr. **8**(1), 1–125 (2014)
14. Khosla, A., Raju, A.S., Torralba, A., Oliva, A.: Understanding and predicting image memorability at a large scale. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2390–2398 (2015)
15. Lee, M.L., Dey, A.K.: Lifelogging memory appliance for people with episodic memory impairment. In: Proceedings of the 10th International Conference on Ubiquitous Computing, pp. 44–53. ACM (2008)
16. Li, H., Li, J., Li, N., Li, B., Wang, P., Zhou, T.: Cognitive intervention for persons with mild cognitive impairment: a meta-analysis. Ageing Res. Rev. **10**(2), 285–296 (2011)
17. Manera, V., Petit, P.D., Derreumaux, A., Orvieto, I., Romagnoli, M., Lyttle, G., David, R., Robert, P.H.: Kitchen and cooking, a serious game for mild cognitive impairment and Alzheimer's disease: a pilot study. Front. Aging Neurosci. **7** (2015)
18. McCallum, S., Boletsis, C.: Dementia games: a literature review of Dementia-related serious games. In: Ma, M., Oliveira, M.F., Petersen, S., Hauge, J.B. (eds.) SGDA 2013. LNCS, vol. 8101, pp. 15–27. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40790-1_2
19. Petersen, R.C., Smith, G.E., Waring, S.C., Ivnik, R.J., Tangalos, E.G., Kokmen, E.: Mild cognitive impairment: clinical characterization and outcome. Arch. Neurol. **56**(3), 303–308 (1999)
20. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. arXiv preprint arXiv:1612.08242 (2016)

21. Robert, P.H., König, A., Amieva, H., Andrieu, S., Bremond, F., Bullock, R., Ceccaldi, M., Dubois, B., Gauthier, S., Kenigsberg, P.A., et al.: Recommendations for the use of serious games in people with Alzheimer's disease, related disorders and frailty. Front. Aging Neurosci. **6** (2014)
22. Salthouse, T.A.: Mental exercise and mental aging: evaluating the validity of the use it or lose it hypothesis. Perspect. Psychol. Sci. **1**(1), 68–87 (2006)
23. Saunders, N.L., Summers, M.J.: Attention and working memory deficits in mild cognitive impairment. J. Clin. Exp. Neuropsychol. **32**(4), 350–357 (2010)
24. Sellen, A.J., Whittaker, S.: Beyond total capture: a constructive critique of lifelogging. Commun. ACM **53**(5), 70–77 (2010)
25. Talavera, E., Strisciuglio, N., Petkov, N., Radeva, P.: Sentiment recognition in egocentric photostreams. In: Alexandre, L.A., Salvador Sánchez, J., Rodrigues, J.M.F. (eds.) IbPRIA 2017. LNCS, vol. 10255, pp. 471–479. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58838-4_52

Journal of Visual Communication and Image Representation 50 (2018) 205–216

Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

journal homepage: www.elsevier.com/locate/jvci

Egocentric video description based on temporally-linked sequences☆

Marc Bolaños[a,b,*], Álvaro Peris[c], Francisco Casacuberta[c], Sergi Soler[a], Petia Radeva[a,b]

[a] Universitat de Barcelona, Barcelona, Spain
[b] Computer Vision Center, Bellaterra, Spain
[c] PRHLT Research Center, Universitat Politècnica de València, València, Spain

ARTICLE INFO

Keywords:
Egocentric vision
Video description
Deep learning
Multi-modal learning

ABSTRACT

Egocentric vision consists in acquiring images along the day from a first person point-of-view using wearable cameras. The automatic analysis of this information allows to discover daily patterns for improving the quality of life of the user. A natural topic that arises in egocentric vision is storytelling, that is, how to understand and tell the story relying behind the pictures.

In this paper, we tackle storytelling as an egocentric sequences description problem. We propose a novel methodology that exploits information from temporally neighboring events, matching precisely the nature of egocentric sequences. Furthermore, we present a new method for multimodal data fusion consisting on a multi-input attention recurrent network. We also release the *EDUB-SegDesc* dataset. This is the first dataset for egocentric image sequences description, consisting of 1339 events with 3991 descriptions, from 55 days acquired by 11 people. Finally, we prove that our proposal outperforms classical attentional encoder-decoder methods for video description.

1. Introduction

Egocentric vision [13,3,4] is a recent topic in the computer vision field, with the goal of analyzing the visual information provided by wearable cameras, which have the capability to acquire images or videos from a first person point-of-view. The analysis of these images can provide useful information about the behavior of the user for several complementary topics like social interactions, scene understanding, time-space-based localization, action recognition, nutritional habits, among others. Thus, enabling to understand the whole story and user's behavior behind the pictures (i.e. automatic storytelling) followed by inferring his/her actions and habits, could lead to a better quality of life for the user. Considering the sheer amount of data wearable cameras provide, there is a need to create automatic algorithms to analyze and summarize them. In this paper, we focus on the specific topic of creating automatic diaries of the life of the user by means of textual descriptions. One of the possible health-related applications for the automatic diary construction could be the treatment of patients with dementia. As proven by Spector et al. [45], Sellen et al. [42], the daily review of egocentric pictures taken by this kind of patients can help them recover partially their cognitive capabilities. The incorporation of additional automatically extracted textual information and comparing it to the user's one could give novel tools to complete cognitive frameworks for memory enhancement.

The egocentric captioning problem can be seen as an instantiation of a video description task [53]: the system receives as input a sequence of images and must produce a sentence as a sequence of words describing it. The problem is challenging due to two main reasons: on the one hand, most wearable cameras have a small field of view the other hand, lifelogging cameras have low temporal resolution (2–3 fpm) that is events are not videos, but collection of temporally ordered images. Hence, the problem is much more difficult than conventional videos and we need powerful techniques and algorithms from computer vision and natural language understanding to address the problem of video description. The recent development of deep learning techniques has allowed a breakthrough in the computer vision and natural language processing fields. The ability of training complex models has pushed forward many research areas and nowadays, Convolutional Neural Networks (CNNs) constitute one of the most powerful tools in the computer vision field. As the problem at hand involves sequence learning or prediction, a natural choice are Recurrent Neural Networks (RNNs), which are powerful sequence modelers. Specifically, the use of gated units, such as Long Short-Term Memory (LSTM) [19] or gated recurrent units [11] allows to properly model sequences with long and

☆ This paper has been recommended for acceptance by Mariella Dimiccoli.

* Corresponding author at: Universitat de Barcelona, Barcelona, Spain.
E-mail addresses: marc.bolanos@ub.edu (M. Bolaños), lvapeab@prhlt.upv.es (Á. Peris), fcn@prhlt.upv.es (F. Casacuberta), ssolerso8@alumnes.ub.ed (S. Soler), petia.ivanova@ub.edu (P. Radeva).

https://doi.org/10.1016/j.jvcir.2017.11.022
Received 21 March 2017; Received in revised form 9 November 2017; Accepted 28 November 2017
Available online 02 December 2017

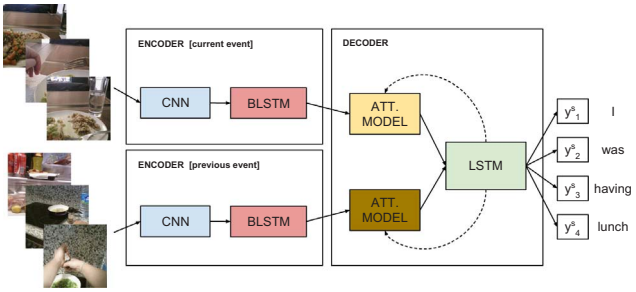

Fig. 1. General outline of the proposed temporally-linked multi-input attention model (TMA).

complex relationships.

The combination of RNNs together with CNNs is therefore widely employed for tackling multi-modal learning tasks, involving the combination of vision and language. Some examples of multi-modal learning problems are image or video description, dense captioning, visual question answering, multi-modal interaction to mention a few.

In this work, we develop a fully-neural end-to-end system for egocentric captioning that we call Temporally-linked Multi-input Attention model (TMA). We hypothesize that, within the egocentric sequences captioning problem, given a day, some of the events that compose it can follow a temporally logical relation. That is, previous actions occurred during a day can influence the following ones. Therefore, we need a model able to capture and learn this relation. Our proposal is able to embed previous information coming from either image or language. In Fig. 1, we show a simplified outline of the proposed method. In this example, we illustrate how previous event's frames can be employed in order to help to the process of video description of the analyzed event. Note that in the example, the user is cooking and next he proceeds to take the food. This temporal relationship between actions aids the model when predicting the current description.

Furthermore, we publish the first egocentric dataset for event captioning, composed of 55 complete days containing nearly 55,000 images in total, acquired by 9 different people.

The main contributions of this work are the following:

- We present a novel captioning model, which incorporates the information from previous events into the current decoding state.
- We present a new LSTM model capable of combining information from multiple inputs and modalities, as well as applying a separate attention mechanism to each of them.
- We conduct experiments on the new dataset and compare our model with other classical captioning architectures. Results show that using information from previous events provides better generalization.
- We present the first dataset for egocentric sequences captioning, named EDUB-SegDesc, based on describing the events that take place along a day.
- We make public both dataset and model, in order to make results reproducible and foster the research in this topic.

The rest of the paper is structured as follows: the related work is reviewed in Section 2. Next, we describe our model and its main components in Section 3. We present the egocentric captioning dataset in Section 4. We set up our experimental framework and show the obtained results in Section 5. Such results are analyzed in Section 6. Finally, conclusions and future research lines are drawn in Section 7.

2. Related work

In recent years, deep learning techniques have provided tremendous advances in the computer vision field. More precisely, CNNs [27] have proved to excel on the task of learning rich image representations and consequently have served as feature extractors, providing record-breaking results in most tasks. On the other hand, RNNs have proved to be powerful sequence modelers. They have been recently used in many sequences learning tasks, including machine translation [47,1], image [54] and video [55,35] captioning, or visual question answering [15,5]. Most of these problems involve tackling multi-modal data (i.e. text and images) [50], which means that, in all of them, both CNNs and RNNs are commonly employed.

Such systems are built under the encoder-decoder framework: an encoder processes the input and computes a meaningful representation of it. Next, the decoder takes this representation and produces the desired output, typically a sentence in natural language. Given their power as feature extractors, CNNs are usually employed as encoders [54,55,15]. Nevertheless, if the input signal has a sequential component (as in machine translation or video captioning), RNNs, solely or together with CNNs, can also act as encoders [47,35].

The vanilla encoder-decoder architecture was enhanced with attention mechanisms [1,54,55], allowing the decoder to selectively focus on parts of the input during the text generation process. Therefore, the system is able to properly deal with long and complex inputs.

Regarding the video description problem, which consists in generating a natural language description given an input video or sequence of images, several proposals have been published in recent years. The general framework used also consists of applying one or several CNNs to the input images, followed by a generative LSTM for sentence generation. Several improvements and variations have been proposed to the main model, some examples being the use of additional CNN representations like optical flow [53,55], Bidirectional LSTMs (BLSTM) in the encoder [35], attention mechanisms in the decoder [55], hierarchical information [32], external linguistic knowledge [52] or multi-modal attention mechanisms [20], among others.

All these works relating video captioning assume a sample-wise independence. This is that a sample is meant to be unrelated to the next one. This may represent a limitation in tasks which aim to model continuous events, arbitrarily split. The task addressed in this work belongs to this class of problems: we aim to describe the whole day of a user. For tackling it, we divide a day into events and describe each one of these events. Due to this division, our samples are conditioned between them. For example, if the user is at the office, it is likely that in the following event he/she uses a computer. Therefore, the classical sample-wise independence assumption becomes excessively severe, as critical information may be potentially lost. We propose a novel model that takes into account information from past events, being suitable for this kind of tasks.

Only one work has recently proposed to take into account mid-term temporal information in conventional videos [26]. Their model extracts C3D features on variable-sized conventional videos and computes several temporal segmentation proposals in short actions. Later, it generates textual descriptions for each action incorporating contextual information from past and future actions belonging to the same video. In comparison to our problem of egocentric lifelogging data, apart from the different perspective of the sequences, their videos are much shorter. Moreover, instead of consisting of several long-term events, theirs contain mid-term short actions belonging to the same event. Even though there are important differences to both problems, the authors prove that incorporating past and future information helps predicting the output of current events or actions.

In the field of egocentric vision [13,3,4], some authors have worked on related problems like activity recognition [22,6] or event classification [7]. The main handicap of these methods is that they provide simple labels from a user-oriented perspective, which provide more limited and predefined information compared to generating free textual descriptions. Regarding any of these methods, the first step needed in order to provide a coherent description of the actions and events happening in egocentric images is the application of an automatic segmentation [12,37,30] of the complete day of the user. After the

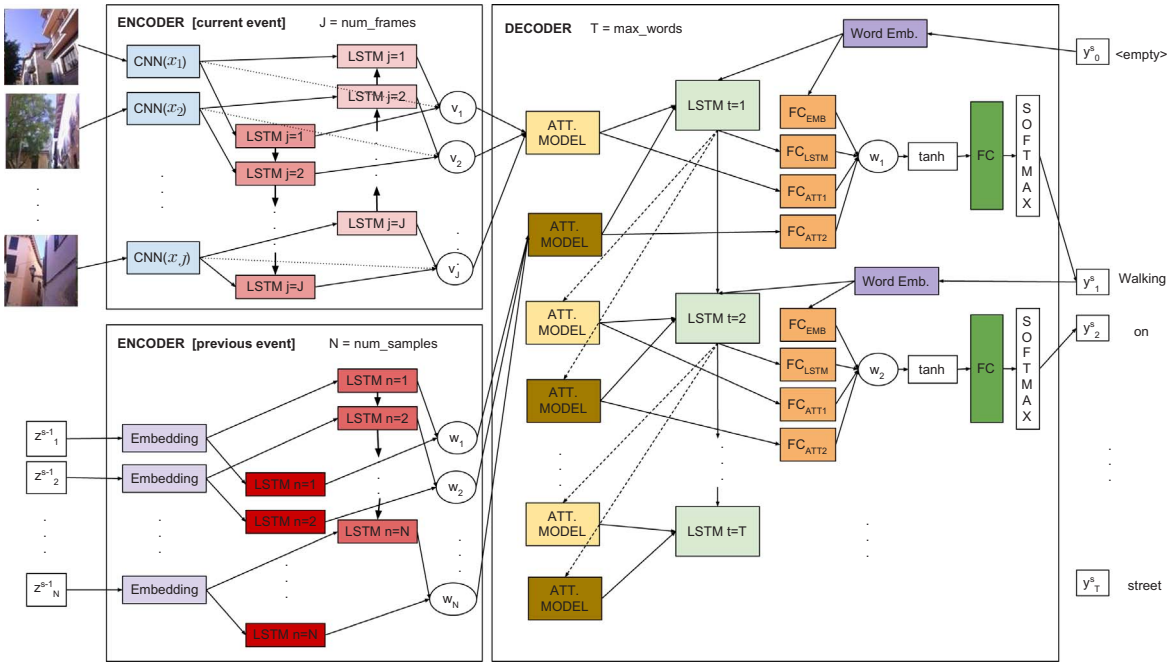

Fig. 2. Architecture of the proposed TMA model. Unlike the traditional approaches, our architecture consists of at least two encoder stages; one for the current sequence and another one (or more) for the previous sequence; and a decoder stage that combines the information of all the previous stages using a multi-input attention LSTM.

application of the day segmentation and having as output the set of events of the day, we can acquire the self-contained units of information needed in order to apply either event captioning, activity recognition or event classification models.

Considering the specific problem at hand, which is the generation of textual descriptions of egocentric events, few works have been proposed in the state-of-the-art. Only one of them tackled it as a video description problem and from an end-to-end perspective. Fan and Crandall [14] explored the problem of creating image diaries in order to apply image retrieval. As a step of its process, the authors proposed an image captioning method that processes one image at a time by applying a CNN for features extraction and a RNN for sentence generation. Later, they proposed grouping the images and applying a sentences fusion technique for providing the final captions for each event along the day. Goel and Naik [17] also focused on the task of image retrieval for both conventional and egocentric videos. It applied a simple method of video description composed of a CNN for feature extraction, a Bi-directional LSTM for image sequence combination and a LSTM in the decoder for applying the final sentence generation for 5-s-long clips of video. The purpose of this method in the work was to provide semantic information of the available data for the posterior retrieval.

In this work, we focus on event textual description. We assume that the events are previously extracted, manually or automatically [12]. To this aim, we propose an end-to-end model specifically trained for egocentric day sequences captioning. As argued before, it is important to take into account relevant information from previous samples, for generating descriptions of the current event. Thus, our model treats the image sequences from different events as temporally-linked units. It jointly models and exploits intra-eventual information (flowing through the frames of a single event) together with inter-eventual information (linking temporal sequences of neighboring events).

3. Methodology

The problem, we face in this work, involves a multi-modal learning task. On the one hand, we have a huge amount of information coming from the sequence of images captured by the wearable camera. On the other hand, each sequence of images has the correspondent caption associated, which describes the event occurring in the life of the user from an egocentric point of view.

Although the sequences of images may be segmented in independent semantic units to some extent (i.e. events), given the nature of the egocentric images, it is obvious that there exists a dependency between temporally neighboring events (i.e. most given events experienced in the day of a person have a relationship to the previous one he/she lived). We aim to exploit this information by incorporating into our model, at a given temporal point s, the information extracted from the previous event $s-1$. This information can be either textual (previous caption), visual (previous sequence of images) or both. Therefore, we develop a model that takes into account the information from both sources: the current sequence of frames together with the information coming from previous events.

3.1. Egocentric captioning

Similarly to the video description problem, in the egocentric task, we have as input a sequence of frames and we want to output a sentence that describes the input. This problem has already been tackled in the literature on conventional videos with multiple variations [32,53,52,55,35]. In the latter work, the frames were encoded by a CNN and a BLSTM network. The representation obtained was fed to an LSTM decoder equipped with an attention mechanism, which generated the corresponding caption. We use this architecture as starting point for developing our system.

The main difference that characterizes the egocentric captioning problem is that, unlike in the classical video description approach, we do have temporally-linked events, which share a relationship (e.g. if in an event the user enters into an office and sits on his/her table, it is highly likely that in the following event he/she will be using a computer).

Thus, we propose a system able to take advantage from both the current sequence of egocentric images and the action happened in the previous event. In Fig. 2 we detail the architecture of our model. The input to the system is a sequence of frames $X^s = x_1,...c,x_J$ and the sequence of information providing from the previous event, $Z^{s-1} = z_1^{s-1},...c,z_N^{s-1}$. This latter sequence can either be the previous

textual description, Y^{s-1} or the previous sequence of frames, X^{s-1}, or both. The current sequence of images is processed in the encoder of the current event, which is composed of a CNN (blue) and of a BLSTM (light red), producing the sequence of visual features $\mathbf{v}_1,...,c,\mathbf{v}_J$. Meanwhile, the data Z^{s-1} is processed in the encoder of the previous event, which is composed of an embedding mechanism (light purple) and of another BLSTM (dark red), which computes the final sequence of representations $\mathbf{w}_1,...c,\mathbf{w}_N$. The embedding mechanism used in this encoder can either be a CNN (blue) when dealing with images or a word embedding matrix (dark purple) when dealing with a textual representation. Note that this word embedding matrix is the same as the one in the decoder. These sequences of feature vectors are given as input to the decoder LSTM by means of two independent attention mechanisms (yellow and brown). In addition, at each time-step t, the decoder takes as input the word embedding of the previously generated word ($\mathbf{E}(y^s_{t-1})$). Finally, a series of skip connections is combined in an element-wise summation after embedding the different modalities by using fully-connected layers (orange). We include a skip connection for each attention mechanism, one for the LSTM hidden state and another one for the previously generated word. The sequence of time-steps for the current caption is indexed by t, while the sequence of events is indexed by s.

We extend the classical encoder-decoder approach by adding different encoders: one for the current sequence of images and another for the information from previous events. Such encoders are built using CNNs and BLSTMs networks, which we detail in Section 3.2. In Section 3.3, we proceed to explain the decoder, which is also extended by adding a multi-input attention LSTM for dealing with input sequences of multiple modalities. Finally, in Section 3.4, we summarize the whole pipeline applied in the proposed TMA model.

3.2. Encoder

3.2.1. Convolutional neural networks

CNNs have proved to be great modelers of image representations. Several models have been proposed in the state-of-the-art (e.g. AlexNet [27], VGG [43], GoogLeNet [48], ResNet [18], etc.), most of them originally for the problem of object recognition or detection [40]. Furthermore, they have been proven to serve as excellent feature extractors for other related tasks in the computer vision field.

Without loss of generality, in our proposal, we make use of the well-known GoogLeNet architecture [48] pre-trained on the ILSVRC challenge data as an extractor for training our complete TMA model. Although it varies depending on the problem and data, GoogLeNet has proven to be one of the best models for feature extraction, offering a trade-off between performance, number of parameters and computational time. We have to note that it is possible to use any alternative state-of-the-art CNN model for dealing with the images representation in our TMA model.

3.2.2. Long short-term memory and bidirectional networks

Since the input of our system is a sequence (of images), we can effectively model it and its relationships by using a RNN. More precisely, in order to learn a representation of the input sequences and its relationships, we use a LSTM network [19,16]. Unlike simple units, LSTM cells feature an additional state, the so-called memory state. The network controls how information flows through the unit by means of three gates, namely input, output and forget gates. Such gates modulate the amount of information that the network will incorporate from the input, from the output at the current time-step, or the amount that it will store for future time-steps. Therefore, LSTM networks are able to model long and complex sequences of information, reducing the vanishing gradient problem [2].

Hence, at a given time-step j, the hidden state $\mathbf{v}_j$ is the memory state $\mathbf{c}_j$ controlled by the output gate $\mathbf{o}_j$. The memory state depends on the previous memory state (modulated by the forget gate $\mathbf{f}_j$) and an updated memory state (modulated by the input gate $\mathbf{i}_j$). The updated state is obtained from the input $\mathbf{x}_j$ and the previous hidden state $\mathbf{v}_{j-1}$. All three gates are also computed from $\mathbf{x}_j$ and $\mathbf{v}_{j-1}$ [16].

With LSTM units, we manage how information flows through the network, and we are able to model long-term relationships. Nevertheless, this information only flows in a time direction. If we have the full day to analyze, we can take profit from relationships from the previous to the next events, but also from information flowing from next to previous events. Bidirectional networks [41] cope with both sequence directions by maintaining two independent recurrent layers: the forward layer processes the sequence from the past to the future and the backward layer processes the sequence reversed in time. Therefore, we can extract relationships flowing in both time directions of our input signal. Once the full sequence has been processed in both directions, forward and backward layers are combined, typically by concatenating their hidden states [1,35].

3.3. Decoder: multi-input attention LSTM

Since we are in a multi-modal scenario dealing with sequences of text and images, we propose to use an LSTM network that accepts multiple inputs—from multiple sources—and combines them after applying an attention mechanism for each input. The combination, as well as the attention mechanisms, are thus learned together with the rest of the model.

Attention mechanisms [1,54] help the decoder to selectively focus on parts of the input sequence, depending on the decoding step. More formally, given a sequence of J input vectors $\mathbf{v}_1,...,\mathbf{v}_J$ previously calculated by the encoder, at each decoding time-step t, the attention mechanism weights them into a single context vector $\mathbf{z}_t$:

$$\mathbf{z}_t = \sum_{j=1}^{J} \alpha_{jt}\mathbf{v}_j, \tag{1}$$

being α_{jt} the weight given to the j-th feature vector at time, t. At each time-step, the weights are calculated according to the scores obtained by a soft-alignment model, which determines how much a feature vector does influence the outcome of the current word. The aligner is implemented by a single-layer perceptron:

$$e_{jt} = \mathbf{w}^{\top}\tanh(\mathbf{W}_a\mathbf{h}_{t-1} + \mathbf{U}_a\mathbf{v}_j), \tag{2}$$

where $\mathbf{w}$,$\mathbf{W}_a$ and $\mathbf{U}_a$ are the perceptron parameters and $\mathbf{h}_{t-1}$ is the decoder hidden state from the previous time-step. For clarity, we omit the bias term in this equation, as well as in the rest of the paper. The aligner is then followed by a softmax function in order to obtain the final normalized weights, α_{jt}:

$$\alpha_{jt} = \frac{\exp(e_{jt})}{\sum_k^J \exp(e_{kt})}. \tag{3}$$

Our multi-input decoder is a natural extension of classical LSTM networks, which is able to process N different inputs. For simplicity, in this section we assume that we have inputs from two different modalities. Such inputs are processed by the respective attention models. Hence, at a given time-step, t, we have two inputs from the attention models ($\mathbf{z}_t$ and $\mathbf{z}'_t$). Furthermore, in order to boost the decoder capabilities, we introduce the word embedding of the previously generated word as an additional input. Fig. 3 illustrates the LSTM cell.

As in regular LSTMs, in our multi-input network, the hidden state depends on the memory state and the output gate:

$$\mathbf{h}_t = \mathbf{o}_t \odot \mathbf{c}_t, \tag{4}$$

where $\odot$ denotes the element-wise multiplication. $\mathbf{c}_t$ is the memory state, defined as:

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \tag{5}$$

where $\mathbf{c}_{t-1}$ is the previous time-step memory state and $\tilde{\mathbf{c}}_t$ is the updated memory state. $\mathbf{f}_t$ and $\mathbf{i}_t$ are the forget and input gates, which modulate

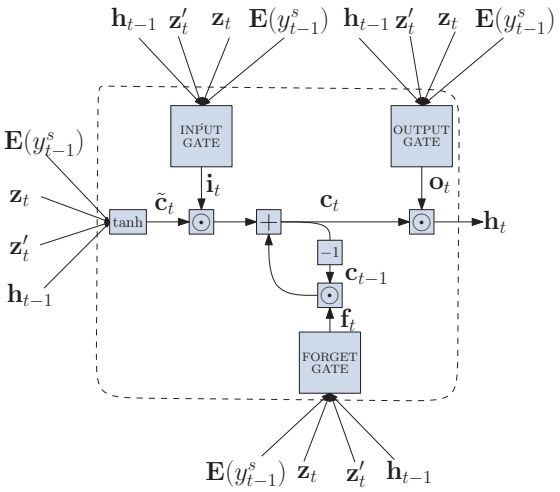

Fig. 3. Architecture of a multi-input LSTM cell at time-step, t. In this case, the inputs to the cell are the word embedding of the previous word ($\mathbf{E}(y^s_{t-1})$), the previous LSTM hidden state, ($\mathbf{h}_{t-1}$) and the context vectors from two different attention mechanisms, ($\mathbf{z}_t$,$\mathbf{z}'_t$).

them. $\tilde{\mathbf{c}}_t$ is computed taking into account the attended representations of the inputs ($\mathbf{z}_t$ and $\mathbf{z}'_t$), the last word generated by the decoder (y_{t-1}) and the previous hidden state ($\mathbf{h}_{t-1}$):

$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_c\mathbf{E}(y_{t-1}) + \mathbf{U}_c\mathbf{h}_{t-1} + \mathbf{A}_c\mathbf{z}_t + \mathbf{B}_c\mathbf{z}'_t), \tag{6}$$

where $\mathbf{E}$ is a word embedding matrix and $\mathbf{E}(y_{t-1})$ denotes the word embedding of the previously generated word. $\mathbf{W}_c$,$\mathbf{U}_c$,$\mathbf{A}_c$ and $\mathbf{B}_c$ are the weight matrices for the word embedding of y_{t-1}, the previous hidden state and both context vectors from the alignment attention models, respectively.

The forget, input and output gates also depend on the context vectors, the previous word and the previous hidden state:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f\mathbf{E}(y_{t-1}) + \mathbf{U}_f\mathbf{h}_{t-1} + \mathbf{A}_f\mathbf{z}_t + \mathbf{B}_f\mathbf{z}'_t), \tag{7}$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_i\mathbf{E}(y_{t-1}) + \mathbf{U}_i\mathbf{h}_{t-1} + \mathbf{A}_i\mathbf{z}_t + \mathbf{B}_i\mathbf{z}'_t), \tag{8}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o\mathbf{E}(y_{t-1}) + \mathbf{U}_o\mathbf{h}_{t-1} + \mathbf{A}_o\mathbf{z}_t + \mathbf{B}_o\mathbf{z}'_t). \tag{9}$$

3.4. Temporary-linked Multi-input attention model

Considering our TMA model as a whole, first we process the sequence of images captured by the camera for the current event s,$X^s = x^s_1,...,c,x^s_J$. As image encoder, we use a CNN that extracts features from X^s. Next, we apply a BLSTM network, in order to capture the temporal relationships existing in the sequence. Finally, for each frame x^s_j, we compute a feature vector $\mathbf{v}_j$ by concatenating the representation extracted by the CNN together with the forward and backward hidden states from the BLSTM.

Simultaneously, we process the previous event (Z^{s-1}). The information of the previous event can be either the previous sequence of egocentric images (X^{s-1}), the previous caption (Y^{s-1}), or both. In case of processing the previous sequence of images, the method is the same as the one applied on the current event: a CNN combined with a BLSTM network. If the information providing from the previous event is the caption $Y^{s-1} = y^{s-1}_1,...,cy^{s-1}_N$, words are projected to the continuous space by means of the embedding matrix, shared with the decoder. Next, an additional BLSTM processes this sequence of word embeddings. We concatenate the forward and backward BLSTM hidden states for obtaining a contextual representation for each input word. In case, we are using both modalities, the model is modified for having three different encoders: (1) current event frames X^s, (2) previous event frames X^{s-1} and (3) previous event output caption Y^{s-1}, as well as three attention mechanisms in the decoder, one for each input. For the sake of clarity and without loss of generality, let us assume from now on that we only have one modality as additional input from the previous event.

Therefore, our encoder produces two sequences of features: $\mathbf{v}_1,...,c\mathbf{v}_J$, referring to the current video and $\mathbf{w}_1,...,c,\mathbf{w}_N$, related to the previous event. We integrate this into the decoder by means of two independent attention mechanisms. Each attention mechanism weights the elements of its respective sequence and computes a joint representation of it, $\mathbf{z}_t$ and $\mathbf{z}'_t$ respectively, taking into account the previous decoding state $\mathbf{h}_{t-1}$ (see Section 3.3). Such representations, together with the previously generated word, are the inputs of the multi-input LSTM (described in Section 3.3).

Finally, at each time-step t, and in order to predict the following word in the output sequence, we define a probability distribution over the task vocabulary as follows:

$$\mathbf{p}_t = \mathrm{softmax}(\mathbf{U}_p\tanh(\mathbf{W}_a\mathbf{h}_t + \mathbf{W}_b\mathbf{z}_t + \mathbf{W}_c\mathbf{z}'_t + \mathbf{W}_d\mathbf{E}(y^s_{t-1}))), \tag{10}$$

where $\mathbf{h}_t$ is the hidden state from the decoder, $\mathbf{z}_t$ and $\mathbf{z}'_t$ are the contextual representations of the current and previous inputs computed by their respective attention models and $\mathbf{E}(y^s_{t-1})$ is the word embedding of the preceding word in the current sequence. $\mathbf{W}_a$,$\mathbf{W}_b$,$\mathbf{W}_c$,$\mathbf{W}_d$ are the skip-connections weight matrices (orange in Fig. 2). $\mathbf{U}_p$ is the last layer matrix (dark green[1] in Fig. 2).

The produced distribution, $\mathbf{p}_t$ represents the probability of a word given the input video X^s, the previous event Z^{s-1}, and the words generated so far in the current sequence $y^s_1,...,y^s_{t-1}$:

$$\mathbf{p}_t = p(y^s_t|X^s,Z^{s-1},y^s_1,...,y^s_{t-1}). \tag{11}$$

We approximate the most likely caption by using a beam search method [47]. The complete model parameters (θ) are jointly estimated over a dataset $\mathscr{S}$, which consists of S image sequences and caption pairs. The training objective is to maximize the log-likelihood of $\mathscr{S}$ with respect to θ:

$$\hat{\theta} = \arg\max_{\theta} \sum_{s=1}^{S} \sum_{t=1}^{T_s} \log(p(y^s_t|X^s,Z^{s-1},y^s_1,...,y^s_{t-1};\theta)), \tag{12}$$

where T_s is the length of the s-th caption. If the previous event, Z^{s-1} of a given image sequence X^s is undefined, we introduce an artificial empty event as Z^{s-1}.

4. EDUB-SegDesc dataset

EDUB-SegDesc[2] is a dataset that can be used either for egocentric events segmentation [12] or for egocentric sequences description. It was acquired by the wearable camera Narrative,[3] taking a picture every 30 s (2 fpm). It consists of 55 days acquired by 9 people. Each day was manually segmented in events or sequences following the same criteria as in Dimiccoli et al. [12]: "An event is a semantically perceptual unit that can be inferred by visual image features, without any prior knowledge of what the camera wearer is actually doing." In Fig. 4 a histogram with the duration (in minutes) of the resulting segments is shown. We can observe a wide variability in duration. Most of the segments (around 65%) have relatively short durations of 15 min or less, but there also exist several long events (around 5%) with a duration longer than 100 min.

The dataset contains a total of 48,717 images, divided in 1339 events (or image sequences) and 3991 captions, and has an average of 3 captions per event. It was divided in training, validation and test splits making sure that all the sequences from the same day should belong to the same data split. The division results in the figures depicted in

[1] For interpretation of color in Fig. 2, the reader is referred to the web version of this article.

[2] http://www.ub.edu/cvub/edub-segdesc.

[3] www.getnarrative.com.

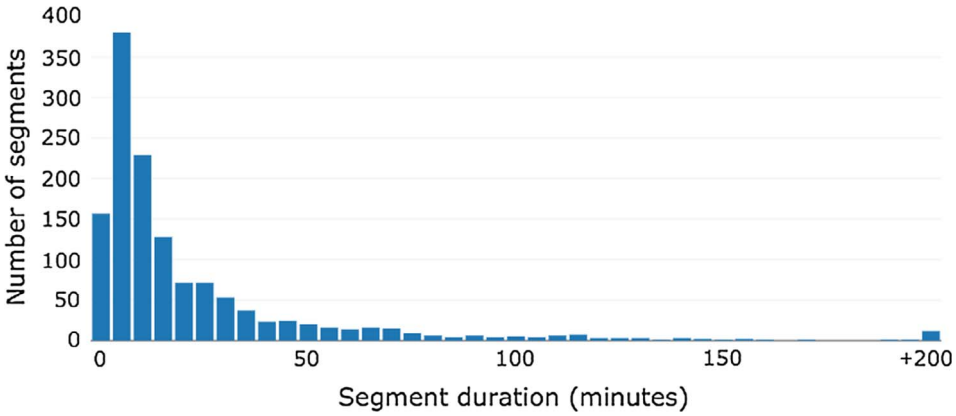

Fig. 4. Histogram depicting the duration of the segments in the EDUB-SegDesc dataset. All segments with a duration equal or greater than 200 min appear grouped in the last bin (+200).

Table 1
Figures of the EDUB-SegDesc dataset, according to each partition, training, validation and test.

EDUB-SegDesc	Training	Validation	Test	Total
#Days	39	7	9	55
#Images	32,664	7301	8752	48,717
#Segments	889	204	246	1339
#Descriptions	2652	598	741	3991

Table 1.

In Fig. 5 we show the number of co-occurrences in consecutive segments from some manually chosen keywords. This highlights the natural relationships found in consecutive events. Some notable examples of concepts appearing in consecutive events include: 'people' in past events followed by 'talked' in current events (social events); 'laptop' followed by 'work' (work-related events); 'street' followed by 'entered' (going from an outdoors to an indoors environment); 'station' followed by 'train' (transport-related events) or 'phone' followed by 'street' (events related to using the mobile phone on the street).

Fig. 6 shows several word-related statistics: Occurrences of the most common words and bigrams and histogram sentence lengths. Note that the number of appearances of the word 'I' is very high both in the single word and the bigram counts given the egocentric nature of the dataset. Other commonly occurring words and bigrams are the ones related to events where the user is 'walking' and/or on the 'street'. Curiously, there is a certain bias in the number of words contained in the annotations, where a considerable number of the sentences are composed of 5 words. Some examples of 5-word sentences are 'I went to my office', 'I worked with my laptop', or 'I walked on the street'.

Finally, we show some examples of the events and sentences contained in the dataset (Fig. 7). The low temporal resolution of the camera used (2 frames per minute) becomes clear in dynamic events, where the user moves and this causes to have highly variable environments. This fact, together with the limited information present in some of the images highlights the difficulty of the problem.

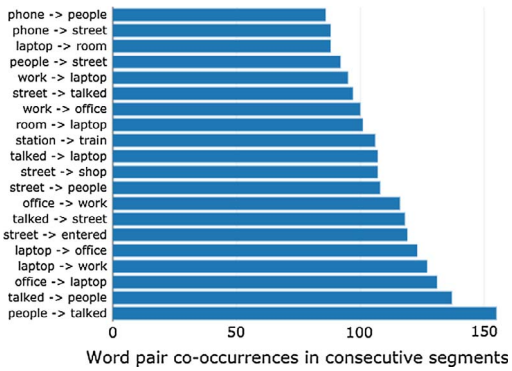

Fig. 5. Number of co-occurrences in consecutive segments of several manually chosen keywords. Word pairs are shown in the following format: `word1 -> word2`, where `word1` appears in the past event and `word2` appears in the current event.

The EDUB-SegDesc dataset was specifically acquired and labeled for the purpose of developing a model for describing and understanding all the events appearing along the day of a person. Thus, the main application of using egocentric sequences for textual descriptions generation is providing a memory aid for MCI patients.

5. Experiments and results

In this section we set up the experimental environment. We also define the metrics employed to evaluate it. Finally, we show the results obtained.

5.1. Metrics and evaluation

We evaluated our proposal using standard image and video captioning metrics. We used the COCO-Caption evaluation package [10] and computed three metrics:

BLEU [33]: compares the ratio of n-gram structures that are shared between the system hypotheses and the reference sentences. We report BLEU-4 in our results.
METEOR [28]: this metric was introduced to solve the lack of the recall component when computing BLEU. It computes the F1 score of precision and recall between hypotheses and references. In addition, it considers exact, stemmed, synonyms and paraphrase matches.
CIDEr [51]: similarly to BLEU, it computes the number of matching n-grams, but penalizing n-grams frequently found in the training set. The CIDEr metric ranges from 0 (minimum quality) to 10 (maximum quality).

5.2. Experimental setup

All the neural models that we compare were built with the Keras[4] and Theano [49] libraries. We release the source code of our implementation for future comparisons.[5] The full model was jointly trained end-to-end on the EDUB-SegDesc dataset, except for the CNN, which was already pre-trained for object detection on ImageNet [39] and remained static during the training of our model.

The main hyper-parameters of the model were selected according to the analysis reported in the video description task [35]. Moreover, since we conducted experiments with pre-trained models, we must keep fixed

[4] www.keras.io.
[5] https://github.com/MarcBS/TMA.

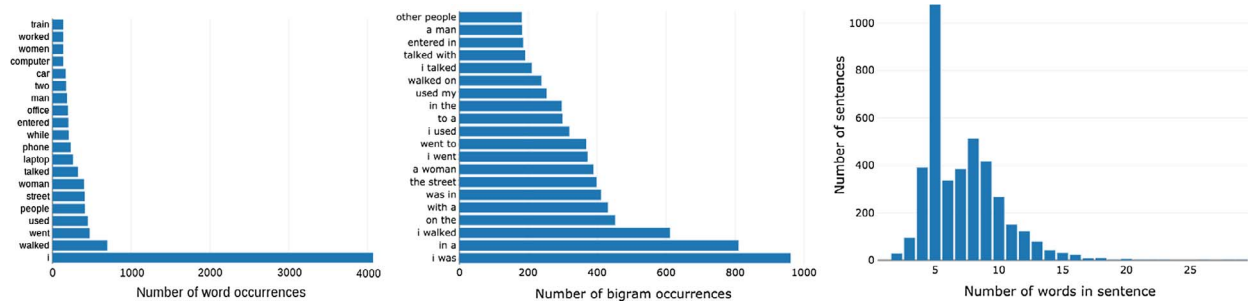

Fig. 6. Word-related statistics of the dataset. Occurrences of the most common words (left), occurrences of the most common bigrams (center), and histogram of number of words in all the sentences (right).

some hyper-parameters. Therefore, we used word embeddings of size 301, the encoder BLSTM had 717 units in the forward and 717 more units in the backward layers and the decoder LSTM had 484 units. The initial state of the decoder LSTM was initialized with the hyperbolic tangent of the mean of the video features obtained by the encoder [54]. We took at most 26 frames evenly distributed from each complete sequence of our dataset.

We trained our model by stochastic gradient descent (SGD). After doing experiments either using Adam or Adadelta optimizers, we observed that for some model variations, the optimal performance was reached either using Adadelta [56] with a learning rate of 1.0 and without learning rate decay; or using Adam [24] with an initial learning rate of 0.001 and a decay of 0.995 at the end of every epoch. Thus, we report the best results on each case. During training, the norm of the gradients was clipped to 10 [34].

In order to prevent over-fitting, we used batch normalization [21]. Contrarily to other works, we observed that the use of dropout [46] combined with batch normalization, produced better performance, ($p = 0.5$). We also applied weight decay (10^{-4}) and Gaussian noise ($\sigma = 10^{-2}$) to the non-recurrent weights.

We set our batch size to 64 and used an early stop criterion on the validation set based on BLEU-4, setting our patience to 20 and checking the performance each 50 updates. The size of the beam during the search was 10.

In order to minimize the influence of randomness in our results, mostly due to the weights random initialization, each experiment was run 5 times, and reported the median value of such runs. The model was trained either from scratch, exclusively using the data from the EDUB-SegDesc dataset or reusing the pre-trained weights from certain layers that were learned in different model variations. We studied the inclusion of word embeddings, obtained using the skip-gram model from Mikolov et al. [31] and trained on part of Google News dataset. We also tested training the decoder as a language model on the 1 Billion words dataset [8], but results in both cases were not better. Finally, we also tested reusing the pre-trained weights from the video captioning model from Peris et al. [35] (trained on the MSVD dataset published by Chen and Dolan [9]), which proved to improve the results under certain model configurations. The Microsoft Research Video Description Corpus (MSVD) dataset [9] contains 1,970 short clips from YouTube annotated by different users, accounting for more than 80,000 training samples. In terms of coverage, approximately the 98% of the words from EDUB-SegDesc are present in the MSVD dataset.

5.3. Results

In this section, we study the influence on the final performance of different architectural proposals. First, we show the performance of a classical video caption system and study the influence of several pre-training methods. Next, we study some of the architectural choices that led to the best TMA system. Finally, we compare the state of the art with our best proposals proposal based on the influence that information from previous events has on the system.

Table 2 shows the results obtained when using different pre-training techniques for the language model. In all the experiments we tackle the problem as classical video captioning without incorporating information from previous events. The captioning model is the same as in Peris et al. [35].

As shown in Table 2, the inclusion of word2vec vectors worsens the performance of the system in terms of BLEU and METEOR. We hypothesize that this is due to the different domains on which the word embeddings are trained. The word2vec vectors were trained on a more general domain and, therefore, their capabilities cannot be exploited to the full in our problem.

If we use the parameters learned from MSVD data, the BLEU score is also lowered, although in a lower extent than with word2vec. Nevertheless, the performance of the MSVD model in terms of METEOR and CIDEr is increased.

In Table 3 we report a further set of comparisons that show several additional tests that did not provide good results. In the first part of the table we compare several models either using or not dropout. It appears that, combining the use of dropout, batch normalization and Gaussian noise in the same model clearly helps obtaining better results in all the cases. The sole use of dropout as regularization strategy produced even worse results. In the second part of the table we compare either using all the images available in the dataset or removing all non-informative images [29] (i.e. images which are either dark, blurry or that point to the sky or ground without showing any object).

The effect of including information from previous events is shown in Table 4. For comparison, we also include the only similar approach in the egocentric video captioning literature, *DeepSeek* [17]. This model consists of a non-attentional BLSTM encoder with two layers. The encoder feeds a similar decoder to the one from Yao et al. [55]. We performed additional tests with several state-of-the-art models for video captioning: *Enc-Dec Global* [55], *hLSTMat* [44] and *ABiViRNet* [35]. Note that none of them considers long-term temporal information from different events.

As detailed in Section 3.3, we tested our TMA model introducing the previous caption, the previous sequence of images or both previous caption and images. As before, we distinguish between training from scratch with the EDUB-SegDesc dataset or start from the weights learned with MSVD pre-trained model. Given the results observed in Table 2, we drop the use of word2vec word embeddings.

According to Table 4, if we compare the results obtained by the state of the art methods (lines 1–4), which only consider the current sequence of images, to the different configurations of our method (lines 5–10), we can see that, in most of the cases, our method outperforms the rest. This behaviour can be explained by the inclusion of information from previous events in the TMA model. Since it considers a broader context, it is able to better understand the given event, and therefore, generally succeeds at increasing the performance of the system. Furthermore, considering the characteristics of egocentric lifelogging photostreams, the data provided by a single event often

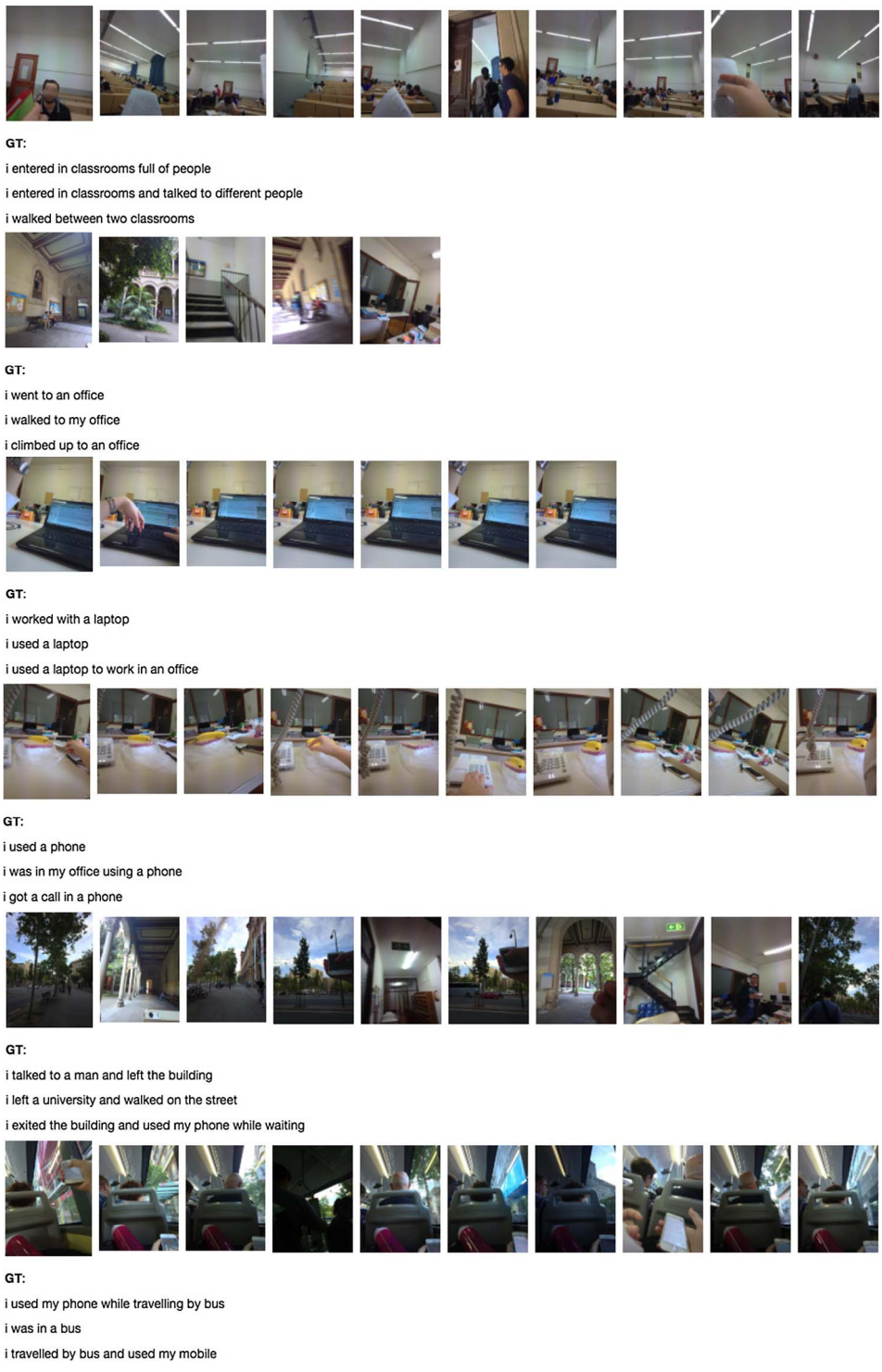

Fig. 7. Subset of a day from the dataset EDUB-SegDesc. We show some consecutive events and their respective GT sentences. The difficulty of the problem is highlighted by the dynamism and instability of the scene, when the user is moving. Particularly difficult examples can be seen in the second event: the GT sentences explain that the user is heading to his/her office, but this aspect is only manifested in the last image of the sequence.

lacks enough information to easily understand what is happening. Thus, providing context from previous events usually improves the captioning results. On the other hand, some state of the art methods outperform certain configurations of our model (see line 3 BLEU-4, and line 4 METEOR and CIDEr on the test set). We understand that this phenomenon can occur, because although providing context from previous events can usually be useful, in some cases noise could be introduced for two reasons: (1) the error of the predictions of previous events can be propagated, and (2) some consecutive events could lack any semantical relationship or have a very low number of samples in the training set.

Considering the differences between different configurations of our TMA model, the inclusion of the previous video event as input yields the best generalization results during test evaluation. The previous caption also enhances the system, but in a lower extent. On the other hand, the inclusion of both previous frames and caption produced better results, but only in the validation set. This phenomenon could imply that although the model has a greater potential, it is also more complex considering the number of parameters, which produces a quicker over-fitting on the training data and the consequent impact produced by the model selection of the validation set [38].

The results obtained with the TMA model show that taking into

Table 2
EDUB-SegDesc validation and test set results for different pre-training techniques on the language model. ABiViRNet refers to the video captioning system from Peris et al. [35]. BLEU and METEOR metrics are given in percentage. We compare the basic model trained from scratch with the same model with certain pre-trained components. #params denotes the number of parameters to estimate, given in millions.

	Validation			Test			#params
	BLEU-4	METEOR	CIDEr	BLEU-4	METEOR	CIDEr	
ABiViRNet	31.2	21.3	0.97	29.6	20.3	0.79	27.3 M
ABiViRNet + word2vec	30.1	21.0	1.04	26.0	20.1	0.90	27.3 M
ABiViRNet + MSVD	32.5	22.0	1.11	28.5	21.2	0.89	35.1 M

Table 3
EDUB-SegDesc validation and test set results for several additional configurations tested that did not provide good enough results. For each bad result we report its counter example configuration with better results. BLEU and METEOR metrics are given in percentage. #params denotes the number of parameters to estimate, given in millions. The best results for each measure are shown in boldface. Results with the symbol ∗ were obtained with the Adam optimizer instead of Adadelta.

	Validation			Test			#params
	BLEU-4	METEOR	CIDEr	BLEU-4	METEOR	CIDEr	
ABiViRNet no dropout	29.3	21.6	1.09	24.8	18.6	0.80	27.3 M
ABiViRNet	31.2	21.3	0.97	29.6	20.3	0.79	27.3 M
ABiViRNet + MSVD no dropout	30.2	21.0	1.05	27.9	19.4	0.89	35.1 M
ABiViRNet + MSVD	32.5	22.0	1.11	28.5	21.2	0.89	35.1 M
TMA previous-caption no dropout	31.9	21.6	1.02	28.1	20.1	0.94	39.1 M
TMA previous-caption	32.6	21.7	1.04	30.6	20.4	0.90	39.1 M
TMA previous-video + MSVD - NonInfo∗	31.7	21.7	1.13	30.3	21.5	0.99	51.0 M
TMA previous-video + MSVD∗	35.4	23.5	1.18	31.9	22.1	1.07	51.0 M

account the previous video event is more effective than considering the previous caption. This effect is partially produced because, at test time, we use as input an output previously generated by the model. Obviously, this output may be erroneous. Therefore, it may lead to the introduction of noise to the system and to error propagation.

From Tables 2–4, it can be concluded that, when fine-tuning from the MSVD model, the METEOR scores are always better than when starting from scratch, but BLEU-4 scores are lowered. This is probably due to vocabulary differences from both tasks. The model pre-trained tends to produce captions with the structure that it learned from the MSVD dataset, while the model trained from scratch on the EDUB-SegDesc generates ad hoc captions for the task at hand. Since BLEU-4 is based on n-gram counts, the latter model generally obtains larger values than the first one, because its captions are more literal. Since the METEOR metric stems and employs synonyms, it is less rigid than BLEU-4. Therefore, pre-trained models are able to score better than those trained from scratch.

6. Discussion

In this section, we review some illustrative examples, in order to understand the weaknesses and strengths of our model, as well as the major difficulties appeared in the task at hand. Fig. 8 shows some examples of the predictions produced by our model on consecutive events in the test set. We can observe the influence that the previous event had in the captioning of the second sample. In sample #2, the *TMA previous-video* model is aware that the user was previously in a restaurant. This conditions the model, which generates the caption *"I went to the bathroom"*, which is a likely action when a person is in a restaurant. The *ABiViRNet* model is unable to capture this information.

The influence of previous actions can also be observed in events #5 and #6. At event #5, the user is shopping and in the next event, he/she continues shopping. The *TMA previous-video* model can distinguish the shopping action occurred in event #6 due to incorporation of the visual information from the previous one. On the other hand, the *ABiViRNet*

Table 4
EDUB-SegDesc validation and test set results for variations of our TMA model compared to the state-of-the-art, which do not consider information from previous events. We use either the previous caption, the previous sequence of images (video) or both for making the current prediction. BLEU and METEOR metrics are given in percentage. #params denotes the number of parameters to estimate, given in millions. The best results for each measure are shown in boldface. Results with the symbol ∗ were obtained with the Adam optimizer instead of Adadelta.

	Validation			Test			#params
	BLEU-4	METEOR	CIDEr	BLEU-4	METEOR	CIDEr	
Enc-Dec Global [55]	30.1	21.9	1.02	28.1	20.8	0.88	44.4 M
hLSTMat [44]	31.6	23.3	1.28	25.6	20.8	0.88	–
ABiViRNet [35]	31.2	21.3	0.97	29.6	20.3	0.79	27.3 M
DeepSeek [17]	33.6	24.1	1.26	27.9	21.4	0.99	27.2 M
TMA previous-caption	32.6	21.7	1.04	30.6	20.4	0.90	39.1 M
TMA previous-caption + MSVD	33.9	23.5	1.21	28.3	21.3	0.94	46.9 M
TMA previous-video∗	34.4	23.7	1.21	31.0	21.4	1.01	43.3 M
TMA previous-video + MSVD∗	35.4	23.5	1.18	**31.9**	**22.1**	**1.07**	51.0 M
TMA previous-video-caption	**36.4**	**24.6**	**1.29**	30.4	21.8	1.00	55.1 M
TMA previous-video-caption + MSVD∗	34.0	23.0	1.19	29.7	**22.1**	0.93	62.8 M

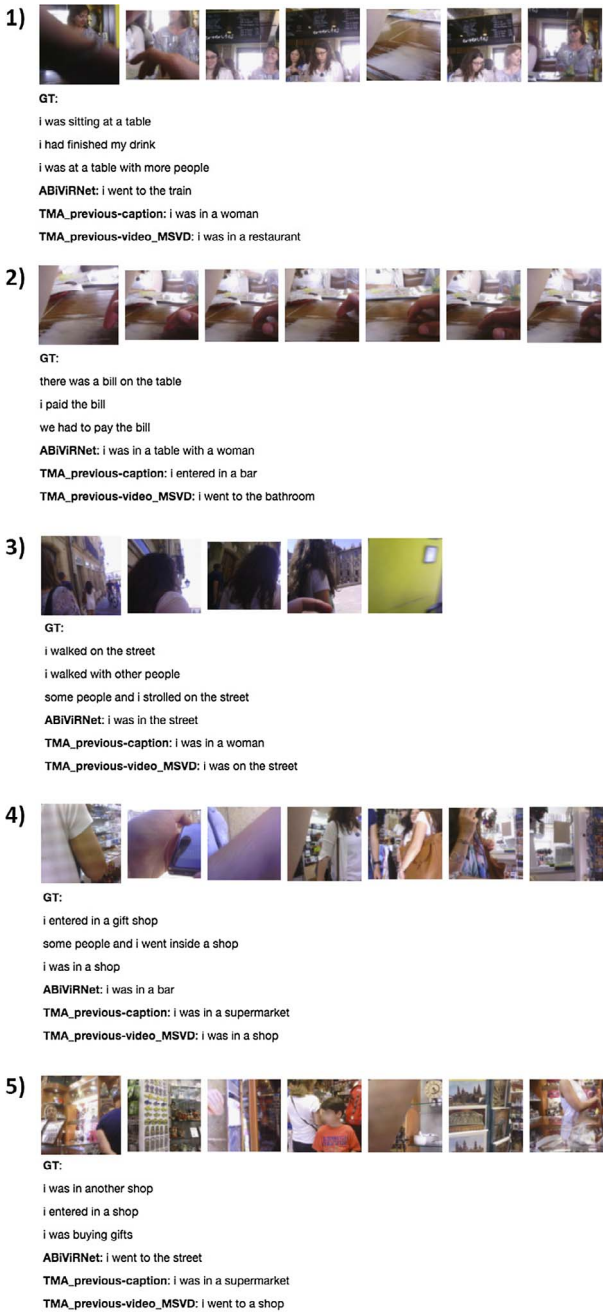

Fig. 8. Comparison of the results obtained by the baseline method ABiViRNet and two of our proposals TMA previous-caption and TMA previous-video + MSVD. A set of consecutive events from the test split is shown.

model is not able to do this inference and deduces that the user is in the street. Similarly, the *TMA previous-caption* model is more prone to error propagation: when fails to understand a certain event, it is more likely to be also wrong in the following one. For example, in sample #4 the model generates *"I was in a supermarket"*, which leads the model to fail also on the consecutive event #5. Fig. 9 depicts some examples of initial events of different days. We can see that, in the case of TMA models, the fact that information from previous events can not be used in this particular cases does not influence the model and obtains better or comparable results than non-TMA models. Fig. 10 shows additional examples of successes/failures of the system. In the success case #1, we can see that the model is able to understand that the user went to the bathroom, although it is not specifically described in the GT and it is hard to distinguish just looking at the images (note the third image

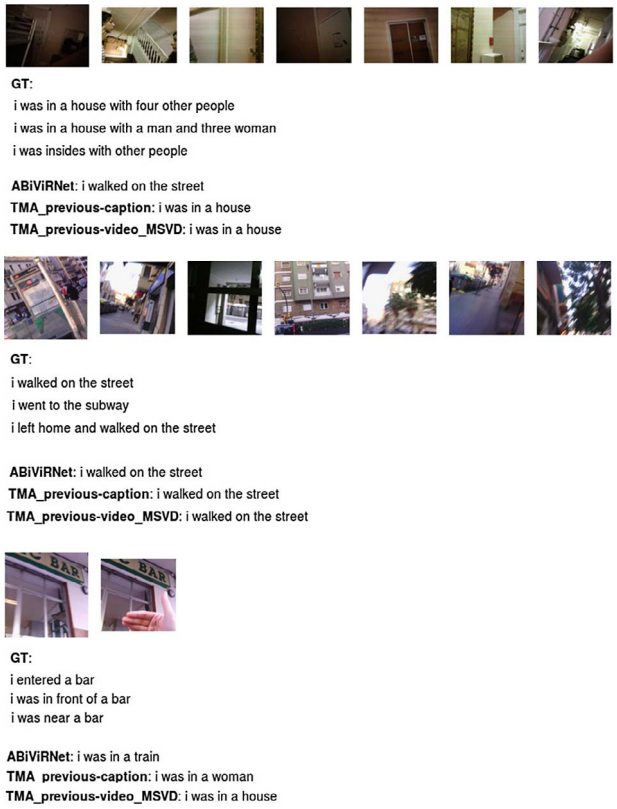

Fig. 9. Comparison of the results obtained by the baseline method ABiViRNet and two of our proposals TMA previous-caption and TMA previous-video + MSVD. Events corresponding to the start of the day are shown.

where the user is drying his hands). In addition, this exemplifies one of the major challenges occurred within the event captioning problem: during an event, many situations may occur. The system may focus only on part of the event, resulting in captions that, although correct, are not found in the GT.

In the success case #2, we can see that both *ABiViRNet* and *TMA previous-video* model correctly describe the first event. But as we move to the next one, the first model is unable to detect that the user is having lunch, while the TMA model is capable to infer that the event has changed. Therefore, it incorporates this information and correctly guesses that the user is having lunch.

The right examples show failure cases caused under certain conditions in the input images. In the first failure case, the model infers that the user is in a supermarket due to the vending machine in the first image. In the second failure case, the model interprets the images as a parade due to the multiple persons and colored lights appearing.

7. Conclusions and future work

In this work, we addressed the challenging problem of egocentric captioning as a video description task considering a natural characteristic of this problem: since the egocentric sequences were captured consecutively along a day, there exists a relationship between a given situation and the previous one. We aimed to include such dependency in an automatic captioning system. For doing this, we developed a natural extension of LSTM networks, able to deal with multiple inputs, even when coming from different modalities.

For assessing our proposal, we also constructed a dataset for egocentric captioning. Image sequences were obtained using a wearable camera and were manually segmented and annotated. Both source code and dataset are made publicly available. We carried out an automatic evaluation of the system, with clear results: the inclusion of previous

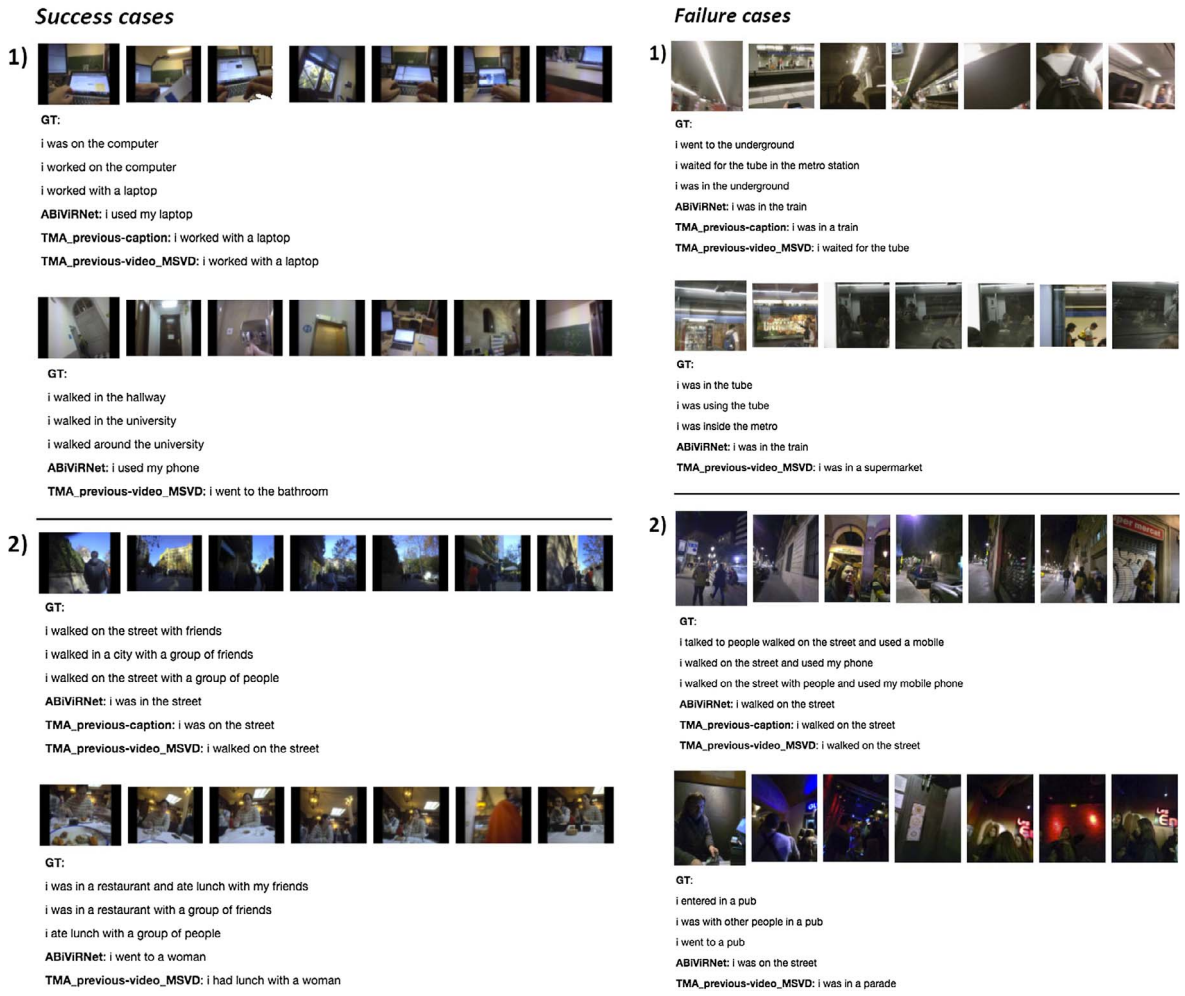

Fig. 10. Two success cases (left) and two failure cases (right) of our model *TMA previous-video + MSVD*. All the examples belong to the test set. For each case, the top sequence is the previous event and the bottom sequence is the event that we are predicting. In success case #1, we can see that our method correctly recognizes that the user went to the bathroom even though it is not specified in the GT sentences and it is not straightforward to see in the images (note the third image using a hand drier). The two samples at the right exemplify cases in which our model fails due to certain images appearing in the sequences. For instance, in the failure case #1 our model considers that the user is in a supermarket due to the first image, where a vending machine appears. In failure case #2, our model considers that the user is seeing a parade due to the multiple people and colored lights appearing in the images.

information effectively enhances the performance of the system.

If we have available the sequence of events of a full day, we could introduce not only previous events to the system, but also the following ones. Since the TMA model defined in this work supports an arbitrary number of inputs, the inclusion of the context coming from following events becomes natural. Therefore, the captioning results could be refined, not only by the previous, but also by incorporating the following information. Furthermore, we could aim to look further than the immediately previous event, incorporating longer-term memory. Very recently, Kaiser et al. [23] proposed a memory-augmented network, able to learn very long-term relationships. We plan to test whether such architectures can deal with our problem in an effective way.

Additionally, in a hypothetical real application, the inclusion of an interactive correction process could be considered. Interactive neural systems provide encouraging results [25,36] in the machine translation field. In such systems, the user corrects the errors committed by the system, the system takes into account these corrections and changes its outputs, to produce a hopefully better caption. Finally, closely related with interactivity, another interesting extension of this work is the application of online learning techniques to the egocentric captioning pipeline. This would allow the development of better user-tailored and adaptive captioning systems, definitely more useful to the final user.

Acknowledgments

This work was partially founded by TIN2015-66951-C2, SGR 1219, CERCA, Grant 20141510 (Marató TV3), PrometeoII/2014/030 and R-MIPRCV network (TIN2014-54728-REDC). Petia Radeva is partially founded by ICREA Academia'2014. Marc Bolaños is partially founded by an FPU fellowship. We gratefully acknowledge the support of NVIDIA Corporation with the donation of a Titan X GPU used for this research. The funders had no role in the study design, data collection, analysis, and preparation of the manuscript.

References

[1] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, 2015. Also available at: arXiv: < 1409.0473 > .
[2] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Networks 5 (2) (1994) 157–166.
[3] A. Betancourt, P. Morerio, C.S. Regazzoni, M. Rauterberg, The evolution of first person vision methods: a survey, IEEE Trans. Circ. Syst. Video Technol. 25 (5) (2015) 744–760.
[4] M. Bolaños, M. Dimiccoli, P. Radeva, Toward storytelling from visual lifelogging: An overview, IEEE Trans. Human-Mach. Syst. 47 (1) (2017) 77–90.
[5] M. Bolaños, Á. Peris, F. Casacuberta, P. Radeva, VIBIKNet: Visual bidirectional kernelized network for visual question answering, 2016. Also available at: arXiv: < 1612.03628 > .
[6] A. Cartas, M. Dimiccoli, P. Radeva, Batch-based activity recognition from egocentric photo-streams, 2017. arXiv preprint Also available at: arXiv: < 1708.07889 > .

[7] D. Castro, S. Hickson, V. Bettadapura, E. Thomaz, G. Abowd, H. Christensen, I. Essa, Predicting daily activities from egocentric images using deep learning, Proceedings of the 2015 ACM International symposium on Wearable Computers, ACM, 2015, pp. 75–82.

[8] C. Chelba, T. Mikolov, M. Schuster, Q. Ge, T. Brants, P. Koehn, T. Robinson, One billion word benchmark for measuring progress in statistical language modeling, 2013. Also available at: arXiv: < 1312.3005 > .

[9] D.L. Chen, W.B. Dolan, Collecting highly parallel data for paraphrase evaluation, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2011, pp. 190–200.

[10] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, C.L. Zitnick, Microsoft COCO captions: Data collection and evaluation server, 2015. Also available at: arXiv: < 1504.00325 > .

[11] K. Cho, B. van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, 2014. Also available at: arXiv: < 1409.1259 > .

[12] M. Dimiccoli, M. Bolaños, E. Talavera, M. Aghaei, S.G. Nikolov, P. Radeva, SR-clustering: Semantic regularized clustering for egocentric photo streams segmentation, Comput. Vis. Image Understand. 155 (2017) 55–69.

[13] A.R. Doherty, S.E. Hodges, A.C. King, A.F. Smeaton, E. Berry, C.J. Moulin, S. Lindley, P. Kelly, C. Foster, Wearable cameras in health, Am. J. Prevent. Med. 44 (3) (2013) 320–323.

[14] C. Fan, D.J. Crandall, Deepdiary: Automatically captioning lifelogging image streams, in: Proceedings of European Conference on Computer Vision, 2016, pp. 459–473.

[15] A. Fukui, D.H. Park, D. Yang, A. Rohrbach, T. Darrell, M. Rohrbach, Multimodal compact bilinear pooling for visual question answering and visual grounding, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2016, pp. 457–468.

[16] F.A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: continual prediction with LSTM, Neural Comput. 12 (10) (2000) 2451–2471.

[17] K. Goel, J. Naik, Deepseek: a video captioning tool for making videos searchable, 2016.

[18] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[19] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[20] C. Hori, T. Hori, T.-Y. Lee, K. Sumi, J.R. Hershey, T.K. Marks, Attention-based multimodal fusion for video description, 2017. Also available at: arXiv: < 1701.03126 > .

[21] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, Proc. Int. Conf. Mach. Learn. 32 (2015) 448–456.

[22] Y. Iwashita, A. Takamine, R. Kurazume, M.S. Ryoo, First-person animal activity recognition from egocentric videos, 22nd International Conference on Pattern Recognition (ICPR), 2014, IEEE, 2014, pp. 4310–4315.

[23] Ł. Kaiser, O. Nachum, A. Roy, S. Bengio, Learning to remember rare events, 2017. Also available at: arXiv: < 1703.03129 > .

[24] D. Kingma, J. Ba, Adam: a method for stochastic optimization, 2014. Also available at: arXiv: < 1412.6980 > .

[25] R. Knowles, P. Koehn, Neural interactive translation prediction, in: Proceedings of the Association for Machine Translation in the Americas, 2016, pp. 107–120.

[26] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, J.C. Niebles, Dense-captioning events in videos, 2017. arXiv preprint Also available at: arXiv: < 1705.00754 > .

[27] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Neural Information Processing Systems Conference, 2012, pp. 1097–1105.

[28] A. Lavie, M.J. Denkowski, The METEOR metric for automatic evaluation of machine translation, Mach. Transl. 23 (2–3) (2009) 105–115.

[29] A. Lidon, M. Bolanos, M. Dimiccoli, P. Radeva, M. Garolera, X. Giró-i Nieto, Semantic summarization of egocentric photo stream events, 2015. Also available at: arXiv: < 1511.00438 > .

[30] Z. Lu, K. Grauman, Story-driven summarization for egocentric video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013 pp. 2714–2721.

[31] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proceedings of the Neural Information Processing Systems Conference, 2013, pp. 3111–3119.

[32] P. Pan, Z. Xu, Y. Yang, F. Wu, Y. Zhuang, Hierarchical recurrent neural encoder for video representation with application to captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1029–1038.

[33] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, BLEU: a method for automatic evaluation of machine translation, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2002, pp. 311–318.

[34] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, Proc. Int. Conf. Mach. Learn. 28 (2013) 1310–1318.

[35] Á. Peris, M. Bolaños, P. Radeva, F. Casacuberta, Video description using bidirectional recurrent neural networks, in: Proceedings of the International Conference on Artificial Neural Networks, 2016, pp. 3–11.

[36] Á. Peris, M. Domingo, F. Casacuberta, Interactive neural machine translation, Comput. Speech Lang. 45 (2017) 201–220.

[37] Y. Poleg, A. Ephrat, S. Peleg, C. Arora, Compact cnn for indexing egocentric videos, IEEE Winter Conference on Applications of Computer Vision, IEEE, 2016, pp. 1–9.

[38] J. Reunanen, Overfitting in making comparisons between variable selection methods, J. Mach. Learn. Res. 3 (Mar) (2003) 1371–1382.

[39] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al.. Imagenet large scale visual recognition challenge, 2014. Also available at: arXiv: < 1409.0575 > .

[40] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[41] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.

[42] A.J. Sellen, A. Fogg, M. Aitken, S. Hodges, C. Rother, K. Wood, Do life-logging technologies support memory for the past?: an experimental study using sensecam, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2007, pp. 81–90.

[43] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014. Also available at: arXiv: < 1409.1556 > .

[44] J. Song, Z. Guo, L. Gao, W. Liu, D. Zhang, H.T. Shen, Hierarchical lstm with adjusted temporal attention for video captioning, 2017. arXiv preprint Also available at: arXiv: < 1706.01231 > .

[45] A. Spector, L. Thorgrimsen, B. Woods, L. Royan, S. Davies, M. Butterworth, M. Orrell, Efficacy of an evidence-based cognitive stimulation therapy programme for people with dementia, Brit. J. Psychiat. 183 (3) (2003) 248–254.

[46] N. Srivastava, G.E. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.

[47] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, in: Proceedings of the Neural Information Processing Systems Conference, Vol. 27, 2014, pp. 3104–3112.

[48] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[49] Theano Development Team, Theano: A Python framework for fast computation of mathematical expressions, 2016.

[50] A.H. Toselli, E. Vidal, F. Casacuberta, Multimodal Interactive Pattern Recognition and Applications, Springer Science & Business Media, 2011.

[51] R. Vedantam, C. Lawrence Zitnick, D. Parikh, Cider: Consensus-based image description evaluation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, p. 4566–75.

[52] S. Venugopalan, L.A. Hendricks, R. Mooney, K. Saenko, Improving LSTM-based video description with linguistic knowledge mined from text, 2016. Also available at: arXiv: < 1604.01729 > .

[53] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, K. Saenko, Sequence to sequence-video to text, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4534–4542.

[54] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: Proceedings of the International Conference on Machine Learning, 2015, pp. 2048–2057.

[55] L. Yao, A. Torabi, K. Cho, N. Ballas, , Pal, C., Larochelle, H., Courville, A., 2015. Describing videos by exploiting temporal structure. In: Proceedings of the International Conference on Computer Vision. pp. 4507–4515.

[56] M.D. Zeiler, Adadelta: an adaptive learning rate method, 2012. Also available at: arXiv: < 1212.5701 > .

Chapter 4

Deep Learning on Food Recognition

In this chapter you can find all the papers related to deep learning on Food Analysis and Food Recognition. All the articles presented can be divided in four different groups.

The four groups contain either general topics like advanced data annotation techniques for acquiring training data samples; food detection, localization and recognition; or advanced recognition techniques like classifiers fusion or multi-task learning. Or more specific topics as in the last group, which is devoted to food recognition on self-service restaurants and to food and restaurant recognition for smartphone users.

Data Annotation
Active Labeling (Bolaños, Garolera, and Radeva, 2013)

Detection, Localization and Recognition
Food Detection (Aguilar, Bolanos, and Radeva, 2017)
Catalan Diet Recognition (Herruzo, Bolaños, and Radeva, 2016)
Simultaneous Food Localization and Recognition (Bolanos and Radeva, 2016)
Ingredients Recognition (Bolaños, Ferrà, and Radeva, 2017)

Advanced Recognition Techniques: Multi-task and Fusion
Fusion of Classifiers (Aguilar, Bolaños, and Radeva, 2017)
Multi-task for Food Analysis (Aguilar, Bolaños, and Radeva, 2019)

Food Recognition in Restaurants
Grab, Eat and Pay (Aguilar et al., 2018)
Where and What Am I Eating? (Bolaños, Valdivia, and Radeva, 2018)

Active Labeling Application Applied to Food-Related Object Recognition

Marc Bolaños
Dept. MAIA, U. de Barcelona
Gran Via de les Corts Catalanes 585
Barcelona 08007, Spain
mark.bs.1991@gmail.com

Maite Garolera
Neeuropychology Unit,
Consorci Sanitari de Terrassa
Ctra.Torrebonica s/n
08227, Terrassa, Spain
mgarolera@cst.cat

Petia Radeva
Dept. MAIA, U. de Barcelona
Gran Via 585, Barcelona
08007, Spain
CVC, Campus UAB, Bellaterra
(Barcelona), Spain
petia.ivanova@ub.edu

ABSTRACT

Every day, lifelogging devices, available for recording different aspects of our daily life, increase in number, quality and functions, just like the multiple applications that we give to them. Applying wearable devices to analyse the nutritional habits of people is a challenging application based on acquiring and analyzing life records in long periods of time. However, to extract the information of interest related to the eating patterns of people, we need automatic methods to process large amount of life-logging data (e.g. recognition of food-related objects). Creating a rich set of manually labeled samples to train the algorithms is slow, tedious and subjective. To address this problem, we propose a novel method in the framework of Active Labeling for constructing a training set of thousands of images. Inspired by the hierarchical sampling method for active learning [6], we propose an Active forest that organizes hierarchically the data for easy and fast labeling. Moreover, introducing a classifier into the hierarchical structures, as well as transforming the feature space for better data clustering, additionally improve the algorithm. Our method is successfully tested to label 89.700 food-related objects and achieves significant reduction in expert time labelling.

Categories and Subject Descriptors

Computing Methodologies [**Image Preprocessing and Computer Vision**]: Scene Analysis—*Time-varying imagery; Color; Object recognition; Shape*

Keywords

Active labelling, food-related object recognition.

1. INTRODUCTION

It is clear that every day, technology is a little bit more present in our daily life. There are unlimited aspects and usual tasks in which Pervasive Computing (Ubiquitous Computing) can improve our quality of life. A way in which this emerging field can help us the most, is based on our feeding habits and all their related aspects: nutrition, physical activities, emotions and social interaction. And one of the most evident problems, for which we could be interested in logging every bit of the diet of a person, would be healthy weight management.

Figure 1: Picture of the wearable camera SenseCam.

An adequate and rich nutrition is clearly an important issue to take into account for anyone who wants to be healthy. Nutrition problems are widely known in our society, although not everyone is concerned about it and does much to solve them. Obesity and anorexia are diseases called "diseases of the XXI century". Given the advantages of keeping a record of feeding habits, interventional psychologists treating obese people, ask to record their lifestyle by writing diaries with annotation of all feeding activities during the days. However, several studies have reported that people tend to underestimate their food intake, meanwhile overestimating their physical activities [18].

An application based on lifelogging could make a big leap to solve this problem. People who clearly need help with their nutrition-related habits, could get an incredible benefit by collecting more explicit and objective the information related to their day-to-day by wearable cameras. Moreover, taking into account the importance of nutrition to prevent diseases, every person could also take a great advantage from a device like that. Hence, life-logging by a wearable camera appears as a natural solution by being able to objectively acquire day-by-day the feeding habits of persons. Starting by recognising objects related to feeding, and ending by being able to analyse the components of every meal, as well as

CEA'13, October 21 2013, Barcelona, Spain.

http://dx.doi.org/10.1145/2506023.2506032 .

defining the eating patterns of people, are important steps towards the success [7]. In this paper, we focus on using lifelogging technology and dealing with the huge amount of data that it produces to recognise objects closely related to nutrition and eating habits, more precisely, automatic recognition of dish objects in lifelogging records. To our knowledge, this work for first time addresses the problem of Active Labeling applied to the field of food-related object recognition.

1.1 Lifelogging and Wearable Devices

Lifelogging refers to the process of capturing large portions of peopleŠs lives by typically wearing computer or other digital devices. There are many different devices that help us logging any kind of information related to our daily routine, or any other sensor that can "sense" what we are doing, where we are, what we are looking at, etc. Even our smart phones can become a powerful lifelogging machine that can feed us with a wide range of useful information. A. Sellen and S. Whittaker in [23] summarized the benefits of pervasive computing, in general, and lifelogging, in particular, as the "Five Rs": recollecting, reminiscing, retrieving, reflecting and remembering intentions.

For logging the nutritional habits of a person, we can use one of the multiple wearable cameras that are available in the market for general users or researchers. In this work, we use the Microsoft's SenseCam (see Figure 1). SenseCam is designed for any lifelogging issues, although, its main research goal, when it was created, was improving the memory retention of Alzheimer's patients. S. Hodges et al. [15] proved that people with memory problems using SenseCam used to delay significantly the progression of the disease by reviewing their records captured by the camera. Since then, SenseCam and other wearable cameras have been successfully applied for multiple purposes [15, 16, 7].

SenseCam is able of taking on average about 4200 images per day. Once switched on, it captures continuously 2 frames per minute, up to 12 hours per day. The camera has a privacy button to be switched-off, when necessary. Once connected to a computer, all pictures are automatically downloaded and removed from the camera.

1.2 Food-Related Object Recognition

Given the problem of monitoring the nutritional habits of an individual, we reformulate it as a problem of recognising objects related to nutrition in the surroundings of our subject. Analysing the photos taken by the SenseCam along the day, we want to know when and for how long people are in contact with food. Thus, similarly as it had been done in [20, 14], we need to construct a robust classifier to automatically detect food-related objects, taking into account the variation of their instances in all of their variants, shapes and positions.

Today, there is a large battery of supervised classification algorithms (KNN, AdaBoost, Decision Trees, Neural Networks, Support Vector Machines, etc.) [9] applied to many problems in computer vision. A common feature for most of them is that they need a large set of training data to achieve high performance. This amount of input data is necessary to learn the important patterns that describe the objects in order to be able to automatically determine if a new object tested by the classifier, is or not an instance of the substance that we are detecting in the images.

The main disadvantages of most supervised classifier techniques are that: 1) we need large amounts of labeled training samples, and 2) the variables (features) related to our objects are too complex in order to develop a highly reliable unsupervised classifier. Active learning is a field of machine learning whose main purpose is guiding the process of manually annotating large amounts of data optimizing it [24, 6]. In contrast to Active learning, where the main goal is to learn in the fastest possible way, the goal of Active labeling is to guide the labeling process of all samples the fastest way possible (minimizing time, effort, clicks, etc.). This problem can be of interest in severl applications such as annotating all samples of a set for training or validation purposes.

1.3 Active Labeling

The basic features of an active learning method are: environments on which we have large numbers of samples to label, and the necessity of an expert to label and validate the training samples [25, 11]. Roughly speaking, we can divide most of the Active Learning approaches into two types [5]:

1. *Classifier-Based Active Learning:* Having a distribution of samples and a subset already labeled by a master, the method determines the next region of data distribution (based on the classifier that we are using) to be inspected by the master, which is where the classifier's uncertainty of unlabeled samples is higher [17].
2. *Data Distribution-Based Active Learning:* In this approach, the data queried is based on information of data distribution instead of classifier performance.

Hierarchical Sampling (HS) method [6, 5] forms a part of the second group. This particular method starts by creating a Hierarchical Cluster binary Tree (HCT) of all the samples to label using their features as guidelines for the partition. The HCT is constructed using the Single Linkage approach [13, 12] and K-Means [19] clustering. Once the HCT has been created, the labels of samples of the different clusters are queried to the master according to a certain criterion. The algorithm uses an uncertainty and purity measure (called score or bound) in order to decide: 1) how pure the clusters (nodes in the HCT) are, according to their known labels, and 2) if their sub-clusters should be considered. Thus, once the master has answered the labels of an iteration, the next queried cluster (node of the HCT) depends on its "purity" (the more similar the samples' labels are, the higher the probability of the cluster being chosen to label will be) [6]. We chose the HS algorithm, as a main skeleton of this work, due to its two main advantages:

1. It guides the labeling process through a previously calculated HCT depending on the impurity (or uncertainty) of the set of labels that reside in that particular level of the HCT, traveling downwards dividing the sets but never upwards.
2. It presents a method to explicitly obtain the error bounds or impurity of each of those clusters after each query step.

Due to these properties, the algorithm optimizes significantly the master's labeling effort and achieves results very competitive to the state-of-the-art, as verified by labeling several public domain databases [6, 5].

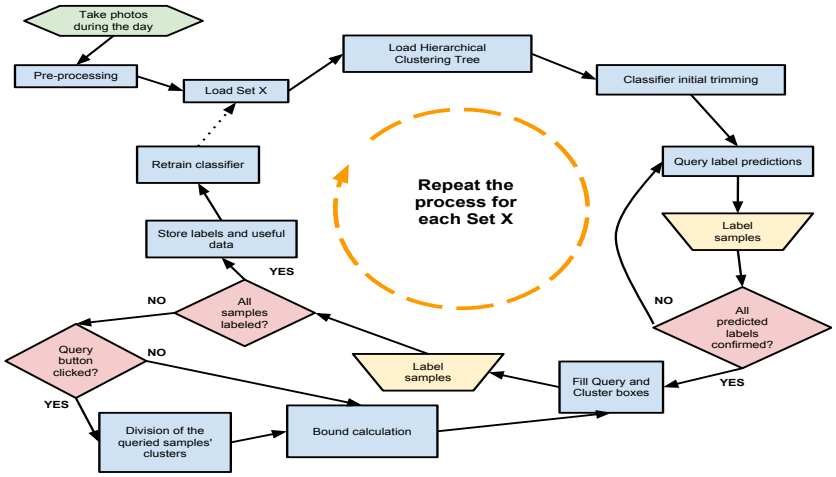

Figure 2: Scheme summarizing the basic functioning of the food-related object recognition application.

2. ACTIVE FOREST METHOD

The technique we propose is for creating a large set of training data for a food-related classifier. It follows the HS approach to treat the images of interest. We extend it by integrating the method used by [8, 21] which gives the master the authority of labeling some queries (Queries) sampled randomly [6] as well as validating a whole cluster (Cluster) with high value of purity[8]. In the latter case, homogeneous clusters are labeled with a single click. Thus, the algorithm achieves to optimize the number of clicks and decrease the time of unlabeled samples annotation.

Given our goal, which is labeling thousands of high dimensional samples in the minimum time possible, the set to label can be so huge that applying HS is getting computationally infeasible. Taking that into account, we made our data come in batches (e.g. corresponding to different days), we propose a novel method, called *Active Forest* which consists in splitting the data into different bags, for which different HCTs are constructed.

Our structure can be visualized as a forest of trees. A natural question arises: once the samples of a tree have been labeled, how do we transfer this knowledge to the next tree. Before constructing the next tree, we train a classifier on the objects of the already labeled data HCTs, and apply it to trim the samples set before constructing its HCT. All samples from the next set that are classified with high confidence, are labeled as a group by the master, and the rest is used to construct the HCT. Thus, in parallel to the labeling process, the Active forest also contains a supervised classifier that can predict the class labels of the data and reduce the samples to be labeled on the next step. Without loss of generality, we tested two of the most popular classifiers: K-NN [22] and AdaBoost [10], although any other supervised classifier can be applied.

Adding the supervised trimming data process as a step before each labeling session of the next tree, gives us the following benefits:

1. Smaller sets for an easier task given to the master.
2. No loss of information between the labeling sessions.
3. Decrease on the labeling time due to the smallest HCT constructed from the trimmed data.

Another issue is the optimal division of each partition set or clusters/nodes of the HCT in subsets using the clustering approach. Originally, in the HS approach it is done by applying unsupervised clustering. Note that moving downwards on the HCT, in each iteration, some of the samples of each partition set are already labeled. In the Active forest, we look for a transformation of the feature space of the current partition set, so as to minimize the distance between samples of the same class and maximize the distance between samples from different classes. In this way, we claim that sub-clusters will tend to be more "pure". To this purpose, we apply the Linear Discriminant Analysis [26, 2, 4], combined with a previously applied Principal Component Analysis [1]). By doing so, we optimize the purity of each cluster and reduce the dimensionality of the original high-dimensional feature space for faster clustering.

The flow diagram is given in Fig.2. Each of the labeling sessions is started by: a) trimming off samples classified with high confidence and confirmed by the master, and b) constructing the HCT to be labeled. Afterwards, the HS+LDA approach is applied in addition of using the Queries and the Cluster boxes for the master to choose. Finally, when all samples are labeled, the classifier is retrained to be used in the next set of labeling. The pseudo-algorithm definition is as follows:

Input: *One of the sets of unlabeled images with their corresponding features.*

Step 1: Initialize an empty tree structure T for keeping track of the pruning followed, the labeled and unlabeled samples that are in each cluster and their purity measure.

Step 2: If we have labeled more sets previously, apply initial trimming by the classifier trained on the previous HCTs.

Step 3: Choose randomly unlabeled samples and query the master.

Step 4: Save labels, set samples to "labeled" and increase the number of clicks.

Step 5: **While** *there is any unlabeled sample:*

Step 6: Get bounds (purity) of each node of the HCT and the most probable class assignment for each one (which will be temporary set as predicted until user's approval).

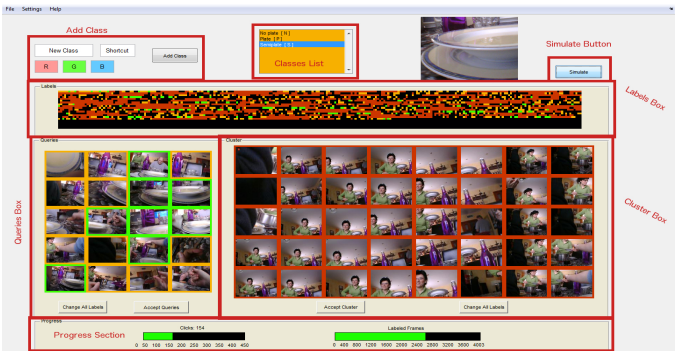

Figure 3: Main screen of our active labeling application with its different sections highlighted.

Step 7: Query N random samples and put them on the window ("Queries") with their predicted labels.

Step 8: Query the first M samples from the purest cluster and put them on the window ("Cluster") with their predicted labels.

Step 9: After user's approval save labels, set samples to "labeled" and increase the number of clicks (only from the chosen window samples).

Step 10: **If** *user selected "Queries": apply LDA and K-means in the new feature space to generate bi-partition of the clusters, where each of the queried samples belonged to.*

Step 11: **End While**

Step 12: Retrain the KNN classifier with the new labels.

Output: *Labeled images corresponding to the current HCT and the trained classifier.*

2.1 Food-Recognition Application for Active Labeling

After taking the photos with our lifelogging device along the day, images are pre-processed by: image crop by sliding window, set division for creating the trees for that day, and image features extraction. For each region, we extract the HOG descriptor and the mean (R,G,B) colors. We start the labeling process that is repeated until all the trees from that day are completely labeled. The Active forest is implemented in a user-friendly application to create the labeling set of the food-related objects. Figure 3 visualizes a scheme of its main window. On the top, the user has the possibility to add a new class and define its label. Bellow, a compacted view of all samples is presented where different colors code the labels of the samples, and black means "unlabeled". In the center, we have two groups of images: on the right, we have a "Cluster" that are the first samples of the cluster with higher purity. On the left, we have the "Queries" that are sampled randomly. Their frame shows the predicted label to be confirmed by the master. On the bottom, we have the statistics in terms of number of "clicks" (corrections) of the master and the percentage of the video that is labeled until the current step.

3. RESULTS

In order to validate the performance of the Active forest approach, we formed a validation set composed by: a public domain and a home-made database. We performed different tests to tune all the possible parameters and to compare

Figure 4: Samples from the three classes: NP (left), P (centre), SP (right).

the time needed for each labeling strategies. Given the aim of recognition of food-related objects, we illustrate the approach on plate recognition where any meal can be available. Without loss of generality, we assumed three different classes or labels for our data (Figure 4):

1. No Plate (NP): The image does not show any plate or it is too far away from the camera.
2. Plate (P): There is a clear plate near the centre of the image and the subject is close to it, or a maximum 5% is out of the field of view.
3. SemiPlate (SP): A plate is only partially visible.

3.1 Data Sets

During the lifelogging acquisition of images, different tasks of the subject's everyday life were recorded. We selected mealtime records of 6 different days (business days or holidays) that led to a total of 408 images. Apart from the SenseCam images, we decided to incorporate plate images from Image-Net.org. To sum up, we used 508 images (408 from SenseCam and 100 from Image-Net) from which, using different scales and crops as a compulsory preprocessing, we obtained 89.709 images (regions) divided in 24 different sets of approximately 4000 images each (basis of the Active Forest technique).

3.2 Sensitivity & Specificity

In order to get the optimal performance of the classifiers, we computed the sensitivity and the specificity of the KNN and the AdaBoost classifiers with different parameters. To perform these tests, we used a 10-fold cross-validation always with balanced classes, on which, as a first result, and due to the low performance using a NP vs P vs SP classifier, we decided to use cascade Combined Classifier [3]. It first discriminates NP vs (P + SP) as a first step; if the sample is classified to (P+SP), a second classifier will discriminate between P vs SP.

In order to compare the results of the AdaBoost and the KNN, we tested the sensitivity and specificity of the first classification step, NP vs (P + SP) (see Figure 5 (top)). In this figure, NT is the number of tests performed, and NR is the number of rounds. Given the opposite nature of sensitivity and specificity, we give the results according to the ratio between both measures. An alternative would be to use the F-measure. Figure 5 (bottom) gives the tests on the classification of P vs SP. We could see that both classifiers give approximately the same performance but to distinguish NP vs (P + SP), KNN is slightly better. Although, AdaBoost is slightly better in discriminating P vs SP, we decided to use the KNN for both classification problems due to its more stable results and less dependence on the parameters. We decided to take a conservative value of k equal to 15 due to its better generalization capability.

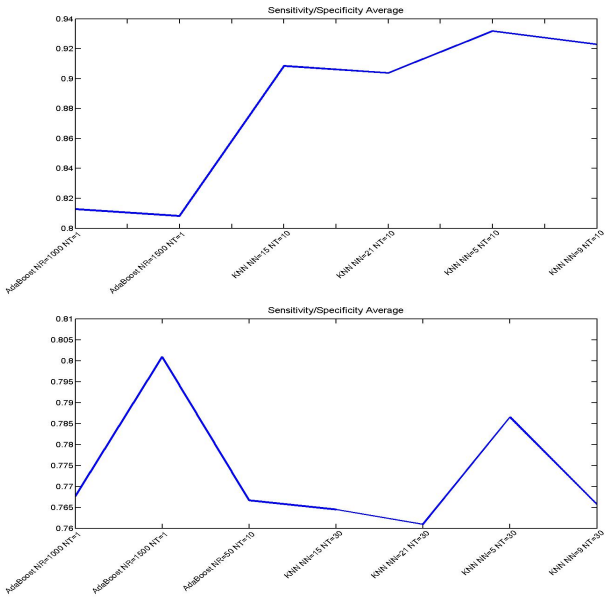

Figure 5: Sensitivity and Specificity of NP vs (P + SP) classifier (top), and Sensitivity and Specificity of P vs SP classifier (bottom)

3.3 Precision and Likelihood

The trimming step separates groups of objects with high likelihood (LH) assigning them a class, so that there is a high probability that the user will not have to correct their predicted labels. So we are interested in finding a likelihood threshold for the groups that optimizes the precision of the classifier (i.e. the groups should contain as much true positives as possible keeping only up to a small % of false positives). Figure 6 illustrates the precision of the KNN as a function of the LH threshold. Considering that a 95-97% of precision would be enough for the predicted labels to be trimmed with high probability to form a pure cluster, we fixed the corresponding LH values (NP: 0.7, P & SP: 0.7, P: 0.9, SP: 0.8) as thresholds to use in the supervised trimming procedure before constructing the next HCT. All samples that fall below these thresholds, will participate in the structure to be labeled by the hierarchical sampling method.

3.4 Active Forest Results

Finally, we present the performance results of our method. We used 89.709 labeled image regions and simulated the labeling on a MacBook Pro 2.66 GHz Intel Core i7 using Matlab running on Windows 7 on a VM. We ran 3 times each of the simulations shown in Figure 7.

Our first result is that the Active forest manages to label the set of 89.709 images, which turned out to be an impossible task for the HS method due to software and hardware limitations when creating the HCT. Moreover, each of its main contributions, results in improvement of the method: Figure 7 shows the results obtained by the Active Forest (blue), Active Forest with LDA (green) and Classifier (KNN) + Active Forest + LDA (red). We can see that, using Active forest allows us to label the images in up to 0.08 s/image, adding LDA improves it achieving 0.049s/image and adding the KNN classifier additionally reduces the time, achieving 0.03s/image, which represents an improvement of

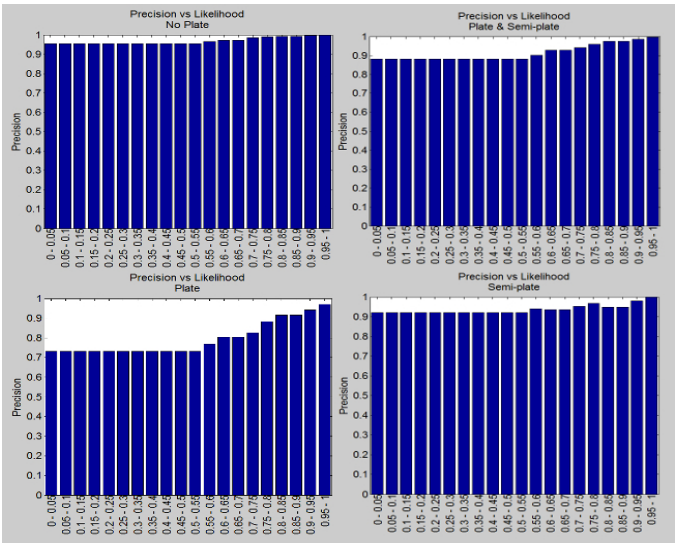

Figure 6: Precision vs Likelihood using KNN classifier for each of the labels of the Combined Classifier.

62% (all representing simulation times). We should note that in this simulation, the time needed by the masterŠs to "click" on an image is not considered, the time is spent in the algorithm execution. Interpreting the results, we can say that splitting the set and constructing several HCTs, the minimal time (that is the inferior limit), despising the master reaction time, will be about 1 hour and 42 minutes. Meanwhile, using the complete Active forest+LDA+KNN will label the set in 40 minutes (plus the time of masterŠs reaction). Since our tests shew that the number of master "clicks" was approximately the same in these cases, the times of the three algorithms would increase by a constant. When tests were done with a real master, the time increased to 0.55s per image.

Note that the theoretical bounds of HS are directly applicable to the Active forest, too. Since the purity estimation of each HCT node depends only on the number of labeled and unlabeled data, as well as the ratio of labels per class, the same estimation can be applied for the Active forest.

4. CONCLUSIONS

In this paper, we propose a novel technique (Active Forest), that allows to label large amounts of data. Our method contributes in three directions: 1) It extends hierarchical sampling method to a forest of hierarchical structures in order to make possible treating a huge volume of data; 2) It uses a trimming process applied by a supervised classifier before each labeling session has started that additionally optimizes the labeling time by 62%; and 3) Applying Linear Discriminant Analysis and a K-Means clusterisation on the nodes of the structure allows a dimensionality reduction of the feature space as well as a better clustering, minimizing the distances between samples of the same class and maximizing distances between different classes. Still, the Active forest maintains the same theoretical bounds as in the HS extracted from the purity of each node of the trees.

We integrated the Active forest method in a user-oriented application (Figure 3) that allows for an easy and fast labeling of huge amounts of data. Our application for Active

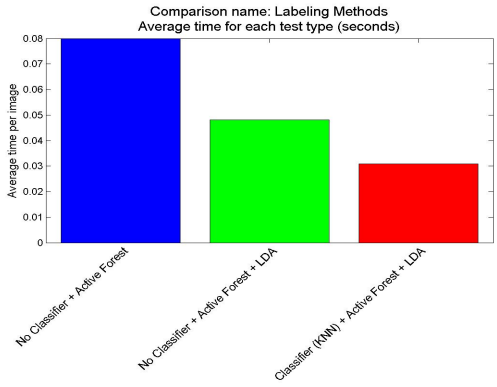

Figure 7: Labeling time of Active forest simulation.

Labeling is not limited to the binary classification and can be easily adapted for labeling multiple types of objects.

5. ACKNOWLEDGMENTS

This work was partially founded by the projects: TIN2009-14404-C02-02, TIN2012-38187-C03-01 and 2009SGR696.

6. REFERENCES

[1] H. Abdi and L. J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[2] S. Balakrishnama and A. Ganapathiraju. Linear discriminant analysis-a brief tutorial. *Institute for Signal and information Processing*, 1998.

[3] X. Baró, S. Escalera, J. Vitrià, O. Pujol, and P. Radeva. Traffic sign recognition using evolutionary adaboost detection and forest-ecoc classification. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):113–126, 2009.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *PAMI, IEEE Trans. on*, 19(7):711–720, 1997.

[5] S. Dasgupta. Two faces of active learning. *Theoretical computer science*, 412(19):1767–1781, 2011.

[6] S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th ICML*, pages 208–215. ACM, 2008.

[7] A. R. Doherty and A. F. Smeaton. Automatically segmenting lifelog data into events. In *Image Analysis for Multimedia Interactive Services, WIAMIS'08*, pages 20–23. IEEE, 2008.

[8] M. Drozdzal, S. Seguí, C. Malagelada, F. Azpiroz, J. Vitrià, and P. Radeva. Interactive labeling of wce images. In *Pattern Recognition and Image Analysis*, pages 143–150. Springer, 2011.

[9] R. O. Duda, P. E. Hart, et al. *Pattern classification and scene analysis*, volume 3. Wiley New York, 1973.

[10] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[11] P. H. Gosselin and M. Cord. Active learning methods for interactive image retrieval. *Image Processing, IEEE Transactions on*, 17(7):1200–1211, 2008.

[12] J. C. Gower and G. Ross. Minimum spanning trees and single linkage cluster analysis. *Applied statistics*, pages 54–64, 1969.

[13] J. A. Hartigan. Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association*, 76(374):388–394, 1981.

[14] H. Hoashi, T. Joutou, and K. Yanai. Image recognition of 85 food categories by feature fusion. In *Multimedia (ISM), 2010 IEEE International Symposium on*, pages 296–301. IEEE, 2010.

[15] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood. Sensecam: A retrospective memory aid. In *UbiComp 2006: Ubiquitous Computing*, pages 177–193. Springer, 2006.

[16] Y. J. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *IEEE Conference on CVPR*, pages 1346–1353. IEEE, 2012.

[17] M. Li and I. K. Sethi. Confidence-based active learning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(8):1251–1261, 2006.

[18] S. W. Lichtman, K. Pisarska, E. R. Berman, M. Pestone, H. Dowling, E. Offenbacher, H. Weisel, S. Heshka, D. E. Matthews, and S. B. Heymsfield. Discrepancy between self-reported and actual caloric intake and exercise in obese subjects. *New England Journal of Medicine*, 327(27):1893–1898, 1992.

[19] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[20] C. Morikawa, H. Sugiyama, and K. Aizawa. Food region segmentation in meal images using touch points. In *Proceedings of the ACM multimedia 2012 workshop on Multimedia for cooking and eating activities*, pages 7–12. ACM, 2012.

[21] P. Radeva, M. Drozdzal, S. Segui, L. Igual, C. Malagelada, F. Azpiroz, and J. Vitria. Active labeling: Application to wireless endoscopy analysis. In *International Conference on HPCS'2012*, pages 174–181. IEEE, 2012.

[22] T. Seidl and H.-P. Kriegel. Optimal multi-step k-nearest neighbor search. In *ACM SIGMOD Record*, volume 27, pages 154–165. ACM, 1998.

[23] A. J. Sellen and S. Whittaker. Beyond total capture: a constructive critique of lifelogging. *Communications of the ACM*, 53(5):70–77, 2010.

[24] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 2010.

[25] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2:45–66, 2002.

[26] M. Welling. Fisher linear discriminant analysis. *Department of Computer Science, University of Toronto*, 2005.

2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Simultaneous Food Localization and Recognition

Marc Bolaños and Petia Radeva
Department of Mathematics and Informatics
Universitat de Barcelona, Barcelona, Spain
Computer Vision Center, Bellaterra, Spain
Email: marc.bolanos@ub.edu

***Abstract*—The development of automatic nutrition diaries, which would allow to keep track objectively of everything we eat, could enable a whole new world of possibilities for people concerned about their nutrition patterns. With this purpose, in this paper we propose the first method for simultaneous food localization and recognition. Our method is based on two main steps, which consist in, first, produce a food activation map on the input image (i.e. heat map of probabilities) for generating bounding boxes proposals and, second, recognize each of the food types or food-related objects present in each bounding box. We demonstrate that our proposal, compared to the most similar problem nowadays - object localization, is able to obtain high precision and reasonable recall levels with only a few bounding boxes. Furthermore, we show that it is applicable to both conventional and egocentric images.**

I. Introduction

The analysis of people's nutrition habits is one of the most important mechanisms for applying a thorough monitorisation of several medical conditions (e.g. diabetes, obesity, etc.) that affect a high percentage of the global population. In most of the cases, interventional psychologists ask people to keep a manual detailed record of the daily meals ingested. However, as proved in [17], usually people tend to underestimate the quantity of food intake up to a 33%. Hence, methods for automatically logging one's meals could not only make the process easier, but also make it objective to the user's point of view and interpretability.

One of the solutions adopted recently that could ease the automatic construction of nutrition diaries is to ask individuals to take photos with their mobile phones [1]. An alternative technique is visual lifelogging [6] that consists of using a wearable camera that automatically captures pictures from the user point of view (egocentric point of view) with the aim to analyse different patterns of his/her daily life and extract highly relevant information like nutritional habits. By developing algorithms for food detection and food recognition that could be applied on mobile or lifelogging images, we can automatically infer the user's eating pattern. However, an important consideration to take into account when working with mobile or egocentric images is that they usually are of lower quality than conventional images due to the lower quality of portable hardware components. In addition, the analysis of egocentric images is harder considering that the pictures are non-intentionally taken and from a lateral point-of-view, causing motion blurriness, important partial occlusions, and bad lighting conditions (Fig. 1).

Fig. 1. Examples of conventional food images (two on the left) and egocentric food images (two on the right).

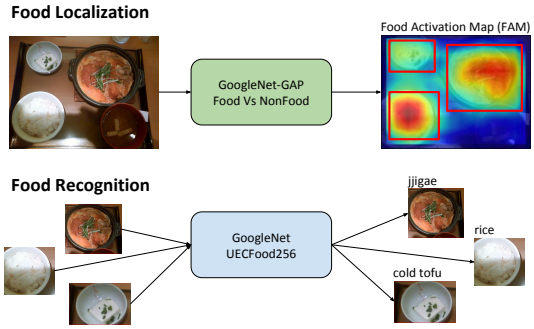

Fig. 2. General scheme of our food localization and recognition proposal.

A relatively recent technology that can leverage the automatic construction of nutrition diaries is Deep Learning, and more precisely, from the Computer Vision side, Convolutional Neural Networks (CNNs) [16]. These networks are able to learn complex spatial patterns from images. Thanks to the appearance of huge annotated datasets, the performance of these models has burst, allowing to improve the state of the art of many Computer Vision problems.

In this paper, we propose a novel and fast approach based on CNNs for detecting and recognizing food in both conventional and egocentric vision pictures. Our contributions are four-fold: 1) we propose the first food-related objects localization algorithm, which is specifically trained to distinguish images containing generic food and has the ability to propose several bounding boxes containing generic food (without particular classes) in a single image, 2) we propose a food recognition algorithm, which learns by re-using food-related knowledge and can be applied on the top of the food localization method, 3) we present the first egocentric dataset for food localization and recognition, and 4) we demonstrate that our methodology is useful for both conventional and egocentric pictures. Our contribution for food localization, inspired by the food detection method in [12], starts by training a binary food/non food CNN classifier for food detection and then, a simple and

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

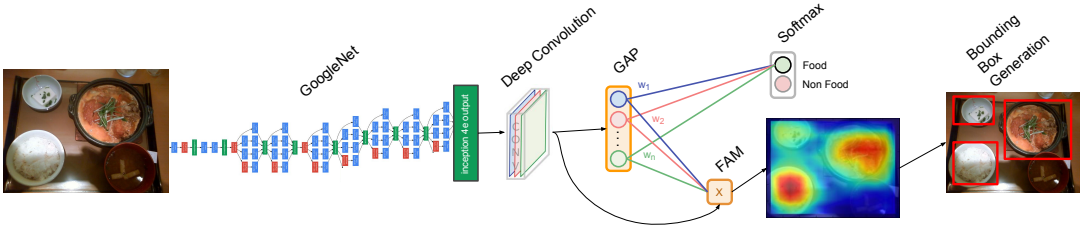

Fig. 3. Our localization method based on Global Average Pooling (GAP), which produces a Food Activation Map (FAM).

easy to interpret mechanism that allows us to generate food probability maps [24] is learned at the top of it. Finally, we propose an optimized method for generating bounding boxes on the obtained maps. Note that, as the desired application of the method is the generation of automatic nutrition diaries, we should not only detect food, but also food-related objects (e.g. bottles, cups, etc.). With this in mind, we collected data from complementary and varied datasets containing either food and non food pictures (see section IV-B). Up to our knowledge, there is no work in the literature that considers these categories. Without loss of generality, we add to the food categories those related to food-related objects, referring all of them as *food*. On the food recognition part, inspired by the findings in [23], we prove that, when we have small datasets for our problem, we can apply transfer learning by performing a chain of fine-tunings on a CNN for getting closer to our target domain (food types or food-related objects recognition) and achieving a better performing network.

The organization of this paper is as follows. In section II, we review the state of the art in food detection/localization and recognition. In section III, we explain the proposed methodology. In section IV, we describe the datasets used, the experimental setup, and present and discuss our results. Finally, in section V, we review the contributions, the limitations of the method and future directions.

II. Related Work

Considering that no works have been presented yet for simultaneous food localization and recognition in the bibliography, following we will review the most recent works devoted to food detection and food recognition, separately.

Food Detection and Localization: the problem of food detection has been typically addressed as a binary classification problem, where the algorithm simply has to distinguish whether a given image is representing food or not [1], [11], [12]. A different approach is applied by several papers [3], [5], [18], [25], where they intend to first segment and separately classify the components or ingredients of food and then apply a joint dish recognition.

The main problem of both approaches is that they assume that the dish was previously localized and therefore it is centered in the image. Instead, in the context of *food localization*, we are interested in finding the precise generic regions (or bounding boxes) in an image where any kind of food is present.

Although no methods have been presented specifically for food localization, several works have focused on generic object localization, usually called object detection, too. These methods could be used as a first step for food localization if they are followed by a food/non food classification applied on the obtained regions. Selective Search [22], considered as one of the best in the state of the art, applies a hierarchical segmentation and grouping strategy to find objects at different scales. The object detection methods, which obtain generic object proposals, intend to detect as many objects in the image as possible for optimizing the recall level, thus, they need to propose hundreds or thousands of candidates, leading to near null precision. An open question is how to obtain straightforward object localization methods that get high precision and recall results at the same time. An alternative to the generic object localization methods are methods trained to localize a set of predetermined objects like Faster R-CNN [19]. The authors propose a powerful end-to-end CNN optimized for localizing a set of 20 specific object classes.

Food Recognition: several authors have recently focused on food recognition. Most of them [4], [7], [9], [11], [14], [15], [23] have analyzed which features and models are more suitable for this problem. In their works, they have tested various methods for obtaining hand-crafted features in addition to exploring the use of different CNNs. One of the best results were obtained in [4] where the authors trained a CNN on the database Food101 [7] with 101 food categories and proved that applying a pre-training and then fine-tuning with in-domain food images can improve the classification performance. The best results on the UECFOOD256 database [13] that contains 256 food categories were obtained by Yanai et al. [23], where they used a network pre-trained on mixed food and object images for improving the final performance on food recognition. Some papers [5], [18] take a step further and use additional information like GPS location for recognizing the restaurant where the picture was taken and improve the classification results.

III. Methodology

In this section, we will describe the proposed methodology (see Fig. 2) in two steps: a) creating a generic food localizer, and b) training a fine-grained food recognition method by applying transfer learning.

A. Generic Food Localization

Our food-specialised algorithm detects image regions containing any kind of food, being reliable enough so that with

a few bounding boxes it is able to keep both high precision and recall. In order to achieve a fast inference, we propose to use a CNN trained on food detection. Then, we adapt it with a Global Average Pooling (GAP) layer [24] capable of generating Food Activation Maps (FAM) (i.e. heat maps of *foodness* probability). Finally, we extract candidates from the FAM in the form of bounding boxes (see pipeline in Fig. 3).

1) Food vs Non Food classifier: the first step to obtain a generic food localizer is to train a CNN for binary food classification. We chose the GoogleNet architecture [21] due to its proven high performance on several Computer vision tasks. We trained the CNN on the Deep Learning framework Keras[1]. For obtaining a faster convergence we applied a fine-tuning for our binary classification of the GoogleNet, which was previously trained on ILSVRC data [20].

2) Fine-tuning for FAM generation: once we had a model capable of distinguishing Food vs Non Food images, we applied the following steps [24]: 1) remove the two last inception modules and the following average pooling layer from the GoogleNet for obtaining a 14x14 pixels resolution (this allows to have a high enough spatial resolution for providing a final spatial classification), 2) introduce a new deep convolutional layer with 1024 kernels of dimensions 3x3 and stride 1, 3) introduce a GAP layer that summarizes the information captured by each kernel, and 4) set a new softmax layer for our binary problem. After getting the architecture ready, we applied an additional fine-tuning for the binary problem, providing a quick learning of the newly introduced layers.

Note that, instead of generating a map per class as done in Zhou et al. [24], we focus on obtaining a food-specific activation map that should be generic for any kind of food.

At inference time, our GoogleNet-GAP Food Vs NonFood network only has to: 1) apply a forward pass deciding whether the image contains food or not (softmax layer) and 2) compute the following equation for FAM generation:

$$\text{FAM}(x, y) = \sum_k w_k \cdot f_k(x, y), \tag{1}$$

where $k = \{1, ..., 1024\}$ identifies each of the kernels in the deep convolutional layer, and w_k and $f_k(x, y)$ are the weighting terms of the softmax layer for the class food, and the activation of the kth kernel at pixel (x, y), respectively.

3) Bounding box generation: as the last step, in order to extract bounding box proposals, we propose to apply a four steps method based on: 1) pick all regions above a certain threshold t, being t a percentage of the maximum FAM value, 2) remove all regions covering less than a certain percentage size s of the original image, 3) generate a bounding box for each of the selected regions, and 4) expand the bounding boxes by a certain percentage, e. All three parameters $\{t, s, e\}$ were estimated through a cross-validation procedure on the validation set (see section IV-D).

[1]https://github.com/MarcBS/keras

B. Transfer Learning for Food Recognition

After obtaining a generic object localizer, the final step in our approach is to classify each of the detected regions as a type of food. Again, for obtaining a high performing network and a faster convergence, we fine-tuned the GoogleNet pre-trained on ILSVRC. In addition, considering that our food recognition network has to overcome the problem of data quantity that most food classification datasets have, we propose applying an additional pre-training to the network. This supervised pre-training should serve as a fine-grained parameters adaptation in which the network should extract valuable knowledge from an extensive food recognition dataset before the final in-domain fine-tuning. For this purpose, we re-trained the GoogleNet, which was previously trained on ILSVRC, on the Food101 dataset [7].

At the end, we fine-tuned the network on the target domain data (either UECFood256 [13] or EgocentricFood). To obtain as little false positives as possible, we added an additional class to the final food recognition network containing Non Food samples, enabling the system to discard false food regions detected by the localization method.

IV. Results

In this section we will describe the different datasets used for performing the tests; the pre-processing applied to them; the metrics used for testing the localization algorithm; the experimental setup and; finally, the results and performance of our localization and recognition techniques.

A. Datasets

Following we describe all the dataset used in this work either for food localization, for food recognition or for both.

PASCAL VOC 2012 [8]: dataset for object localization consisting of more than 10,000 images with bounding boxes of 20 different classes (none of them related to food).

ILSVRC 2013 [20]: dataset similar to PASCAL with more than 400,000 images and 1,000 classes for training and validation (with a subset of classes related to food).

Food101 [7]: dataset for food recognition that consists of 101 classes of typical foods around the world, having each class 1,000 different samples.

UECFood256 [13]: dataset for food localization and recognition. It consists of 256 different international dishes with at least 100 samples each. The dataset was collected by the authors from images on the web, which means that they can be captured either by conventional cameras or by smartphones.

Egocentric Food: the first dataset of egocentric images for food-related objects localization and recognition. This self-made dataset was collected using the wearable camera Narrative Clip and consists of 9 different classes (glass, cup, jar, can, mug, bottle, dish, food, basket) with a total of 5038 images and 8573 bounding boxes.

B. Data Pre-processing

Following we detail the different data pre-processing applied for each of the learning steps and classifiers.

Food Vs Non Food training: we used three different datasets: *Food101*, where all the images were treated as positive samples (class Food). We used the training split provided by the authors for generating a training (80%) and a validation (20%) splits balanced along all classes; *PASCAL*, where an object detector [2] was used to extract 50 object proposals per image on the 'trainval' set. All the resulting bounding boxes were treated as negative samples (class Non Food). Again, we divided the data in 80/20% for training and validation; and *ILSVRC*, where we selected the 70 classes (or synsets) of food or food-related objects available. In this case, we only used the training/validation split provided by the authors. The bounding boxes were extracted and used as positive samples (class Food).

Food Recognition training: we used the Food101 dataset as the first dataset for fine-tuning the food recognition network pre-trained on ILSVRC. The previously applied 80/20% split of the training set provided by the authors was used for training and validation, respectively. The test set provided was used for testing. On the second fine-tuning, the same preprocessing was applied on both UECFood256 and EgocentricFood: a random 70/10/20% split of images was applied for training/validation/testing on each class separately and the bounding boxes were extracted.

Joint Localization and Recognition tests: the previous 70/10/20% split was also used on the localization and recognition test. We made sure that any image containing more than one instance was included only in one split.

C. Localization Metrics

The metric used for evaluating the results of a localization algorithm is the Intersection over Union (IoU). This metric defines how precise is the predicted bounding box (bb) with respect to the ground truth (GT) annotation, and is defined as:

$$\text{IoU(bb)} = \frac{GT \cap bb}{GT \cup bb}, \tag{2}$$

where usually a bounding box is considered valid when its IoU ≥ 0.5. The other evaluation metrics used are: Precision $= \frac{TP}{TP+FP}$, Recall $= \frac{TP}{TP+FN}$, and Accuracy $= \frac{TP}{TP+FP+FN}$, where the true positives (TP) are the bounding boxes correctly localized, the false positives (FP) are the predicted bounding boxes that do not exist in the ground truth, and the false negatives (FN) are the ground truth samples that are lost by the model. Note that given the convention from [8], if more than one bounding box overlaps the same GT object, only one will be considered as TP, the rest will be FPs.

D. Experimental Setup

The Food vs Non Food binary network used for *food localization* was trained during 24,000 iterations with a batch size of 50 and a learning rate of 0.001. A decay of 0.1 was applied every 6,000 iterations. The final validation accuracy achieved on the binary problem was 95.64%. During localization, the bounding box generation is applied on the FAM only if the image was classified as containing food by the softmax

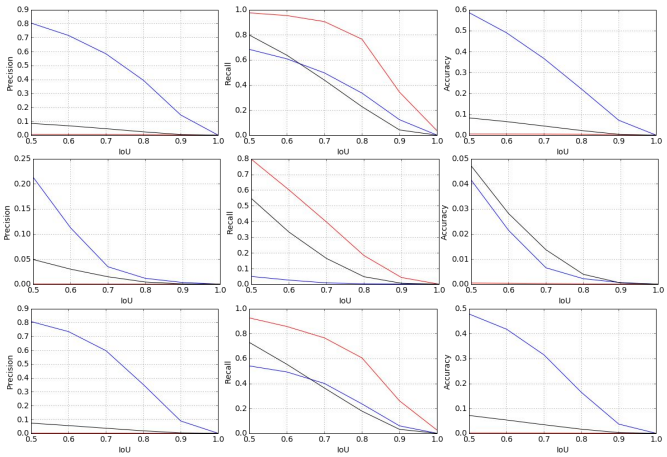

Fig. 4. Curves of Precision vs IoU (left), Recall vs IoU (centre) and Accuracy vs IoU (right) on the test sets of UECFood256 (top), EgocentricFood (middle) and both combined (bottom). Our method is shown in blue, Selective Search in red and Faster R-CNN in black.

(see Fig. 3). A grid search was applied on the localization-validation set for choosing the best hyperparameters $\{t, s, e\}$ for localization (named threshold, size, and expansion percentages, respectively). The values tested were from 0.2 to 1 in increments of 0.2 for both t and e, and from 0.0 to 0.1 in increments of 0.02 for s.

Considering that no food localization methods currently exist, we used Selective Search [22] and Faster R-CNN [19] as baselines for being two of the top performing object localization methods. The former obtains generic objects and the latter is optimized for localizing PASCAL's classes (although we will treat its predictions as generic proposals).

For the *food recognition* models, first, the GoogleNet-ILSVRC model was re-trained on Food101 using Caffe [10], achieving the best validation accuracy after 448,000 iterations. A batch size of 16 and a learning rate of 0.001 with a decay of 0.5 every 50,000 iterations were used. The model was converted to Keras before applying the final fine-tuning to the respective datasets UECFood256 or EgocentricFood.

During the *joint localization and recognition* tests, a bounding box is only considered TP if and only if it is both correctly localized (with a minimum IoU value of 0.5) and correctly recognized.

E. Food Localization

Taking into account that some of the tested methods [22] lack the capability of providing a localization score for each region, we are not enable to calculate a Precision-Recall curve. For this reason, we chose the accuracy as our guideline for comparison, which enables a trade-off between the capabilities of the methods to find all the objects present (Recall) and produce as little miss-localizations as possible (Precision). We chose the best $\{t, s, e\}$ parameters on the combined validation set (UECFood256 and EgocentricFood) in terms of the average accuracy value among all the IoU scores, resulting in $t = 0.4$, $s = 0.1$ and $e = 0.2$.

TABLE I
FOOD RECOGNITION RESULTS ON EACH DATASET. BEST TOP-1 RESULTS ARE SHOWN IN BOLDFACE.

Dataset	Pre-training	Validation Accuracy		Test Accuracy	
		Top-1	Top-5	Top-1	Top-5
Food101	ILSVRC	**74.75**	91.11	**79.20**	94.11
UECFood256	ILSVRC	52.72	78.61	51.60	78.26
	Food101	**65.71**	86.40	**63.16**	85.57
EgocentricFood	ILSVRC	**91.50**	99.80	90.77	99.37
	Food101	90.85	99.65	**90.90**	99.37

In Fig. 4 we can see the precision, recall and accuracy curves obtained by the different localization methods.

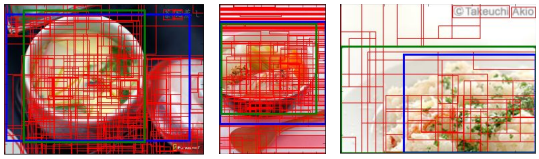

Fig. 5. Examples of localization on UECFood256. Ground truth in green, our method in blue and Selective Search in red.

Comparing the methods in terms of precision, it can be appreciated that ours outperforms the other methods in all cases. This pattern is easy to explain given that any generic object localization method (Selective Search in this case) usually outputs several thousands of proposals per image (see some examples in Fig. 5), causing it to get a lot of FPs. In comparison, Faster R-CNN only provides some tens of proposals per image given that it is optimized for finding bounding boxes of the specific classes in the PASCAL dataset. This means that it can focus on the most interesting proposals per class, which is a great advantage compared to Selective Search and makes its precision higher. Even though, it is still far from the optimum considering that usually there are less than 10 food-related elements in an image. Note that, curiously, Faster R-CNN is able to find food-related objects even without being optimized to do so. Comparing the methods in terms of recall, the Selective Search, in contrast to our method and Faster R-CNN, is clearly the best given that its goal is to find any object appearing in the image even if it is necessary to *sacrifice* the precision of the method. We can see that, although on most of the cases our method and Faster R-CNN are paired, in EgocentricFood the latter is better. This can be explained by the fact that the purpose of Faster R-CNN, which is to localize objects, is more aligned with the annotations found in EgocentricFood, which are of food-related objects. If we compare the methods in terms of accuracy, we can see that our proposal, which is able to obtain more balanced precision-recall results, outperforms both state of the art methods in UECFood256 and the combined datasets, and is paired with Faster R-CNN on EgocentricFood.

As we saw, a great part of the proposed bounding boxes are correctly predicted by our method. Although, we could say that this ability is also its weak point in terms of recall, where it obtains lower values considering it is not always able to find all the food-related elements in the image, mostly when they are very close or overlapping.

Additionally, comparing them in terms of execution time, Selective Search needs an average of 0.8s per image, Faster R-CNN needs 0.2s and our localization method needs only 0.06s using a GPU and a batch size of 25. Thus, it is able to apply a near real-time inference.

F. Food Recognition

From the food recognition side, the results on the different trainings performed can be seen on Table I. Note that the results are comparable to the state of the art on food recognition: either on Food101 [23], or in UECFood256, where an alternative would be to apply the method on [4]. We can see that, when fine-tuning on a model which is already adapted for food recognition, we can obtain better accuracy. The difference is more remarkable on UECFood256 because all the samples in the dataset are different types of food, while EgocentricFood is more focused on food-related objects.

G. Localization and Recognition

Finally, we test the whole localization and recognition pipeline proposed. We present the final results fixing the minimum IoU to 0.5 in Table II. To take into account the results of both steps at the same time, we evaluated the precision, recall and accuracy separately for each class and applied a final mean over all the classes. Note that when combining both datasets, we have a total of 265 classes (256 on UECFood256 and 9 on EgocentricFood). Our method is able to find most of the food-related objects in the UECFood256 dataset with only a few bounding boxes (usually at most 5). On the EgocentricFood dataset the difficulty of the problem becomes clear, where there are three additional issues to overcome: 1) the quality of the pictures is lower and objects are taken in a lateral point of view, 2) some classes are ambiguous and difficult to distinguish from non food-related objects and, 3) a great part of the samples are occluded and far from the camera wearer (see examples in Fig. 1 and 6).

Finally, in Fig. 6 we show some examples of the complete method. In some cases, the GT ambiguity produces recognition or localization misclassification. For instance, in the first image at the bottom right zone we can see a glass (GT) with a lemon (food prediction) inside, and in the second one, we can see a dish in the foreground (GT) and a bounding box of bread in the dish (food prediction).

V. CONCLUSION

We proposed the first methodology for simultaneous food localization and recognition. Our method is applicable to conventional and to egocentric point-of-view images. We

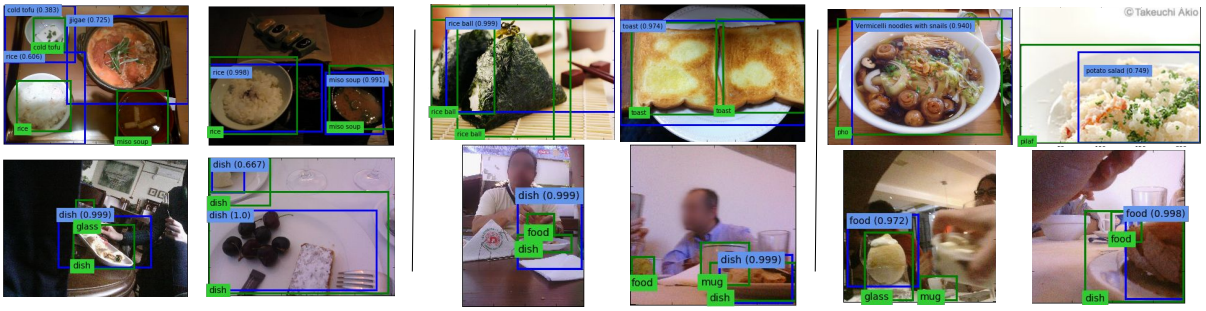

Fig. 6. Examples of localization and recognition on UECFood256 (top) and EgocentricFood (bottom). Ground truth is shown in green and our method in blue (recognition score between parenthesis). Some overall good (left), good recognition, but bad localization (centre) and good localization, but bad recognition (right) examples are shown.

TABLE II
SIMULTANEOUS TEST LOCALIZATION AND RECOGNITION.

Dataset	Precision	Recall	Accuracy
UECFood256	54.33	50.86	36.84
EgocentricFood	17.38	8.72	6.41
Combined	53.58	49.26	35.82

have proven that this methodology outperforms the baseline achieved by generic object localizers. As future work, we will focus on the ability of the method to distinguish very close or overlapping food-related objects.

ACKNOWLEDGMENT

Work partially funded by TIN2015-66951-C2-1-R, SGR 1219 and an ICREA Academia2014 grant. We acknowledge NVIDIA for the donation of a GPU and M. Ángeles Jiménez for her collaboration.

REFERENCES

[1] Kiyoharu Aizawa, Yuto Maruyama, He Li, and Chamin Morikawa. Food balance estimation by using personal dietary tendencies in a multimedia food log. *Multimedia, IEEE Transactions on*, 15(8):2176–2185, 2013.

[2] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 73–80. IEEE, 2010.

[3] Marios Anthimopoulos, Joachim Dehais, Peter Diem, and Stavroula Mougiakakou. Segmentation and recognition of multi-food meal images for carbohydrate counting. In *Bioinformatics and Bioengineering (BIBE), 2013 IEEE 13th International Conference on*, pages 1–4, 2013.

[4] Shuang Ao and Charles X Ling. Adapting new categories for food recognition with deep representation. In *2015 IEEE International Conference on Data Mining Workshop*, pages 1196–1203, 2015.

[5] Vinay Bettadapura, Edison Thomaz, Aman Parnami, Gregory D Abowd, and Irfan Essa. Leveraging context to support automated food recognition in restaurants. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 580–587. IEEE, 2015.

[6] Marc Bolaños, Mariella Dimiccoli, and Petia Radeva. Towards storytelling from visual lifelogging: An overview. *arXiv preprint arXiv:1507.06120*, 2015.

[7] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014*, pages 446–461. Springer, 2014.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[9] ZongYuan Ge, Chris McCool, Conrad Sanderson, and Peter Corke. Modelling local deep convolutional neural network features to improve fine-grained image classification. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 4112–4116. IEEE, 2015.

[10] Yangqing J., E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678, 2014.

[11] H. Kagaya, K. Aizawa, and M. Ogawa. Food detection and recognition using convolutional neural network. In *Proceedings of the ACM International Conference on Multimedia*, pages 1085–1088, 2014.

[12] Hokuto Kagaya and Kiyoharu Aizawa. Highly accurate food/non-food image classification based on a deep convolutional neural network. In *ICIAP 2015 Workshops*, pages 350–357.

[13] Y. Kawano and K. Yanai. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In *Proc. of ECCV Workshop on Transferring and Adapting Source Knowledge in Computer Vision (TASK-CV)*, 2014.

[14] Yoshiyuki Kawano and Keiji Yanai. Real-time mobile food recognition system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2013.

[15] Yoshiyuki Kawano and Keiji Yanai. Foodcam-256: a large-scale real-time mobile food recognition system employing high-dimensional features and compression of classifier weights. In *Proceedings of the ACM International Conference on Multimedia*, pages 761–762. ACM, 2014.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[17] Steven W Lichtman, Krystyna Pisarska, Ellen Raynes Berman, Michele Pestone, Hillary Dowling, Esther Offenbacher, Hope Weisel, Stanley Heshka, Dwight E Matthews, and Steven B Heymsfield. Discrepancy between self-reported and actual caloric intake and exercise in obese subjects. *New England Journal of Medicine*, 327(27):1893–1898, 1992.

[18] Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. Im2calories: towards an automated mobile vision food diary. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1233–1241, 2015.

[19] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[22] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

[23] Keiji Yanai and Yoshiyuki Kawano. Food image recognition using deep convolutional network with pre-training and fine-tuning. In *Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on*, pages 1–6. IEEE, 2015.

[24] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *arXiv preprint arXiv:1512.04150*, 2015.

[25] Fengqing Zhu, Marc Bosch, TusaRebecca Schap, Nitin Khanna, David S Ebert, Carol J Boushey, and Edward J Delp. Segmentation assisted food classification for dietary assessment. In *IS&T/SPIE Electronic Imaging*, pages 78730B–78730B. International Society for Optics and Photonics, 2011.

Can a CNN Recognize Catalan Diet?

Pedro Herruzo[a)], Marc Bolaños[b)] and Petia Radeva[c)]

Universitat de Barcelona. Barcelona, Spain.
Computer Vision Center. Bellaterra, Spain.

[a)]pherrusa7@alumnes.ub.edu
[b)]marc.bolanos@ub.edu
[c)]petia.ivanova@ub.edu

Abstract. Nowadays, we can find several diseases related to the unhealthy diet habits of the population, such as diabetes, obesity, anemia, bulimia and anorexia. In many cases, these diseases are related to the food consumption of people. Mediterranean diet is scientifically known as a healthy diet that helps to prevent many metabolic diseases. In particular, our work focuses on the recognition of Mediterranean food and dishes. The development of this methodology would allow to analise the daily habits of users with wearable cameras, within the topic of lifelogging. By using automatic mechanisms we could build an objective tool for the analysis of the patient's behaviour, allowing specialists to discover unhealthy food patterns and understand the user's lifestyle.

With the aim to automatically recognize a complete diet, we introduce a challenging multi-labeled dataset related to Mediterranean diet called FoodCAT. The first type of label provided consists of 115 food classes with an average of 400 images per dish, and the second one consists of 12 food categories with an average of 3800 pictures per class. This dataset will serve as a basis for the development of automatic diet recognition. In this context, deep learning and more specifically, Convolutional Neural Networks (CNNs), currently are state-of-the-art methods for automatic food recognition. In our work, we compare several architectures for image classification, with the purpose of diet recognition. Applying the best model for recognising food categories, we achieve a top-1 accuracy of 72.29%, and top-5 of 97.07%. In a complete diet recognition of dishes from Mediterranean diet, enlarged with the Food-101 dataset for international dishes recognition, we achieve a top-1 accuracy of 68.07%, and top-5 of 89.53%, for a total of 115+101 food classes.

INTRODUCTION

Technology that helps track health and fitness is on the rise, in particular, automatic food recognition is a hot topic for both, research and industry. People around us have at least 2 devices, such as tablets, computers, or phones, which are used daily to take pictures. These pictures are commonly related to food; people upload dishes to social networks such as Instagram, Facebook, Foodspotting or Twitter. They do it for several reasons, to share a dinner with a friend, to keep track of a healthy diet or to show their own recipes. This amount of pictures is really attractive for companies, who are already putting much effort to understand people's diet, in order to offer personal food assistance and get benefits.

Food and nutrition are directly related to health. Obesity, diabetes, anemia, and other diseases, are all closely related to food consumption. Looking at food habits, the Mediterranean diet is scientifically known as a healthy diet. For example, a growing number of scientific researches has been demonstrating that olive oil, operates a crucial role on the prevention of cardiovascular and tumoral diseases, being related with low mortality and morbidity in populations that tend to follow a Mediterranean diet [1]. Many doctors tell patients to write a diary of their diet, trying to make them aware of what they are eating. Usually people do not care too much about that, annotating all the meals often is getting boring. An alternative is to make the food diary by pictures with the phone, or even better, to take the pictures automatically with a small wearable camera. It can be very useful in order to analyse the daily habits of users with wearable cameras. It appears as an objective tool for the analysis of patient's behaviour, allowing specialists to discover unhealthy food patterns and understand user's lifestyle. However, automatic food recognition and analysis are still challenges to solve for the computer vision community.

Deep learning and more specifically, Convolutional Neural Networks (CNNs) are actually the technologies within

FIGURE 1. Examples of Catalan cuisine in *FoodCAT* dataset: sauteed beans, paella, strawberries with vinegar, cuttlefish with peas, roasted snails and beans with sausage.

the state-of-the-art for automatic food recognition. The *GoogleNet* [2] was responsible for setting the state of the art for classification and detection in the ImageNet Large-Scale Visual Recognition Challenge in 2014 *ILSVRC14* [3]. Another widely used model is *VGG* [4], which secured the first and the second places also for the ImageNet *ILSVRC14* competition [3], in the localization and classification tasks respectively. One of the most popular food dataset is the *Food-101* dataset [5], containing 101 food categories, with 101.000 images. Another well known is the *UEC FOOD 256* dataset [6], which contains 256 types of food. Many researchers have been working with these datasets achieving very good results on food recognition [7], or in both food localisation and recognition [8] [9]. Another food related classification task that we are interested in, is to classify food categories, e.g. we should be able to classify a paella picture into the category of rice. In our case, we will do it following a robust classification of Catalan diet proposed in the book *El Corpus del patrimoni culinari català* [10]. Other related works on that topic classify 85 food classes [11] or 50 dishes [12]. Hence, we construct our dataset from the Catalan cuisine as a good representative of the Mediterranean food.

In this paper we focus on developing automatic algorithms to recognize Catalan food using deep learning techniques. For this purpose we build a dataset and enlarge it with the public domain dataset *Food-101*. Our work is organized in three steps:

1. Build a dataset including healthy food: The current food datasets are built in order to achieve a good performance in the general challenge of recognizing pictures automatically. Our goal is to present a method for food recognition of extended dataset based on Catalan food, as it is scientifically supported as a healthy diet (see Fig. 1 for some examples). Therefore, we present a new dataset based on Catalan food, which we call *FoodCAT*. This dataset has been classified following two different approaches. On one side, the images have been classified based on dishes, and on the other side, in a more general food categories. As an example, our system will recognize a dish with chickpeas with spinach as the food class 'chickpeas with spinach', but also as food category 'Legumes'.

2. Recognize food dishes with Convolutional Neural Networks: We are interested in applying a Convolutional Neural Network to recognize the new built healthy dataset together with the dataset *Food-101* [5]. We use pre-trained models over the large dataset *ImageNet*, such as *GoogleNet* [2] and the VGG [4]. Moreover, in order to recognize food categories, we compare the differences between fine-tuning a pre-trained model over all the layers, versus the same model trained only for the last fully-connected layer.

3. Improve the quality of the dataset and the recognition task with Super-Resolution: It has been proven that large image resolution improves recognition accuracy [13]. Therefore, we will base on a new method to increase the resolution of the images, based on a Convolutional Neural Network, known as Super-Resolution (SR) [14]. With that, our goal is to get a better performance in the image recognition task.

METHODOLOGY

The image classification problem is the task of assigning a label from a predefined set of categories to an input image. In order to tackle this task for the Catalan diet problem, we propose taking a data-driven approach. After collecting a dataset for the problem at hand, we are going to train a CNN for automatically learning the appearance of each class and classifying them.

The collected dataset, named *FoodCAT*, when compared to the most widely used dataset for food classification *Food-101*, presents a lower image resolution which, as we prove in our experiments, leads to a data bias and a lower performance when training a CNN on the combined datasets. In order to solve this problem, we must increase the resolution to at least 256x256 pixels, which is the usual input size to CNNs. Thus, we propose using the method known as *Super-Resolution* and consequently improve the accuracy in the food recognition task.

Model

In order to apply food classification, we propose using the *GoogleNet* architecture, which has proven to obtain very high performance in several classification tasks [7] [8] [15].

We train the *GoogleNet* model using an image crop of 224x224x3 pixels as input. During training, in order to perform data augmentation, we extract random crops from the images after unifying their resolution to 256x256x3. During the testing procedure, we use the central image crop. The *GoogleNet* convolutional neural network architecture is a replication of the model described in the *GoogleNet* publication [2]. The network is 22 layers deep when counting only layers with parameters (or 27 layers if we also count pooling layers). As the authors explain in their paper [2], two of the features that made this net so powerful are : *Auxiliary classifiers connected to the intermediate layers*: which was thought to combat the vanishing gradient problem given the relatively large depth of the network. During training, their loss gets added to the total loss of the network with a discount weight. In practice, the auxiliary networks effect is relatively minor (around 0.5%) and it is required only one of them to achieve the same effect. *Inception modules*: the main idea for it is that in images, correlations tend to be local. Therefore, in each of the 9 modules, they use convolutions of dimension 1x1, 3x3, 5x5, and pooling layers of 3x3. Then, they put all outputs together as a concatenation. Note that to reduce the depth of the volume, convolutions 3x3 and 5x5 are performed after applying a 1x1 convolution with less filters, and pooling 3x3 is also followed by a convolution 1x1. This makes the model more efficient reducing the number of parameters in the net.

Super-Resolution

The image dimensions of *FoodCAT* dataset are on average smaller than 256x256. Motivated by the fact that larger images improve recognition accuracy [13], we propose increasing the resolution with a state-of-the-art method instead of applying a common upsampling through bilinear interpolation. To increase the size of the images, we use the method called Super-Resolution [14]. In this paper, the authors propose a technique for obtaining a High-Resolution (HR) image from a Low-Resolution (LR) one. To this end, they use a Sparse Coding based Network (SCN) based on the Learned Iterative Shrinkage and Thresholding Algorithm (LISTA) [16]. Notable improvements are achieved over the generic CNN model in terms of both recovery accuracy and human perception. The implementation is based on recurrent layers that merge linear adjacent ones, allowing to jointly optimize all the layer parameters from end to end. It is achieved by rewriting the activation function of the LISTA layers as follows:

$$[h_\theta(a)]_i = sign(a_i)\theta_i(\|a_i\|/\theta_i 1)_+ = \theta_i h_1(a_i/\theta_i)$$

Fig. 2 shows the visual difference of a randomly chosen FoodCAT image compared to its SR version. In this example, the original image is 402x125, so the SR was applied with a factor of 3 to assure that both dimensions are bigger than 256.

RESULTS

In this section, we describe the datasets, metrics used for evaluating and comparing each model, and results for each of the image recognition tasks: dishes and food categories.

FIGURE 2. Left shows the SR decreased to 256x256 and right shows the original increased to 256x256.

Dataset

Our dataset, *FoodCAT* has two different labels for each image: Catalan dish, and Catalan food category. Although the total number of Catalan dishes of our datasets are 140, we selected only the set of classes with at least 100 images for our experiments, resulting in a total of 115 classes. Some examples of the available dishes are: sauteed beans, paella, strawberries with vinegar, cuttlefish with peas, roasted snails or beans with sausage. In addition, the images are also labeled in 12 general food categories. Table 1 shows a summary of the general statistics of the dataset, including the number of dishes and images that we have tagged for each food category.

TABLE 1. First column lists the categories, second and third column show the number and the percentage of dishes, and the fourth one shows the amount of pictures by category.

	# dishes	**%**	**# images**
Desserts and sweets	34	24,28	11.933
Meats	26	18,57	7.373
Seafood	25	17,85	5.977
Pasta, rice and other cereals	11	7,85	4.728
Vegetables	11	7,85	3.007
Salads and cold dishes	5	3,57	2.933
Soups, broths and creams	8	5,71	2.857
Sauces	4	2,85	2.462
Legumes	6	4,28	1.920
Eggs	5	3,57	615
Snails	3	2,14	470
Mushrooms	2	1,42	438
Total	140	100	44.713

Implementation

There are several frameworks with high capabilities for working on the field of Deep Learning such as TensorFlow, Torch, Theano, Caffe, Neon, etc. We choose Caffe, because it tracks the state-of-the-art in both code and models and is fast for developing. We also decided to use it, because it has a large community giving support on the Caffe-users group and Github, uploading new pre-trained models that people can use for different purposes.

A competitive alternative of the *GoogleNet* model is the *VGG-19*, which we also use in our experiments. This net has 5 blocks of different depth convolutions (64, 128, 256, 512, and 512 consecutively) and 3 FC layers. The first 2 blocks contain 2 different convolutions each and the last 5 contain 4 different convolutions each. It has a total of $2 \times 2 + 3 \times 4 + 3 = 19$ layers. All convolutions have a kernel size of 3x3 with a padding of 1 pixel, i.e. the spatial resolution is preserved after each convolution. Finally, after each convolutional block a max pooling is performed over a 2x2 pixel window with stride 2, i.e. reducing by a factor of 2 the spatial size after each block. As the VGG-19 paper [4] shows, small-size convolution filters are the key to outperform the GoogleNet in ILSVRC14 [3] in terms of the single-network classification accuracy.

Evaluation Metrics

Many metrics can be considered to measure the performance of a classification task. In the literature, mainly three methods are used: Accuracy Top-1 (AT1), Accuracy Top-5 (AT5), and the Confusion Matrix (CM). In real-world applications, usually the dataset contains unbalanced classes and the above measures can hide the misclassification of classes with fewer samples. Hence, we consider the Normalized Accuracy Top-1 (NAT1), that gives us the information of how good the classifier is no matter how many samples each class has. Let us define formally each metric.

Let N be the total number of classes with images to test, let N_i be the number of images of the i-th class, and set $n = \sum_{i=0}^{N-1} N_i$, as the total number of images to test. Let $\hat{y}^k_{i,j}$ be the `top-k` predicted classes of the j-th image of the i-th class, and $y_{i,j}$ the corresponding true class. Let us also define $\mathbf{1}_A : X \to \{0, 1\}$ as the indicator function as follows:

$$\mathbf{1}_A(x) := \begin{cases} 1 & \text{if } x_i \in A, \text{ for some } i, \\ 0 & \text{if } x_i \notin A, \text{ for all } i. \end{cases}$$

Then, the definitions of the metrics are as follows:

$$\texttt{AT1} = \frac{1}{n} \sum_{i,j} 1_{y_{i,j}}(\hat{y}^1_{i,j}), \quad \texttt{AT5} = \frac{1}{n} \sum_{i,j} 1_{y_{i,j}}(\hat{y}^5_{i,j}), \quad \texttt{NAT1} = \frac{1}{N} \sum_{i=0}^{N-1} \frac{1}{N_i} \sum_{j=0}^{N_i-1} 1_{y_{i,j}}(\hat{y}^1_{i,j}).$$

Super Resolution application

For all *FoodCAT* images, we applied the SR method in order to make both image dimensions, width and height, bigger or equal to 256. In Fig. 3, we show the behaviour of the SR algorithm applied on a Food-101 image. On the left, we show the original image (512x512) resized to the network's input 256x256, and on the right, we show the same image after resizing it to a smaller resolution than the network's input and applying the SR method for also obtaining a results of 256x256. Thus, we simulate the result of the SR procedure on FoodCAT images: first, improvement through SR and second, resizing to the network's input. We can see that, from a human perception perspective, applying the SR to a low resolution image does not affect the result. Also, when computing the histogram of both images (see Fig. 4), one can see that the difference between them is negligible.

Experimental Results

We need to test the performance of the convolutional neural network on both: dish and food category recognition. **Dish recognition**: One of the richest public domain datasets is the *Food-101* dataset. Since there is small intersection of both datasets, we decided to combine the *FoodCAT* and the *Food-101* dataset in order to build a joint classification model for several types of food. However, in this case we must deal with the differences in image resolution. In order to tackle this problem, we compared the classification on three different dataset configurations (see Fig. 5).

a) *Food-101+FoodCAT*: in this experiment, we use the original images. While all pictures in *Food-101* dataset have similar dimension (width or height) equal to 512, the pictures in *FoodCAT* have a huge diversity in resolutions and do not follow any pattern. On average, their resolution is below 256x256.

b) *Food-101* halved+*FoodCAT*: in this experiment, we decreased the resolution of all images in *Food-101* to make them more alike *FoodCAT*.

c) *Food-101+FoodCAT* with SR: in this experiment, we increased the resolution of all images in *FoodCAT* with the SR technique. Therefore, augmenting the resolution allows to reach a higher fidelity than increasing it with a standard resizing method.

FIGURE 3. Example of SR used in a high resolution image. Left: original image 512x512 resized to 256x256. Right: original image reduced at 40% 230x230, then increased by the SR two times to 460x460, and finally resized to 256x256.

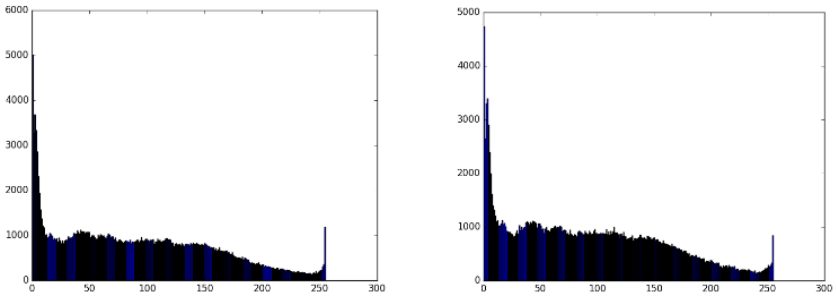

FIGURE 4. Histograms of the original image (left), and the SR (right).

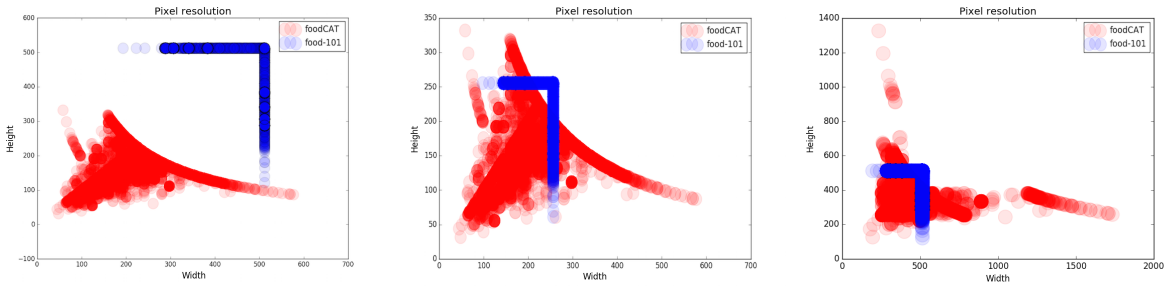

FIGURE 5. Plots of image dimension distributions: left: *Food-101+FoodCAT*; center: *Food-101* halved+*FoodCAT* with resolution halved, and right: *Food-101+FoodCAT* with SR.

Another of the problems, we have to deal with, when joining two different datasets is the unbalance of classes. Table 2 shows the number of images per learning phase either when using all images (top row) or a maximum of 500 images per class for balance (bottom row).

As a result, dish recognition is performed over *FoodCAT* and *Food-101*, having 115+101 classes to classify respectively. We study the network performance depending on image resolutions and balanced/unbalanced classes. The 6 different experiments are listed below, denoting *GoogleNet* as 'G' and *VGG-19* 'V':

1. G: *Food-101* + *FoodCAT* with SR.
2. G: *Food-101* + *FoodCAT* with SR, all balanced.
3. G: *Food-101* halved + *FoodCAT*.

TABLE 2. Number of images per learning phase (training, validation and testing) over the complete dataset and the balanced one. The values are presented giving the total number of images in addition to the relative contribution of each dataset in brackets (*Food-101+FoodCAT*).

	training	**validation**	**testing**	**total**
Complete	116.248 (80.800+35.448)	14.540 (10.100+4.440)	14.516 (10.100+4.416)	145.304 (101.000+44.304)
Balanced	73.085 (40.400+32.685)	9.143 (5.050+4.093)	9.124 (5.050+4.074)	91.352 (50.500+40.852)

4. G: *Food-101* halved + *FoodCAT*, all balanced.
5. V: *Food-101* + *FoodCAT*.
6. V: *Food-101* + *FoodCAT*, all balanced.

For all the experiments, we fine-tune our networks after pre-training them on the ImageNet dataset.

Table 3 organises the results of all the 6 different experiments applied either on both datasets ('A, B') or on *FoodCAT* only ('B'). We set the best $AT1$, $AT5$, and $NAT1$ in bold, for each of the tested datasets (*Food-101+FoodCAT* or *FoodCAT*). We can see that the best results for the dataset *FoodCAT* (columns 'B') are achieved by a CNN trained from the original dataset (without SR) with balanced classes (experiment 6). It shows the importance of the balanced classes to recognize, with similar accuracy, different datasets with a single CNN. Furthermore, the results of the test in both datasets together (columns 'A, B') are better, when we use all samples in both datasets during the training phase with the method SR applied for the *FoodCAT*. This CNN is the one used in experiment 1, and it also achieves the second best result for the $AT1$ over the *FoodCAT* dataset, with a score of 50.02, just 0.57 less than the balanced datasets with VGG (experiment 6). Moreover, adding all scores for the accuracy $AT1$ and $AT5$, over the two tests 'A, B' and 'B', experiment 1 has the highest value of 289.44 followed by experiment 6 with value 288.09.

With all this data, we conclude that the best model is the *GoogleNet* trained from all samples of both datasets, with the SR method applied for *FoodCAT*, corresponding to experiment 1.

TABLE 3. Results of the experiments from 1 to 6. A=*Food-101*, B=*FoodCAT*.

Experiment	**1**		**2**		**3**		**4**		**5**		**6**	
Datasets	A, B	B	A, B	B	A, B	B	A, B	B	A, B	B	A, B	B
$AT1$	**68.07**	50.02	62.41	48.94	67.16	49.66	61.28	48.85	67.74	48.12	65.16	**50.59**
$AT5$	**89.53**	81.82	86.81	81.63	89.27	82.07	86.52	80.92	89.28	81.03	88.94	**83.40**
$NAT1$	59.08	44.25	57.91	44.44	58.57	44.31	56.99	44.44	58.18	42.34	**60.74**	**46.53**

Food categories recognition: The recognition of food categories is performed over the *FoodCAT* dataset by fine-tuning the *GoogleNet* CNN trained previously with the large dataset ImageNet. We study the network performance depending on if we train all layers or only the last one, the fully-connected layer. Table 4 shows the results obtained for this task. First, if we have a limited machine or limited time, we show that fine-tuning just the fully-connected layer over a model previously trained on a large dataset as *ImageNet* [17], it can give a good enough performance. Training all layers, we achieve recognition of food categories over Catalan food with $AT1 = 72.29$ and $AT5 = 97.07$. Taking care of the difference of samples on each class, the normalized measure also gives a high performance, with $NAT1 = 65.06$.

TABLE 4. Performance and learning time, fine-tuning the *GoogleNet* model over the food categories labels. We show the results for two experiments done: training all layers, and only training the last fully-connected.

	AT1	**AT5**	**NAT1**	**# Iterations**	**Best iteration**	**Time executing**
FC	61.36	93.39	50.78	1.000.000	64.728	12h
All layers	**72.29**	**97.07**	**65.06**	900.000	49.104	24h

Figure 6 shows the normalized Confusion Matrix for the *GoogleNet* model trained over all layers. It is not surprising that 'Desserts and sweets' is the category that the net can recognize better, as it is also the class with more samples in the dataset with 11.933 images, followed by 'Meats' with 7.373. We also must note that the classes with

less samples in our dataset are 'Snails' and 'Mushrooms', but those specific classes can also be found in the *ImageNet* (the dataset used for the pre-trained model that we are using) that explains the good performance of the network on them.

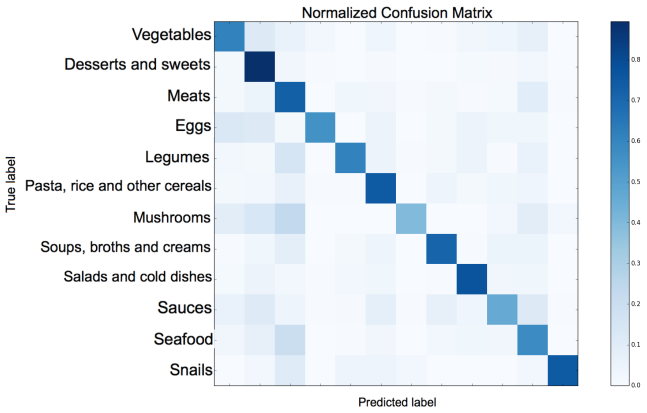

FIGURE 6. Normalized CM of *GoogleNet* model trained over the all layers to recognize food categories.

CONCLUSIONS

In this paper, we presented the novel and challenging multi-labeled dataset related to the Catalan diet called *FoodCAT*. For the first kind of labels, the dataset is divided into 115 food classes with an average of 400 images per dish. For the second kind of labels, the dataset is divided into 12 food categories with an average of 3800 images per dish.

We explored the food classes recognition and found that the best model is obtained by fine-tuning the GoogleNet network on the datasets *FoodCAT*, after increasing the resolution with the Super-Resolution method and *Food-101*. This model achieves the highest accuracy top-1 with 68.07%, and top-5 with 89.53%, testing both datasets together, and top-1 with 50.02%, and top-5 with 81.82%, testing only *FoodCAT*. Regarding the food categories recognition, we achieved the highest accuracy top-1 with 72.29% and top-5 with 97.07%, after fine-tuning the *GoogleNet* model for all layers. Our next steps are to increase the dataset and explore other architectures of convolutional neural networks for food recognition.

ACKNOWLEDGMENTS

This work was partially funded by TIN2015-66951-C2-1-R, La Marató de TV3, project 598/U/2014 and SGR 1219. P. Radeva is supported by an *ICREA Academia* grant. Thanks to the University of Groningen for letting us use the Peregrine HPC cluster.

REFERENCES

[1] F. Monteiro-Silva. Olive oil's polyphenolic metabolites - from their influence on human health to their chemical synthesis. *ArXiv e-prints 1401.2413*, January 2014.

[2] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014. URL `http://arxiv.org/abs/1409.4842`.

[3] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, arXiv:1409.1556, 2014.

[5] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

[6] Y. Kawano and K. Yanai. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In *Proc. of ECCV Workshop on Transferring and Adapting Source Knowledge in Computer Vision (TASK-CV)*, 2014.

[7] Atsushi Tatsyma and Aono Masaki. Food image recognition using covariance of convolutional layer feature maps. *IEICE TRANSACTIONS on Information and Systems*, 99(6):1711–1715, 2016.

[8] Marc Bolaños and Petia Radeva. Simultaneous food localization and recognition. *In Proceedings of the International Conference on Pattern Recognition (in press)*, 2016. URL `http://arxiv.org/abs/1604.07953`.

[9] Yuji Matsuda, Hajime Hoashi, and Keiji Yanai. Recognition of multiple-food images by detecting candidate regions. In *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo, ICME 2012, Melbourne, Australia, July 9-13, 2012*, pages 25–30, 2012. doi: 10.1109/ICME.2012.157. URL `http://dx.doi.org/10.1109/ICME.2012.157`.

[10] Institut Catalá de la Cuina. *Corpus del patrimoni culinari catalá*. Edicions de la Magrana, 2011. ISBN 9788482649498.

[11] Hajime Hoashi, Taichi Joutou, and Keiji Yanai. Image recognition of 85 food categories by feature fusion. In *12th IEEE International Symposium on Multimedia, ISM 2010, Taichung, Taiwan, December 13-15, 2010*, pages 296–301, 2010. doi: 10.1109/ISM.2010.51. URL `http://dx.doi.org/10.1109/ISM.2010.51`.

[12] Taichi Joutou and Keiji Yanai. A food image recognition system with multiple kernel learning. In *Proceedings of the 16th IEEE International Conference on Image Processing*, ICIP'09, pages 285–288, Piscataway, NJ, USA, 2009. IEEE Press. ISBN 978-1-4244-5653-6. URL `http://dl.acm.org/citation.cfm?id=1818719.1818816`.

[13] Ren Wu, Shengen Yan, Yi Shan, Qingqing Dang, and Gang Sun. Deep image: Scaling up image recognition. *CoRR*, arXiv:1501.02876, 2015. URL `http://arxiv.org/abs/1501.02876`.

[14] Zhaowen Wang, Ding Liu, Jianchao Yang, Wei Han, and Thomas Huang. Deep networks for image super-resolution with sparse prior. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 370–378, 2015.

[15] X. Jin, Y. Chen, J. Dong, J. Feng, and S. Yan. Collaborative Layer-wise Discriminative Learning in Deep Neural Networks. *ArXiv e-prints*, July 2016.

[16] Johannes Fürnkranz and Thorsten Joachims, editors. *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, 2010. Omnipress.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Food Ingredients Recognition Through Multi-label Learning

Marc Bolaños[1,2( )], Aina Ferrà[1], and Petia Radeva[1,2]

[1] Universitat de Barcelona, Barcelona, Spain
{marc.bolanos,aferrama10.alumnes,petia.ivanova}@ub.edu
[2] Computer Vision Center, Bellaterra, Spain

Abstract. Automatically constructing a food diary that tracks the ingredients consumed can help people follow a healthy diet. We tackle the problem of food ingredients recognition as a multi-label learning problem. We propose a method for adapting a highly performing state of the art CNN in order to act as a multi-label predictor for learning recipes in terms of their list of ingredients. We prove that our model is able to, given a picture, predict its list of ingredients, even if the recipe corresponding to the picture has never been seen by the model. We make public two new datasets suitable for this purpose. Furthermore, we prove that a model trained with a high variability of recipes and ingredients is able to generalize better on new data, and visualize how it specializes each of its neurons to different ingredients.

1 Introduction

People's awareness about their nutrition habits is increasing either because they suffer from some kind of food intolerance; they have mild or severe weight problems; or they are simply interested in keeping a healthy diet. This increasing awareness is also being reflected in the technological world. Several applications exist for manually keeping track of what we eat, but they rarely offer any automatic mechanism for easing the tracking of the nutrition habits [2]. Tools for automatic food and ingredient recognition could heavily alleviate the problem.

Since the reborn of Convolutional Neural Networks (CNNs), several works have been proposed to ease the creation of nutrition diaries. The most widely spread approach is food recognition [8]. These proposals allow to recognize the type of food present in an image and, consequently, could allow to approximately guess the ingredients contained and the overall nutritional composition. The main problem of these approaches is that no dataset covers the high amount of existent types of dishes worldwide (more than 8,000 according to Wikipedia).

On the other hand, a clear solution for this problem can be achieved if we formulate the task as an ingredients recognition problem instead [6]. Although tens of thousands of types of dishes exist, in fact they are composed of a much smaller number of ingredients, which at the same time define the nutritional composition of the food. If we formulate the problem from the ingredients recognition perspective, we must consider the difficulty of distinguishing the presence

S. Battiato et al. (Eds.): ICIAP 2017 International Workshops, LNCS 10590, pp. 394–402, 2017.
https://doi.org/10.1007/978-3-319-70742-6_37

of certain ingredients in cooked dishes. Their visual appearance can greatly vary from one dish to another (e.g. the appearance of the ingredient 'apple' in an 'apple pie', an 'apple juice' or a 'fresh apple'), and in some cases they can even be invisible at sight without the proper knowledge of the true composition of the dish. An additional benefit of approaching the problem from the ingredients recognition perspective is that, unlike in food recognition, it has the potential to predict valid outputs on data that has never been seen by the system.

In this paper, we explore the problem of food ingredients recognition from a multi-label perspective by proposing a model based on CNNs that allows to discover the ingredients present in an image even if they are not visible to the naked eye. We present two new datasets for tackling the problem and prove that our method is capable of generalizing to new data that has never been seen by the system. Our contributions are four-fold. (1) Propose a model for food ingredients recognition; (2) Prove that by using a varied dataset of images and their associated ingredients, the generalization capabilities of the model on never seen data can be greatly boosted; (3) Delve into the inner layers of the model for analysing the ingredients specialization of the neurons; and (4) Release two datasets for ingredients recognition.

This paper is organized as follows: in Sect. 2, we review the state of the art; in Sect. 3, explain our methodology; in Sect. 4, we present our proposed datasets, show and analyse the results of the experiments performed, as well as interpret the predictions; and in Sect. 5, we draw some conclusions.

2 Related Work

Food analysis. Several works have been published on applications related to automatic food analysis. Some of them proposed food detection models [1] in order to distinguish when there is food present in a given image. Others focused on developing food recognition algorithms, either using conventional hand-crafted features, or powerful deep learning models [8]. Others have applied food segmentation [11]; use multi-modal data (i.e. images and recipe texts) for recipe recognition [15]; tags from social networks for food characteristics perception [9]; food localization and recognition in the wild for egocentric vision analysis [3], etc.

Multi-Label learning. Multi-label learning [13] consists in predicting more than one output category for each input sample. Thus, the problem of food ingredients recognition can be treated as a multi-label learning problem. Several works [14] argued that, when working with CNNs, they have to be reformulated for dealing with multi-label learning problems. Some multi-label learning works have already been proposed for restaurant classification. So far, only one paper [6] has been proposed related to ingredients recognition. Their dataset, composed of 172 food types, was manually labelled considering visible ingredients only, which limits it to find 3 ingredients on average. Furthermore, they propose a double-output model for simultaneous food type recognition and multi-label ingredients recognition. Although, the use of the food type for optimizing the model limits

its capability of generalization only to seen recipes and food types. This fact becomes an important handicap in a real-world scenario when dealing with new recipes. As we demonstrate in Sects. 4.3 and 4.4, unlike [6], our model is able to: (1) recognize the ingredients appearing in unseen recipes (see Fig. 1b); (2) learn abstract representations of the ingredients directly from food appearance (see Fig. 2); and (3) infer invisible ingredients.

Interpreting learning through visualization. Applying visualization techniques is an important aspect in order to interpret what has been learned by our model. The authors in [17] have focused on proposing new ways of performing this visualization. At the same time, they have proven that CNNs have the ability to learn high level representations of the data and even hidden inter-related information, which can help us when dealing with ingredients that are apparently invisible in the image.

3 Methodology

Deep multi-ingredients recognition. Most of the top performing CNN architectures have been originally proposed and intended for the problem of object recognition. At the same time, they have been proven to be directly applicable to other related classification tasks and have served as powerful pre-trained models for achieving state of the art results. In our case, we compared either using the InceptionV3 [12] or the ResNet50 [7] as the basic architectures for our model. We pre-trained it on the data from the ILSVRC challenge [10] and modified the last layer for applying a multi-label classification over the N possible output ingredients. When dealing with classification problems, CNNs typically use the softmax activation in the last layer. The softmax function allows to obtain a probability distribution for the input sample x over all possible outputs and thus, predicts the most probable outcome, $\hat{y}_x = \arg\max_{y_i} P(y_i|x)$.

The softmax activation is usually combined with the categorical cross-entropy loss function L_c during model optimization, which penalizes the model when the optimal output value is far away from 1:

$$L_c = -\sum_x \log(P(\hat{y}_x|x)). \tag{1}$$

In our model, we are dealing with ingredients recognition in a multi-label framework. Therefore, the model must predict for each sample x a set of outputs represented as a binary vector $\hat{Y}_x = \{\hat{y}_x^1, ..., \hat{y}_x^N\}$, where N is the number of output labels and each $\hat{y}_x^i$ is either 1 or 0 depending if it is present or not in sample x. For this reason, instead of softmax, we use a sigmoid activation function:

$$P(y_i|x) = \frac{1}{1 - \exp^{-f(x)_i}} \tag{2}$$

which allows to have multiple highly activated outputs. For considering the binary representation of $\hat{Y}_x$, we chose the binary cross-entropy function L_b [5]:

$$L_b = -\sum_{x}\sum_{i}^{N}(\hat{y}_x^i \cdot log(P(y_i|x)) + (1-\hat{y}_x^i) \cdot log(1-P(y_i|x))) \quad (3)$$

which during backpropagation rewards the model when the output values are close to the target vector $\hat{Y}_x$ (i.e. either close to 1 for positive labels or close to 0 for negative labels).

4 Results

In this section, we describe the two datasets proposed for the problem of food ingredients recognition. Later we describe our experimental setup and at the end, we present the final results obtained both for ingredients recognition on known classes as well as recognition results for generalization on samples never seen by the model.

4.1 Datasets

In this section we describe the datasets proposed for food ingredients recognition and the already public datasets used.

Food101 [4] is one of the most widely extended datasets for food recognition. It consists of 101,000 images equally divided in 101 food types.

Ingredients101[1] is a dataset for ingredients recognition that we constructed and make public in this article. It consists of the list of most common ingredients for each of the 101 types of food contained in the Food101 dataset, making a total of 446 unique ingredients (9 per recipe on average). The dataset was divided in training, validation and test splits making sure that the 101 food types were balanced. We make public the lists of ingredients together with the train/val/test split applied to the images from the Food101 dataset.

Recipes5k[2] is a dataset for ingredients recognition with 4,826 unique recipes composed of an image and the corresponding list of ingredients. It contains a total of 3,213 unique ingredients (10 per recipe on average). Each recipe is an alternative way to prepare one of the 101 food types in Food101. Hence, it captures at the same time the intra-class variability and inter-class similarity of cooking recipes. The nearly 50 alternative recipes belonging to each of the 101 classes were divided in train, val and test splits in a balanced way. We make also public this dataset together with the splits division. A problem when dealing with the 3,213 raw ingredients is that many of them are sub-classes (e.g. 'sliced tomato' or 'tomato sauce') of more general versions of themselves (e.g. 'tomato').

[1] http://www.ub.edu/cvub/ingredients101/.
[2] http://www.ub.edu/cvub/recipes5k/.

Thus, we propose a simplified version by applying a simple removal of overly-descriptive particles[3] (e.g. 'sliced' or 'sauce'), resulting in 1,013 ingredients used for additional evaluation (see Sect. 4.3).

We must note the difference between our proposed datasets and the one from [6]. While we consider any present ingredient in a recipe either visible or not, the work in [6] only labelled manually the visible ingredients in certain foods. Hence, a comparison between both works is infeasible.

4.2 Experimental Setup

Our model was implemented in Keras[4], using Theano as backend. Next, we detail the different configurations and tests performed. **Random prediction**: (baseline) a set of K labels are generated uniformly distributed among all possible outputs. K depends on the average number of labels per recipe in the corresponding dataset. **InceptionV3 + Ingredients101**: InceptionV3 model pre-trained on ImageNet and adapted for multi-label learning. **ResNet50 + Ingredients101**: ResNet50 model pre-trained on ImageNet and adapted for multi-label learning. **InceptionV3 + Recipes5k**: InceptionV3 model pre-trained on InceptionV3 + Ingredients101. **ResNet50 + Recipes5k**: ResNet50 model pre-trained on ResNet50 + Ingredients101.

4.3 Experimental Results

In Table 1, we show the ingredient recognition results on the Ingredients101 dataset. In Fig. 1a some qualitative results are shown. Both the numerical results and the qualitative examples prove the high performance of the models in most of the cases. Note that although a multi-label classification is being applied, considering that all the samples from a food class share the same set of ingredients, the model is indirectly learning the inherent food classes. Furthermore, looking at the results on the Recipes5k dataset in Table 2 (top), we can see that the very same model obtains reasonable results even considering that it was

Table 1. Ingredients recognition results obtained on the dataset Ingredients101. Prec stands for *Precision*, Rec for *Recall* and F_1 for F_1 *score*. All measures reported in %. The best test results are highlighted in boldface.

	Validation			Test		
	Prec	Rec	F_1	Prec	Rec	F_1
Random prediction	2.05	2.01	2.03	2.06	2.01	2.04
InceptionV3 + Ingredients101	80.86	72.12	76.24	83.51	**76.87**	80.06
ResNet50 + Ingredients101	84.80	67.62	75.24	**88.11**	73.45	**80.11**

[3] https://github.com/altosaar/food2vec.
[4] www.keras.io.

(a) Ingredients101 samples.

(b) Recipes5k using the fine-grained 3,213 ingredients (left), and using the 1,013 simplified ingredients (right).

Fig. 1. Our method's results. TPs in green, FPs in red and FNs in orange. (Color figure online)

Table 2. Ingredients recognition results on Recipes5k (top) and on Recipes5k simplified (bottom). Prec stands for *Precision*, Rec for *Recall* and F_1 for F_1 *score*. All measures reported in %. Best test results are highlighted in boldface.

	Validation			Test		
	Prec	Rec	F_1	Prec	Rec	F_1
Random prediction	0.33	0.32	0.33	0.54	0.53	0.53
InceptionV3 + Ingredients101				23.80	18.24	20.66
ResNet50 + Ingredients101				26.28	16.85	20.54
InceptionV3 + Recipes5k	36.18	20.69	26.32	35.47	**21.00**	**26.38**
ResNet50 + Recipes5k	38.41	19.67	26.02	**38.93**	19.57	26.05
Random prediction	6.27	6.29	6.28	6.14	6.24	6.19
InceptionV3 + Ingredients101				44.01	34.04	38.39
ResNet50 + Ingredients101				47.53	30.91	37.46
InceptionV3 + Recipes5k	56.77	31.40	40.44	55.37	31.52	40.18
ResNet50 + Recipes5k	56.73	28.07	37.56	**58.55**	28.49	38.33
InceptionV3 + Recipes5k simplified	53.91	42.13	47.30	53.43	**42.77**	**47.51**

not specifically trained on that dataset. Note that only test results are reported for the models trained on Ingredients101 because we only intend to show its generalization capabilities on new data.

Comparing the results with the models specifically trained on Recipes5k, it appears that, as expected, a model trained on a set of samples with high variability of output labels is more capable of obtaining high results on never seen recipes. Thus, it is more capable of generalizing on unseen data.

Table 2 (bottom) shows the results on the Recipes5k dataset with a simplified list of ingredients. Note that for all tests, the list was simplified only during the evaluation procedure for maintaining the fine-grained recognition capabilities of the model, with the exception of *Inception V3 + Recipes5k simplified*, where the simplified set was also used for training. The simplification of the ingredients list enhances the capabilities of the model when comparing the results, reaching more than 40% in the F_1 metric and 47.5% also training with them.

Figure 1b shows a comparison of the output of the model either using the fine-grained or the simplified list of ingredients. Overall, although usually only a single type of semantically related fine-grained ingredients (e.g. 'large eggs', 'beaten eggs' or 'eggs') appears at the same time in the ground truth, it seems that the model is inherently learning an embedding of the ingredients. Therefore, it is able to understand that some fine-grained ingredients are related and predicts them at once in the fine-grained version (see waffles example).

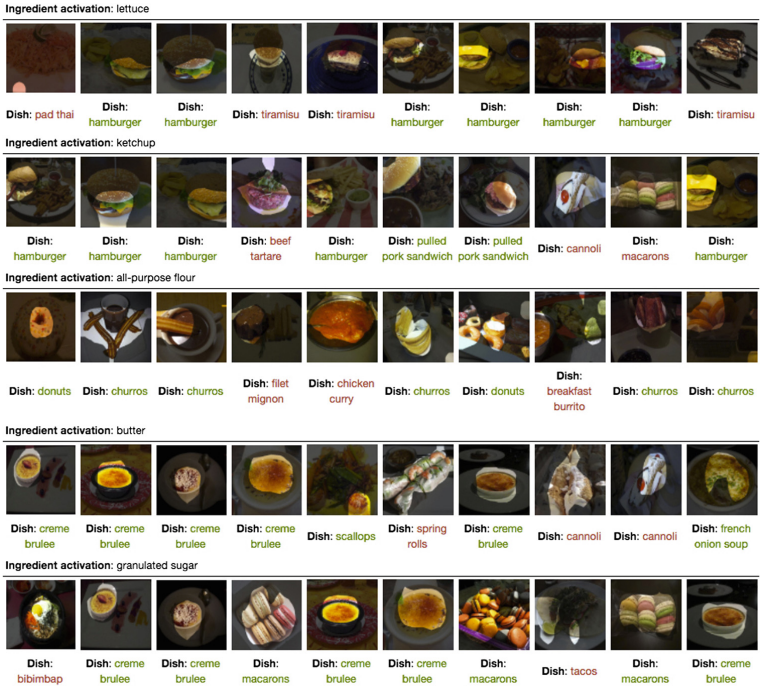

Fig. 2. Visualization of neuron activations. Each row is associated to a specific neuron from the network. The images with top activation are shown as well as the top ingredient activation they have in common. The name of their respective food class is only for visualization purposes and is displayed in green if the recipe contains the top ingredient. Otherwise, it is shown in red. (Color figure online)

4.4 Neuron Representation of Ingredients

When training a CNN model, it is important to understand what it is able to learn and interpret from the data. To this purpose, we visualized the activations of certain neurons of the network in order to interpret what is it able to learn.

Figure 2 shows the results of this visualization. As we can see, it appears that certain neurons of the network are specialized to distinguish specific ingredients. For example, most images of the 1st and 2nd rows illustrate that the characteristic shape of a hamburger implies that it will probably contain the ingredients 'lettuce' and 'ketchup'. Also, looking at the 'granulated sugar' row, we can see that the model learns to interpret the characteristic shape of *creme brulee* and *macarons* as containing sugar, although it is not specifically seen in the image.

5 Conclusions and Future Work

Analysing both the quantitative and qualitative results, we can conclude that the proposed model and the two datasets published offer very promising results for the multi-label problem of food ingredients recognition. Our proposal allows to obtain great generalization results on unseen recipes and sets the basis for applying further, more detailed food analysis methods. As future work, we will create a hierarchical structure [16] relationship of the existent ingredients and extend the model to utilize this information.

References

1. Aguilar, E., Bolaños, M., Radeva, P.: Exploring food detection using CNNs. In: Proceedings of the 16th International Conference on Computer Aided Systems Theory, pp. 242–243. Springer (2017)
2. Aizawa, K., Ogawa, M.: FoodLog: multimedia tool for healthcare applications. IEEE MultiMedia **22**(2), 4–8 (2015)
3. Bolaños, M., Radeva, P.: Simultaneous food localization and recognition. In: Proceedings of the 23rd International Conference on Pattern Recognition (ICPR) (2016)
4. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 446–461. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_29
5. Buja, A., Stuetzle, W., Shen, Y.: Loss functions for binary class probability estimation and classification: structure and applications. Working draft, November 2005
6. Chen, J., Ngo, C.-W.: Deep-based ingredient recognition for cooking recipe retrieval. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 32–41. ACM (2016)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

8. Martinel, N., Foresti, G.L., Micheloni, C.: Wide-slice residual networks for food recognition. arXiv preprint arXiv:1612.06543 (2016)
9. Ofli, F., Aytar, Y., Weber, I., al Hammouri, R., Torralba, A.: Is saki# delicious? the food perception gap on instagram and its relation to health. arXiv preprint arXiv:1702.06318 (2017)
10. Russakovsky, O., Deng, J., Hao, S., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
11. Shimoda, W., Yanai, K.: CNN-based food image segmentation without pixel-wise annotation. In: Murino, V., Puppo, E., Sona, D., Cristani, M., Sansone, C. (eds.) ICIAP 2015. LNCS, vol. 9281, pp. 449–457. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23222-5_55
12. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
13. Tsoumakas, G., Katakis, I.: Multi-label classification: an overview. Int. J. Data Warehouse. Min. **3**(3), 1–13 (2006)
14. Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., Xu, W.: CNN-RNN: a unified framework for multi-label image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2285–2294 (2016)
15. Wang, X., Kumar, D., Thome, N., Cord, M., Precioso, F.: Recipe recognition with large multimodal food dataset. In: 2015 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1–6. IEEE (2015)
16. Wu, H., Merler, M., Uceda-Sosa, R., Smith, J.R.: Learning to make better mistakes: semantics-aware visual food recognition. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 172–176. ACM (2016)
17. Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., Lipson, H.: Understanding neural networks through deep visualization. arXiv preprint arXiv:1506.06579 (2015)

Exploring Food Detection Using CNNs

Eduardo Aguilar( ), Marc Bolaños, and Petia Radeva

Universitat de Barcelona and Computer Vision Center, Barcelona, Spain
{eduardo.aguilar,marc.bolanos,petia.ivanova}@ub.edu

Abstract. One of the most common critical factors directly related to the cause of a chronic disease is unhealthy diet consumption. Building an automatic system for food analysis could enable a better understanding of the nutritional information associated to the food consumed and thus, help taking corrective actions on our diet. The Computer Vision community has focused its efforts on several areas involved in visual food analysis such as: food detection, food recognition, food localization, portion estimation, among others. For food detection, the best results in the state of the art were obtained using Convolutional Neural Networks. However, the results of all different approaches were tested on different datasets and, therefore, are not directly comparable. This article proposes an overview of the last advances on food detection and an optimal model based on the GoogLeNet architecture, Principal Component Analysis, and a Support Vector Machine that outperforms the state of the art on two public food/non-food datasets.

Keywords: CNN · PCA · GoogLeNet · SVM · Food detection

1 Introduction

In the last decades, the amount of people with overweight and obesity is progressively increasing [1], whom generally maintain an excessive unhealthy diet consumption. Additionally to the physical and psychological consequences involved to their condition, these people are more prone to acquire chronic diseases such as heart diseases, respiratory diseases, and cancer [2]. Consequently, it is highly necessary to build tools that offer high accuracy in nutritional information estimation from ingested foods, and thus, improve the control of food consumption and treat people with nutritional problems.

Recently, the computer vision community has focused its efforts on several areas devoted to developing automated systems for visual food analysis, which usually involve using a food detection method [3–6]. These methods, also called food/non-food classification, have as purpose to determine the presence or absence of food in an image. Generally, they are applied as a pre-processing prior to food analysis, and can also be useful for selecting food images from huge datasets acquired from the WEB or from wearable devices.

R. Moreno-Díaz et al. (Eds.): EUROCAST 2017, Part II, LNCS 10672, pp. 339–347, 2018.
https://doi.org/10.1007/978-3-319-74727-9_40

Food detection has been investigated in the literature in different works [3,6–9], where it has been proven that the best results obtained are based on Convolutional Neural Networks (CNN). The first method based using this technique was proposed by [3], which achieved a 93.8% using AlexNet model [10] on a dataset composed of 1,234 food images and 1,980 non-food images acquired from social media sources. They proved that using a CNN provided a 4% higher accuracy compared to using hand crafted features [7]. In [11], the authors improved the accuracy on this dataset to 99.1% using the NIN model [12]. In addition, they evaluated their model on other datasets, IFD and FCD, obtaining 95% and 96% of accuracy, respectively. Evaluation on a huge dataset with over 200,000 images constructed from Food101 [13] and ImageNet Challenge was done in [5], where the authors achieved 99.02% using an efficient CNN model based on inception module called GoogLeNet [14]. The same model was used in [4], the authors obtained 95.64% of accuracy on a dataset composed of Food101; food-related images extracted from the ImageNet Challenge dataset; and Pascal [15] (used as non-food images). Evaluation of different CNN models and settings was proposed by [9] on a dataset that we call RagusaDS. The authors obtained the best results using AlexNet for feature extraction and Binary SVM [16] for classification. In terms of accuracy, they achieved 94.86%. In [6], the authors apply fine-tuning on the last six layers of a GoogLeNet obtaining high accuracy, but tested their model on a balanced dataset of only 5,000 images (Food-5k). Since the proposed models were evaluated on different datasets, the results obtained are not directly comparable. Therefore, in order to compare our results with the state of the art, we selected the available datasets with more than 15,000 images.

Furthermore, we explored the food detection problem using the GoogLeNet, because this CNN model presented the best results in the classification of objects in the ILSVRC challenge [17]. In particular for food detection it has also presented good results on multiplies datasets with images acquired in different conditions [4–6]. Specifically, we propose a food detection model combining GoogLeNet for feature extraction, PCA [18] for feature selection and SVM for classification, which prove the best accuracy in the state of the art with respect to the previous works on the same datasets.

This article is organized as follows: in Sect. 2, we present our methodology. In Sect. 3, we present and discuss the datasets used and the results obtained. Finally, in Sect. 4, we present conclusions and future work.

2 Methodology for Food Detection

We propose a methodology for food detection, which involves the use of the GoogLeNet model for feature extraction, PCA for feature selection and SVM for classification. In Fig. 1, we show the pipeline of our food detection approach which will be explained below.

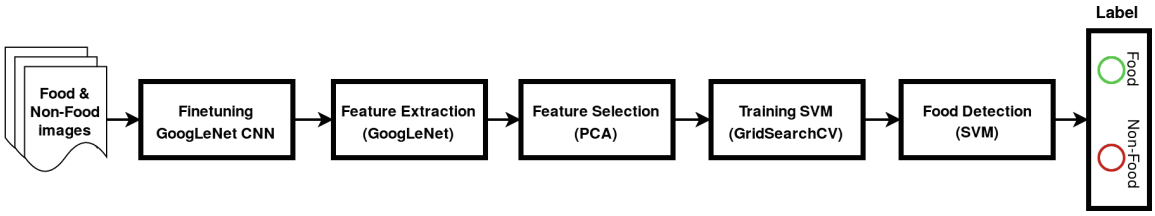

Fig. 1. Method overview for our food detection approach.

2.1 GoogLeNet for Feature Extraction

The first step in our methodology consists in training the GoogLeNet CNN model. For this purpose, we pre-train the GoogLeNet model on ImageNet [17], as a base model, and then we change the number of classes in the output layer for our binary classification problem (food/non-food). Then, GoogLeNet is fine-tuned on the last two layers until the accuracy on training set stops to increase, then we choose the model that gives the best accuracy on the validation set.

Once GoogLeNet is fine-tuned, we use the resulting model as a feature extractor. The feature vector for each image is extracted using the penultimate layer, with which a 1024-dimensional vector is obtained for each image. Then, we calculate a transformation that distributes normally the data through a Gaussian distribution function with zero mean and unit variance, by means of the feature vectors obtained from the training set. Finally, we normalize the data, in a range of $[-1, 1]$, by applying this transformation to each extracted feature vector.

2.2 PCA for Feature Selection

The following step in our methodology consists in reducing the dimensions of the feature vectors obtained in the previous steps by means of Principal Component Analysis (PCA) [18], which transforms the data to a new coordinate system leaving the greatest variance of the images in the first axes (principal components). We apply PCA on all feature vectors normalized from the training set and then the principal components are analyzed to select the first dimensions that retain the most discriminant information. To do this, we selected the features based on the Kaiser Criterion [19], which consists of retaining those components with eigenvalues greater than 1. The feature vectors reduced are used during the training of SVM and also during classification.

2.3 SVM for Classification

An SVM is a classification algorithm that optimizes a boundary $f(x) = Wx + b$ for maximizing the margin between two types of data, where the maximum margin is found solving a quadratic optimization problem. The dual SVM formulation is defined as follows:

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$s.t. \quad 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, n, \quad \sum_{i=1}^{n} \alpha_i yi = 0.$$

where x_i are the reduced feature vectors calculated from the n training set images, y_i the images class identified by the label -1 or 1, α_i are the Lagrange multipliers, and $K(x_i, x_j) = \tan h(\gamma x_i^T x_j)$ is the chosen kernel function, as in [9], for SVM classifier. The parameters C and γ are obtained by means of the GridSearchCV strategy, and the parameters α_i are adjusted during the optimization.

With the solution of the optimization problem, we compute the parameters of the objetive function f(x), where $W = \sum_{i=1}^{n} \alpha_i y_i x_i$, and b is the bias. Finally, the class is predicted using the $sgn(f(x))$ function, which returns $+1$ when $f(x)$ is positive and -1 when $f(x)$ is negative.

3 Experiments

3.1 Datasets

In this section, we present the selected datasets for the evaluation of the proposed model and comparison of the results. Both datasets, FCD and RagusaDS, were selected because they contain a significant amount of images, at least 15,000, and also they have free access to the images.

FCD was constructed from two public datasets widely used: Food-101 [13] and Caltech-256 [20] for food and non-food images, respectively (see Fig. 2). Food-101 is a dataset for food recognition, which contains 101 international food categories with 1,000 images each one. Caltech-256 contains 256 categories of objects with a total of 30,607 images, in which each object has a minimum of 80 images. For the construction of FCD, not all images of these datasets were considered. To balance the amount of food and non-food, we selected 250 images for each category in the Food-101. The selection was based on the color histogram of the images, keeping those with the highest color variance within the same category and thus, keeping the most highly variable set, obtaining a total of 25,250 food images. In Caltech-256, all images were selected except the food-related ones, resulting in 28,211 non-food images. To evaluate our approach, we used 64% of the images for training, 16% validation and 20% test.

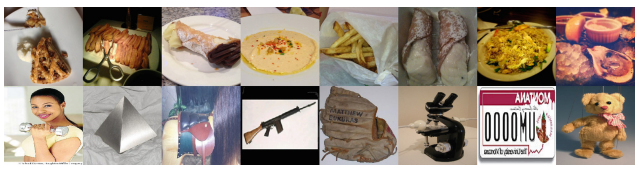

Fig. 2. Example of images contained in the FCD dataset. Top row shows food images from Food101 and bottom row shows non-food images from Caltech256.

RagusaDS consists of three datasets acquired in different conditions: UNICT-FD889 [21] and Flickr-Food [8] for positive; and Flickr-NonFood [8] for

negative samples. UNICT-FD889 is a dataset composed of 3,583 images of meals of 889 different dishes acquired from multiple perspectives with the same device in real-world scenarios, where images were acquired from a top view avoiding the presence of other objects. The Flickr images datasets were manually labeled as being food or non-food images. These datasets, which are called Flickr-Food and Flickr-NonFood, contain 4,805 images of food and 8,005 of non-food, respectively. Compared to UNICT-FD889, they contain less restricted images, and specifically for Flickr-Food the images can contain additional objects as well as food and were taken from different points of view. In total, the dataset contains 8,388 images of food and 8,005 of non-food (see Fig. 3). From the UNICT-FD889 dataset we split 80% of the data for training and 20% for validation. The first 3,583 images of Flickr-NonFood were also used for validation, and the remaining 4,422 as well as all images from Flickr-Food were used for testing.

Fig. 3. Example of images contained in the RagusaDS dataset. Top row shows food images from UNICT-FD889, middle row shows food images from Flickr-Food, and bottom row shows non-food images from Flickr-NonFood.

3.2 Experimental Setup

We used Caffe [22] for training our CNN model. We fine-tuned the last two layers of our model applying ten times the default learning rate. We set a learning rate of 1×10^3, with a decay of 0.96 every 5,000 iterations and a batch size of 32. We pre-processed the images by resizing them to 256×256 pixels, subtracted the training average of ImageNet and maintained the original color pixels scale. During training, data augmentation was applied by using horizontal mirroring and random crops of 224×224 pixels. During prediction, a center crop is applied.

The GoogLeNet was fine-tuned during 10 epochs, in the case of FCD, and during 40 epochs for RagusaDS. Training of the models was stopped when accuracy converged in the training set. The feature vector extracted from each image is reduced selecting the principal components based on the Kaiser Criterion, resulting in 186 dimensions on RagusaDS and 206 dimensions on FCD. As for the optimization of C and γ parameters, we applied a 3-fold cross validation. We defined a range of 14 values uniformly distributed on a base-10 logarithmic scale. In the case of the C parameter, we used a range from 1×10^{-4} to 1×10^2 and for γ parameter from 1×10^{-8} to 1×10^{-2}. Finally, the best parameters are used to train the SVM from scratch with all the training set.

3.3 Metrics

We used different metrics to evaluate the performance of our approach, namely: overall Accuracy (ACC), True Positive rate (TPr) and True Negative rate (TNr), which are defined as follows: $ACC = \frac{TP+TN}{T}$, where TP (True Positive) and TN (False Negative) are the amount of correctly classified images as Food and Non-Food, respectively; $TPr = \frac{TP}{TP+FN}$, where FN (False Negative) is the amount of misclassified images as Non-Food; $FNr = \frac{FN}{FP+TN}$, where FP (False Positive) is the amount of images misclassified as food.

3.4 Results

In this section, we present the results obtained during the experiments. In Table 1, the first two rows correspond to the state of the art algorithms that gave the best prediction on RagusaDS and FCD datasets, respectively. The last three methods are variations of our proposal, which is based on the GoogLeNet. The results show the ACC, the TPr and TNr obtained when evaluating each method on the FCD and RagusaDS datasets. In the case of the FCD, it can be seen that the model obtains a high precision in the global classification and maintains a slightly higher performance on TNr, which may be due to the small imbalance between food and non-food images of this dataset. On the other hand, for RagusaDS the difference between TPr and TNr is about 7% better for TNr. We believe that this occurs considering that food images used during training are very different from those used for evaluation and therefore the model is not able to recover enough discriminant information that allows to generalize over a sample acquired under different conditions. GoogLeNet + PCA-SVM is selected for the next experiment given that it achieved the best results on both datasets.

Table 1. Results obtained by models based on CNN on RagusaDS and FCD datasets on the food detection task. All results are reported in %.

	RagusaDS			FCD		
	ACC	TPr	TNr	ACC	TPr	TNr
AlexNet + SVM [9]	94.86	**94.28**	95.50	-	-	-
NIN [11]	-	-	-	96.4	96	97
GoogLeNet	94.66	91.53	98.06	98.87	98.48	**99.22**
GoogLeNet + SVM	94.95	91.53	**98.67**	98.96	**98.85**	99.06
GoogLeNet + PCA-SVM	**94.97**	91.57	**98.67**	**99.01**	**98.85**	99.15

Following, we trained the best model and evaluated its performance using RagusaDS and FCD datasets together, maintaining the same sets of training, validation and test, which we named RagusaDS + FCD. Table 2 shows the results obtained by training our approach using the training sets from RagusaDS + FCD

Table 2. Results obtained when GoogLeNet + PCA-SVM is trained on both datasets together (RagusaDS + FCD) and evaluated separately and jointly.

Test dataset	ACC	TPr	TNr
RagusaDS	95.78%	93.65%	98.10%
FCD	98.81%	98.60%	99.01%
RagusaDS + FCD	97.41%	96.19%	98.61%

and evaluating on the test sets from RagusaDS + FCD, RagusaDS and FCD. The results show that, when the model is trained on RagusaDS + FCD, it improves the classification significantly on RagusaDS although it presents a slight decrease on FCD. We believe that the improvement on RagusaDS is mainly due to an increase in the detection of food-related images. We deduce that by combining the training datasets, our method is able to extract features from various types of food acquired in different conditions, which allows to have a more robust classifier achieving a better generalization on the test set of RagusaDS dataset.

Some FPs FNs obtained in both datasets are shown in Fig. 4. Analyzing the FNs, we can observe that in the case of RagusaDS most errors occurred in images in which food was a liquid (drink, coffee, etc.). The reason for this is because the training set contains a wide variety of dishes but none of these correspond to beverages and therefore the classifier does not recognize them as food. In addition, other factors that influence classification are poorly labeled images such as food and also the cases where in the same image there are a lot of dishes. In the case of FCD, there are also some errors caused by wrong labels in both categories.

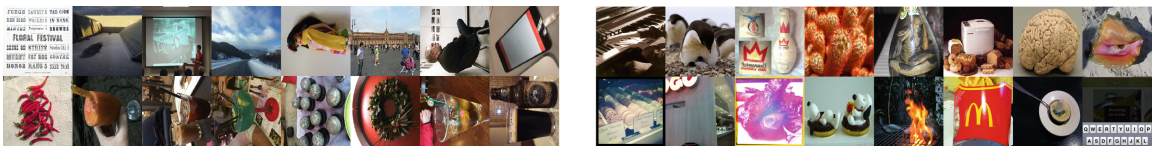

Fig. 4. FP (top) and FN (bottom) on RagusaDS (left) and FCD (right) datasets.

4 Conclusions

In this paper, we addressed the food detection problem and proposed a model that uses GoogLeNet for feature extraction, PCA for feature selection and SVM for classification. Furthermore, we applied a benchmark on the two more widely used publicly available datasets. From the results obtained, we observed that the best accuracy is achieved in both datasets with our proposed approach. Specifically, the improvement in the overall accuracy is more than 2% on FCD and about 1% for RagusaDS, when both datasets are combined for training and evaluated on the respective datasets. In addition, the overall accuracy when

combining both datasets is 97.41%. As a conclusion, we explored the problem of food detection comparing the last works in the literature and our proposed approach provides an improvement on the state of art with respect to both public datasets. Moreover, models based on GoogLeNet, independently of the settings, gave the highest accuracy on the food detection problem. As future work, we will evaluate the performance of CNN-based models on larger datasets containing a much wider range of dishes and beverages such as food images and diversity of environments for non-food images.

Acknowledgement. This work was partially funded by TIN2015-66951-C2, SGR 1219, CERCA, *ICREA Academia'2014*, CONICYT Becas Chile, FPU15/01347 and Grant 20141510 (Marató TV3). The funders had no role in the study design, data collection, analysis, and preparation of the manuscript. We acknowledge Nvidia Corporation for the donation of a Titan X GPU.

References

1. Ng, M., et al.: Global, regional, and national prevalence of overweight and obesity in children and adults during 1980–2013: a systematic analysis for the global burden of disease study 2013. Lancet **384**, 766–781 (2014)
2. World Health Organization: Diet, nutrition and the prevention of chronic diseases. WHO Technical Report Series, vol. 916, p. 149 (2003)
3. Kagaya, H., Aizawa, K., Ogawa, M.: Food detection and recognition using convolutional neural network. In: ACM Multimedia, pp. 1085–1088 (2014)
4. Bolaños, M., Radeva, P.: Simultaneous food localization and recognition. In: ICPR (2016)
5. Myers, A., et al.: Im2Calories: towards an automated mobile vision food diary. In: ICCV (2015)
6. Singla, A., Yuan, L., Ebrahimi, T.: Food/non-food image classification and food categorization using pre-trained GoogLeNet model. In: Proceedings of the 2nd International Workshop on MADiMa (2016)
7. Kitamura, K., Yamasaki, T., Aizawa, K.: FoodLog. In: Proceedings of the ACM Multimedia 2009 Workshop on Multimedia for Cooking and Eating Activities (2009)
8. Farinella, G.M., Allegra, D., Stanco, F., Battiato, S.: On the exploitation of one class classification to distinguish food vs non-food images. In: Murino, V., Puppo, E., Sona, D., Cristani, M., Sansone, C. (eds.) ICIAP 2015. LNCS, vol. 9281, pp. 375–383. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23222-5_46
9. Ragusa, F., et al.: Food vs non-food classification. In: Proceedings of the 2nd International Workshop on MADiMa (2016)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25, p. 19 (2012)
11. Kagaya, H., Aizawa, K.: Highly accurate food/non-food image classification based on a deep convolutional neural network. In: Murino, V., Puppo, E., Sona, D., Cristani, M., Sansone, C. (eds.) ICIAP 2015. LNCS, vol. 9281, pp. 350–357. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23222-5_43
12. Lin, M., Chen, Q., Yan, S.: Network in network. arXiv Preprint, p. 10 (2013)

13. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 446–461. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_29
14. Szegedy, C., et al.: Going deeper with convolutions. In: CVPR (2015)
15. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**, 303–338 (2010)
16. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**, 273–297 (1995)
17. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**, 211–252 (2015)
18. Jollie, I.T.: Principal component analysis. J. Am. Statist. Assoc. **98**, 487 (2002)
19. Kaiser, H.F.: The application of electronic computers to factor analysis. Edu. Psychol. Measur. **20**, 141–151 (1960)
20. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Caltech mimeo **11**, 20 (2007)
21. Farinella, G.M., Allegra, D., Stanco, F.: A benchmark dataset to study the representation of food images. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8927, pp. 584–599. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16199-0_41
22. Jia, Y. et al.: Caffe: convolutional architecture for fast feature embedding. arXiv Preprint (2014)

Food Recognition Using Fusion of Classifiers Based on CNNs

Eduardo Aguilar( ), Marc Bolaños, and Petia Radeva

Universitat de Barcelona & Computer Vision Center, Barcelona, Spain
{eduardo.aguilar,marc.bolanos,petia.ivanova}@ub.edu

Abstract. With the arrival of Convolutional Neural Networks, the complex problem of food recognition has experienced an important improvement recently. The best results have been obtained using methods based on very deep Convolutional Neural Networks, which show that the deeper the model, the better the classification accuracy is. However, very deep neural networks may suffer from the overfitting problem. In this paper, we propose a combination of multiple classifiers based on Convolutional models that complement each other and thus, achieve an improvement in performance. The evaluation of our approach is done on 2 public datasets: Food-101 as a dataset with a wide variety of fine-grained dishes, and Food-11 as a dataset of high-level food categories, where our approach outperforms the independent Convolutional Neural Networks models.

Keywords: Food recognition · Fusion classifiers · CNN

1 Introduction

In the field of computer vision, food recognition has caused a lot of interest for researchers considering its applicability in solutions that improve people's nutrition and hence, their lifestyle [1]. In relation to the healthy diet, traditional strategies for analyzing food consumption are based on self-reporting and manual quantification [2]. Hence, the information used to be inaccurate and incomplete [3]. Having an automatic monitoring system and being able to control the food consumption is of vital importance, especially for the treatment of individuals who have eating disorders, want to improve their diet or reduce their weight.

Food recognition is a key element within a food consumption monitoring system. Originally, it has been approached by using traditional approaches [4,5], which extracted ad-hoc image features by means of algorithms based mainly on color, texture and shape. More recently, other approaches focused on using Deep Learning techniques [5–8]. In these works, feature extraction algorithms are not hand-crafted and additionally, the models automatically learn the best way to discriminate the different classes to be classified. As for the results obtained, there is a great difference (more than 30%) between the best method based on hand-crafted features compared to newer methods based on Deep Learning, where the best results have been obtained with Convolutional Neural Networks (CNN) architectures that used inception modules [8] or residual networks [7].

S. Battiato et al. (Eds.): ICIAP 2017, Part II, LNCS 10485, pp. 213–224, 2017.
https://doi.org/10.1007/978-3-319-68548-9_20

Fig. 1. Example images of Food-101 dataset. Each image represents a dish class.

Food recognition can be considered as a special case of object recognition, being a very active topic in computer vision lately. The specific part is that dish classes have a much higher inter-class similarity and intra-class variation than usual Imagenet objects (cars, animals, rigid objects, etc.) (see Fig. 1). If we analyze the last accuracy increase in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [9], it has been improved thanks to the depth increase of CNN models [10–13] and also to the fusion of CNNs models [11,13]. The main problem of CNNs is the need of large datasets to avoid overfitting the network as well as the need of high computational power for training them.

Considering the use of different classifiers, in general, trained on the same data, one can observe that patterns misclassified by the different models would not necessarily overlap [14]. This suggests that they could potentially offer complementary information that can be used to improve the final performance [14]. An option to combine the outputs of different classifiers was proposed in [15], where the authors used what they call a decision templates scheme instead of simple aggregation operators such as the product or average. As they showed, this scheme maintains a good performance using different training set sizes and is also less sensitive to particular datasets compared to the other schemes.

In this article, we integrate the fusion concept into the CNN framework, with the purpose of demonstrating that the combination of the classifiers' output, by using a decision template scheme, allows to improve the performance on the food recognition problem. Our contributions are the following: (1) we propose the first food recognition algorithm that fuses the output of different CNN models, (2) we show that our CNNs fusion approach has better performance compared to the use of CNN models separately, and (3) we demonstrate that our CNNs Fusion approach keeps a high performance independently of the target (dishes, family of dishes) and dataset validating it on 2 public datasets.

The organization of the article is as follows. In Sect. 2, we present the CNNs Fusion methodology. In Sect. 3, we present the datasets, the experimental setup and discuss the results. Finally, in Sect. 4, we describe the conclusions.

2 Methodology

In this section, we describe the CNN Fusion methodology (see Fig. 2), which is composed of two main steps: training K CNN models based on different architectures and fusing the CNN outputs using the decision templates scheme.

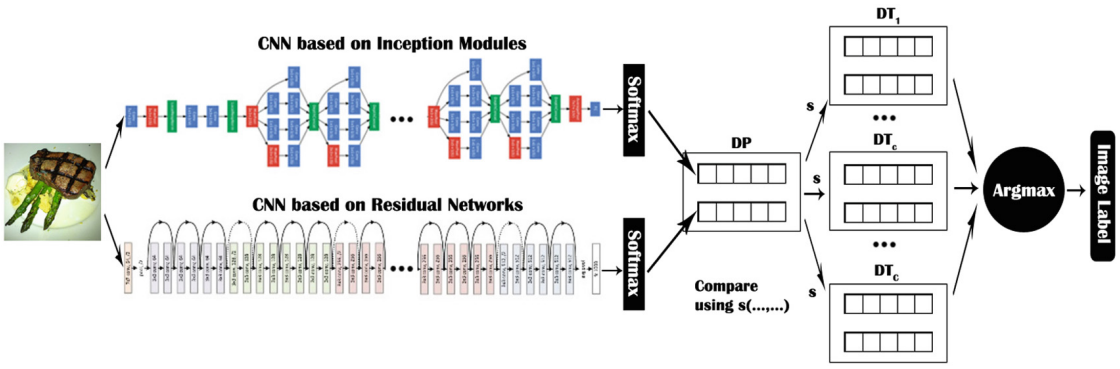

Fig. 2. General scheme of our CNNs fusion approach.

2.1 Training of CNN Models

The first step in our methodology involves separately training two CNN models. We chose two different kind of models winners of the ILSVRC in the object recognition task. Both models won or are based on the winner of the challenges made in 2014 and 2015 proposing novel architectures: the first based its design on "inception models" and the second on "residual networks". First, each model was pre-trained on the ILSVRC data. Later, all layers were fine-tuned by a certain number of epochs, selecting for each one the model that provides the best results in the validation set and that will be used in the fusion step.

2.2 Decision Templates for Classifiers Fusion

Once we trained the models on the food dataset, we combined the softmax classifier outputs of each model using the Decision Template (DT) scheme [15].

Let us annotate the output of the last layer of the k-th CNN model as $(\omega_{1,k}, \ldots, \omega_{C,k})$, where $c = 1, \ldots, C$ is the number of classes and $k = 1, \ldots K$ is the index of the CNN model (in our case, K= 2). Usually, the softmax function is applied, to obtain the probability value of model k to classify image x to a class c: $p_{k,c}(x) = \frac{e^{\omega_{k,c}}}{\sum_{c=1}^{C} e^{\omega_{k,c}}}$. Let us consider the k-th decision vector D_k:

$$D_k(x) = [p_{k,1}(x), p_{k,2}(x), \ldots, p_{k,C}(x)]$$

Definition [15]**:** A **Decision Profile,** DP for a given image x is defined as:

$$DP(x) = \begin{bmatrix} p_{1,1}(x) & p_{1,2}(x) & \ldots & p_{1,C}(x) \\ & \ldots & & \\ p_{K,1}(x) & p_{K,2}(x) & \ldots & p_{K,C}(x) \end{bmatrix} \tag{1}$$

Definition [15]**:** Given N training images, a **Decision Template** is defined as a set of matrices $DT = (DT^1, \ldots, DT^C)$, where the c-th element is obtained as the average of the decision profiles (1) on the training images of class c:

$$DT^c = \frac{\sum_{j=1}^{N} DP(x_j) \times Ind(x_j, c)}{\sum_{j=1}^{N} Ind(x_j, c)},$$

where $Ind(x_j, c)$ is an indicator function with value 1 if the training image x_j has a crisp label c, and 0, otherwise [16].

Finally, the resulting prediction for each image is determined considering the similarity $s(DP(x), DT^c(x))$ between the decision profile $DP(x)$ of the test image and the decision template of class c,$c = 1, \ldots, C$. Regarding the arguments of the similarity function $s(.,.)$ as fuzzy sets on some universal set with $K \times C$ elements, various fuzzy measures of similarity can be used. We chose different measures [15], namely 2 measures of similarity, 2 inclusion indices, a consistency measure and the Euclidean Distance. These measures are formally defined as:

$$S_1(DT^c, DP(x)) = \frac{\sum_{k=1}^{K}\sum_{i=1}^{C}\min(DT^c_{k,i}, DP_{k,i}(x))}{\sum_{k=1}^{K}\sum_{i=1}^{C}\max(DT^c_{k,i}, DP_{k,i}(x))},$$

$$S_2(DT^c, DP(x)) = 1 - \sup_u\{\left|DT^c_{k,i} - DP_{k,i}(x)\right| : c = 1, \ldots, C, k = 1, \ldots, K\},$$

$$I_1(DT^c, DP(x)) = \frac{\sum_{k=1}^{K}\sum_{i=1}^{C}\min(DT^c_{k,i}, DP_{k,i}(x))}{\sum_{k=1}^{K}\sum_{i=1}^{C}DT^c_{k,i}},$$

$$I_2(DT^c, DP(x)) = \inf_u\{\max(\overline{DT^c_{k,i}}, DP_{k,i}(x)) : c = 1, \ldots, C, k = 1, \ldots, K\},$$

$$C(DT^c, DP(x)) = \sup_u\{\min(DT^c_{k,i}, DP_{k,i}(x)) : c = 1, \ldots, C, k = 1, \ldots, K\},$$

$$N(DT^c, DP(x)) = 1 - \frac{\sum_{k=1}^{K}\sum_{i=1}^{C}(DT^c_{k,i} - DP_{k,i}(x))^2}{K \times C},$$

where $DT^c_{k,i}$ is the probability assigned to the class i by the classifier k in the DT^c, $\overline{DT^c_{k,i}}$ is the complement of $DT^c_{k,i}$ calculated as $1 - DT^c_{k,i}$, and $DP_{k,i}(x)$ is the probability assigned by the classifier k to the class i in the DP calculated for the image, x. The final label, L is obtained as the class that maximizes the similarity, s, the inclusion index, the consistency measure or the Euclidean distance between $DP(x)$ and DT^c: $L(x) = argmax_{c=1,\ldots,C}\{s(DT^c, DP(x))\}$.

3 Experiments

3.1 Datasets

The data used to evaluate our approach are two public datasets of very different images: Food-11 [17] and Food-101 [4], which are chosen in order to verify that the classifiers fusion provides good results regardless of the different properties of the target datasets, such as intra-class variability (the first one is composed of many dishes of the same general category, while the second one is composed of specific fine-grained dishes), inter-class similarity, number of images, number of classes, images acquisition condition, among others.

Food-11 is a dataset for food recognition [17], which contains 16,643 images grouped into 11 general categories of food: bread, dairy products, dessert, egg, fried food, meat, noodle/pasta, rice, seafood, soup and vegetable/fruit

(see Fig. 3). The images were collected from existing food datasets (Food-101, UECFOOD100, UECFOOD256) and social networks (Flickr, Instagram). This dataset has an unbalanced number of images for each class with an average of 1,513 images per class and a standard deviation of 702. For our experiments, we used the same data split, images and proportions, provided by the authors [17]. These are divided as 60% for training, 20% for validation and 20% for test, that is 9,866, 3,430 and 3,347 images for each set, respectively.

Fig. 3. Images from the Food-11 dataset. Each image corresponds to a different class.

Food-101 is a standard to evaluate the performance of visual food recognition [4]. This dataset contains 101.000 real-world food images downloaded from foodspotting.com, which were taken under unconstrained conditions. The authors chose the top 101 most popular classes of food (see Fig. 1) and collected 1,000 images for each class: 75% for training and 25% for testing. With respect to the classes, these consist of very diverse and fine-grained dishes of various countries, but also with highly intra-class variation and inter-class similarity in most occasions. In our experiments, we used the same data splits provided by the authors. Unlike Food-11, and keeping the procedure followed by other authors [5,7,8], we validate and test our model on the same data split.

3.2 Experimental Setup

As usually, every CNN model was pre-trained on the ILSVRC dataset. Following, we adapted them by changing the output of the models to the number of classes for each target dataset and fine-tuned the models using the new images. For the training of the CNN models, we used the Deep Learning framework Keras[1]. The models chosen for Food-101 dataset due to their performance-efficiency ratio were InceptionV3 [18] and ResNet50 [13]. Both models were trained during 48 epochs with a batch size of 32, and a learning rate of 5×10^{-3} and 1×10^{-3}, respectively. In addition, we applied a decay of 0.1 during the training of InceptionV3 and of 0.8 for ResNet50 every 8 epochs. The parameters were chosen empirically by analyzing the training loss.

As to the Food-11 dataset, we kept the ResNet50 model, but changed InceptionV3 by GoogLeNet [12], since InceptionV3 did not generalize well over Food-11. We believe that the reason is the small number of images for each class not sufficient to avoid over-fitting; the model quickly obtained a good result on the training set, but a poor performance on the validation set. GoogLeNet and Resnet50 were trained during 32 epochs with a batch size of 32 and 16,

[1] www.keras.io.

respectively. The other parameters used for the ResNet50 were the same used for Food-101. In the case of GoogLeNet, we used a learning rate of 1×10^{-3} and applied a decay of 0.1 during every 8 epochs, that turned out empirically the optimal parameters for our problem.

3.3 Data Preprocessing and Metrics

The preprocessing made during the training, validation and testing phases was the following. During the training of our CNN models, we applied different preprocessing techniques on the images with the aim of increasing the samples and to prevent the over-fitting of the networks. First, we resized the images keeping the original aspect ratio as well as satisfying the following criteria: the smallest side of the resulting images should be greater than or equal to the input size of the model; and the biggest side should be less than or equal to the maximal size defined in each model to make random crops on the image. In the case of InceptionV3, we set to 320 pixels as maximal size, for GoogLeNet and ResNet50 the maximal size was defined as 256 pixels. After resizing the images, inspired by [8], we enhanced them by means of a series of random distortions such as: adjusting color balance, contrast, brightness and sharpness. Finally, we made random crops of the images, with a dimension of 299×299 for InceptionV3 and of 224×224 for the other models. Then, we applied random horizontal flips with a probability of 50%, and subtracted the average image value of the ImageNet dataset. During validation, we applied a similar preprocessing, with the difference that we made a center crop instead of random crops and that we did not apply random horizontal flips. During test, we followed the same procedure than in validation (1-Crop evaluation). Furthermore, we also evaluated the CNN using 10-Crops, which are: upper left, upper right, lower left, lower right and center crop, both in their original setup and also applying an horizontal flip [10]. As for 10-Crops evaluation, the classifier gets a tentative label for each crop, and then majority voting is used over all predictions. In the cases where two labels are predicted the same number of times, the final label is assigned comparing their highest average prediction probability.

We used four metrics to evaluate the performance of our approach, overall Accuracy (ACC), Precision (P), Recall (R), and F_1 score.

3.4 Experimental Results on Food-11

The results obtained during the experimentation on Food-11 dataset are shown in Table 1 giving the error rate (1 - accuracy) for the best CNN models, compared to the CNNs Fusion. We report the overall accuracy by processing the test data using two procedures: (1) a center crop (1-Crop), and (2) using 10 different crops of the image (10-Crops). The experimental results show an error rate of less than 10 % for all classifiers, achieving a slightly better performance when using 10-Crops. The best accuracy is achieved with our CNNs Fusion approach, which is about 0.75% better than the best result of the classifiers evaluated separately. On the other hand, the baseline classification on Food-11 was given by their

authors, who obtained an overall accuracy of 83.5% using GoogLeNet models fine-tuned in the last six layers without any pre-processing and post-processing steps. Note that the best results obtained with our approach have been using the pointwise measures (S2, I2). The particularity of these measures is that they penalize big differences between corresponding values of DTs and DP being from the specific class to be assigned as the rest of the class values. From now on, in this section we only report the results based on the 10-Crops procedure.

Table 1. Overall test set error rate of Food-11 obtained for each model. The distance measure is shown between parenthesis in the CNNs Fusion models.

Authors	Model	1-Crop	10-Crops	N/A
[17]	GoogLeNet	-	-	16.5%
us	GoogLeNet	9.89%	9.29%	-
us	ResNet50	6.57%	6.39%	-
us	CNNs Fusion (S_1)	6.36%	5.86%	-
us	CNNs Fusion (S_2)	6.12%	**5.65**%	-
us	CNNs Fusion (I_1)	6.36%	5.89%	-
us	CNNs Fusion (I_2)	6.30%	**5.65**%	-
us	CNNs Fusion (C)	6.45%	6.07%	-
us	CNNs Fusion (N)	6.36%	5.92%	-

As shown in Table 2, the CNNs Fusion is able to properly classify not only the images that were correctly classified by both baselines, but in some occasions also when one or both fail. This suggests that in some cases both classifiers may be close to predicting the correct class and combining their outputs can make a better decision.

Table 2. Percentage of images well-classified and misclassified on Food-11 using our CNNs Fusion approach, distributed by the results obtained with GoogLeNet (CNN_1) and ResNet50 (CNN_2) models independently evaluated.

CNNs Fusion	CNNs evaluated independently			
	Both wrong	CNN_1 wrong	CNN_2 wrong	Both fine
Well-classified	3.08%	81.77%	54.76%	99.97%
Misclassified	96.92%	18.23%	45.24%	0.03%

Samples misclassified by our model are shown in Fig. 4, where most of them are produced by mixed items, high inter-class similarity and wrongly labeled images. We show the ground truth (top) and the predicted class (bottom) for each sample image.

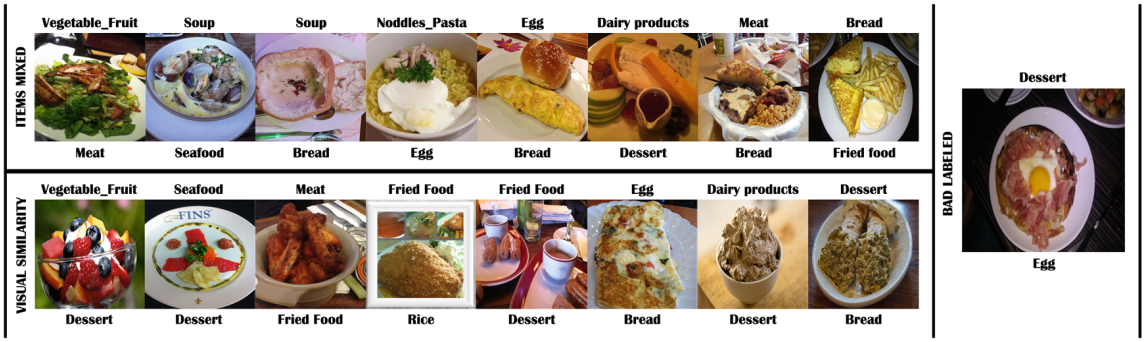

Fig. 4. Misclassified Food-11 examples: predicted labels (on the top), and the groundtruth (on the bottom).

In Table 3, we show the precision, recall and F_1 score obtained for each class separately. By comparing the F_1 score, the best performance is achieved for the class Noodles_Pasta and the worst for Dairy products. Specifically, the class Noddles_Pasta only has one image misclassified, which furthermore is a hard sample, because it contains two classes together (see items mixed in Fig. 4). Considering the precision, the worst results are obtained for the class Bread, which is understandable considering that bread can sometimes be present in other classes (e.g. soup or egg). In the case of recall, the worst results are obtained for Dairy products, where an error greater than 8% is produced for misclassifying several images as class Dessert. The cause of this is mainly, because the class Dessert has a lot of items in their images that could also belong to the class Dairy products (e.g. frozen yogurt or ice cream) or that are visually similar.

Table 3. Some results obtained on the Food-11 using our CNNs Fusion approach.

Class	#Images	Precision	Recall	F1
Bread	368	**88.95%**	91.85%	90.37%
Dairy products	148	89.86%	**83.78%**	**86.71%**
Meat	432	94.12%	92.59%	93.35%
Noodles_Pasta	147	**100.00%**	**99.32%**	**99.66%**
Rice	96	94.95%	97.92%	96.41%
Vegetable_Fruit	231	98.22%	95.67%	96.93%

3.5 Experimental Results on Food-101

The overall accuracy on Food-101 dataset is shown in Table 4 for two classifiers based on CNN models, and also for our CNNs Fusion. The overall accuracy is obtained by means of the evaluation of the prediction using 1-Crop and 10-Crops. The experimental results show better performance (about 1% more) using 10-Crops instead of 1-Crop. From now on, in this section we only report

the results based on the 10-Crops procedure. In the same way as observed in Food-11, the best accuracy obtained with our approach was by means of point-wise measures S2, I2, where the latter provides a slightly better performance. Again, the best accuracy is also achieved by the CNNs Fusion, which is about 1.5% higher than the best result of the classifiers evaluated separately. Note that the best performance on Food-101 (overall accuracy of 90.27%) was obtained using WISeR [7]. In addition, the authors show the performance by another deep learning-based approaches, in which three CNN models achieved over a 88% (InceptionV3, ResNet200 and WRN [19]). However, WISeR, WRN and ResNet200 models were not considered in our experiments since they need a multi-GPU server to replicate their results. In addition, those models have 2.5 times more parameters than the models chosen, which involve a high cost computational especially during the learning stage. Following the article steps, our best results replicating the methods were those using InceptionV3 and ResNet50 models used as a base to evaluate the performance of our CNNs Fusion approach.

Table 4. Overall test set accuracy of Food-101 obtained for each model.

Author	Model	1-Crop	10-Crops	N/A
[8]	InceptionV3	-	-	88.28%
[7]	ResNet200	-	88.38%	-
[7]	WRN	-	88.72%	-
[7]	WISeR	-	90.27%	-
us	ResNet50	82.31%	83.54%	-
us	InceptionV3	83.82%	84.98%	-
us	CNNs Fusion (S_1)	85.52%	86.51%	-
us	CNNs Fusion (S_2)	86.07%	86.70%	-
us	CNNs Fusion (I_1)	85.52%	86.51%	-
us	CNNs Fusion (I_2)	85.98%	**86.71**%	-
us	CNNs Fusion (C)	85.24%	86.09%	-
us	CNNs Fusion (N)	85.53%	86.50%	-

As shown in Table 5, in this dataset the CNNs Fusion is also able to properly classify not only the images that were correctly classified for both classifiers, but also when one or both fail. Therefore, we demonstrate that our proposed approach maintains its behavior independently of the target dataset.

Table 6 shows the top five worst and best classification results on Food-101 classes. We highlight the classes with the worst and best results. As for the worst class (Steak), the precision and recall achieved are 60.32% and 59.60%, respectively. Interestingly, about 26% error in the precision and 30% error in the recall is produced with only three classes: Filet mignon, Pork chop and Prime rib. As shown in Fig. 5, these are fine-grained classes with high inter-class similarities

Table 5. Percentage of images well-classified and misclassified on Food-101 using our CNNs Fusion approach, distributed by the results obtained with InceptionV3 (CNN_1) and ResNet50 (CNN_2) models independently evaluated.

CNNs Fusion	CNNs evaluated independently			
	Both wrong	CNN_1wrong	CNN_2wrong	Both fine
Well-classified	1.95%	73.07%	64.95%	99.97%
Misclassified	98.05%	26.93%	35.05%	0.03%

Table 6. Top 3 better and worst classification results on Food-101.

Class	Precision	Recall	F1
Spaghetti Bolognese	94.47%	95.60%	95.03%
Macarons	97.15%	95.60%	96.37%
Edamame	**99.60%**	**100.00%**	**99.80%**
Steak	**60.32%**	**59.60%**	**59.96%**
Pork Chop	75.71%	63.60%	69.13%
Foie Gras	72.96%	68.00%	70.39%

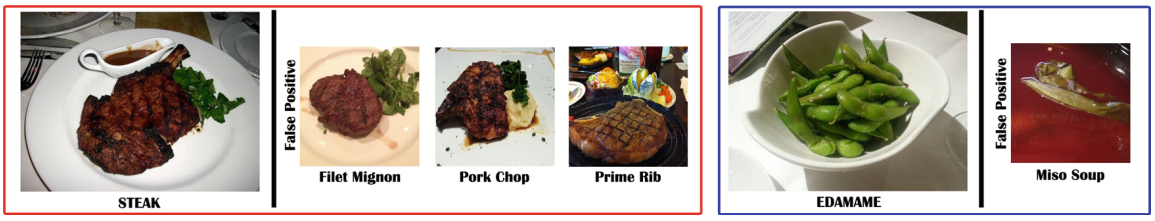

Fig. 5. Misclassified examples for the Food-101 classes that obtained the worst (steak) and best (edamame) classification results by F1 score (groundtruth label - bottom).

that imply high difficulty for the classifier, because it should identify small details that allow to determine the corresponding class of the images. On the other hand, the best class (Edamame) was classified achieving 99.60% of precision and 100% of recall. Unlike Steak, Edamame is a simple class to classify, because it has a low intra-class variation and low inter-class similarities. In other words, the images in this class have a similar visual appearance and they are quite different from the images of the other classes. Regarding the only one misclassified image, its visual appearance is close to the class Edamame as for the shape and color.

4 Conclusions

In this paper, we addressed the problem of food recognition and proposed a CNNs Fusion approach based on the concepts of decision templates and decision profiles and their similarity that improves the classification performance with respect to using CNN models separately. Evaluating different similarity measures, we show

that the optimal one is based on the infinimum of the maximum between the complementary of the decision templates and the decision profile of the test images. On Food-11, our approach outperforms the baseline accuracy by more than 10% of accuracy. As for Food-101, we used two CNN architectures providing the best state of the art results where our CNNs Fusion strategy outperformed them again. As a future work, we plan to evaluate the performance of the CNN Fusion strategy as a function of the number of CNN models.

Acknowledgement. This work was partially funded by TIN2015-66951-C2, SGR 1219, CERCA, *ICREA Academia'2014*, CONICYT Becas Chile, FPU15/01347 and Grant 20141510 (Marató TV3). The funders had no role in the study design, data collection, analysis, and preparation of the manuscript. We acknowledge Nvidia Corporation for the donation of 3 Titan X GPUs.

References

1. Waxman, A., Norum, K.R.: WHO global strategy on diet, physical activity and health. Food Nutr. Bull. **25**, 292–302 (2004)
2. Shim, J.-S., Oh, K., Kim, H.C.: Dietary assessment methods in epidemiologic studies. Epidemiol. Health **36**, e2014009 (2014)
3. Rumpler, W.V., Kramer, M., Rhodes, D.G., Moshfegh, A.J., Paul, D.R.: Identifying sources of reporting error using measured food intake. Eur. J. Clin. Nutr. **62**, 544–552 (2008)
4. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101-mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 446–461. Springer, Cham (2014). doi:10.1007/978-3-319-10599-4_29
5. Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., Ma, Y.: DeepFood: deep learning-based food image recognition for computer-aided dietary assessment. In: Chang, C.K., Chiari, L., Cao, Y., Jin, H., Mokhtari, M., Aloulou, H. (eds.) ICOST 2016. LNCS, vol. 9677, pp. 37–48. Springer, Cham (2016). doi:10.1007/978-3-319-39601-9_4
6. Yanai, K., Kawano, Y.: Food image recognition using deep convolutional network with pre-training and fine-tuning. In: ICMEW, pp. 1–6 (2015)
7. Martinel, N., Foresti, G.L., Micheloni, C.: Wide-slice residual networks for food recognition. arXiv Preprint (2016)
8. Hassannejad, H., Matrella, G., Ciampolini, P., De Munari, I., Mordonini, M., Cagnoni, S.: Food image recognition using very deep convolutional networks. In: Proceedings of the 2nd International Workshop on MADiMa, pp. 41–49 (2016)
9. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**, 211–252 (2015)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25, pp. 1–9 (2012)
11. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). doi:10.1007/978-3-319-10590-1_53

12. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR, pp. 1–9 (2015)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)
14. Kittler, J., Hatef, M.: On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **20**, 226–239 (1998)
15. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision templates for multiple classifier fusion: an experimental comparison. Pattern Recogn. **34**, 299–314 (2001)
16. Kuncheva, L.I., Kounchev, R.K., Zlatev, R.Z.: Aggregation of multiple classification decisions by fuzzy templates. In: EUFIT, pp. 1470–1474 (1995)
17. Singla, A., Yuan, L., Ebrahimi, T.: Food/non-food image classification and food categorization using pre-trained googlenet model. In: Proceedings of the 2nd International Workshop on MADiMa, pp. 3–11 (2016)
18. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR, pp. 2818–2826 (2016)
19. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv Preprint (2016)

Grab, Pay, and Eat: Semantic Food Detection for Smart Restaurants

Eduardo Aguilar, Beatriz Remeseiro, Marc Bolaños, and Petia Radeva, *Fellow, IAPR*

***Abstract*—The increase in awareness of people toward their nutritional habits has drawn considerable attention to the field of automatic food analysis. Focusing on self-service restaurants environment, automatic food analysis is not only useful for extracting nutritional information from foods selected by customers, it is also of high interest to speed up the service solving the bottleneck produced at the cashiers in times of high demand. In this paper, we address the problem of automatic food tray analysis in canteens and restaurants environment, which consists in predicting multiple foods placed on a tray image. We propose a new approach for food analysis based on convolutional neural networks, we name Semantic Food Detection, which integrates in the same framework food localization, recognition and segmentation. We demonstrate that our method improves the state-of-art food detection by a considerable margin on the public dataset UNIMIB2016, achieving about 90% in terms of F-measure, and thus provides a significant technological advance toward the automatic billing in restaurant environments.**

***Index Terms*—Food tray analysis, food recognition, semantic segmentation, convolutional neural networks.**

I. Introduction

HAVING a poor routine of physical exercises and poor nutritional habits are two of the main possible causes of people's health-related issues like obesity or diabetes, among others. For these reasons, nowadays people are more concerned about these aspects of their daily life. Therefore, the need for applications that allow to keep track of both physical activities and nutrition habits are rapidly increasing, a field in which the automatic analysis of food images plays an important role. Focusing on self-service restaurants, food recognition algorithms could enable both monitoring of food consumption and the automatic billing of the meal grabbed by the customer. The latter is quite relevant because remove the need for a manual selection of the chosen dishes, allowing to speed-up the service offered by these restaurants.

From the computer vision side, several approaches have been proposed to tackle the problem, most of them using Convolutional Neural Networks (CNNs) [1]–[4]. Several of the published work consider the development of methods for food recognition, i.e., being able to recognize the dish depicted in a picture in which a single plate is shown. An important consideration to take into account when modeling visual food-related information is its fine-grained nature, meaning that specially in the problem of food analysis the intra and inter-class similarity are hardly making difficult the problem of obtaining robust food recognition methods.

Several works in the literature have proposed methods for food intake self-monitoring [5], [6], in which the user should take pictures of each meal and the system would consequently track any nutritional information associated. Other approaches related to the problem of food intake include food portion estimation by using two images acquired by mobile devices [7]; food ingredients recognition from recipes using CNNs as multi-label predictors [8], [9]; multimodal multitask deep belief networks for learning both visual information and image-ingredient representation [10]; bayesian models for analyzing similarities between cuisines [11]; or cross-modal learning for multi-attribute recognition and recipe retrieval [12].

Instead of applying personalized tracking, there are several contexts where social monitoring or recognition is required. A clear example is food tray detection in public spaces [13], [14], where the sample consists of a tray picture that includes all the food that a user is about to consume (see Fig. 1) and the model is intended to process all pictures from any possible users taking food at the same restaurant. The development of a system able to apply food tray detection in a controlled, but social and public environment could enable several applications. The most straightforward context of applicability would be automatic billing in self-service restaurants, where the system could solve the need for a person selecting what the customer grabbed before paying. A different application could consider the design of smart trays [15], which could provide food

Manuscript received October 30, 2017; revised March 9, 2018; accepted April 14, 2018. Date of publication April 30, 2018; date of current version November 15, 2018. This work was supported in part by TIN2015-66951-C2-1-R, in part by SGR 1219, in part by CERCA Programme/Generalitat de Catalunya, and in part by the NVIDIA Corporation with the donation of the Titan Xp GPU. The work of E. Aguilar was supported by CONICYT Becas Chile. The work of B. Remeseiro was supported by the Spanish Ministry of Economy and Competitiveness under *Juan de la Cierva* Program (ref. FJCI-2014-21194). The work of M. Bolaños was supported by an FPU fellowship (ref. FPU15/01347). The work of P. Radeva was supported by ICREA Academia 2014. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jianfei Cai. *(Corresponding author: Beatriz Remeseiro.)*

E. Aguilar is with the Departamento de Ingeniería de Sistemas y Computación, Universidad Católica del Norte, Antofagasta 0610, Chile (e-mail: eaguilar02@ucn.cl).

B. Remeseiro is with the Departament de Matemàtiques i Informàtica, Universitat de Barcelona, Barcelona 08007, Spain, and also with the Department of Computer ScienceUniversidad de Oviedo, Gijón 33203, Spain (e-mail: bremeseiro@uniovi.es).

M. Bolaños and P. Radeva are with the Departament de Matemàtiques i Informàtica, Universitat de Barcelona, Barcelona 08007, and Computer Vision Center, Cerdanyola 08007 (Barcelona), Spain (e-mail: marc.bolanos@ub.edu; petia.ivanova@ub.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2018.2831627

Fig. 1. Example of images used in traditional approaches to food analysis (left) and food tray analysis (right).

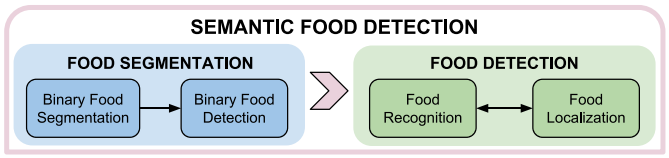

Fig. 2. Main tasks of our Semantic Food Detection framework.

recommendations depending on what the customer is selecting. The provided recommendations could be based on calorie counting, healthy food, specific nutritional composition, etc. In addition, if we also consider a system able to log the food consumed by every individual along time, it could provide health-related recommendations in a long-term way.

There are several aspects that make the food tray analysis a challenging problem [14]: 1) multiple foods placed on the same placemat, 2) different foods served in the same dish, 3) visual distortions and illumination changes due to shadows, and 4) objects placed on a tray that do not correspond to any type of food. On the other hand, unlike traditional approaches to food analysis, difficulties due to intra-class variability have less influence on the problem of food tray detection.

In this work, we propose a novel method that unifies the problems of food detection, localization, recognition and segmentation into a new framework that we call Semantic Food Detection. As Fig. 2 shows, we integrate the information extracted by two main approaches: a) food segmentation and b) object detection trained for food detection, by taking advantage of the benefits provided by both algorithms in a CNN framework. The first one allows us to determine where the food is in terms of pixel and bounding boxes. The second one allows us to locate and recognize the foods present in the images. The Semantic Food Detection framework combines the information that both algorithms provide in order to prevent false food detections and thus provide a better performance.

Our main contributions are: 1) a novel framework that integrates the problems of food detection, localization, recognition and segmentation; and 2) a novel approach to address the problem of food tray analysis, that integrates a fully convolutional network for semantic segmentation and a convolutional neural network for object detection through a probabilistic approach and a custom non-maximum suppression. Our method achieves about 90% in terms of F2-score, and it is able to outperform the state-of-art methods by more than 10% and 20% with respect to recall and mean average accuracy.

The remainder of this paper is organized as follows: Section II includes an overview of the related work, Section III presents the proposed Semantic Food Detection approach, Section IV shows the experimental results and discussion, and Section V closes with the conclusions and future research.

II. Related Work

Nowadays, there is a great interest in conducting research for visual food analysis, mainly in its applicability for diet monitoring based on the intrinsic nutritional information contained in food images. In this field, researchers have focused on different several aspects related to automatic food analysis.

The most basic aspect tackled in the literature is the *binary food detection* problem that determines the presence or absence of food in an image. This problem is also called food/non-food classification or food detection [16]. The first approximation was proposed by Kitamura *et al.* [17], who combine a BoF model and a SVM achieving a high accuracy on a tiny dataset of 600 images. An improvement of about 4% is achieved in terms of overall accuracy using a CNN-based method [16]. From this, numerous researchers have proposed CNN-based models either for feature extraction [2], [3] or for the whole recognition process [1], [4]. The best results obtained on public datasets with more than 15,000 images [1], [2] have been reported in [3] through the combination of CNN GoogLeNet for feature extraction, PCA for dimension reduction and SVM for classification. As for its applicability, this problem has commonly been investigated for indexing WEB images [17] or as a pre-processing method for an automatic food recognition system [1], [4]. It has been also used to detect bounding boxes in an food images [18], and to automate the process of image cleaning required when gathering images of a food dataset [19].

In food analysis, once images containing food are identified, *food recognition* is usually the next step to apply. Again, CNN-based models have been able to progressively improve the results of food recognition models reaching an accuracy of about 90% in datasets with around 100 different food classes [20]. In general, the best proposals are based on the winning models of the ILSVRC challenge [21], and a fine-tuning process is usually applied either making some architectural model changes (e.g., addition or removal of layers) [22], [23] or not [24]. Several datasets have been proposed to tackle this problem: a) datasets including fine-grained classes (e.g., apple pie, pork chop, pizza), like UECFOOD-256 [25] or Food-101 [26]; and b) datasets based on high-level categories (e.g., dessert, meat, soup), like Food-11 [4]. The best result when using fine-grained classes was achieved by the WISeR model [20], which combines the food traits and the vertical structure of some food, extracted by the standard squared convolutional kernel and the proposed slice convolutional kernel, respectively. Regarding the high-level categories, the best results were obtained by [27] through a novel approach that fuses several CNN models, achieving a 10% improvement in terms of accuracy with respect to the baseline method.

Most of the approaches focused on food recognition only exploit the visual content, but they ignore the context. However, geolocation and other information have also been explored in the literature for restaurant-oriented food recognition: on-line

restaurant information is used in [28], similarly to [29] in which nutritional information is also retrieved; whilst the menu, the location and user images of dished are used in [30]. On the other hand, Herranz *et al.* [31] go a step further since their target is not only to improve both classification performance and efficiency, but also to better model contextual data and its relation with the other elements.

To date, most food recognition algorithms and datasets focus on classifying images that include only one dish [20], [23], [24]. However, in some cases, there may be more than one dish in the image and, in some cases, the dish can contain several kinds of food. *Food localization* and *food segmentation* are two tasks intended to cope with these problems. The former consists in extracting the regions of the images where the food is located. Up to our knowledge, the only available approach that does not require segmenting the food before extracting the bounding boxes is the one proposed by [18]. The task of food segmentation consists in classifying each pixel of the images representing a food. The latest research for food segmentation proposes an automatic weakly supervised methods [32], [33], which are based on Deep Convolutional Neural Networks and Distinct Class-specific Saliency Maps, respectively.

Regarding image segmentation for general purposes, fully convolutional networks (FCNs) [34] are the state-of-art in semantic segmentation. They are composed of convolutional layers only, i.e. they do not have any fully-connected layer. They consist of a down-sampling path and an up-sampling path, which allow to take input images of arbitrary size and produce outputs of equivalent size, by means of an efficient inference and learning process. Several FCN models can be found in the literature applied to semantic segmentation. SegNet [35] is a deep FCN that consists of a VGG16-based encoder, a decoder and a final pixel-wise classification layer. DeepLab [36] uses *atrous convolutions* in the up-sampling path, allowing to incorporate larger context with no increase in parameters. RefineNet [37] is a multi-path refinement network that allows to obtain high-resolution predictions by using residual connections. PSPNet [38] is a pixel-level prediction framework that includes a pyramid pooling module to exploit the capability of global context information. Tiramisu [39] is an extension of Densely Connected Convolutional Neural Networks (DenseNets) for semantic segmentation, based on the idea of connecting each layer to every other layer in a feed-forward fashion. Its main benefits include a more accurate and easier training, with much less parameters.

According to the experimentation presented in the respective manuscripts, PSPNet [38] and Tiramisu [39] are the most competitive models. PSPNet is based on Residual Networks (ResNets), whilst Tiramisu is based on DenseNets. DenseNets can be seen as an extension of ResNets, with some characteristics that make them very appropriate for semantic segmentation problems: parameter efficiency, implicit deep supervision, and feature reuse. For all these reasons, Tiramisu will be the model of reference in our research.

In this manuscript, we deal with the identification of different foods placed on a food tray, by integrating the four food analysis problems mentioned above. To the best of our knowledge, only one approach with this purpose has been evidenced in the literature [14]. The authors introduced an additional food dataset composed of images taken in a canteen environment named UNIMIB2016. In addition, they proposed a pipeline for food recognition that performs classification based on the candidate regions obtained by combining two separate images segmentation processes, through saturation and color texture (JSEG). The best result was achieved by combining global and local (patch-based) classification approaches. Regarding the classification, for each region, they are carried out both in a sub-image (global strategy) and in several image patches (local strategy), the feature extraction using a CNN model based in AlexNet, and then the classification by an SVM. Furthermore, in the local strategy, an additional post-processing phase is needed to merge the labels of all image patches of the respective region. Then, the classification obtained by both approaches is combined by exploiting the sum of posterior probabilities to judge the final classification decision. Our approach differs mainly in three aspects: 1) we perform semantic segmentation by learning the best discriminant features between different foods from the dataset instead of using a segmentation approach based on generic image processing methods; 2) we locate and classify simultaneously all the foods placed in the tray by considering the context instead of performing the classification for each region individually, which implies a significant improvement in both result and processing time; and 3) we integrate the outputs of both methods to avoid false detections and thus make better decisions, instead of performing the classification directly based on the segmentation results. Additionally, our method is able to perform the food segmentation and detection processes in parallel, allowing to speed up the processing time.

III. Semantic Food Detection

This work proposes a method for food tray semantic detection that integrates food vs non-food semantic segmentation with food localization and recognition. Fig. 3 depicts the pipeline of our approach, subsequently explained in detail.

A. Food Segmentation

Food segmentation deals here with the problem of separating the food and food-related items, from the tray and other background elements, thus obtaining a binary image. For this purpose, we apply semantic segmentation techniques that work in a supervised learning framework, unlike the most segmentation methods that focus on image properties (e.g., color or texture). Notice that semantic segmentation could be used to directly segment the input image into the different food categories. However, the most recent methods in this field provide great results with datasets that contain a relatively low number of classes, such as CamVid with 11 semantic classes or Gatech with 8 [39]. The number of categories used in food analysis is much higher, thus increasing the difficulty of the task and providing not so satisfactory results [34].

Among the FCN models found in the literature applied to semantic segmentation, the Tiramisu model was considered [39], as mentioned in Section II. Its down-sampling and up-sampling

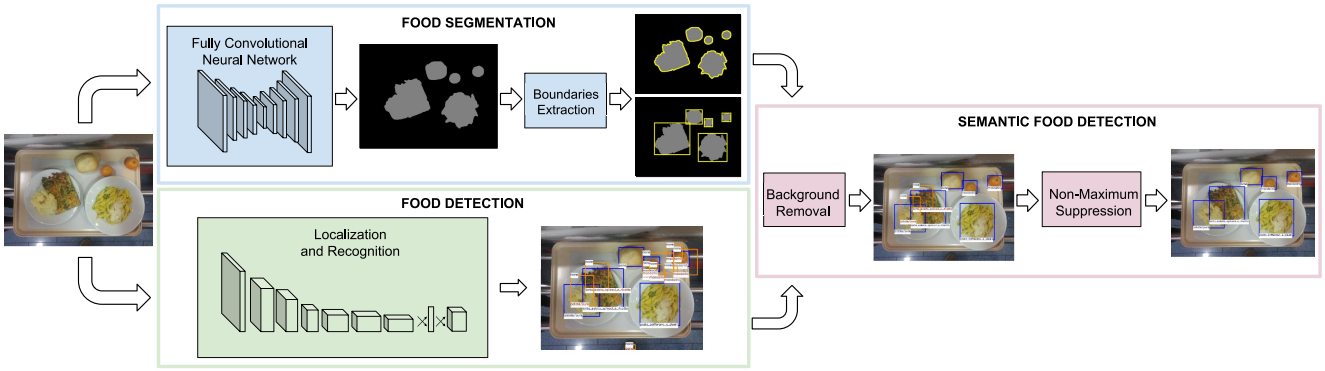

Fig. 3. Detailed workflow of the proposed Semantic Food Detection method: food segmentation and food detection methods are applied in parallel, before combining them for a final detection on food tray images.

Fig. 4. Food segmentation output (left to right): binary prediction by FCN, regions boundaries and food bounding boxes.

paths are connected by skip connections, and its architecture is composed of dense blocks, each one of them containing a set of concatenated layers for a better training.

After training our FCN model with food tray images, the binary images predicted by it are used in the next step, which aims at tracing the exterior boundaries of the food regions, avoiding the holes inside them. In this manner, small holes that may appear inside regions are discarded and thus the regions are homogenized. For this task, we use the Moore-Neighbor tracing algorithm modified by Jacob's stopping criteria [40].

Once the boundaries are traced, the bounding boxes that contain the regions are determined, thus obtaining a binary food detection. As small regions may also appear in the predicted images, and they usually correspond to false positives, this step also includes their elimination by considering a threshold criterion. Fig. 4 illustrates an example of the outputs obtained in the food segmentation procedure, including the binary image provided by the FCN model, the boundaries extracted and the bounding boxes generated.

B. Food Detection

In this work, following the definition of the object detection problem [21], we consider as Food Detection the localization and recognition of food. For this purpose, we propose retraining an object detection algorithm to apply food detection instead. In particular, we chose one of the best object detection approaches in the state-of-art, YOLOv2 [41], [42]. As for the model, the authors propose a new FCN called Darknet-19, composed by 19 convolutional layers and 5 max pooling layers to tackle the recognition task. They modified this network for object detection by removing the last convolutional layer and adding four convolutional layers for producing 13×13 feature maps. At each cell on the output feature maps, the network predicts B bounding boxes with five coordinates for each, among them is the confidence score t_o, and $c = 1, \ldots, C$ conditional class probabilities, $Pr(Class_c|Object)$. Predictions are obtained from the last convolutional layer having a size equal to 1×1 and F filters, where the number of filters is calculated as: $F = (B \times (5 + C))$. From this, it is possible to determine the class-specific confidence score, CS_c for each bounding box as follows:

$$CS_c = Pr(Class_c|Object) * \sigma(t_o) \tag{1}$$

where $\sigma(.)$ stands for a logistic activation to constrain the predictions to fall in the range between 0 and 1. Note that, in the experiment, we use the original setting of B (equal to 5).

C. Semantic Food Detection

In object detection, one of the most common errors are false positives, which can be classified based on the type of error: localization error, confusion with similar objects, confusion with dissimilar objects, and confusion with background [43]. Our Semantic Food Detection proposal focuses on reducing two of the most common errors of object detectors [41]: localization errors, specifically those corresponding to duplicate detections; and errors produced by the confusion with the background. For this purpose, we propose the following procedure that integrates the detection and segmentation algorithms:

1) Background Removal: The first step involves the application of both boundaries extracted (contour and bounding box) from the Food Segmentation procedure in order to remove the background detections. Let $Y = \{b_1^Y, \ldots, b_N^Y\}$ be the set of bounding boxes obtained with the detection method, $S_1 = \{b_1^S, \ldots, b_L^S\}$ and $S_2 = \{c_1^S, \ldots, c_L^S\}$ the set of bounding boxes and contours extracted by the Food Segmentation method, respectively. Considering each element belonging to the sets named above as a set of points (x, y) that defines a polygon, we calculate the probability of a bounding box, b_i^Y to belong to the background Bkg as follows:

$$Pr(Bkg|b_i^Y) = \min(CS_c(\overline{b_i^Y}), \max(Pr(\overline{S_1|b_i^Y}), Pr(\overline{S_2|b_i^Y})))$$

where $CS_c(\overline{b_i^Y})$ is the complement of the confidence score, $1 - CS_c(b_i^Y)$ for the i-th detection, $Pr(\overline{S_1|b_i^Y})$ is the probability

that b_i^Y is a false detection on the extracted boxes, S_1:

$$Pr(\overline{S_1|b_i^Y}) = 1 - \max_{j=1,\ldots,L} \frac{|b_i^Y \cap b_j^S|}{|b_i^Y|}$$

where $|.|$ stands for the cardinality of a set of pixels corresponding to an image region, and $Pr(\overline{S_2|b_i^Y})$ the probability that b_i^Y does not intersect with any contour in S_2:

$$Pr(\overline{S_2|b_i^Y}) = \min_{j=1,\ldots,L} Ind(b_i^Y \cap c_j^S = \emptyset)$$

where $Ind(*)$ is an indicator function with value 1 if the condition is true, and 0 otherwise.

Bounding boxes with a probability higher than 50% to be background ($Pr(Bkg|b_i^Y) > T, T = 0.5$) are considered to be false detections, and are therefore removed.

2) Non-Maximum Suppression: The second step involves the application of a greedy procedure to eliminate duplicate detections by non-maximum suppression [44]. Once the Background Removal is applied, the remaining detections $Y' \subseteq Y$ are sorted in descending order by the confidence score $CS_c(b_j^Y)$ and grouped into C sets $Y^1, \ldots, Y^C \subset Y'$, where C is the number of classes. Then, for each $Y^c, c = 1, ...C$, we greedily select the highest scoring bounding boxes while removing detections that are lower in the ranking and their maximum intersection ratio (MIR) with respect to the i-th previously selected bounding boxes is more than 50%, where MIR score for the j-th bounding box is calculated as:

$$MIR_j = \max_{\forall i, i<j} \frac{|b_i^Y \cap b_j^Y|}{\min(|b_i^Y|, |b_j^Y|)}$$

Notice that the chosen food detection method already incorporates a non-maximum suppression procedure. In our framework, we propose an additional personalized non-maximum suppression that differs mainly in two aspects: 1) we consider the predicted classes for the bounding boxes, and 2) we propose a MIR score instead of the traditional IoU. The last one was applied because in some cases the overlapped predictions for the same class could have a completely different dimension and proportion, and then, the IoU score will be very small even if one bounding box is completely inside the other.

IV. Experimental Results

In this section, we first describe the dataset used to evaluate the proposed approach, which is composed of images taken in self-service restaurants. Then, we describe the evaluation measures used and present the results obtained with the different methods and model configurations.

A. Dataset

UNIMIB2016 [14] is a food dataset that has been collected in a self-service canteen. Each image includes a tray with some food placed both on plates and placemats. The acquisition process was performed on a semi-controlled environment using a Samsung Galaxy S3 smartphone. As a result, images acquired have a resolution of 3264 × 2448 in RGB, and present visual distortions and variable illuminations, making them challenging for any task of automatic food analysis.

Fig. 5. A representative sample of the UNIMIB dataset [14]: original image (left) and food annotations (right).

The dataset is composed of 1,027 images that include a total of 73 food categories. Among them, only 1,010 images and 65 categories were used for experimentation, as suggested in [14] due to the low number of samples of the categories not considered. For experimental purposes, the dataset has been split in training and test sets: the former contains 650 images (≈64%), whilst the latter contains 360 (≈36%).

The annotations included in the dataset contain, for each food item: the polygon defining its boundaries, the bounding box and the food label. Fig. 5 illustrates an image of the UNIMIB2016 dataset with its corresponding annotations.

B. Food Segmentation

Metrics. In order to evaluate the different food segmentation approaches, several performance measures have been used. First, two pixel-wise metrics commonly used in semantic segmentation problems have been considered [34]:

- *Global pixel accuracy (GA).* The pixel-wise accuracy computed over all the pixels of the dataset.
- *Intersection over Union (IoU).* Also known as Jaccard index, it is defined as:

$$IoU(c) = \frac{\sum_i t_i == c \wedge p_i == c}{\sum_i t_i == c \vee p_i == c} \quad (2)$$

  where c is a class, i represents all the pixels of the dataset, t_i are the target labels, and p_i are the predicted labels. Note that this metric is calculated for each single class c, and then the mean across the classes is computed.

To perform a fair comparison with [14], three region-based metrics have been also considered [45]:

- *Covering (CO).* The covering of the ground truth (GT) by the segmented (S) images measures the level of overlapping between each pair of regions (R and R'):

$$C(S \rightarrow GT) = \frac{1}{N} \sum_{R \in GT} |R| \cdot \max_{R' \in S} \frac{|R \cap R'|}{|R \cup R'|} \quad (3)$$

  where N is the number of pixels of the image.
- *Rank index (RI).* It compares the compatibility of assignments between pairs of elements in the ground truth (GT)

TABLE I
RESULTS OBTAINED WITH OUR FOOD SEGMENTATION APPROACH IN TEST SET

	No.	Pixel-wise		Region-based		
	param	GA	IoU	CO	RI	VI
JSEG [47]	-	-	-	0.385	0.389	3.106
Ciocca et al. [14]	-	-	-	0.916	0.931	0.429
Classic Upsam.	12.7M	0.991	0.962	0.984	0.982	0.125
Tiramisu56	1.4M	0.992	0.967	0.986	0.984	0.112
Tiramisu67	3.5M	**0.993**	**0.971**	**0.987**	**0.986**	**0.105**
Tiramisu103	9.4M	0.992	0.968	0.986	0.984	0.111

and the segmented (S) images:

$$RI(S,GT) = \frac{1}{\binom{N}{2}} \sum_{i<j} [\mathbb{I}(t_i == t_j \wedge p_i == p_j) + \mathbb{I}(t_i \neq t_j \wedge p_i \neq p_j)] \quad (4)$$

where $\binom{N}{2}$ is the number of possible unique pairs among the N pixels of each image, and $\mathbb{I}$ is the identity function.

- *Variation of information (VI)*. It measures the distance between the ground truth (GT) and the segmented (S) images in terms of their average conditional entropy:

$$VI(S,GT) = H(S) + H(GT) - 2 \cdot MI(S,GT) \quad (5)$$

where H and MI are, respectively, the entropy and the mutual information. In this case, the lower the better.

Notice that these three metrics are calculated for each single image, and then the mean across images is computed.

Experimental setup. Regarding the methods used for semantic segmentation, we trained three networks based on Tiramisu [39]: 1) *Tiramisu56*: 56 layers, with 4 layers per dense block and a growth rate of 12; 2) *Tiramisu67*: 67 layers, with 5 layers per dense block and a growth rate of 16; and 3) *Tiramisu103*: 103 layers, with a variable number of layers per dense block (from 12 to 4 in the downsampling path, and from 4 to 12 in the upsampling) and a growth rate of 16. Additionally, the *Classic Upsampling*, which uses standard convolutions in the upsampling path instead of dense blocks [46], has been also considered for comparative purposes.

All the FCN models were trained with the UNIMIB2016 dataset [14] (images resized to 360×480), and two-target labels: food vs non-food. The models were initialized with He-Uniform and trained with RMSprop [39]. The training process consists of two steps: first, the models were trained with cropped images (224×224) for data augmentation and batch size 3, with an initial learning rate of $1e-3$ and an exponential decay of 0.995 per epoch; and second, their parameters were fine-tuned with full size images (360×480) and batch size 1, using a learning rate of $1e-4$. The outputs were monitored using the global accuracy and the IoU, with a patience of 100 during pre-training and 50 during fine-tuning.

Table I includes the results achieved with the four networks for semantic segmentation, as well as with the two segmentation methods from [14]: the JSEG algorithm [47], and the segmentation pipeline proposed in [14]. With respect to the pixel-wise measures, all the networks produced competitive results (over 0.96). The Tiramisu models outperformed the Classic Upsampling, thanks to the dense blocks, despite a lower number of parameters used. In general, the Tiramisu model benefits from having more parameters and depth. However, in this binary problem the Tiramisu103 produced overfitting whilst the Tiramisu67 achieved the best results, with a good trade-off between depth and performance. Regarding the region-based measures, all the FCNs provided better results than the two approaches from [14], which demonstrated the adequacy of the proposed methods for our problem.

C. Semantic Food Detection Performance

Metrics. In order to evaluate food recognition and localization, we chose three standard measures commonly used in multi-class object recognition problems:

- *Recall (Rec)*. The proportion of true positives detected.
- *Precision (Pre)*. The proportion of the true positives against all the positive results.
- *F_β-measure*. A weighted average of precision and recall. We use $\beta = 2$ (F_2) to place more emphasis on wrong classified or undetected foods.

For comparative purposes, the measures used by Ciocca *et al.* [14] were also considered:

- *Standard Accuracy (SA)*. It is equivalent to the recall.
- *Macro Average Accuracy (MAA)*. The proportion of correctly classified foods, but taking into account the class imbalance of the dataset:

$$MAA = \frac{1}{C} \sum_{c=1}^{C} \frac{TP_c}{NF_c} \quad (6)$$

where C is the number of classes, TP_c is the number of correctly classified foods of class c, and NF_c is the total number of foods of class c.

- *Tray Accuracy (TA)*. The percentage of trays for which all the foods contained are correctly recognized:

$$TA = \frac{1}{T} \sum_{t=1}^{T} Ind\left(\frac{TP_t}{NF_t} = 1\right) \quad (7)$$

where T is the number of food tray images, TP_t is the number of correctly classified foods on the tray t, and NF_t is the total number of foods on the tray t.

Experimental setup. YOLOv2 was first pre-trained on the ILSVRC dataset. Following, we adapted it by changing the output of the model to 65 classes and applied a fine-tuning using UNIMIB2016 images (resized to 416×416). For training the model, we used the framework Darknet [48]. The models were trained during 4000 iterations with a batch size of 32, and a learning rate of $1e-3$. In addition, we applied a decay of 0.9 to the iterations 3000 and 3500. To avoid overfitting, we use standard data augmentation procedures with random crops and distortions in the HSV color space [42].

Once YOLOv2 training is completed, the next step is to determine the confidence threshold to be used during localization and recognition of the food. A low confidence threshold implies

TABLE II
RESULTS OBTAINED BY YOLOV2 AND THE PROPOSED APPROACH IN TRAINING SET USING DIFFERENT CONFIDENCE THRESHOLDS

		1/65	1/32	1/16	1/8	1/4	1/2
YOLOv2 [42]	Pre	**0.511**	0.687	0.832	0.926	0.968	0.994
	Rec	0.999	0.998	0.997	0.995	0.988	0.966
	MAA	0.999	0.997	0.995	0.992	0.981	0.952
	TA	0.997	0.992	0.988	0.982	0.960	0.895
Proposed	Pre	**0.918**	0.952	0.973	0.984	0.991	0.996
	Rec	0.998	0.997	0.996	0.994	0.987	0.965
	MAA	0.999	0.999	0.996	0.994	0.981	0.951
	TA	0.995	0.992	0.988	0.982	0.957	0.894

a greater number of detections, which maximizes the likelihood that all the foods present in the image will be detected. At the same time, it also increases the chances of obtaining false detections. Taking into account that the confidence defined by the detection method considers two factors (the fit of the bounding box to the object and the predicted class), we chose the minimum threshold according to the number of classes. Given that the target dataset has 65 classes, the minimum threshold chosen is $\frac{1}{65}$. With this value, it can be interpreted that the bounding boxes extracted will have a recognition probability greater than a random value when the detected bounding box fits the object perfectly. Following the interpretation given, we chose $\frac{1}{2}$ as maximum threshold, which implies a high probability, at least 50%, that the localized object is correctly classified.

Table II shows the results obtained in the training set using different confidence thresholds. The tested thresholds range from the minimum and maximum values mentioned above. As can be observed, when the threshold increases, the precision also increases considerably, whilst the rest of the indicators are hardly affected. When comparing the results obtained between YOLOv2 and the proposed method, for the minimum threshold, it can be observed that a significant improvement in precision is obtained (≈40%) with only a slight decrease in the other indicators (0.1%-0.2%). Another interesting aspect to highlight is the comparison of results when using the maximum threshold, since they are practically identical for both methods. This means that, for a threshold of $\frac{1}{2}$, there are almost no false detections that can be reduced with our procedure. For the remaining experiments, the minimum threshold was chosen for two main reasons: 1) it obtains the best results for the $Recall$, MAA and TA indicators; and 2) it allows us to discard the false positives that appear when combining the results with the food segmentation procedure.

The Semantic Food Detection results on the test set are shown in Table III. In order to see the performance of the different parts of our pipeline, we group the results of this table in three rows: the first one corresponds to the results obtained with YOLOv2 retrained for food detection, and our proposed framework without considering the information extracted from the segmentation method to perform the classification (YOLOv2 + III-C2); the second one corresponds to the results of the baseline method [14], our framework without considering the personalized non-maximum suppression procedure (YOLOv2 + III-C1), and our proposed framework; and the third row is similar to the second one, but replacing the segmentation method by the ground truth segmentation. As for the results achieved, it should be highlighted that our proposal outperforms the food recognition, with respect to the state-of-art method (Ciocca *et al.* [14]) in a 12.4% for *Recall* and 20.9% for *MAA*. Regarding *TA*, a decrease of 2.5% is observed. However, we consider that this measure does not reflect how well the recognition works mainly due to the imbalance in the quantity of food in the trays, which varies between 1 and 9 [see Fig. 6(a)], as well as because *TA* measures the amount of food trays in which all positive samples have been correctly predicted, but does not penalize when there are false positives.

In order to apply a complete comparison, we also replicated the evaluation proposed by [14], in which the authors considered a perfect segmentation using the ground truth (GT) and applied their detection method (bottom section of Table III). In our case, there is no significant improvement with respect to the use of the proposed semantic segmentation, because our proposal considers the integration of the extracted information with the segmentation to refine the predictions already obtained by the object detection method. In contrast, Ciocca *et al.* [14]

TABLE III
TRAY FOOD ANALYSIS RESULTS, FROM TOP TO BOTTOM: FOOD DETECTION METHOD 1) WITHOUT SEGMENTATION, 2) WITH SEGMENTATION, AND 3) WITH GROUND-TRUTH SEGMENTATION TO PERFORM THE RECOGNITION

	F_2	Pre	Rec	MAA	TA
YOLOv2 [42]	0.786	0.489	**0.927**	**0.850**	**0.769**
YOLOv2 + III-C2	**0.856**	**0.659**	**0.925**	**0.849**	0.772
Ciocca et al. [14]	-	-	0.798	0.636	**0.789**
YOLOv2 + III-C1	0.844	0.628	**0.923**	**0.846**	0.761
Proposed	**0.905**	**0.841**	**0.922**	**0.845**	0.764
Mezgec et al. [19]	-	-	0.864	-	-
Ciocca et al. [14]	-	-	0.891	0.684	**0.871**
YOLOv2 + III-C1	0.854	0.651	**0.926**	**0.850**	0.769
Proposed	**0.911**	**0.856**	**0.926**	**0.850**	0.775

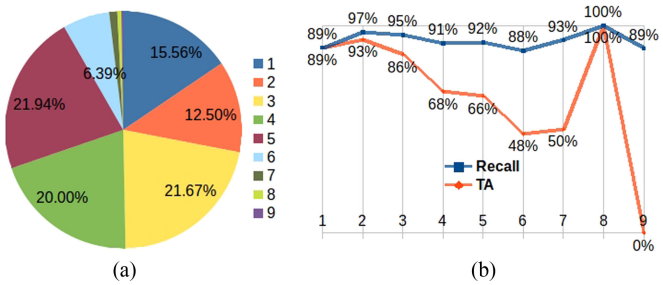

Fig. 6. a) Distribution of the trays according to the number of foods that are placed in them. b) Results in terms of *Recall* (blue) and *TA* (orange), for each item of the distribution.

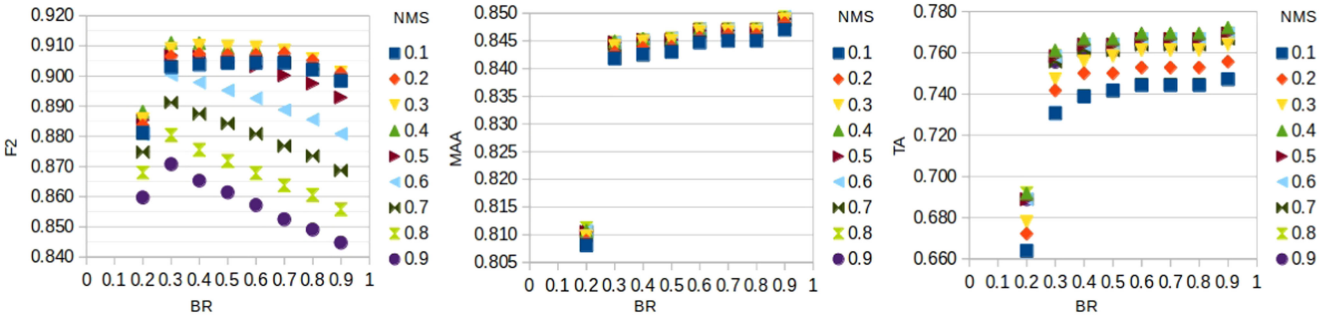

Fig. 7. Results of our proposal when varying the background removal (BR) and non-maximum-suppression (NMS) thresholds.

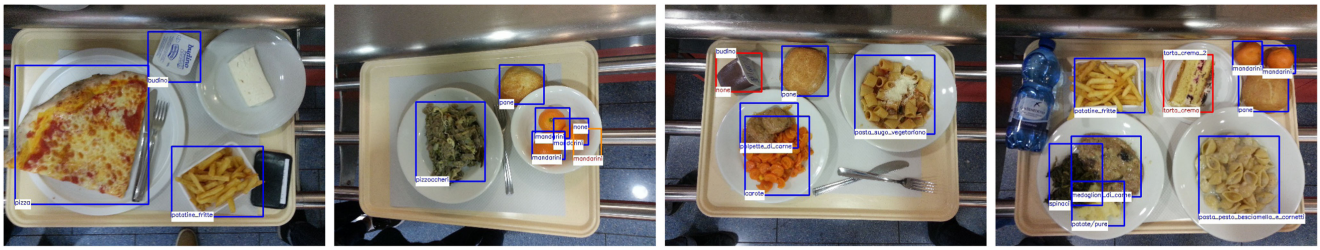

Fig. 8. Some samples of the results obtained using the proposed approach, from left to right: food tray with all the objects correctly detected (blue), false detection sample (orange), and two samples with one misclassified object (red).

performed the recognition directly on the segmented objects. Comparing to the results obtained in [14], we can see that their method improves significantly in terms of *Recall* using the GT for segmentation, achieving to match our results. However in terms of *MAA*, despite improving its performance, our results are still about 16% better. A low *MAA* with a high *Recall* implies that the classifier has a strong bias towards the classes that have a greater amount of instances. Therefore, even if we consider a perfect segmentation to contrast the results, our proposal keeps a better performance for recognition and a lower bias towards the dominant classes.

The results obtained with the proposed approach based on the number of objects to be classified per food tray is shown in Fig. 6(b). As expected, the *TA* measure tends to decrease as the number of objects increases, however there is no clear trend for the *Recall*. One of the lowest results in both measures is obtained in trays containing 6 foods, whereby we can determine that the errors correspond to 17 misclassified objects along 12 trays, that is, an average error of 1.42 objects per incorrectly classified tray. Despite having a low *TA* (0.478), the results are good considering the *Recall* obtained, since it is preferable to minimize the number of errors per tray if we think of a semi-automatic food billing system, in which the operator would make minor corrections if necessary.

When reviewing the overall mean of errors by misclassified trays, we can see that our classifier has an average of 1.09 errors along 85 trays classified incorrectly, compared to [14] that has an average of 3.33 errors along 76 trays classified incorrectly. That said, even though the baseline method achieves to completely classify 9 trays more than our proposal, due to its overall performance, the misclassified trays have about three times as many objects wrongly classified per tray.

The results achieved with our approach consider a value of 0.5 for the thresholds used in both procedures, Background Removal (BR) and Non-Maximum Suppression (NMS). However, our approach achieves a good performance not only with a unique combination of values, but also with a wide range of them. Specifically, for the problem at hand, we can obtain results close to the ones described with any value in the ranges [0.3–0.6] for BR and [0.3–0.5] for NMS thresholds (see Fig. 7). The flexibility of choosing threshold values in a wide range suggests that our approach is robust with respect to its parameters. Furthermore, considering the F2 score, any value of the parameters for our proposed method produces better results than YOLOv2 + III-C1.

Finally, some examples of the results obtained by means of our proposed Semantic Food Detection method are shown in Fig. 8. In general terms, the classifier achieves a good performance in a variety of food items, where the main difficulties encountered are due to the following issues: 1) unlabeled food items, because they are not part of the 65 classes (e.g. fresh cheese) or because they are not belonging to the same tray and that have been recognized by our algorithm; 2) the same food items placed very close (e.g. mandarine); 3) foods ignored because they are not clearly distinguishable whether correspond to a meal or not (e.g. pudding); and 4) confusions with classes corresponding to different kinds of cakes (e.g. torta_cream), meats, pastas, among others.

V. Conclusion

We present a novel system that performs Semantic Food Detection applied to the problem of food tray analysis in self-service restaurants. More precisely, we integrate both techniques, food/non-food semantic segmentation with food

detection, through the application of two procedures: a probabilistic procedure that allow us remove the background detections, and a custom non-maximum suppression procedure to avoid the occurrence of duplicate detections.

Regarding the architecture, we deal with the problem at hand using two pathways in parallel for food detection and semantic segmentation. The purpose of applying this separate computation is to take advantage of the benefits of each method separately to later combine them. In this manner, they do not condition each other, but reinforce themselves. In particular, if we propose an end-to-end architecture which directly feeds the segmentation output into the detection, the segmentation errors could not be recovered and, therefore, they could negatively influence the detection performance.

As for the results, our proposal significantly outperforms the state-of-art in terms of recall and mean average accuracy. Furthermore, our model is less sensitive to class imbalance and the mean of errors per foods placed on a tray is about 1, when the classifier is not able to recognize the whole tray well. The latter is quite relevant if our approach is applied in a semi-automatic billing system, in which the cashier would have to make only small changes to generate the final bill, and in this way to streamline the process involved in a self-service restaurant of *grab a meal, pay, and eat*. Furthermore, our proposed approach takes less than 0.5 seconds to predict all foods present in a image, considering the use of a personal computer with a low performance GPU (GeForce 940MX).

Our future research is focused on semantic detection of food ingredients and completely automating the self-service billing by integrating the restaurant menu by geolocalization.

REFERENCES

[1] H. Kagaya and K. Aizawa, "Highly accurate food/non-food image classification based on a deep convolutional neural network," in *Proc. Int. Conf. Image Anal. Process.*, 2015, pp. 350–357.

[2] F. Ragusa, V. Tomaselli, A. Furnari, S. Battiato, and G. M. Farinella, "Food vs non-food classification," in *Proc. Int. Workshop Multimedia Assisted Dietary Manag.*, 2016, pp. 77–81.

[3] E. Aguilar, M. Bolaños, and P. Radeva, "Exploring food detection using CNNs," in *Proc. Comput. Aided Syst. Theory EUROCAST*, 2018, pp. 339–347.

[4] A. Singla *et al.*, "Food/non-food image classification and food categorization using pre-trained GoogLeNet model," in *Proc. Int. Workshop Multimedia Assisted Dietary Manag.*, 2016, pp. 3–11.

[5] K. Aizawa, Y. Maruyama, H. Li, and C. Morikawa, "Food balance estimation by using personal dietary tendencies in a multimedia food log," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 2176–2185, Dec. 2013.

[6] G. Waltner *et al.*, "Personalized dietary self-management using mobile vision-based assistance," in *Proc. Int. Workshop Multimedia Assisted Dietary Manag.*, 2017.

[7] J. Dehais, M. Anthimopoulos, S. Shevchik, and S. Mougiakakou, "Two-view 3D reconstruction for food volume estimation," *IEEE Trans. Multimedia*, vol. 19, no. 5, pp. 1090–1099, May 2017.

[8] M. Bolaños, A. Ferrà, and P. Radeva, "Food ingredients recognition through multi-label learning," in *Proc. Int. Conf. Image Anal. Process.*, 2017, pp. 394–402.

[9] J. Chen and C.-W. Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," in *Proc. ACM Multimedia Conf.*, 2016, pp. 32–41.

[10] W. Min, S. Jiang, J. Sang, H. Wang, X. Liu, and L. Herranz, "Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration," *IEEE Trans. Multimedia*, vol. 19, no. 5, pp. 1100–1113, May 2017.

[11] W. Min, B. Bao, S. Mei, Y. Zhu, Y. Rui, and S. Jiang, "You are what you eat: Exploring rich recipe information for cross-region food analysis," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 950–964, Apr. 2018.

[12] J. Chen, C. Ngo, and T. Chua, "Cross-modal recipe retrieval with rich food attributes," in *Proc. ACM Multimedia Conf.*, 2017, pp. 1771–1779.

[13] G. Ciocca, P. Napoletano, and R. Schettini, "Food recognition and leftover estimation for daily diet monitoring," in *Proc. Int. Conf. Image Analy. Process.*, 2015, pp. 334–341.

[14] G. Ciocca, P. Napoletano, and R. Schettini, "Food recognition: A new dataset, experiments, and results," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 3, pp. 588–598, May 2017.

[15] G. Raimato, "The design of a smart tray with its canteen users: A formative study," in *Proc. Int. Conf. Methodologies Intell. Syst. Technol. Enhanced Learn.*, vol. 617, pp. 36–43, 2017.

[16] H. Kagaya, K. Aizawa, and M. Ogawa, "Food detection and recognition using convolutional neural network," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 1085–1088.

[17] K. Kitamura, T. Yamasaki, and K. Aizawa, "FoodLog: Capture, analysis and retrieval of personal food images via web," in *Proc. Workshop Multimedia Cooking Eating Activities*, 2009, pp. 23–30.

[18] M. Bolanos and P. Radeva, "Simultaneous food localization and recognition," in *Proc. Int. Conf. Pattern Recognit.*, 2016, pp. 3140–3145.

[19] S. Mezgec and B. Koroušić Seljak, "Nutrinet: A deep learning food and drink image recognition system for dietary assessment," *Nutrients*, vol. 9, no. 7, p. 657, 2017.

[20] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," *arXiv:1612.06543*, 2016.

[21] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[22] K. Yanai and Y. Kawano, "Food image recognition using deep convolutional network with pre-training and fine-tuning," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops*, 2015, pp. 1–6.

[23] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma, "Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment," in *Proc. Int. Conf. Smart Homes Health Telematics*, 2016, pp. 37–48.

[24] H. Hassannejad, G. Matrella, P. Ciampolini, I. De Munari, M. Mordonini, and S. Cagnoni, "Food image recognition using very deep convolutional networks," in *Proc. Int. Workshop Multimedia Assisted Dietary Manag.*, 2016, pp. 41–49.

[25] Y. Kawano and K. Yanai, "Automatic expansion of a food image dataset leveraging existing categories with domain adaptation," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2014, pp. 3–17.

[26] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101–mining discriminative components with random forests," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 446–461.

[27] E. Aguilar, M. Bolaños, and P. Radeva, "Food recognition using fusion of classifiers based on CNNs," in *Proc. Int. Conf. Image Anal. Process.*, 2017, pp. 213–224.

[28] V. Bettadapura, E. Thomaz, A. Parnami, G. D. Abowd, and I. Essa, "Leveraging context to support automated food recognition in restaurants," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2015, pp. 580–587.

[29] O. Beijbom, N. Joshi, D. Morris, S. Saponas, and S. Khullar, "Menu-match: Restaurant-specific food logging from images," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2015, pp. 844–851.

[30] R. Xu, L. Herranz, S. Jiang, S. Wang, X. Song, and R. Jain, "Geolocalized modeling for dish recognition," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1187–1199, Aug. 2015.

[31] L. Herranz, S. Jiang, and R. Xu, "Modeling restaurant context for food recognition," *IEEE Trans. Multimedia*, vol. 19, no. 2, pp. 430–440, Feb. 2017.

[32] W. Shimoda and K. Yanai, "CNN-based food image segmentation without pixel-wise annotation," in *Proc. Int. Conf. Image Anal. Process.*, 2015, pp. 449–457.

[33] W. Shimoda and K. Yanai, "Foodness proposal for multiple food detection by training of single food images," in *Proc. Int. Workshop Multimedia Assisted Dietary Manag.*, 2016, pp. 13–21.

[34] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[36] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.
[37] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1925–1934.
[38] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.
[39] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional DenseNets for semantic segmentation," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 1175–1183.
[40] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB.* Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.
[41] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.
[42] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6517–6525.
[43] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors," *Eur. Conf. Comput. Vis.*, pp. 340–353, 2012.
[44] P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1–20, Sep. 2009.
[45] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
[46] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.
[47] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 8, pp. 800–810, Aug. 2001.
[48] J. Redmon, "Darknet: Open source neural networks in C," 2016. [Online.] Available: http://pjreddie.com/darknet/

Authors' photographs and biographies not available at the time of publication.

Where and What Am I Eating? Image-Based Food Menu Recognition

Marc Bolaños[1,2( )], Marc Valdivia[1], and Petia Radeva[1,2]

[1] Universitat de Barcelona, Barcelona, Spain
marc.bolanos@ub.edu
[2] Computer Vision Center, Bellaterra, Spain

Abstract. Food has become a very important aspect of our social activities. Since social networks and websites like Yelp appeared, their users have started uploading photos of their meals to the Internet. This phenomenon opens a whole world of possibilities for developing models for applying food analysis and recognition on huge amounts of real-world data. A clear application could consist in applying image food recognition by using the menu of the restaurants. Our model, based on Convolutional Neural Networks and Recurrent Neural Networks, is able to learn a language model that generalizes on never seen dish names without the need of re-training it. According to the Ranking Loss metric, the results obtained by the model improve the baseline by a 15%.

Keywords: Multimodal learning · Computer vision · Food recognition

1 Introduction

Food and nutrition is one of the main activities in people's lives. Nowadays, food does not only cover a basic need, but it has become a really important aspect of our social life. Since social networks appeared and, with them, food-focused applications (like TripAdvisor, Yelp, etc.), their users have started uploading photos of their meals to the Internet. It seems to be a strong and visible tendency in today's society to share pictures of absolutely every piece of food that we taste; exotic or local, fancy-looking or ordinary. Moreover, people post, on many different social media channels, plenty of videos of special restaurants where they eat. Every single day, thousands of people use social media to make recommendations, promote a particular place or give their friends a warning about a nearby restaurant. That is why, tags and location opportunities were introduced for all social media users to make their posts easier and faster to create. The creation of automatic tools for food recognition based on images could enable an easier generation of content, create food diaries for improving nutrition habits or even create personal food profiles for offering personalized recommendations.

The purpose of this work is to explore a problem that we call image-based food menu recognition, which consists in, given an image, determine its correct

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11134, pp. 590–605, 2019.
https://doi.org/10.1007/978-3-030-11024-6_45

Fig. 1. Example of the Food Menu Recognition problem, where we have to retrieve the correct food name from a list of menu items.

menu item corresponding to the restaurant where it was taken (see Fig. 1). By being able to match the picture to an item of the menu it would be easier to retrieve the exact nutritional information of the food or any other data stored by the restaurant owners. Some of the main applications for this model would be creating a personalised profile with food preferences or a personal food diary for improving the eating habits.

The proposed methodology does not need to train a new model for each restaurant [32], instead it will learn to understand meal names in relation to a set of examples by learning a language model. We should point out the difficulty of the problem because of the context where we are working in. Restaurants usually use fancy names to refer to the dishes just to get the attention of their customers. Additionally, food presentation is different in every restaurant, having a high intra-class variability. Chefs try to surprise the customers by using unusual combinations of ingredients, colorful plates and/or sauces.

1.1 Health and Leisure

The work in [22] introduces the relationship that exists between food consumption and people's health. In Europe, despite being a first-world region, more than 4 million people die each year due to chronic diseases linked to unhealthy lifestyles. In many of these cases, the lack of basic knowledge or awareness is a crucial factor in all problems, most people simply do not pay much attention to their eating habits. Furthermore, as it is mentioned in [24], a great number of deaths related to coronary heart diseases are caused by a group of major risk factors among which bad eating habits are at the top.

On the other hand, for a lot of people being and feeling healthy is considered a must. Thanks to social networks, people share their healthy lifestyle on social media on a daily basis. Nowadays, going out for dinner and enjoying a cosy atmosphere in a restaurant is not enough. The healthier (and better looking)

your food is, the better. Because of this important fact, today's restaurants are really visible online and they tend to use many different Internet channels to remain in the center of their customers' attention. Food-based applications like Yelp, help their users find opinions on the quality of the service in the place they plan to visit, and all the data introduced is generated by the users with their smartphones.

1.2 Food Analysis and Deep Learning

Considering the huge number of pictures of meals that people upload on the Internet, food analysis has become popular in the Deep Learning field. That is the reason why several public datasets have appeared. Some examples of public well-known datasets are Food-101 [8], UEC Food256 [16], or Vireo-Food 172 [9]. The most basic problem related to food explored in the literature is food detection [1], which consists in determining if any kind of food appears in an image. Food recognition is one of the most popular problems nowadays [2]. It consists in recognizing the food present on a picture given a pre-defined set of classes (dishes). Other applications of food analysis are food localization, which consists in detecting multiple dishes in a picture [7], calories estimation [13], ingredients detection [5], or multi-dishes recognition for self-service restaurants [3], which combines several of the aforementioned problems.

1.3 Restaurant Food Recognition

Several applications are focused on understanding customers' experiences in restaurants. Some sites like Yelp have plenty of information, but they are not able to classify a picture in the restaurant's menu automatically. It is the user who must do this manually. For this reason, we propose a model to solve this specific problem: locate the restaurant where customers are eating and recognize the meal that they chose from the menu [32]. Solving this problem would allow to create automatic personalized food diaries or personal food preferences, among other applications. The novelties of our work are the following:

- We propose a model that determines the similarity between a picture of food and the dish name provided in the restaurant's menu. Thanks to the language model learned, the system is able to detect the most probable food item in the menu using semantic information from LogMeal's API.
- We propose the first model for food menu recognition applicable to any restaurant. The system does not need previous information of a specific restaurant or a set of examples for a specific class to perform the prediction.
- We make public a dataset collected from Yelp[1]. Our dataset contains 53,877 images, from 313 restaurants and 3,498 different dishes.
- The results obtained over the collected data improve the baseline by a 15%.

[1] http://www.yelp.com.

In the context of the dataset, although ours is equivalent to the one proposed in [32], which is in Chinese, we were not able to perform tests on their dataset due to language issues. A critical component of our methodology is the language model, which allows to generalize for any restaurant, but considering the lack of embedding models pre-trained on Chinese, it is not possible to directly apply it.

This paper is organized as follows. In the related work (see Sect. 2), we explain previous papers published in relation with the problem that we want to solve. Our proposed model is introduced in the methodology (see Sect. 3). The dataset section (see Sect. 4) introduces the data used to train our model and how it was collected. In results (see Sect. 5), we explain and discuss the set of experiments done to choose the best parameters of the proposed model and their performance. Finally, we draw some conclusions and future work (see Sect. 6).

2 Related Work

Deep learning and Convolutional Neural Networks (CNNs) [17] have played a major role in the development of food-related methods in the last years. The huge amount of images related to food available on the internet in websites like Google Images, Instagram or Pinterest have allowed to collect large-scale datasets useful for training deep learning architectures. Even though, challenges inherent to the culinary world like intra-class variability (e.g. apple pie) and inter-class similarity (e.g. different types of pasta), demand the use of complex and smart algorithms. In this section we review the literature on works related to food analysis problems, some important works on multi-modal learning and food, and the application of these techniques in the restaurants context.

2.1 Food Analysis

In the literature there exist several problems and topics related to the analysis of food images. One of the most notable topics is food detection [1,23], where the goal is to detect whether a given image contains any food-related information/element. In a similar way, food recognition [2,20,25] is a widely explored topic, being the goal in this case to classify the image into a set of pre-defined list of classes related to food (usually prepared meals).

Other problems explored in the literature that are related to food analysis are calorie counting and monitoring or volume estimation, like in [19,31], where the authors present a mobile phone-based calories monitoring system to track the calories consumption for the users. Or focused on diabetes, Li et al. [18] estimate the amount of carbohydrate present in a meal from an image.

Other works have treated problems like food localization. In [7] the authors introduce the use of egocentric images to perform food detection and recognition. Food ingredients recognition [5,9] uses a state of the art CNN to predict a list of ingredients appearing in the meal. Food localization and recognition on self-service restaurants is presented in [3].

2.2 Multi-modal Food Analysis

Some times, food analysis uses context or additional information to improve the accuracy of the predictions. This complementary data can be of several types (e.g. images or text). Multi-modal Deep Learning [21] solves this particular problem, learning features over multiple modalities. The paper in [26] introduces a new large-scale dataset with more than 800.000 images and 1.000 recipes. The predictive model presented in the paper tries to join images and recipes through a retrieval task. The proposed solution generates two vectors. One of the vectors represents the image and the other one represents the recipe (text). For optimizing the model, they use the cosine similarity loss, which determines if a given recipe-image pair represents the same food.

The problem that we face also has two different inputs: we need to compare an image and a text sequence, so it could also be formulated as an image retrieval problem. The main differences of our proposal is that, instead of using a general purpose CNN to generate the features vector of the image, we use a semantic-based system for generating food categories that will be structured as a feature vector. Additionally, we use the dish name (text) instead of the recipe and intend to classify the input image into a set of menu items, being a problem more related to restaurant food recognition.

2.3 Restaurant Food Recognition

Seeing food analysis from a different perspective, in [4] the authors propose an automatic food logging system using smartphones. They use state of the art computer vision techniques and add context information of the restaurant to predict the food being consumed by the costumer. The system in [31] creates a calorie estimation from web video cameras in fast food restaurants across the United States. They focused on a reduced group of restaurants to understand the obesity problem. Similarly to our proposal, Xu et al. [32] introduces the context of the pictures to recognize the dish appearing in the image. Using the GPS information provided by the smart-phones they can determine a set of possible restaurants where the picture has been taken. This reduces the search space, which is really important when you try to determine the restaurant and menu item that appear in the picture taken by the user.

The system in [32] needs to train a discriminative model for each pair of restaurants in the dataset comparing their menus and images. Another common problem present in food recognition (or object recognition in general) is that it is limited to a predefined set of classes. This means that if the model was not trained to recognize a specific type of food, it will never provide it as a possible output. Furthermore, the complexity in the restaurants' food recognition resides in the need of training a different model for each restaurant. These models could be very accurate, but the number of outputs is also limited to the restaurant's menu. In this paper, we propose a model that solves these problems. It learns a language model considering a great amount of possible names and associates them to their corresponding pictures. Thus, our algorithm should be able to

take a completely new restaurant's menu (never seen before) and a totally new picture associated to one of the menu's items and find out the correct menu item given the list. Thus, implying that the proposed model does not need to specifically learn every meal.

3 Image-Based Food Menu Recognition: Our Model

Figure 2 shows a scheme of our proposed model, which is based on image retrieval. Given two inputs: an image, and a dish name, it gives an output value based on their similarity. By using this, the prediction process consists in running the predictive model for each menu item and a single meal picture. The generated results produce a ranked list based on the most-similar-first criterion.

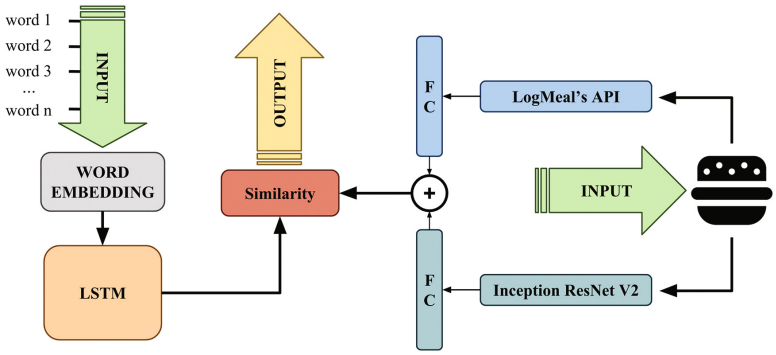

Fig. 2. Image-based food menu recognition model. On the one hand, the system gets an image and applies two different CNNs to generate the feature vectors. Each one is connected to a different fully connected layer to generate comparable structures and are combined performing an addition operation. On the other hand, the text sequence is processed by a word embedding and a Long Short Term Memory. Finally, we compute the similarity between the two inputs using the Euclidean similarity.

3.1 Image and Dish Name Embedding

Our method takes two different inputs, one in the form of an image, that will be transformed in two vectors of different modalities: a low-level vector and a high-level semantic vector, and the other in the form of text. Which means that they must be treated differently before embedding them into the system.

First, the image is converted in two vectors in parallel. One of them provides a low-level description of the food image by using the penultimate layer of the InceptionResNetV2 [29] CNN, composed by a vector of 1,536 values. This CNN is pre-built in the Keras [10] framework and trained using the ImageNet [12] dataset. The other vector provides a high-level semantic description of the food appearing in the image by using LogMeal's API[2]. This API provides three different CNNs that predict the dish [2], food group (or family) [2] and the ingredients

[2] http://www.logmeal.ml.

detected in the image [5]. More precisely, LogMeal's API provides (during the development of this paper) as output the probabilities of the image of belonging to 11 food groups (e.g. meat, vegetables, fish, soup, etc.), 200 dishes (e.g. pizza, spaghetti alla carbonara, etc.) and 1.092 ingredients (e.g. tomato, cheese, salt, garlic, etc.). In the implementation of our model, we are not using the ingredients output because, as we observed, the large dimensionality of the output and the noise that this group introduces to the system does not help obtaining better results. This, in order to build the semantic high-level vector, we concatenate the probabilities vector of the food groups together with the probabilities vector of the dishes.

Second, the text sequence input representing the meal's name is encoded using a word embedding. The inputs of our dataset are, in most of the cases, in English or Spanish. For this reason, and in order to make our model converge quicker, we need a word2vector pre-trained system supporting multiple languages. This is why we chose ConceptNet [28], which generates vectors of 300 features. The words that do not appear in ConceptNet's vocabulary are initialized using a vector of random values.

Unlike the two vectors extracted from the images, which are pre-computed and used as inputs to our system, the word embedding matrix is considered in the optimization procedure and trained together with the rest of the model.

3.2 Model Structure

More details about the image feature vectors generation and embedding can be seen in Fig. 3. One of them comes from LogMeal's API response and the other from the InceptionResNetV2. Later, each of them is inputted to the system and linked to a fully connected (FC) layer of 300 neurons. This layer transforms the feature vectors to the same size, so we can combine them applying an addition operation, which has been proven to be a simple yet effective way of multi-modal information merging [6].

Considering the text sequence that encodes the meal's name, it is generated using a Long Short Term Memory (LSTM) [15] network (Fig. 4) that encodes and joins the sequence of word embedding vectors generated in the first step. In order to match the dimensions of the image vector, the output size of the LSTM is also set to 300 neurons.

3.3 Similarity and Ranking

The last part of the model consists in processing the vectors provided from the image side and the text side in order to calculate their similarity, which will be a value between 0 and 1. Nevertheless, given a certain image and all the list of items in a restaurant's menu, we use the generated similarity values in order to build a sorted ranked list. It means that we need to run the model for each item in the menu on the same picture. The similarity function used to build the algorithm is an adaptation of the Euclidean distance $\frac{1}{1+\|q-p\|}$.

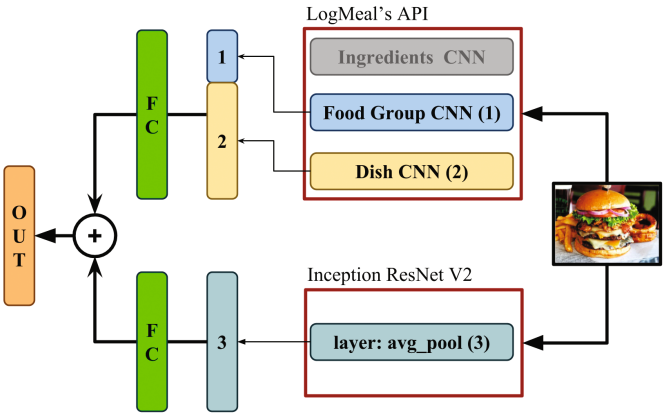

Fig. 3. Image processing part of our model. The system uses the food group and dish recognition outputs of LogMeal's API to create a semantic vector and connect it to a FC layer. The penultimate layer of the InceptionResNetV2 CNN is also used in parallel as a low-level feature vector which is connected to another FC layer. Finally, both partial results are combined performing an addition.

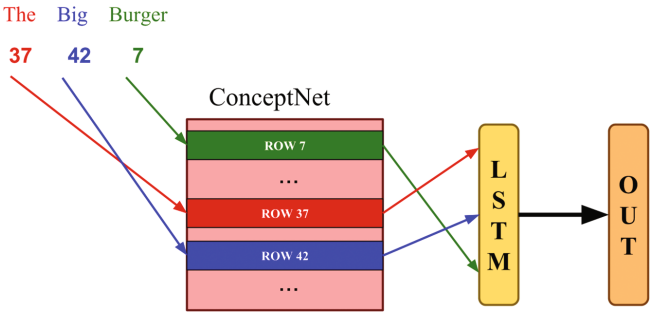

Fig. 4. The text sequence is encoded using a Word Embedding matrix, which is initialized using ConceptNet [28]. The generated vectors are connected to an LSTM.

4 Dataset

The dataset presented in this work was built using Yelp as the source of the information. We scraped the available public information of each restaurant, which consists in a list of menus for each restaurant, a list of dishes for each menu, and a list of images for each dish.

4.1 Dataset Characteristics

The dataset was built from restaurants located in California. We chose this location because of the amount of active Yelp users in this area. We make the dataset publicly available[3].

Analyzing the response of LogMeal's API, we decided to remove the ingredients information. Analyzing the outputs for images of the same dish name, we observe that they have similar activation points, and at the same time they are

[3] Available after paper publication due to blind review process.

different for images that represent different meals. Nevertheless, the ingredients recognition is noisy and does not give enough relevant information. Leading to an increase in the dimensionality of the input and a decrease of performance.

Table 1 (right) shows the number of images, dishes and restaurants in the dataset. The dataset dishes' vocabulary is composed of 1,584 different words. Figure 5 shows an histogram of the number of dishes per restaurant (left), and the number of images per dish (right). Observing the figures, the number of restaurants with just only one dish in their menu is considerably high, that is because we only retrieve the dishes containing some image. Additional problems that we found during dataset collection include the language of the dishes. Due to the location of the restaurants, there is a high probability of finding dishes in both English and Spanish, which introduces a problem: special characters. We encoded the text using the UTF-8 format, but there are some cases where the characters were represented by an empty symbol (_). We decided to remove these samples from the dataset in order to avoid errors during the word embedding.

4.2 Dataset Split

The dataset is split in three groups: training, validation and testing. Previously to the split process, we cleaned the data. This means removing the dishes encoded in a not valid format or the ones that do not have more than 5 images. The dishes are randomly split into three groups: the training group contains 80% of the dishes, 8% is included in validation and 12% of the meals are in the testing split. The number of images of the groups are shown in Table 1 (left).

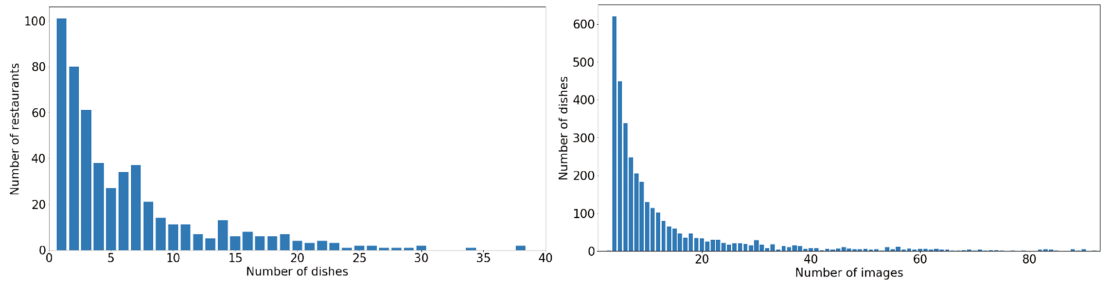

Fig. 5. Histogram of the number of dishes (with images) per restaurant in the dataset (left). Histogram of the number of images per dish in the dataset (right).

Table 1. Number of images in each split of the dataset (left). Number of images, dishes and restaurants of the dataset (right).

Split	# of images
Training	37,956
Validation	7,721
Test	10,794

Split	# of samples
# of images	53,877
# of dishes	3,498
# of restaurants	313

Considering that our model encodes the similarity of the image and text inputs, we need to provide both positive and negative samples in order to train it. The information downloaded from Yelp only contains positive examples, for this reason a set of negative samples has been generated for training (becoming a 50% of the total training samples). The negative examples have been generated assigning a wrong dish name to every image of the dataset. The validation and test splits are built randomizing the set of selected dishes in the menu together with the correct one. The groups of dishes where formed by randomly selecting between 10 and 20 dishes per menu. We generate a random list instead of using the menus of the restaurants to avoid restaurants that have few dishes in their menus.

5 Results

In this chapter we present the results obtained in our work, introduce the metrics used to evaluate the system and show the set of experiments created to find the best combination configuration of our model.

5.1 Ranking Loss and Accuracy Top-1 Distance

In order to compare the performance of the different methods, we use the Ranking Loss [30]. The lower the ranking loss is, the closer is the right value to the top of the list.

To complement the ranking loss error metric, we introduce our own accuracy metric in Eq. 1, which we call accuracy top-1 distance. This measure evaluates how close the ranked result is to the top. The difference with the ranking loss is that our metric only takes in consideration the distance from the position of the predicted class to the top of the ranking. We normalize the output between 0 and 1 using the number of labels in our ranking.

$$\text{accuracy top-1 distance} = \frac{n_{labels} - 1 - ranking_{position}}{n_{labels} - 1} \tag{1}$$

5.2 Experimental Setup

There are several components of our methodology that need to be tuned for finding the best configuration. The selection of the best combination of components was done using a forward propagation-grid search, and the policy we follow to choose the best parameter uses the ranking loss error over the test. The configurations to test where grouped in *similarity measures*, *losses*, *CNN features* and *sample weight*. For each step in the grid search, we select the configuration that obtains the best performance for each of the groups. Each configuration was calculated training the model 5 times. The representative model for each configuration was chosen considering the median value of the 5 runs. The results of the best configuration were obtained at the first epoch with a batch size of

64 samples and without applying any data augmentation or normalization process. Following, we detail the different model variants that we compare in the experimental section.

Similarity Measures: We tested two similarity function candidates. (a) the *Euclidean* similarity, which consists on a normalized version of the euclidean distance; and (b) the *Pearson similarity* (see Eq. 2), which is the absolute value of the Pearson correlation. Using the absolute value we get values between 0 and 1.

$$\rho = \left| \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y} \right| \tag{2}$$

Losses: We tested: (a) the binary cross-entropy (BCE) [27], which is a commonly used loss function for binary classification problems; and (b) the contrastive loss (CL) [14], which is usually used for Siamese networks [11]. The contrastive loss is a distance-based system and tries to minimize the separation between examples of the same semantic topic.

CNN Features: We also tested different CNN feature extraction configurations in our model: (a) using only the features from LogMeal's API (*LM*); (b) combination of the vectors from LogMeal and the InceptionResNetV2 CNN (*LM+Inc*); and (c) InceptionResNetV2 only (*Inc*).

Sample Weight: The last configuration to test is the sample weight. It indicates whether we want to assign a weight value to each dish in relation with the amount of images that it contains with respect to the total number of images in the dataset. This kind of weighing is usually useful when the dataset is unbalanced, giving more importance to the samples that are less frequent.

5.3 Experimental Results

Table 2 shows the results of the grid search. The last row of the table displays the baseline error (based on a random selection of an item in the menu) and accuracy value over validation and test. We have to consider that the values of the ranking loss follow the rule, the lower the better. Meanwhile, the accuracy has the opposite behavior, we want to achieve the higher possible value. The first two rows of the table compare the two similarity measures. Both similarity measures are tested with the same loss optimizer, CNN and sample weight values to be comparable. The error of the Euclidean similarity is 0.033 points better than the one using the Pearson function. Comparing the loss functions, we can see that even though the contrastive loss is usually used for similarity-based CNN models, in this case the binary one works better. If we compare the different CNN feature extraction methods, LM and Inc, the fist one works better. It is because LogMeal's models are trained using food images. Despite this considerations, the best results are obtained by the model using the combination of the two CNNs, meaning that both networks complement each other. Finally, we see that we obtain better results if we deactivate the sample weights. The cause for this

Table 2. Comparison of results for the different model configurations. CNN feat. indicates the combination of CNNs used in the model (LogMeal's API and InceptionResNetV2). The weight column indicates if the systems is using sample weight or not. The ranking loss is indicated with *r.loss* (the lower the better), and the accuracy top-1 distance is *acc.* (the higher the better). For each vertical section, a different configuration is tested. When a certain configuration is fixed it is shown in boldface.

				val		**test**	
similarities	**losses**	**CNN feat.**	**weight**	**r. loss**	**acc.**	**r. loss**	**acc.**
euclidean	binary	LM	NO	0.384	0.623	**0.362**	0.671
pearson	binary	LM	NO	0.416	0.602	0.395	0.639
euclidean	binary	LM	NO	0.384	0.623	**0.362**	0.671
euclidean	contrastive	LM	NO	0.405	0.398	0.375	0.664
euclidean	**binary**	LM	NO	0.384	0.623	0.362	0.671
euclidean	**binary**	LM+Inc	NO	0.372	0.641	**0.351**	0.678
euclidean	**binary**	Inc	NO	0.443	0.572	0.413	0.598
euclidean	**binary**	**LM+Inc**	NO	0.372	0.641	**0.351**	0.678
euclidean	**binary**	**LM+Inc**	YES	0.396	0.612	0.378	0.668
euclidean	**binary**	**LM+Inc**	**NO**	**0.372**	**0.641**	**0.351**	**0.678**
random selection (baseline)				0.5	0.5	0.5	0.5

might be that we do not have a set of pre-stablished classes, but instead we have a language model that links them semantically. This component of our architecture is able to better learn the importance of each sample without the need of forcing a specific weight during optimization. Concluding the table analysis, the best combination of parameters for our model improves the baseline by a 15%. The best ranking loss for the test group is 0.351 and the accuracy top-1 distance is 0.678. It means an improvement of 0.149 and 0.178 points respectively over the baseline.

5.4 Visual Results Analysis

In Figs. 6 and 7 we show some visualizations of the results obtained by our model. The visualization contains a picture of the meal, the ranked results of our system and the true prediction for the image. Figure 6 shows that the cases where the system works better is when the picture presents a single piece of food and the image is clear and centered as well as contains a common dish (with enough samples in the training set). Figure 7 shows examples of failure cases, where the images contain multiple meals on them, making the recognition harder. Additionally, it is appreciable that the dishes with long names are usually at the bottom of the ranking. It is because these meals do not contain a lot of images and are not very popular in the restaurants. So, the model is not able to learn them and retrieve good predictions.

Another problem that we encountered was that, even being uncommon, the data tagged by Yelp's users is misclassified because the pictures uploaded to the site are not verified. Sometimes, the users take photos of their dishes including

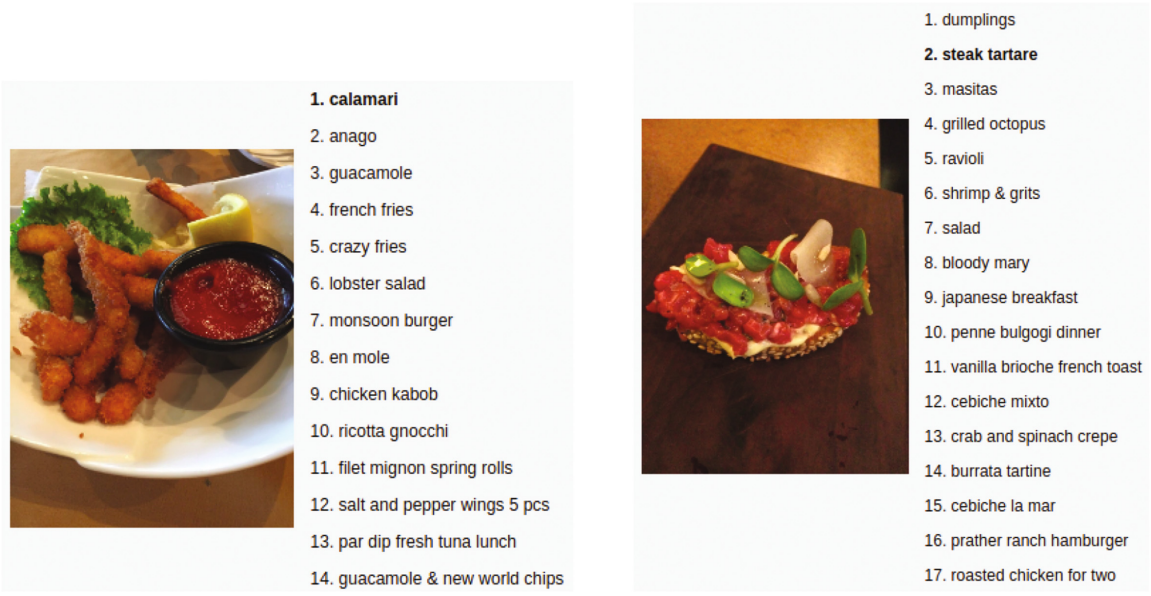

Fig. 6. Examples of ranked lists produced by our algorithm for images of the dishes 'calamari' (left) and 'steak tartare' (right). We observe the good results obtained when the names of the dishes are common enough.

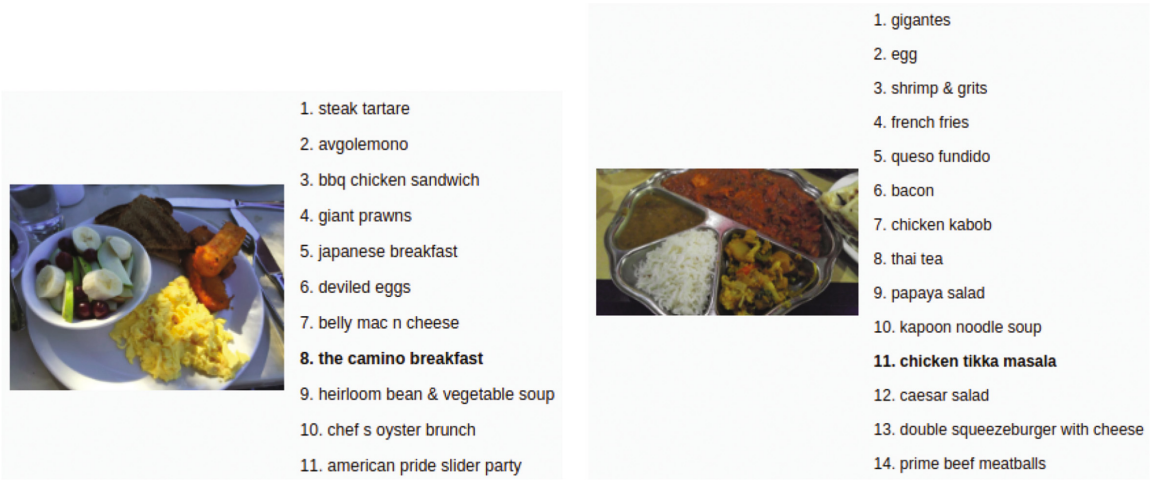

Fig. 7. Examples of ranked lists produced by our algorithm for images of the dishes 'the camino breakfast' (left) and 'chicken tikka masala' (right). Worse results are obtained when the names of the dishes are rare.

context information, and it is a possibility that this information includes other people's meals, which makes more difficult to classify the sample. The main difficulty for the algorithm is dealing with a high variety of names. The restaurants have some speciality dishes that they name at their own. These meals are really difficult to classify, even for a human. Visualizing the results and analyzing the responses of a random selection of the predictions, we have found some properties that usually work better in our system. The meals that contain common food names tend to get better results than the ones with exotic names. This fact is due to two main reasons: the first one is that the dataset has a lot of examples

with common names and can learn them better, and the second one is that the exotic names do not tend to appear at the word embedding matrix, so the system has no initial information of them. Moreover, these names are present in just a few restaurants, so the system does not have enough examples to learn from.

6 Conclusions and Future Work

We can conclude that it is possible to build a model for food restaurant menu recognition that generalizes for any restaurant available, without the need of learning a different model per restaurant or restaurant pairs. This result is achieved thanks to learning a language model that jointly embeds the information from all the dishes available together with low and high-level (semantic) information coming from the images. The contributions that we have done to the scientific community are the following:

- We introduce the use of a language model for dishes and semantic image information by means of LogMeal's API to perform menu items recognition from restaurants.
- We propose a new model that determines the similarity between a food image and a menu item of a restaurant without the need of re-training for each restaurant, which improves the baseline by a 15%.
- We present a new dataset composed by the dishes and images of the restaurant's menu collected from Yelp. The dataset contains 53,877 images, 3,498 dishes and 313 restaurants.

One of the main issues to take into consideration in the future is the treatment of dishes with exotic names, which can not be easily learned by our language model. Furthermore, in the future we plan to introduce the GPS information of the images. The location of the user gives us a list of two or three candidate restaurants where they are eating. Combining the menus of these restaurants and applying the proposed system we would be able to determine where and what a person is eating.

References

1. Aguilar, E., Bolanos, M., Radeva, P.: Exploring food detection using CNNs. arXiv preprint arXiv:1709.04800 (2017)
2. Aguilar, E., Bolaños, M., Radeva, P.: Food recognition using fusion of classifiers based on CNNs. In: Battiato, S., Gallo, G., Schettini, R., Stanco, F. (eds.) ICIAP 2017. LNCS, vol. 10485, pp. 213–224. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68548-9_20
3. Aguilar, E., Remeseiro, B., Bolaños, M., Radeva, P.: Grab, pay and eat: semantic food detection for smart restaurants. arXiv preprint arXiv:1711.05128 (2017)
4. Bettadapura, V., Thomaz, E., Parnami, A., Abowd, G.D., Essa, I.: Leveraging context to support automated food recognition in restaurants. In: 2015 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 580–587. IEEE (2015)

5. Bolaños, M., Ferrà, A., Radeva, P.: Food ingredients recognition through multi-label learning. In: Battiato, S., Farinella, G.M., Leo, M., Gallo, G. (eds.) ICIAP 2017. LNCS, vol. 10590, pp. 394–402. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70742-6_37
6. Bolaños, M., Peris, Á., Casacuberta, F., Radeva, P.: VIBIKNet: visual bidirectional kernelized network for visual question answering. In: Alexandre, L.A., Salvador Sánchez, J., Rodrigues, J.M.F. (eds.) IbPRIA 2017. LNCS, vol. 10255, pp. 372–380. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58838-4_41
7. Bolanos, M., Radeva, P.: Simultaneous food localization and recognition. In: 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 3140–3145. IEEE (2016)
8. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 446–461. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_29
9. Chen, J., Ngo, C.W.: Deep-based ingredient recognition for cooking recipe retrieval. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 32–41. ACM (2016)
10. Chollet, F., et al.: Keras (2015). https://keras.io
11. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 539–546. IEEE (2005)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: CVPR 2009 (2009)
13. Ege, T., Yanai, K.: Simultaneous estimation of food categories and calories with multi-task CNN. In: 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), pp. 198–201. IEEE (2017)
14. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1735–1742. IEEE (2006)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
16. Kawano, Y., Yanai, K.: Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8927, pp. 3–17. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16199-0_1
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
18. Li, H.C., Ko, W.M.: Automated food ontology construction mechanism for diabetes diet care. In: 2007 International Conference on Machine Learning and Cybernetics, vol. 5, pp. 2953–2958. IEEE (2007)
19. Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., Ma, Y.: DeepFood: deep learning-based food image recognition for computer-aided dietary assessment. In: Chang, C.K., Chiari, L., Cao, Y., Jin, H., Mokhtari, M., Aloulou, H. (eds.) ICOST 2016. LNCS, vol. 9677, pp. 37–48. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39601-9_4
20. Martinel, N., Foresti, G.L., Micheloni, C.: Wide-slice residual networks for food recognition. arXiv preprint arXiv:1612.06543 (2016)

21. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the 28th International Conference on Machine Learning (ICML 2011), pp. 689–696 (2011)
22. Organization, W.H., et al.: Food and health in Europe: a new basis for action. World Health Organization, Regional Office for Europe (2004)
23. Ragusa, F., Tomaselli, V., Furnari, A., Battiato, S., Farinella, G.M.: Food vs non-food classification. In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management, pp. 77–81. ACM (2016)
24. Rozin, P., Fischler, C., Imada, S., Sarubin, A., Wrzesniewski, A.: Attitudes to food and the role of food in life in the usa, japan, flemish belgium and france: possible implications for the diet-health debate. Appetite **33**(2), 163–180 (1999)
25. Salvador, A., et al.: Learning cross-modal embeddings for cooking recipes and food images. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017 (2017)
26. Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I., Torralba, A.: Learning cross-modal embeddings for cooking recipes and food images. Training **720**, 619–508 (2017)
27. Shore, J., Johnson, R.: Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. IEEE Trans. Inform. Theor. **26**(1), 26–37 (1980)
28. Speer, R., Lowry-Duda, J.: Conceptnet at semeval-2017 task 2: extending word embeddings with multilingual relational knowledge. arXiv preprint arXiv:1704.03560 (2017)
29. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI, vol. 4, p. 12 (2017)
30. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining multi-label data. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 667–685. Springer, Boston (2009). https://doi.org/10.1007/978-0-387-09823-4_34
31. Wu, W., Yang, J.: Fast food recognition from videos of eating for calorie estimation. In: IEEE International Conference on Multimedia and Expo, ICME 2009, pp. 1210–1213. IEEE (2009)
32. Xu, R., Herranz, L., Jiang, S., Wang, S., Song, X., Jain, R.: Geolocalized modeling for dish recognition. IEEE Trans. Multimed. **17**(8), 1187–1199 (2015)

J. Vis. Commun. Image R. 60 (2019) 360–370

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Regularized uncertainty-based multi-task learning model for food analysis ☆

Eduardo Aguilar [a,b,*], Marc Bolaños [b,c], Petia Radeva [b,c]

[a] Departamento de Ingeniería de Sistemas y Computación, Universidad Católica del Norte, Avenida Angamos 0610, Antofagasta, Chile
[b] Departament de Matemàtiques i Informàtica, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, 08007 Barcelona, Spain
[c] Computer Vision Center, Bellaterra (Barcelona) 08193, Spain

ARTICLE INFO

Article history:
Received 30 July 2018
Revised 7 December 2018
Accepted 8 March 2019
Available online 11 March 2019

Keywords:
Multi-task models
Uncertainty modeling
Convolutional neural networks
Food image analysis
Food recognition
Food group recognition
Ingredients recognition
Cuisine recognition

ABSTRACT

Food plays an important role in several aspects of our daily life. Several computer vision approaches have been proposed for tackling food analysis problems, but very little effort has been done in developing methodologies that could take profit of the existent correlation between tasks. In this paper, we propose a new multi-task model that is able to simultaneously predict different food-related tasks, e.g. dish, cuisine and food categories. Here, we extend the homoscedastic uncertainty modeling to allow single-label and multi-label classification and propose a regularization term, which jointly weighs the tasks as well as their correlations. Furthermore, we propose a new Multi-Attribute Food dataset and a new metric, Multi-Task Accuracy. We prove that using both our uncertainty-based loss and the class regularization term, we are able to improve the coherence of outputs between different tasks. Moreover, we outperform the use of task-specific models on classical measures like accuracy or F_1.

1. Introduction

Food is a basic component in our daily life, and it is not simply a source of nourishment, but also an important cultural component and a central element on several community celebrations, holidays and leisure activities. It is one of the central pillars both for our pleasure and for our health. Being aware of our daily life, diet is an important issue if we want to follow balanced and healthy lifestyle. The amount of people that suffer from nutrition-related health conditions like obesity or cardiovascular diseases is increasing among our society [1]. Moreover, people that have any of these medical conditions, tend to spend more money on medical care and usually have a shorter lifespan. Thus, the need for increasing the awareness of the importance of following a healthy diet is vital.

Considering the importance of food in our daily life, together with the proliferation of social networks, new trends arose. A rather extended trend related to nutrition consists in taking pictures of food in restaurants or other special events of the daily life. Several users that follow this tendency, are used to share pictures on social networks like Instagram or Pinterest. Considering the huge amount of food-related pictures that are already available *online* due to these routinary actions, it is natural to think of using them for developing automatic methods for food recognition and analysis.

Given the importance of food and the challenging image analysis problem, the computer vision community proposed different food analysis methods like food detection [2], recognition [3,4], localization [5], ingredients recognition [6], multi-attributes recognition [7], recipes retrieval [8], calorie counting [9], food-tray recognition [10], or portion estimation [11], among others. Some of the purposes of these methods could include: (1) easing the tracking of our daily nutrition intake, which would lead to offering recommendations for improvement; (2) providing alerts when recognizing a product that could contain potential allergens; (3) provide cooking recommendations; (4) automatic billing in self-service restaurants; just to mention a few.

Although the potential of food analysis systems is clear, several challenges need to be solved. In particular, if we compare food images to other visual analysis problems like object recognition, food classes have much higher inter-class similarity and intra-class variability, making any food analysis problem very difficult to solve. Considering the particular problems of food-related tasks,

☆ This paper has been recommended for acceptance by Dr. Zicheng Liu.

* Corresponding author at: Departamento de Ingeniería de Sistemas y Computación, Universidad Católica del Norte, Avenida Angamos 0610, Antofagasta, Chile.
E-mail address: eaguilar02@ucn.cl (E. Aguilar).

https://doi.org/10.1016/j.jvcir.2019.03.011
1047-3203/

ingredients recognition can be of high difficulty considering that some ingredients can be present in several textures and shapes, and others can be invisible.

In this paper, we argue that most tasks related to food analysis can be interconnected (e.g. swiss cuisine, fondue and cheese). To this aim, we explore the problem of food multi-task learning and propose a Regularized Uncertainty based Multi-Task Learning model in which different food-related characteristics that are correlated can be predicted from an input image: (a) dish, (b) cuisine, (c) categories and (d) ingredients (see Fig. 1). One of the problems of most multi-task methods is that they do not necessarily consider the correlations between tasks. Note that for multi-label classification problems, the relationship between labels have been studied extensively [12–15]. Instead, as far as we know, there are no proposals for multi-task learning case, where the relationship of the labels between the different tasks being single-label or multi-label are used to regularize the predictions. Furthermore, we argue that different tasks should influence in different degree to the overall model according to their uncertainty. To this purpose, we propose in this paper the following contributions:

1. We extend the model for Multi-Task Learning with Homoscedastic Uncertainty [16] to allow single-label and multi-label classification.
2. We propose a Multi-Task regularization term that weighs existent negative and positive correlations between classes belonging to different tasks.
3. We publish a new Multi-Attribute Food dataset (MAFood-121) with more than 20.000 images and multi-task annotations, including: dish, cuisine and food categories.
4. We propose a new metric called Multi-Task Accuracy (MTA), which measures the prediction agreement and coherence between tasks.

We prove that our model outperforms the state of the art techniques on MAFood-121 and VIREO Food-172 not only using traditional evaluation metrics, but also on our newly proposed metric that ensures the coherence of the different tasks.

The paper is organized as follows: in the following section, Section 2, we review the state of the art related to the most relevant food analysis problems on the computer vision field. In Section 3, we present our proposal for multi-task learning, detailing both our model and the loss functions that we introduce for training. Then, in Section 4, we introduce the multi-attribute food datasets, the MTA metric and compare the results obtained by different methods. Finally, in Section 5, we present the concluding remarks.

2. Related work

In recent years, there have been several studies focused on recognition of food categories.

Food Recognition is one of the most active topics on food-related problems. Most of the initial studies tackling this task proposed the use of hand-crafted features based on color, texture and shape [17–21]; with the exception of [22], that exploited relationships between ingredients, using statistics of pairwise local features to perform the recognition. All of them were evaluated mainly on datasets with too specific, or a limited number of images. Moreover, most of them were taken in restricted conditions [19,22] or with up to 50 different dishes [19,18,21].

Lately, the emergence of Convolutional Neural Networks (CNN) made possible to tackle the problem on more challenging food datasets with a large number of images and a wide diversity of dishes [23,24]. A significant increase of the performance was shown compared to hand-crafted features [23,25]. Most of the results on CNNs have been obtained using fine-tuned models based on winners of the ImageNet challenges [26], being [27,28,25] notable examples that improved the results by a great margin.

Instead of just fine-tuning, a different approach is proposed by [3], which fused several CNN models and explored different fuzzy similarity measures, evidencing better results than using a single network. Finally, Martinel et al. [4] proposed a wide-slide residual network, which provided the best performance in two benchmark datasets, 90.27% on Food-101, and 83.15% on UECFood-256.

Food Group Recognition can serve as a way to represent a group of dishes instead of a particular dish. Food recognition is a complex problem due to its fine-grained nature, which has the particularity of being a problem with a high intra-class variability and high inter-class similarity. In [29], a food dataset with 8 categories related to restaurant menus was proposed. In their dataset, images depicting mixed food (e.g. second course and side dish) were labeled with multiple labels. Instead, the work in [30,31] adopted a semantic categorization of food, in [31] the authors proposed a three-leveled hierarchy of food in order to make better mistakes on the food recognition problem. The authors of [30] proposed a new dataset with 11 categories considering the major food groups defined by the United States Department of Agriculture. Regarding the results on Food-11, Aguilar et al. [3] achieved the best performance by fusing three classifiers based on CNNs.

Ingredients Recognition can be a possible solution to the high diversity of dishes existent in the food recognition problem. Right now, there are more than 8000 dishes (according to Wikipedia) [32]. On the other hand, certain ingredients are not visible or are indistinguishable in the image. Taking into account all these challenges, few works considered the ingredients recognition problem so far. Chen et al. in [6] proposed a model that simultaneously recognizes the dish and the visible ingredients. Bolaños et al. [32] tackled the ingredients recognition problem by considering any present ingredient either visible or not. In addition, they opted for a model with a single output to avoid the loss of generalization capabilities on unseen recipes/dishes.

Cuisine Recognition can be characterized as capturing the set of ingredients, cooking methods and presentation of a certain region in the world. Sajadmanesh et al. [33] showed that, due to the geographical locality of the ingredients, some of them are representative to a specific cuisine, and thus, they play a main role to classify the cuisine [34,33,35,36]. From the computer vision point of view, an early model is proposed by [36], which is based on high-level features detecting 16 ingredients characteristic of certain cuisines. A more recent work [34] took into account that the information of the ingredients, in some cases is not sufficient to classify the cuisine. Therefore, it proposed a multi-modal framework which simultaneously modeled the visual content and textual ingredients to tackle the problem.

Taking into account that on an image we can perform different recognition tasks, our main goal is to propose a highly performing model that boosts prediction using the relationship between different food recognition tasks.

2.1. Multi-task learning based on CNNs

Recently, MTL approaches using CNNs have been adopted by several works to jointly learn related tasks. Several works when applied on face or pedestrian appearance analysis tasks (e.g. facial landmark detection, human pose estimation, etc.), have shown a significant improvement in performance [37,38,34,39–42]. A novel architecture that utilizes the hierarchical features from different

Fig. 1. Food-related tasks predicted by our model.

tasks was proposed in [42]. The authors suggest that features of different CNN levels from multiple tasks are not identical when the tasks are loosely related. They tackle this problem by concatenating the features of different tasks at different CNN levels. Later, they apply a discriminative dimensionality reduction in order to learn a discriminative feature embedding that satisfies the channel size of the following CNN layers. Instead, in our case we deal with highly related tasks. Additionally, we force the joint learning of the features from the different tasks within the loss function by weighing tasks according to their uncertainty and applying a class regularization term. The good results evidenced in the literature has inspired the emergence of new food analysis works that deal with the simultaneous prediction of two or more tasks. Zhang et al. [7] proposed a MTL based on AlexNet CNN, to recognize three tasks: food, cooking method and ingredients. Chen et al. [6] recognized food and ingredients simultaneously, and the authors in [9] applied simultaneous food recognition and calories estimation. An interesting aspect to highlight is that with the exception of [40], previous works weighted uniformly the loss of each individual task [37,39] or manually tuned them [6,9,40,7]. In contrast, Yin et al. [40] proposed a dynamic-weighting scheme that fixed the main task with a weight equal to 1, and learned the weights for each side-task. Although not in the food analysis field, a completely dynamic-weighting is proposed by [16], where a multi-task loss function was derived based on maximizing the Gaussian likelihood with homoscedastic uncertainty for both regression and classification tasks.

3. Regularized uncertainty-based multi-task learning model for food analysis

The Regularized Uncertainty-based Multi-Task Learning model (RUMTL) that we propose addresses the problem of multi-attribute food prediction. Our goal is to generate coherent and simultaneous recognition of different food-related attributes that are correlated: (1) dish, (2) cuisine, (3) food categories and (4) ingredients. We propose tackling the problem in a MTL deep learning framework.

3.1. Model design and activation functions

We adopted the ResNet-50 [43] as a base for our RUMTL architecture due to the good performance demonstrated in computer vision problems related to object detection and, in particular, for food-related problems [3,4,8]. For our purpose, we modified the original design of ResNet-50 by removing the last Fully Connected layer (FC), and instead connected as many FC layers, as tasks we have, with a FC shared layer located at the top of the network containing 2048 neurons (see model architecture in Fig. 2).

Taking into consideration that we have two different types of outputs: (a) single-label for dish and cuisine; and (b) multi-label for categories and ingredients, we need to use different activation functions in the outputs layers.

For the Single-Label tasks (SL), we apply the softmax activation function, due to its ability to obtain a probability distribution that enhances the single most probable class, defined as:

$$softmax(f^{W^t}(x)_i) = \frac{\exp(f^{W^t}(x)_i)}{\sum_j \exp(f^{W^t}(x)_j)},$$

where $f^{W^t}(x)_i$ is the final activation of the network for the sample x on the i-th label and the t-th task, $t = 1,2; i = 1,\ldots,|C|^t$; where $|C|^t$ is the number of classes in task t, which has a set of weights W^t. On the other hand, for the Multi-Label tasks (ML), we apply the sigmoid activation function:

$$sigmoid(f^{W^t}(x)_i) = \frac{1}{1+\exp(-f^{W^t}(x)_i)},$$

which provides an independent probability for each class and enables the prediction of multiple classes [32]. From these activations, we can determine the probability that the output y^t corresponds to the target label(s) $\hat{y}^t$ for the task t given the image x, as follows:

1. SL probability:

$$p(y^t = \hat{y}^t|x) = softmax(f^{W^t}(x)_{\hat{y}^t})$$

2. ML probability:

$$\begin{aligned} p(y^t = \hat{y}^t|x) &= \prod_i^{|C|^t} p(y_i^t = \hat{y}^t{}_i|x) \\ &= \prod_i^{|C|^t} sigmoid(f^{W^t}(x)_i)^{\hat{y}^t{}_i} \times (1 - sigmoid(f^{W^t}(x)_i))^{1-\hat{y}^t{}_i} \end{aligned}$$

3.2. Multi-task uncertainty-based loss

Regarding the loss functions for the model optimization, we apply the categorical cross-entropy for the dish and cuisine tasks, and binary cross-entropy loss for the food categories and ingredients. Note that, for SL and ML tasks the loss function is calculated as: $-\log p(y^t = \hat{y}^t|x)$.

One of the important issues in MTL concerns on how to combine multiple objectives and train them jointly in order to capture the existent relationships between tasks. Using a manually defined weighing procedure is expensive to tune, increasingly difficult and makes the training process both very sensitive to the parameters and also computationally inefficient. To cope with these problems, the authors in [16] proposed to use a weighing procedure for learning and combining different tasks based on probabilistic uncertainty modeling. In Bayesian modeling, there are two main

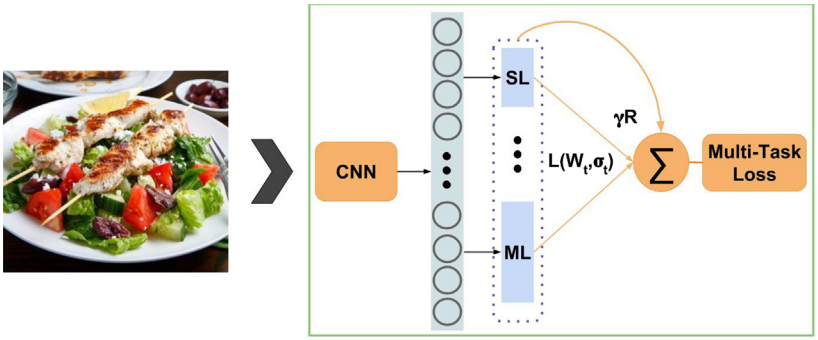

Fig. 2. Regularized uncertainty-based multi-task learning model for classification of food-related tasks.

types of uncertainty [44]: epistemic uncertainty, related to the lack of training data, and aleatoric uncertainty related to the missing information that our data cannot explain. The later can be divided into 2 subcategories: heteroscedastic uncertainty (data-dependent) that is predicted as a model output and homoscedastic (task-dependant) that varies between different tasks.

The authors in [16] proposed a MTL loss function maximizing the Gaussian likelihood with homoscedastic uncertainty. Let $f^W(x)$ be the output of a neural network with weights W and input x. In the case of MTL, the likelihood is expressed as: $p(y_1, y_2, \ldots, y_T)$ considering $f^W(x)$ as a sufficient statistics. Assuming a Gaussian distribution, the loss expressed as a negative log likelihood $L(W, \sigma)$ for each SL classification probabilistic output, modeling the homeoscedastic uncertainty σ, can be approximated by [44]:

$$L(W,\sigma) \approx \frac{1}{\sigma^2} L(W) + \log \sigma^2,$$

using the softmax function as an activation function. The authors defined the MTL loss function integrating the uncertainty along tasks as a sum of the individual multi-task losses: $L(W, \sigma_1, \ldots, \sigma_T) = \sum_{t=1}^{T} L(W, \sigma_t)$.

In contrast to [16] in our case, we have SL as well as ML classification tasks. A straightforward alternative to solve the ML classification is to transform the N possible outputs of the multi-label classification to N independent single-label binary classifiers. With this transformation every output value could become an independent classification task with two labels - relevant or irrelevant (binary relevance method [45]). Then, the loss function with uncertainty based on the categorical cross-entropy might be used for the loss of each binary classification, treating them as independent single-label tasks. However, the binary relevance method assumes independence between outputs, which is not true in our case. Thus, their relationship is not taken into account during the learning. Instead, to avoid suboptimal results due to the ignorance of label correlations, we propose a loss function with uncertainty for the binary cross-entropy. To this purpose, we derive the uncertainty ML classification loss as follows:

$$\begin{aligned}
L(W,\sigma) &= -\log p(y^t = \hat{y}^t | x, \sigma) \\
&= -\log \prod_i p(y_i^t = \hat{y}^t{}_i | x, \sigma) \\
&= -\sum_i \log(sigmoid(f^{W^t}(\tfrac{x}{\sigma^2})_i)^{\hat{y}^t{}_i} (1 - sigmoid(f^{W^t}(\tfrac{x}{\sigma^2})_i))^{1-\hat{y}^t{}_i}) \\
&= \tfrac{1}{\sigma^2} L(W) + \sum_i \log(\tfrac{\exp \frac{1}{\sigma^2} f^{W}(x)_i + 1}{(\exp f^{W}(x)_i + 1)^{\frac{1}{\sigma^2}}}) \\
&\approx \tfrac{1}{\sigma^2} L(W) + K \log \sigma^2
\end{aligned}$$

where L(W) is the sigmoid binary cross-entropy, and K corresponds to the number of labels. Note that in order to simplify the objective function, similar to [16], in the last transition we introduce the assumption that $\frac{1}{\sigma^2}(\exp \frac{f^W(x)_i}{\sigma^2} + 1) \approx (\exp f^{W}(x)_i + 1)^{\frac{1}{\sigma^2}}$, becomes an equality when $\sigma^2 \to 1$.

Therefore, we extend the multi-task objective with homoscedastic task uncertainty for SL and ML classification task as:

$$L(W, \sigma_1, \ldots, \sigma_i) = \sum_i \frac{1}{\sigma_i^2} L(W) + K \log \sigma_i^2 \quad (1)$$

where for SL classification task, K takes a value of 1 and $L(W)$ corresponds to the categorical cross entropy, which coincides with the SL loss expression in Kendall's work [16].

The advantage of this model is that it learns the relative weights, σ_t in a well-founded way. The loss is smoothly differentiable and prevents the task weights σ_t from converging to 0. The parameters σ_t model the observational noise, that is, they capture how much noise is manifested in the output. During the training process, the log likelihood is maximized with respect to the model parameters W and the observation noise parameters $\sigma_t, t = 1, \ldots T$.

3.3. Multi-task class regularization

In our case of food analysis, it is natural that some of the classes from different tasks can be correlated negatively (for example, *fondue* is not typical for Japanese cuisine).

To this purpose, we create a **task-exclusion matrix**, $T^{kl} = \{t_{ij}^{kl}\}, i = 1, \ldots |C|^k$ and $j = 1, \ldots |C|^l$ where:

$$t_{ij}^{kl} = \begin{cases} -1, & \text{if class } i \text{ and class } j \text{ are exclusive,} \\ 1, & \text{otherwise.} \end{cases}$$

To determine the class exclusions, we explored the ground-truth of the training data imposing the condition that if there are no examples of images from class i in task k and class j in task l, then it will be an exclusive relationship.

In our RUMTL model, we penalize when the multi-task classification infers classes of different tasks that are mutually exclusive (e.g. the dish *goulash* and the cuisine *Japanese*). Thus, we define the following penalizing term:

$$R_-^{k,l} = \frac{1}{s^{kl}} \sum_{i=1}^{|C|^k} \sum_{j=1}^{|C|^l} \max(0, -t_{ij}^{kl}) \cdot p(y_i^k | x) \cdot p(y_j^l | x)$$

where s^{kl} is a normalization factor defined as follows:

$$s^{kl} = \begin{cases} 1, & k \text{ and } l \text{ are SL}, \\ \sum_{j=1}^{|C|^l} \max(0, -\min_i(t_{ij}^{kl})), & k \text{ is SL and } l \text{ is ML}, \\ \sum_{i=1}^{|C|^k} \max(0, -\min_j(t_{ij}^{kl})), & l \text{ is SL and } k \text{ is ML}, \\ \sum_{i=1}^{|C|^k}\sum_{j=1}^{|C|^l} \max(0, -t_{ij}^{kl}), & k \text{ and } l \text{ are ML}. \end{cases}$$

On the other hand, when the classes are not exclusive (e.g. the category *pasta* and the cuisine *italian*), the optimization process should push the likelihood of both classes to be as high as possible. Note that this case covers as positive correlation as no correlation between classes. To this purpose, we introduce a second term defined as follows:

$$R_+^{kl} = \frac{\sum_{i=1}^{|C|^k}\sum_{j=1}^{|C|^l} \hat{y}_i^k \hat{y}_j^l (1 - p(y_i^k|x) \cdot p(y_j^l|x))}{\sum_{i=1}^{|C|^k}\sum_{j=1}^{|C|^l} \hat{y}_i^k \hat{y}_j^l}$$

where

$$\hat{y}_c^t = \begin{cases} 1, & \text{if } x \text{ belongs to class } c \text{ from task } t, \\ 0, & \text{otherwise}. \end{cases}$$

Finally, we define the regularization term as a weighed sum of R_- and R_+:

$$R = \binom{T}{2}^{-1} \sum_{k=1}^{T-1}\sum_{l=k+1}^{T} (\alpha R_-^{kl} + \beta R_+^{kl})$$

where α and β look for a trade-off between both terms. In our case, without loss of generality, we used $\alpha = 0.5$ and $\beta = 0.5$. Note that due to the regularization term, if two classes from different tasks are not likely to occur together (that is in some way they are exclusive and in this case, $t_{i,j}^{k,l} = -1$), but the classifiers obtain high probabilities ($p(y_i^k|x) \cdot p(y_j^l|x)$ is high), the regularization term will penalize the loss function. On the other hand, if for a given sample x, there is a clear relation between the classes and at the same time there is no exclusive relation between classes ($t_{i,j}^{k,l} = 1$), the regularization term will try to push the likelihood of the most probable classes close to 1. The final definition of the loss function becomes:

$$L(W, \sigma, \gamma) = \sum_{t=1}^{T} \left(\frac{1}{\sigma_t^2} L_t(W_t) + K \log \sigma_t^2\right) + \gamma R \quad (2)$$

where γ is a parameter to weigh the importance of the regularization term ($\gamma = 1$, in our case).

To sum up, the proposed regularization allows to guide the joint learning of the different tasks considering two factors: the negative (R−) and non-negative (R+) correlation between the classes of different tasks.

- For R−, first of all we must build the task-exclusion matrix from the training data. With this we can determine during the training, if two classes of different tasks are correlated or not. In case the prediction of the model obtains uncorrelated classes, the regularization factor will increase the value of the loss function. As it can be inferred, the model will tend to generate fewer incoherent predictions, but it does not assure us that the prediction really will be correct to the image. For this, we propose the second term of the R+ regularization.
- The term, R+ does not need to use the exclusivity matrix, but considers the ground truth of the image. The aim is to obtain the highest possible probability given by the model to the ground truth of the image for the different tasks, so it will be penalized as long as this probability for the ground truth labels is less than 1.
- When using both terms together we have found a balance between the improvement of the performance and the coherence of the output between the different tasks.

4. Experimental results

In this section, we first present the datasets used, second we describe evaluation measures, third we present the experimental setup and then we describe the results obtained with the proposed RUMTL approach compared to using single-task models.

4.1. Multi-attribute food datasets

In order to justify the results and prove the advantages of our newly proposed multi-task framework, we need a challenging datasets with multiple food-related attributes. In [46], different sets of data with annotations of different tasks were used. However, this is not suitable for building a multi-task model, which requires a single set of data with the annotations for all the tasks. The only publicly available multi-task food dataset is VIREO Food-172 [47], which has annotations for two tasks (dish and ingredients). To highlight the benefits of our approach considering at the same time more realistic multi-task problems, we propose a new dataset with three complementary tasks that have not been exploited so far, two single-label tasks (dish and cuisine) and one multi-label task (food group/categories).

4.1.1. MAFood-121

We built and make public here a dataset of food images comprising 3 different tasks: (a) dish, (b) cuisine and (c) categories. It consists of 21.175 images, distributed as 72.5% for training, 12.5% for validation and 15% for test. We named the resulting dataset as *MAFood-121*.[1] Both dish and cuisine can take only one value per image, while categories have multi-label annotations.

In order to choose the images to include, we selected the top 11 most popular cuisines in the world according to Google Trends (http://www.google.com/trends) (see Fig. A.1). For each cuisine, 11 traditional dishes were chosen. In total, the dataset consists of 121 dishes, where each one belongs to at least one of the following 10 food categories: Bread, Egg, Fried food, Meat, Noodles/Pasta, Rice, Seafood, Soup, Dumpling and VegeTable 8 of the 10 categories used coincide with those proposed by [30].

Regarding the food classes, the images were collected from 4 different sources, 3 of them were the public datasets Food-101 [23], UEC-Food256 [24] and Turk-15 [48]. To reduce the bias that could be present due to the different amount of images per dish, we selected a maximum of 250 images for each one. Regarding Food-101, 250 images were collected from the original test set. As for UEC-FOOD256 and Turk-15, the images were randomly chosen.

For our purpose, it was necessary to collect 38 new dishes along 5 different cuisines from Google Search Engine, because the existing public food datasets consist mainly of food images of a specific geographic region [23,48,24]. We restricted the search to the country that represents a specific cuisine and also restricted their minimum sizes to 256 pixels per side. For each dish, 200 images were downloaded and, following the procedure in [23], all images were rescaled to have a maximum size of 512 pixels. The images acquired were reviewed to remove those with explicit copyright, duplicates, and also those that are not representative of the

[1] http://www.ub.edu/cvub/mafood121/

respective dish. To achieve this, we first applied the automatic food/non-food classifier proposed by [2], and then we manually inspected the remaining images. Consequently, an average of 119 images were obtained per dish. After gathering the images from 11 cuisines and their respective 121 different dishes, we manually labeled the food categories for each image considering the visible ingredients.

4.1.2. VIREO Food-172

This public multi-task dataset consists of 172 popular chinese dishes collected by Baidu and Google image search. VIREO Food-172 has annotations for two tasks: dishes and ingredients. For the last one, the authors only consider the annotation of visible ingredients among 353 ingredients labels, with an average of 3 ingredients per image. In total, the dataset contains 110,241 images corresponding to 172 dishes and 353 ingredient labels, distributed as 60% for training, 10% for validation and the remaining 30% for testing.

4.2. Metrics

In order to evaluate the performance for each individual task, we chose four standard measures frequently used: *Overall Accuracy (Acc)*, for SL; and *Recall (Rec)*, *Precision (Pre)* and F_1*-score* (F_1), for ML. However, these measures do not reflect the quantity of samples asserted by all the tasks at the same time. To this purpose, we propose a new and more *strict* measure, we name *Multi-Task Accuracy (MTA)*, which represents the ratio of images correctly classified by all tasks at the same time (i.e. measures the prediction agreement and coherence between tasks). We formally define the *MTA* measure as follows:

$$MTA = \frac{1}{N}\sum_{i=1}^{N}\prod_{t=1}^{T}\frac{|y_{i,t}^{true} \cap y_{i,t}^{pred}|}{|y_{i,t}^{true} \cup y_{i,t}^{pred}|},$$

where $|.|$ stands for the cardinality for both label sets ground-truth (y^{true}) and predictions (y^{pred}), N and T denote the number of images and tasks, respectively. Note that the denominator specially has sense for the ML data, since it allows to obtain 1 for a special i and j, iff all ML predicted and ground-truth labels coincide.

Note that MTA allows us to evaluate the performance of Multi-task classifiers as a whole. If we have different classifiers with the same performance with respect to the individual tasks, it does not ensure that they provide the same results because the errors of each task can come from different images. We believe that it is of interest to have classification that concentrates the tasks errors in the same images in order to get consistent outputs. In this sense, MTA can help to identify better classifiers even when $F1$ and accuracy are similar for each task. We illustrate it with the following example: suppose we have classifiers C_1 and C_2, and images I_1 and I_2, both with the same accuracy for tasks T_1 and T_2, but with different classification results. $C1$ mis-classifies both tasks for $I1$, and C_2 mis-classifies I_1 for T_1 and I_2 for T_2. Thus, C_1 obtains 50% and C_2 a 0%, with respect to the MTA metric. Therefore, despite having the same accuracy, the classifier C_1 produces more coherent results among tasks. Let us consider another example: C_1 (accuracies 0.8 and 0.8) and C_2 (accuracies 1.0 and 0.6). Let us assume that both tasks are single-label and focus on which data/images the error is produced. In this case, $MTA(C_1)$ will be equivalent to $MTA(C_2)$ only when the 0.2% Acc. error of each task occurs in different images. In the rest of the cases, $MTA(C_1) > MTA(C_2)$, while $MeanAcc(C_1) = MeanAcc(C_2)$. In other words, if we consider that 100 images were predicted by both algorithms, C2 will have inconsistencies in 40 of them, and in the best case C1 will produce only 20 completely misclassified images and all the rest will be correctly classified. Therefore, if we did not consider the data (images) and only looked at the global results, with C1 we would have less misclassified data when considering the coherence between the tasks predictions. Note that the MTA is a lower bound metric of the classical definition of the accuracy of multi-task learning (Mean Acc) i.e. improving MTA leads to improve the accuracy too. In other words, the maximum MTA value achievable is equal to the Mean Acc. When classifications for all tasks are correct for most of the samples: MTA is close to Mean Acc. However in the case that at least one of the tasks is misclassified for most of the samples: MTA tends to 0.

Furthermore, considering the human factor when analysing the results obtained by a multi-task classifier, MTA allows us to obtain a higher acceptance rate by the end users. To illustrate it, imagine a scenario where we should show the performance of our multi-task model to an end user having two tasks (T1 and T2). On the one hand, if our classifier is wrong on at least one task (MTA = 0.0, Mean Acc = 0.5), the user usually perceives the outputs as wrong. For instance, a sample with GT dish (T1) = "spaghetti carbonara" and food group (T2) = "pasta" is correctly predicted as dish (T1): "spaghetti carbonara" but misclassified as food group (T2): "meat"; it would be difficult that the end-user perceives the model as well-performing. On the other hand, if our classifier obtains coherent predictions among tasks, both T1 and T2 will be either both correct or both incorrect (MTA = 0.5, Mean Acc = 0.5). Thus, the user will perceive as correct at least the predictions that provide task coherence, providing a better user acceptance rate of the model.

Summarizing, MTA is a stricter measure of performance that opens room for new research distinguishing data where all classifiers have difficulties to learn from data where some of the classifiers achieve to learn, that could be an indicator for the quality of the data. Hence, providing more specific metrics with respect to the performance of tasks and classifiers (like MTA) will allow to study in the future the different kinds of errors present (epistemic, homoscedastic or heteroscedastic) and thus, ease the improvement of the model.

4.3. Experimental setup

We trained end-to-end a CNN architecture (see Fig. 3) using the proposed multi-loss function in Eq. (2) optimized with Adam. Note that the number and type of output layers of our model depend on the attributes considered for each dataset (MAFood-121, Vireo Food-172). In total, four multi-task configurations applying different σ and γ parameters were evaluated. Additionally, single models focused on a specific task were trained for comparative purpose. The models are named as follows:

1. *Single-task models*: These models trained independently for each task are considered as a baseline.
2. *MTL* ($\sigma_i = 1, \gamma = 0$): "Base" MTL model with the weights of tasks losses uniformly distributed.
3. *RMTL* ($\sigma_i = 1, \gamma = 1$): "Base" MTL model with the weights of tasks losses uniformly distributed and the proposed regularizing term.
4. *UMTL* ($\sigma_i = d, \gamma = 0$): MTL model with uncertainty-modeling weights.
5. *RUMTL* ($\sigma_i = d, \gamma = 1$): MTL model with uncertainty-modeling weights and the proposed regularizing term.

As for training, first all models were pre-trained on the ILSVRC dataset. Then, we re-trained the models during 100 epochs with a batch size of 20, and a learning rate of $2e-4$. In addition, we

applied a decay of 0.2 every 8 epochs. The training was done using Keras with Theano as backend.

Regarding the weighting of losses, in order to smooth out the difference in the magnitudes of the losses between the SL and ML tasks, we assign an initial weight of 1 for SL tasks and 0.1 for ML task in MAFood-121 and 1 for SL task and 0.02 for ML task in VIREO Food-172. Specifically for the models *UMTL* and *RUMTL*, the loss weights are updated dynamically considering the uncertainty obtained by minimizing the target function $L(W, \sigma_t, \gamma)$ in each epoch using Adam optimization. Note that the σ_t were initialized to 1.

4.4. Results

In order to evaluate the performance of our approach in MaFood-121 dataset, we trained four variants of our model by changing the σ and γ values. In addition, we trained a set of single-task models for each task as baseline models. When σ takes the value 1, it implies a naive weighted sum of losses. On the other hand, when $\sigma = d$, it implies the use of uncertainty weighted of the losses. As for γ, it can take the values 1 or 0, whether we apply the proposed regularizing term or not. The results obtained for all the models for each task are shown in Table 1. Different

Table 1
Results of the different variants of our model compared to the use of single-task models on MAFood-121 dataset. Boldface indicates the best result for the MTA metric.

	Dish	Cuisine	Categories			
	Acc	*Acc*	F_1	*Pre*	*Rec*	*MTA*
Single-task	82.50%	85.71%	82.98%	84.15%	81.83%	62.45%
MTL	83.73%	88.23%	83.74%	86.38%	81.26%	67.05%
R_MTL	83.76%	88.79%	83.81%	87.29%	80.59%	67.22%
R+MTL	82.85%	88.01%	84.09%	86.16%	82.11%	67.24%
RMTL	83.35%	88.86%	84.27%	86.57%	82.09%	67.65%
UMTL	83.47%	88.92%	84.55%	86.61%	82.58%	68.22%
RUMTL	83.82%	88.35%	85.02%	86.40%	83.69%	**68.85%**

Table 2
Results of our proposed model compared to the use of single-task models on VIREO-172 dataset. Boldface indicates the best result for the MTA metric.

	Dish	Ingredients			
	Acc	F_1	*Prec*	Rec	*MTA*
VGG [47]	80.41%	60.81%	–	–	–
ResNet-50	84.67%	76.62%	81.35%	72.40%	62.96%
Arch-D [47]	82.06%	67.17%	–	–	–
MTL	84.34%	70.79%	79.28%	63.94%	56.69%
RUMTL	85.19%	76.87%	81.30%	72.89%	**64.99%**

Fig. 3. Success (top 3) and failure (bottom) cases of RUMTL on MAFood-121.

Fig. 4. Success (top 3) and failure (bottom) cases of RUMTL on Vireo-172.

measures were used depending on the nature of the tasks. In the case of ML, we show the results in terms of *Pre*, *Rec* and F_1 score. However, these measures mean the same in the case of SL multi-class approaches [49]. Therefore instead, we show the results with overall accuracy (*Acc*) measure. In addition, we show the coherence of the results obtained among the individual tasks based on the MTA measure. The latter is applied on the multi-task models and also on all single-task models considering them as a unified model. In general terms, the variants of multi-task models were able to improve the performance on all tasks. Furthermore, according to the MTA metric, the application of the regularization term proposed together with the uncertainty weights allows us to jointly guide the learning of the tasks, with which we obtained better and more coherent results, achieving a total improvement of around 7% with respect to the single task models. Note that, the influence of the both terms of our proposed regularization can be easily seen from R_MTL and R_+MTL when we compare the results obtained with respect to MTL in the food categories task for precision and recall metrics. In the case of R_MTL, the precision increases, but the recall decreases, on the contrary, for R_+MTL the recall increases, but the precision decreases. Integrating both terms (*RMTL* model) we are able to balance their contributions achieving an improvement with respect to the *MTL* of 0.6% considering the MTA score. The benefit of the regularization also is evidenced when the uncertainty is included (*UMTL* vs *RUMTL*), in this case the performance is improved by 0.63%.

The results obtained with our proposed model RUMTL in VIREO Food-172 dataset can be seen in Table 2. In the same way to the results obtained in MaFood-121, our proposed model shows a better performance when it is evaluated for each task independently. Furthermore, despite the slight difference in the performance of the individual tasks of our proposal compared to single models based on ResNet-50 for each tasks, when we evaluate the joint classification (MTA metric), the result is about 2% better for our model RUMTL. Keep in mind that the MTA metric allows us to know what the behavior of the models is considering the performance achieved for all tasks in each image. Therefore, despite of single tasks are slightly worse, it is possible to have a large difference in MTA when the errors for each task are present in different images. This claims that if we want a system to simultaneously predict more than one task at once, RUMTL provides much better results than using single-task models.

We illustrate the results obtained for our proposed model on MAFood-121 (see Fig. 3) and VIREO Food-172 (see Fig. 4) datasets. In both cases, the good performance achieved (top 3 dishes) is observed on very diverse dishes. Furthermore, when our method fails, the results of the different tasks maintain a coherence (bottom image on MaFood-121) or are wrong but provide coherent predictions considering the image at hand (bottom image on VIREO Food-172). Finally, in Appendix B, we show the loss functions evolution and uncertainty obtained for each task during training of the RUMTL model on MAFood-121 (see Fig. B.1) and VIREO Food-172 (see Fig. B.2) datasets.

5. Conclusions

We proposed a new model for MTL that improves the coherence of the outputs compared to the single-task models by testing it on two food datasets. The improvement is achieved by modeling the uncertainty in the proposed MTL method, extending the proposal of Kendall [16] to allow SL and ML classification, as well as the incorporation of a MTL regularization term in the loss function. RUMTL allows an end-to-end training with automatic optimization of tasks weights. Furthermore, we publish a new dataset for multi-attribute food analysis and a new metric for measuring tasks coherence to serve as a basis for future works in MTL and food analysis fields.

Acknowledgement

This work was partially funded by TIN2015-66951-C2-1-R, 2017 SGR 1742, Nestore, 20141510 (La MaratoTV3) and CERCA Programme/Generalitat de Catalunya. E. Aguilar acknowledges the support of CONICYT Becas Chile and M. Bolaños acknowledges the support of an FPU fellowship (Ref. FPU15/01347). P. Radeva is partially supported by ICREA Academia 2014. We acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU.

Appendix A. Examples of images from MAFood121 dataset

Fig. A.1.

Fig. A.1. Examples of images from the MAFood-121 dataset divided in the 11 different cuisines considered.

Appendix B. Losses and task uncertainty

Figs. B.1 and B.2.

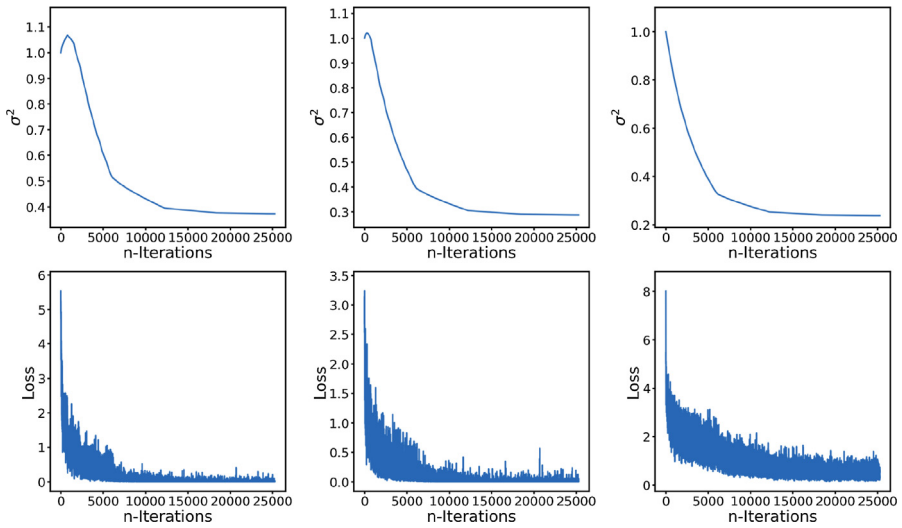

Fig. B.1. From left to right: loss values and task uncertainty (σ^2) for the dish, cuisine and food families tasks obtained during the training of the RUMTL model on MAFood-121.

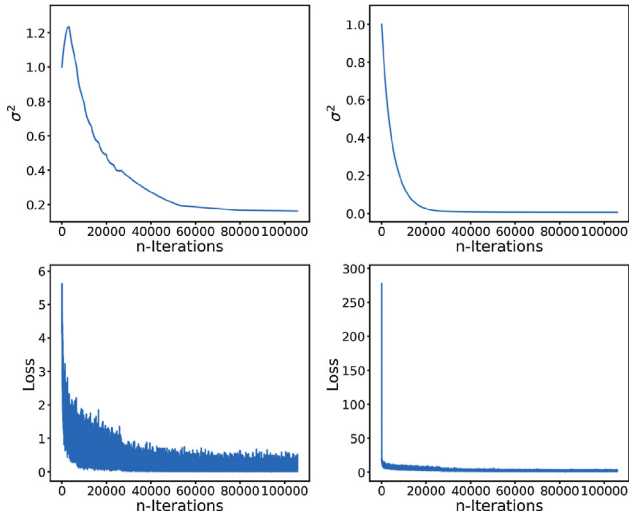

Fig. B.2. Loss values and task uncertainty (σ^2) for the dish (left) and ingredients (right) tasks obtained during the training of the RUMTL model on VIREO Food-172.

References

[1] A. Waxman, K.R. Norum, Why a global strategy on diet, physical activity and health? The growing burden of non-communicable diseases, Public Health Nutr. 7 (3) (2004) 381.

[2] E. Aguilar, M. Bolaños, P. Radeva, Exploring food detection using cnns, in: EUROCAST 2017, 2018, pp. 339–347.

[3] E. Aguilar, M. Bolaños, P. Radeva, Food recognition using fusion of classifiers based on cnns, in: International Conference on Image Analysis and Processing, Springer, 2017, pp. 213–224.

[4] N. Martinel, G.L. Foresti, C. Micheloni, Wide-slice residual networks for food recognition, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018, pp. 567–576.

[5] M. Bolaños, P. Radeva, Simultaneous food localization and recognition, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 3140–3145.

[6] J. Chen, C.-W. Ngo, Deep-based ingredient recognition for cooking recipe retrieval, in: Proceedings of the 2016 ACM on Multimedia Conference, ACM, 2016, pp. 32–41.

[7] X.-J. Zhang, Y.-F. Lu, S.-H. Zhang, Multi-task learning for food identification and analysis with deep convolutional neural networks, J. Comput. Sci. Technol. 31 (3) (2016) 489–500.

[8] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber, A. Torralba, Learning cross-modal embeddings for cooking recipes and food images, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[9] T. Ege, K. Yanai, Simultaneous estimation of food categories and calories with multi-task cnn, in: 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), IEEE, 2017, pp. 198–201.

[10] E. Aguilar, B. Remeseiro, M. Bolaños, P. Radeva, Grab, pay and eat: Semantic food detection for smart restaurants, IEEE Trans. Multimedia. https://doi.org/10.1109/TMM.2018.2831627.

[11] R. Dinic, M. Domhardt, S. Ginzinger, T. Stütz, Eatar tango: portion estimation on mobile devices with a depth sensor, in: Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services, ACM, 2017, p. 46.

[12] Y. Li, Y. Song, J. Luo, Improving pairwise ranking for multi-label image classification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[13] H. Yang, J.T. Zhou, J. Cai, Improving multi-label learning with missing labels by structured semantic correlations, in: European Conference on Computer Vision, Springer, 2016, pp. 835–851.

[14] W. Liu, I.W. Tsang, Large margin metric learning for multi-label prediction, AAAI, vol. 15, 2015, pp. 2800–2806.

[15] X. Li, F. Zhao, Y. Guo, Conditional restricted Boltzmann machines for multi-label learning with incomplete labels, in: Artificial Intelligence and Statistics, 2015, pp. 635–643.

[16] A. Kendall, Y. Gal, R. Cipolla, Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, arXiv preprint arXiv:1705.07115.

[17] M. Chen, K. Dhingra, W. Wu, L. Yang, R. Sukthankar, J. Yang, Pfid: Pittsburgh fast-food image dataset, in: 2009 16th IEEE International Conference on Image Processing (ICIP), IEEE, 2009, pp. 289–292.

[18] T. Joutou, K. Yanai, A food image recognition system with multiple kernel learning, in: 2009 16th IEEE International Conference on Image Processing (ICIP), IEEE, 2009, pp. 285–288.

[19] M. Bosch, F. Zhu, N. Khanna, C.J. Boushey, E.J. Delp, Combining global and local features for food identification in dietary assessment, in: 2011 18th IEEE International Conference on Image Processing (ICIP), IEEE, 2011, pp. 1789–1792.

[20] Y. Matsuda, H. Hoashi, K. Yanai, Recognition of multiple-food images by detecting candidate regions, in: 2012 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2012, pp. 25–30.

[21] Y. Kawano, K. Yanai, Real-time mobile food recognition system, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2013, pp. 1–7.

[22] S. Yang, M. Chen, D. Pomerleau, R. Sukthankar, Food recognition using statistics of pairwise local features, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 2249–2256.

[23] L. Bossard, M. Guillaumin, L. Van Gool, Food-101–mining discriminative components with random forests, in: European Conference on Computer Vision, Springer, 2014, pp. 446–461.

[24] Y. Kawano, K. Yanai, Automatic expansion of a food image dataset leveraging existing categories with domain adaptation., in: ECCV Workshops (3), 2014, pp. 3–17.

[25] K. Yanai, Y. Kawano, Food image recognition using deep convolutional network with pre-training and fine-tuning, in: 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), IEEE, 2015, pp. 1–6.

[26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vision 115 (3) (2015) 211–252.

[27] H. Hassannejad, G. Matrella, P. Ciampolini, I. De Munari, M. Mordonini, S. Cagnoni, Food image recognition using very deep convolutional networks, in: International Workshop on Multimedia Assisted Dietary Management, 2016, pp. 41–49.

[28] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, Y. Ma, Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment, in: International Conference on Smart Homes and Health Telematics, 2016, pp. 37–48.

[29] G.M. Farinella, D. Allegra, M. Moltisanti, F. Stanco, S. Battiato, Retrieval and classification of food images, Comput. Biol. Med. 77 (2016) 23–39.

[30] A. Singla, L. Yuan, T. Ebrahimi, Food/non-food image classification and food categorization using pre-trained googlenet model, in: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management, ACM, 2016, pp. 3–11.

[31] H. Wu, M. Merler, R. Uceda-Sosa, J.R. Smith, Learning to make better mistakes: semantics-aware visual food recognition, in: Proceedings of the 2016 ACM on Multimedia Conference, ACM, 2016, pp. 172–176.

[32] M. Bolaños, A. Ferrà, P. Radeva, Food ingredients recognition through multi-label learning, in: International Conference on Image Analysis and Processing, Springer, 2017, pp. 394–402.

[33] S. Sajadmanesh, S. Jafarzadeh, S.A. Ossia, H.R. Rabiee, H. Haddadi, Y. Mejova, M. Musolesi, E.D. Cristofaro, G. Stringhini, Kissing cuisines: exploring worldwide culinary habits on the web, in: Proceedings of the 26th International Conference on World Wide Web Companion, International World Wide Web Conferences Steering Committee, 2017, pp. 1013–1021.

[34] W. Min, S. Jiang, J. Sang, H. Wang, X. Liu, L. Herranz, Being a supercook: joint food attributes and multimodal content modeling for recipe retrieval and exploration, IEEE Trans. Multimedia 19 (5) (2017) 1100–1113.

[35] H. Su, T.-W. Lin, C.-T. Li, M.-K. Shan, J. Chang, Automatic recipe cuisine classification by ingredients, in: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, ACM, 2014, pp. 565–570.

[36] M.M. Zhang, Identifying the cuisine of a plate of food, University of California San Diego, Tech. Rep.

[37] S. Li, Z.-Q. Liu, A.B. Chan, Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 482–489.

[38] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, R. Feris, Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[39] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, J. Kim, Rotating your face using multi-task deep neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 676–684.

[40] X. Yin, X. Liu, Multi-task convolutional neural network for pose-invariant face recognition, IEEE Trans. Image Process.

[41] Z. Zhang, P. Luo, C.C. Loy, X. Tang, Facial landmark detection by deep multi-task learning, in: European Conference on Computer Vision, Springer, 2014, pp. 94–108.

[42] Y. Gao, Q. She, J. Ma, M. Zhao, W. Liu, A. Yuille, Nddr-cnn: layer-wise feature fusing in multi-task cnn by neural discriminative dimensionality reduction. corr abs/1801.0 (2018) (2018).

[43] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[44] A. Kendall, Y. Gal, What uncertainties do we need in bayesian deep learning for computer vision?, in: Advances in Neural Information Processing Systems, 2017, pp 5580–5590.

[45] M.-L. Zhang, Z.-H. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowled. Data Eng. 26 (8) (2014) 1819–1837.

[46] A. Meyers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, K.P. Murphy, Im2calories: towards an

automated mobile vision food diary, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1233–1241.

[47] J. Chen, C.-W. Ngo, Deep-based ingredient recognition for cooking recipe retrieval, in: Proceedings of the 2016 ACM on Multimedia Conference, ACM, 2016, pp. 32–41.

[48] C. Güngör, F. Baltaci, A. Erdem, E. Erdem, Turkish cuisine: a benchmark dataset with turkish meals for food recognition, in: 2017 25th Signal Processing and Communications Applications Conference (SIU), 2017, pp. 1–4, https://doi.org/10.1109/SIU.2017.7960494.

[49] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (Oct) (2011) 2825–2830.

Chapter 5

Results

In this chapter you can find a summary that contains the most relevant results and contributions obtained from the variety of topics treated in this thesis.

The results are mainly divided in egocentric vision and food recognition. For seeing in-depth results on each of the topics please, refer to the attached papers in the previous chapters.

5.1 Egocentric Vision

You can find in-depth results of each of the sub-topics of this section in Chapter 3.

5.1.1 Segmentation

The topic of segmentation in egocentric vision usually refers to video or event segmentation, where the purpose is to divide all the photo streams acquired along the day of the user into accurate and semantically meaningful events or situations.

The most up-to-date results in this section where obtained in (Dimiccoli et al., 2017). In this paper we presented a novel technique that, based on both global image CNN features and semantic features together with a concept detector, generated a temporal photo streams segmentation.

Table 5.1 compares our results to the other state of the art techniques. We can see that we managed to outperform them by more than a 12% on two publicly available datasets. In Fig. 5.1 you can see an example of the qualitative results obtained.

Furthermore, we presented a novel dataset (EDUB-Seg) that was made publicly available.

5.1.2 Object Discovery

This field consists in, from the one side, recognizing objects that appear in the images acquired by the user and, at the same time, identifying relevant or frequently appearing objects in his/her daily life in different parts of the stream.

In (Bolaños and Radeva, 2015) we proposed an iterative method composed of four steps that allowed us to detect and recognize newly appearing relevant objects on the photo-streams. The main steps of the method are: object candidates generation based on an object detector; candidates characterization based on CNN features extraction; false objects filtering based on an SVM; and instances classification based on clustering and one-class SVMs.

We proved that our proposal achieved substantially better results than the other state of the art method for egocentric data on both F1-score (see Table 5.2) and in the total number of unique objects that it was able to discover (see examples of objects in Fig. 5.2).

	AIHS	Huji EgoSeg	EDUB-Seg P1	EDUB-Seg
(Bolanos, Garolera, and Radeva, 2014)	0.66		0.34	
(Lee and Grauman, 2015)	0.60		0.37	0.50
(Talavera et al., 2015)	**0.79**		0.55	
ADW-ImaggaD	0.35	0.59	0.55	0.36
AC-ImaggaD	0.72	**0.88**	0.53	0.61
(Dimiccoli et al., 2017)	0.78	**0.88**	**0.69**	**0.66**

TABLE 5.1: Average F1-score results of the state of the art works on the publicly available egocentric datasets. AC stands for Agglomerative Clustering, ADW for ADWIN and ImaggaD is our proposal for semantic features, where D stands for Density Estimation. Our proposal corresponds to the last row of the table. The dataset AIHS was presented in (Jojic, Perina, and Murino, 2010), and Huji EgoSeg in (Poleg, Arora, and Peleg, 2014). Huji EgoSeg was modified to simulate a low-temporal resolution similar to the other datasets used for comparison. EDUB-Seg P1 stands for EDUB-Seg part 1 only.

FIGURE 5.1: A set of events as well as the semantic concepts obtained are presented for a day of the evaluated user.

	(Lee and Grauman, 2011)	CNN + Refill	CNN + Refill + Filter
MSRC	0.121	**0.431**	0.410
PASCAL	0.002	0.145	**0.179**
EDUB	0.072	**0.285**	0.250
Average	0.065	0.287	0.280

TABLE 5.2: F1-score comparison for the three datasets, the state of the art (Lee and Grauman, 2011) and our best test settings from (Bolaños and Radeva, 2015) (CNN + Refill and CNN + Refill + Filter).

Furthermore, we presented a novel dataset (EDUB) that was made publicly available.

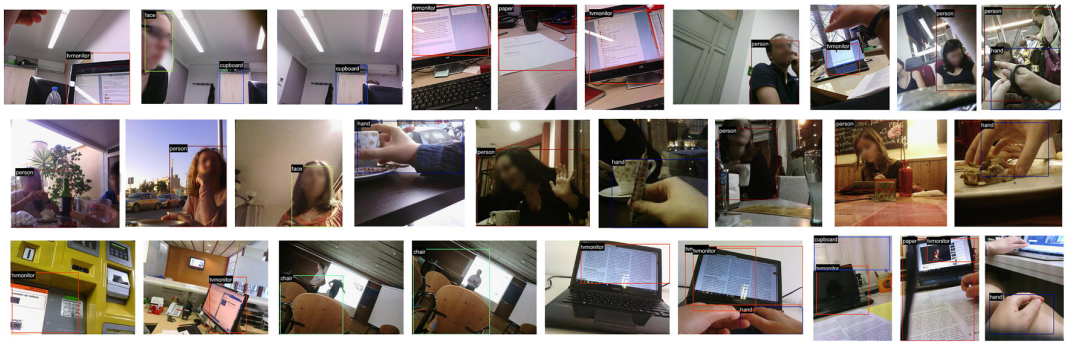

FIGURE 5.2: Examples of discovered objects by our proposed ego-object discovery technique.

5.1.3 Visual Summarization

The best performing method on visual summarization was proposed in (Lidon et al., 2017), where a summarization based on semantics was presented. The method is capable of generating a keyframe-based summarization for egocentric photo-streams. It consists of different preliminary modules, each of which is in charge of detecting a relevant characteristic from the images, as well as a final fusion and ranking-based methods that choose the most relevant pictures for the final summary. The set of relevant characteristics assessed are:

- Informativeness (based on a CNN for detecting informative vs non-informative images).
- Saliency (based on detecting salient regions).
- Objectness (based on an object detector).
- Face Detection (based on a CNN for face detection).

The final results where compared against a uniform sampling baseline as well as the ground truth by designing blind taste tests with volunteers that had to give their opining about which was the best summary for each of the photo-streams. Table 5.3 shows the results of the comparison.

Uniform Sampling	**(Lidon et al., 2017)**	Ground Truth
3.99	4.57	4.94

TABLE 5.3: Mean opinion score of the blind taste test comparing the three keyframe-based summaries.

5.1.4 Textual Description

The task of generating meaningful and grammatically correct sentences in natural language from a set of videos is a complex problem. Added to the fact that egocentric images have a considerably higher degree of complexity than regular images means that the task at hand is very challenging.

In order to solve the problem in (Bolaños et al., 2018) we proposed a multimodal method that combines CNN features, attention mechanisms and LSTMs, and at the same time integrates information from the current as well as from previous events in order to better generalize and provide more meaningful sentences (see examples in Fig. 5.3).

Numerically comparing our proposal to the state of the art methods we managed to outperform all of them on the three most widely extended metrics for textual description generation (BLEU, METEOR and CIDEr). See Table 5.5 with the results.

Furthermore, we presented a novel dataset (EDUB-SegDesc) that was made publicly available.

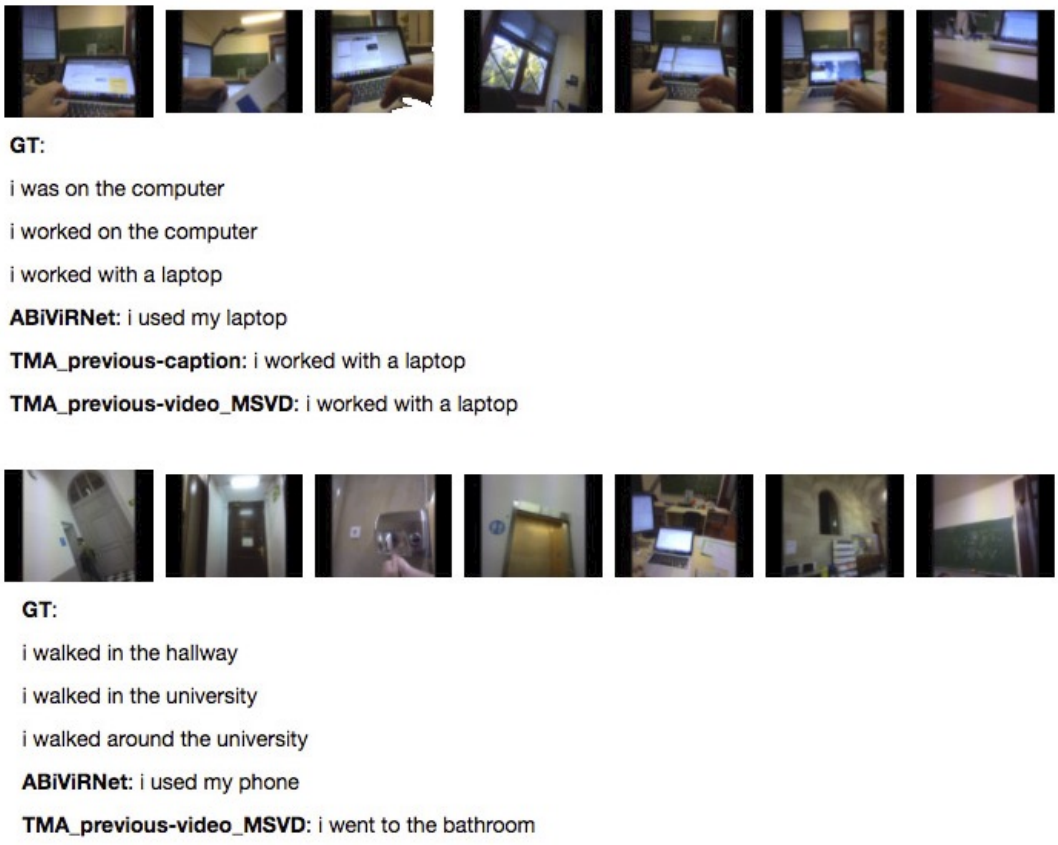

FIGURE 5.3: Examples of events and sentences generated by our method (TMA) compared to the state of the art. ABiViRNet corresponds to the method in (Peris et al., 2016).

	BLEU-4	METEOR	CIDEr
(Yao et al., 2015)	28.1	20.8	0.88
(Song et al., 2017)	25.6	20.8	0.88
(Peris et al., 2016)	29.6	20.3	0.79
(Goel and Naik, 2016)	27.9	21.4	0.99
(Bolaños et al., 2018)	**31.9**	**22.1**	**1.07**

TABLE 5.4: Test set results on the EDUB-SegDesc dataset for variations of our TMA model compared to the state of the art, which do not consider information from previous events. BLEU and METEOR metrics are given in percentage.

5.2 Food Recognition

You can find in-depth results of each of the sub-topics of this section in Chapter 4.

5.2.1 Detection, Localization and Recognition

The best results obtained in this section have to be complemented by two different papers from the ones presented in this thesis.

The first paper, oriented to simultaneous food localization and recognition (Bolanos and Radeva, 2016), proposes a two-stepped process. In the first step, an activation map based on a CNN with global average pooling is used for detecting regions on the image that contain food. The second one, based on a CNN trained on food recognition is in charge of separately classifying each of the regions previously extracted. The proposed method was applied on both egocentric images and on regular images (see some resulting examples in Fig. 5.4).

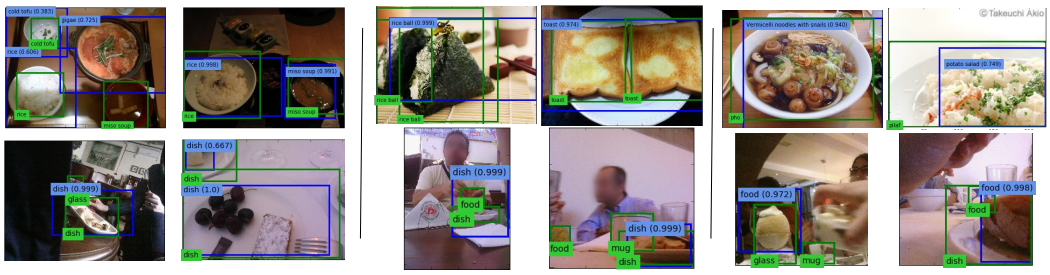

FIGURE 5.4: Results obtained on the proposed simultanous food localization and recognition method on both regular (top) and egocentric images (bottom).

The second paper, aiming at ingredients recognition from images of prepared dishes (Bolaños, Ferrà, and Radeva, 2017), proposes a multi-label framework for recognizing multiple classes corresponding to ingredients from a single image (see samples of the provided results in Fig. 5.5). This work, which targets the ingredients recognition problem for both visible and invisible ingredients works on improving the results with respect to previous state of the art proposals that were focused on visible ingredients only.

Furthermore, we presented and made public a new dataset (Recipes5k) that contained five thousand images together with their recipes as well as Ingredients101, which contains the recipes for each of the 101 classes present in the public dataset Food101 (Bossard, Guillaumin, and Van Gool, 2014).

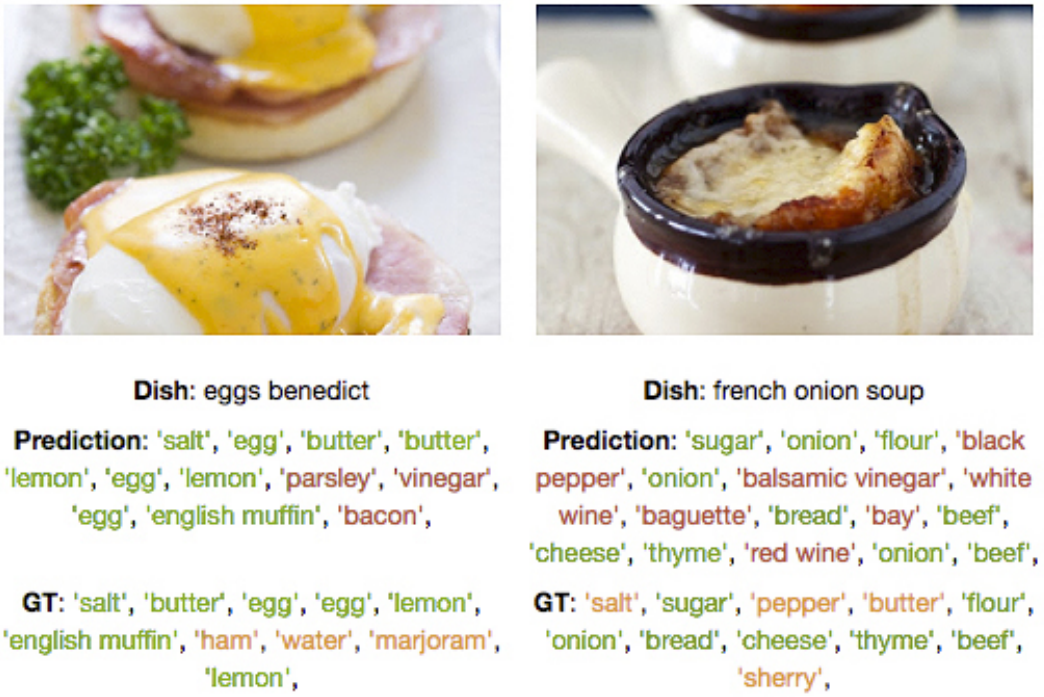

FIGURE 5.5: Results obtained by our Ingredients Recognition method on the Recipes5k dataset. True Positives are shown in green, False Positives in red and False Negatives in orange.

5.2.2 Multi-task and Fusion

The most relevant results regarding the advanced deep learning techniques section are the ones presented in our multi-task food analysis approach (Aguilar, Bolaños, and Radeva, 2019). In this paper we developed a novel methodology that could take profit of the existent correlations between food tasks that are usually taken as independent problems (e.g. food recognition, ingredients recognition, food groups detection, etc.). With this purpose in mind, we proposed first, a loss for weighing the homoscedastic (task-dependant) uncertainty (Kendall, Gal, and Cipolla, 2018) that can be inferred from the predictions generated by the model; and second, a multi-attribute dataset for food analysis.

The quantitative results of the proposed method (RUMTL) can be seen in Table **??**, and the qualitative results are displayed in Fig. 5.6. In both cases the results correspond to the Vireo 172 dataset (Chen and Ngo, 2016).

	Dish Acc.	Ingredients F1-score	MTA
VGG (Chen and Ngo, 2016)	80.41	60.81	-
Arch-D (Chen and Ngo, 2016)	82.06	67.17	-
ResNet 50	84.67	76.62	62.96
(Aguilar, Bolaños, and Radeva, 2019)	**85.19**	**76.87**	**64.99**

TABLE 5.5: Results on the dataset Vireo-172 for different state of the art single-task approaches compared to our approach. MTA corresponds to our proposed metric, which penalizes the incoherence of the predictions applied on different tasks.

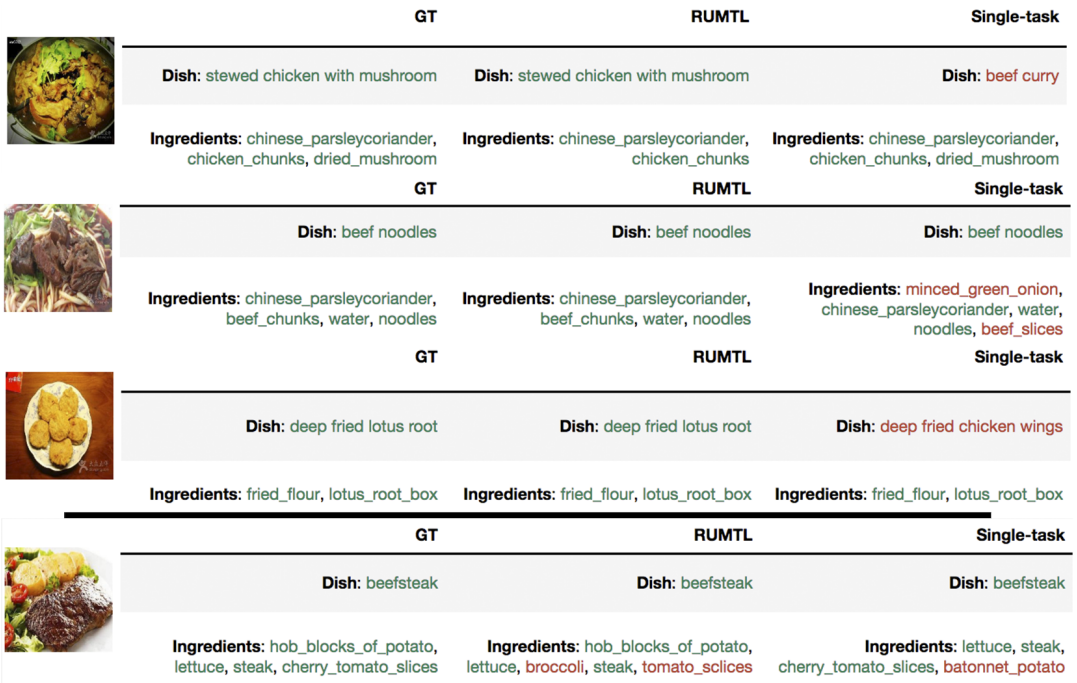

FIGURE 5.6: Comparison of the results of our proposed model (RUMTL) compared to using separate single-task models on the Vireo-172 dataset.

5.2.3 Food Recognition in Restaurants

The last but not the least topic in which we have made a relevant contribution is food recognition in restaurants. The results presented in (Aguilar et al., 2018), which applies food recognition on food trays from self-service canteens, have set the basis for the food recognition self-checkout system that we developed at the start-up LogMeal Food Recognition With a Snap! [1].

In this paper we proposed a semantic food detection method based on three steps, the two initial ones are applied in parallel on the input image and consist in 1) semantic segmentation and 2) food detection and recognition. The last step consists in merging the results from the two previous ones by removing false positive detections.

[1] www.logmeal.es

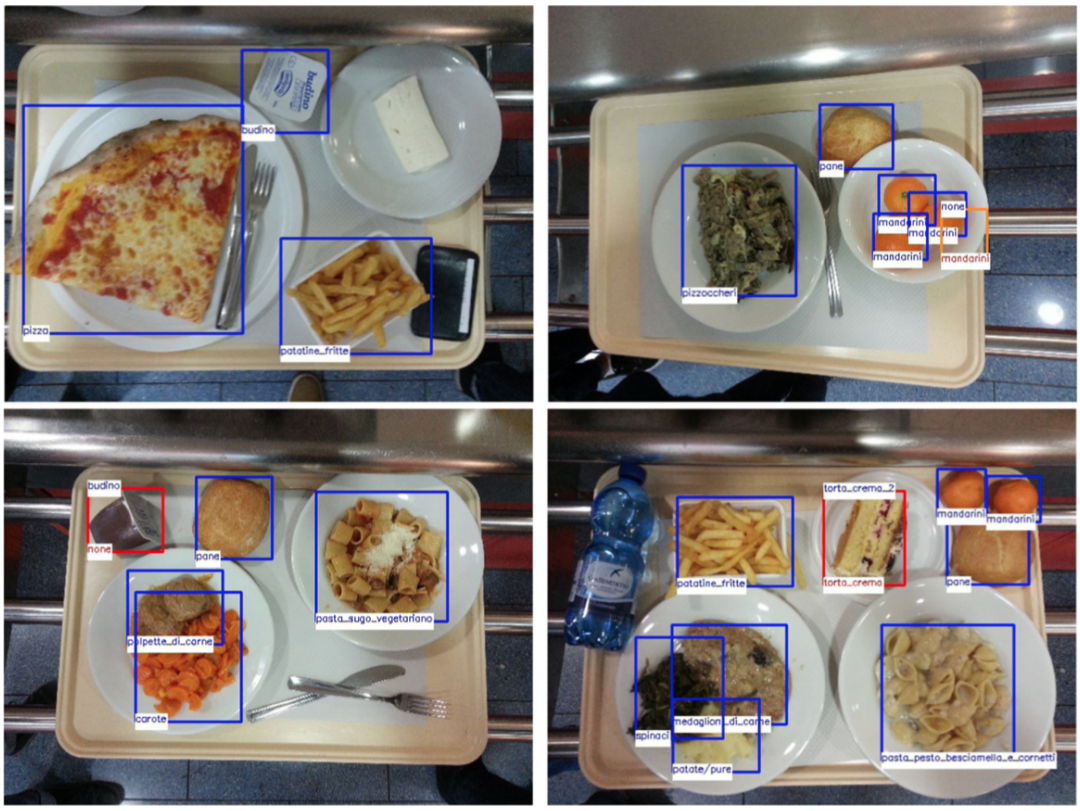

FIGURE 5.7: Examples of detections of our method on a set of food tray images.

Chapter 6

Conclusions and Thoughts for the Future

We have arrived at the end of this thesis, and several conclusions can be drawn from multiple perspectives. At the same time, I would like to provide my point of view about possible future lines that could be explored in order to advance further in the main topics treated in this thesis: Deep Multimodal Learning, Egocentric Vision and Food Recognition.

From the deep and multimodal learning perspective, we have explored multiple points of view and multiple ways to combine several textual and visual data modalities providing from varied sources (e.g. egocentric photo streams, food images, textual descriptions for videos or images, etc.). Complex semantic models have been proposed with the purpose of inferring highly performing outputs that have the ability to understand high-level concepts providing from different data sources.

With the purpose of further improving the understanding and learning capabilities of multimodal models, the appearance of both datasets and techniques with richer and high-level representation capabilities could boost the results in the field. Researchers should aim at incorporating time-related information as we did in (Bolaños et al., 2018) as well as data samples with multiple data modalities from different tasks. Also, following the same line of thought, the development of easier, and possibly high-scale, labeling collaborative techniques could increase the quantity and quality of datasets for multimodal and deep learning.

From the egocentric vision perspective, we worked towards boosting the representativeness capabilities and semantics from many angles (e.g. object discovery, segmentation, summarization, people detection, etc.). The generation of a semantically-rich photo streams representation proved to be relevant for the goal of summarizing and semantically indexing the data from the users in order to build powerful tools for patients with dementia.

In my opinion two main issues need to be tackled in the future for the field to finally reach market-level applicability. The first one is the appearance of interactive and semantic-based browsing tools possibly based on voice commands or other simple techniques for managing huge quantities of self-reported data. Secondly, from an ethics perspective there is still too many people that is reticent for sharing his/her data, even anonymously to algorithms that only infer and predict trends from their data. More powerful and useful tools for improving the quality of life of the population will be possible when (if) people change their perspective on ethics. Similarly, the same line of thought applies to other completely unrelated machine learning fields that work with data in which potential personal data appears.

Finally, from the food recognition perspective, topics like food detection or recognition, ingredients recognition, food groups/categories recognition, cuisines recognition or food localization, among others, have been broadly explored in this

thesis. Furthermore, also as a result of all this work, a start-up dedicated to food recognition offering multiple services for food analysis is being launched. This means that the basis for applying food analysis in a large scale has been set. The applicability of these algorithms reaches several fields and markets at many levels of depth, from creating automated personal food logs for people with nutritional problems to specific image-based applications for restaurants of all kinds.

As future lines, the merge of both egocentric vision and food recognition could enable the complete automation of food logs, where the user would only need to normally live his/her life and a food diary would be automatically created with the complete statistics of what he/she ate. These completely personalized logs would enable a new world of recommender-based systems that could be based not only on the diet of the person, but also on nutrigenetic traits.

Appendix A

Research Papers and Contributions

In this appendix you can find a summary of the list of contributions that were generated as an outcome of this thesis. The contributions are divided in different groups and subgroups (e.g. Commercial Solutions, Journals, Conference Proceedings, Participation in Funded Projects, etc.).

The complete set of papers that are presented accumulate a total of more than 560 citations at the date of presentation of this thesis. Throughout the presentation of the list of contributions, the citations accumulated so far by each of the most relevant articles will also be specified.

A.1 Journal Papers

- Marc Bolanos, Mariella Dimiccoli, and Petia Radeva (2016). "Toward storytelling from visual lifelogging: An overview". In: *IEEE Transactions on Human-Machine Systems* 47.1, pp. 77–90 (**Google Scholar citations: 103**)
- Mariella Dimiccoli, Marc Bolaños, Estefania Talavera, Maedeh Aghaei, Stavri G Nikolov, and Petia Radeva (2017). "Sr-clustering: Semantic regularized clustering for egocentric photo streams segmentation". In: *Computer Vision and Image Understanding* 155, pp. 55–69 (**Google Scholar citations: 41**)
- Eduardo Aguilar, Beatriz Remeseiro, Marc Bolaños, and Petia Radeva (2018). "Grab, Pay, and Eat: Semantic Food Detection for Smart Restaurants". In: *IEEE Transactions on Multimedia* 20.12, pp. 3266–3275 (**Google Scholar citations: 36**)
- Marc Bolaños, Álvaro Peris, Francisco Casacuberta, Sergi Soler, and Petia Radeva (2018). "Egocentric video description based on temporally-linked sequences". In: *Journal of Visual Communication and Image Representation* 50, pp. 205–216
- Eduardo Aguilar, Marc Bolaños, and Petia Radeva (2019). "Regularized uncertainty-based multi-task learning model for food analysis". In: *Journal of Visual Communication and Image Representation*. Vol. 60, pp. 360–370

A.2 Conference Proceedings

- Marc Bolanos, Maite Garolera, and Petia Radeva (2014). "Video segmentation of life-logging videos". In: *International Conference on Articulated Motion and Deformable Objects*. Springer, pp. 1–9
- Marc Bolaños, Maite Garolera, and Petia Radeva (2015). "Object discovery using CNN features in egocentric videos". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 67–74

- Estefania Talavera, Mariella Dimiccoli, Marc Bolaños, Maedeh Aghaei, and Petia Radeva (2015). "R-clustering for egocentric video segmentation". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 327–336 (**Google Scholar citations: 28**)
- Álvaro Peris, Marc Bolaños, Petia Radeva, and Francisco Casacuberta (2016). "Video description using bidirectional recurrent neural networks". In: *International Conference on Artificial Neural Networks*. Springer, pp. 3–11 (**Google Scholar citations: 24**)
- Pedro Herruzo, Marc Bolaños, and Petia Radeva (2016). "Can a CNN recognize Catalan diet?" In: *AIP Conference Proceedings*. Vol. 1773. 1. AIP Publishing, p. 020002
- Marc Bolanos and Petia Radeva (2016). "Simultaneous food localization and recognition". In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, pp. 3140–3145 (**Google Scholar citations: 51**)
- Marc Bolaños, Álvaro Peris, Francisco Casacuberta, and Petia Radeva (2017). "VIBIKNet: Visual bidirectional kernelized network for visual question answering". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 372–380
- Eduardo Aguilar, Marc Bolanos, and Petia Radeva (2017). "Exploring food detection using CNNs". In: *International Conference on Computer Aided Systems Theory*. Springer, pp. 339–347
- Eduardo Aguilar, Marc Bolaños, and Petia Radeva (2017). "Food recognition using fusion of classifiers based on cnns". In: *International Conference on Image Analysis and Processing*. Springer, pp. 213–224 (**Google Scholar citations: 30**)
- Eduardo Aguilar, Bhalaji Nagarajan, Rupali Khatun, Marc Bolaños, and Petia Radeva (2020). "Uncertainty Modeling and Deep Learning Applied to Food Image Analysis". In: *Proc. of the 13th Int. Joint Conf. on Biomedical Engineering Systems and Technologies - Volume 3 BIOINFORMATICS: BIOSTEC*, pp. 9–16

A.3 Workshop Proceedings

- Marc Bolaños, Maite Garolera, and Petia Radeva (2013). "Active labeling application applied to food-related object recognition". In: *Proceedings of the 5th international workshop on Multimedia for cooking & eating activities*. ACM, pp. 45–50
- Marc Bolanos, Ricard Mestre, Estefanía Talavera, Xavier Giró-i Nieto, and Petia Radeva (2015). "Visual summary of egocentric photostreams by representative keyframes". In: *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, pp. 1–6 (**Google Scholar citations: 35**)
- Aniol Lidon, Marc Bolaños, Mariella Dimiccoli, Petia Radeva, Maite Garolera, and Xavier Giro-i Nieto (2017). "Semantic summarization of egocentric photo stream events". In: *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*. ACM, pp. 3–11

- Gabriel Oliveira-Barra, Marc Bolaños, Estefania Talavera, Adrián Dueñas, Olga Gelonch, and Maite Garolera (2017). "Serious Games Application for Memory Training Using Egocentric Images". In: *International Conference on Image Analysis and Processing*. Springer, pp. 120–130

- Marc Bolaños, Aina Ferrà, and Petia Radeva (2017). "Food ingredients recognition through multi-label learning". In: *International Conference on Image Analysis and Processing*. Springer, pp. 394–402 (**Google Scholar citations: 25**)

A.4 Book Chapters

- Gabriel Oliveira-Barra, Marc Bolaños, Estefania Talavera, Olga Gelonch, Maite Garolera, and Petia Radeva (2019). "Lifelog retrieval for memory stimulation of people with memory impairment". In: *Multimodal Behavior Analysis in the Wild*. Elsevier, pp. 135–158

A.5 Participation in Challenges

- Gabriel de Oliveira Barra, Alejandro Cartas Ayala, Marc Bolaños, Mariella Dimiccoli, Xavier Giró Nieto, and Petia Radeva (2016). "Lemore: A lifelog engine for moments retrieval at the ntcir-lifelog lsat task". In: *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*

- Aniol Lidon, Xavier Giró Nieto, Marc Bolaños, Petia Radeva, Markus Seidl, and Matthias Zeppelzauer (2015). "UPC-UB-STP@ MediaEval 2015 diversity task: iterative reranking of relevant images". In: *MediaEval 2015 Multimedia Benchmark Workshop*. CEUR-WS. org

- Marc Bolaños, Álvaro Peris, Francisco Casacuberta, and Petia Radeva (2017). "VIBIKNet: Visual bidirectional kernelized network for visual question answering". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 372–380

A.6 Commercial Solutions

Out of the most relevant outcomes of this thesis we must not forget about the commercial contributions. More specifically, LogMeal[1], a start-up to be, devoted to food recognition.

The company currently offers two distinguished solutions for the B2B market. The first one, LogMeal API is a set of services and cloud-based algorithms that offer a semantic analysis at different levels of specificity: Food Types detection, Food Groups recognition, Single and Several-dishes recognition, Ingredients detection and Nutritional Information analysis.

The second one, called LogMeal SmartTray provides a hardware solution that uses a set of cloud-based algorithms for localizing and recognizing the food items present on a food tray. This solution also provides a self-checkout system for self-service restaurants.

[1] https://www.logmeal.es

A.7 Participation in Funded Projects

- Private Project with the Company Pulso Knowledge in Health "D-Coach" at Universitat de Barcelona (October 2019 – September 2021).
- Private Project with the Research Center Eurecat "CarpeDiem" at Universitat de Barcelona (May 2019 – April 2020).
- European Project EIT Health "Validithi: Validation of an Innovative Dietary Intake Tool for Healthcare Implementation" at Universitat de Barcelona (January 2019 – ongoing).
- Private Project with the Company Pulso Knowledge in Health "Diacare" at Universitat de Barcelona (December 2018 – November 2019).
- Private Project with the Company Serunion "Self-checkout System in Catering Restaurants" at Universitat de Barcelona (May 2018 – May 2019).
- European Project "Nestore: Novel Empowering Solutions and Technologies for Older people to Retain Everyday life activities." at Universitat de Barcelona (September 2017 – September 2020).
- La Marató de TV3 2014: "A Cognitive training tool based on life-logging in Mild Cognitive Impairment (Re-memory)" (2015 - 2018).
- "Revisiting representation models for visual recognition: Objects and Events", TIN2012-38187-C03-01 (2015).
- European Project "Real Time monitoring of SEA contaminants by an autonomous lab-on-a-chip biosensor (SEA-on-a-CHIP)" (2014 - 2016).
- "Evaluation of Intestinal Motility by Endoluminal Image Analysis", Given Imaging, Israel (2013).

A.8 Public Datasets

- EDUB-Obj[2]
- EDUB-Seg[3]
- EgocentricFood[4]
- EDUB-SegDesc[5]
- Recipes 5k[6]
- Ingredients 101[7]

[2]http://www.ub.edu/cvub/edub-obj/
[3]http://www.ub.edu/cvub/egocentric-dataset-of-the-university-of-barcelona-segmentation-edub-seg//
[4]http://www.ub.edu/cvub/egocentricfood/
[5]http://www.ub.edu/cvub/edub-segdesc/
[6]http://www.ub.edu/cvub/recipes5k/
[7]http://www.ub.edu/cvub/ingredients101/

A.9 Other contributions

- Marc Bolaños and Petia Radeva (2015). "Ego-object discovery". In: *arXiv preprint arXiv:1504.01639*
- Petia Radeva, Marc Bolaños, and Estefania Talavera (2018). "Tutorial: Deep Learning and Applications to Activity Recognition from Egocentric Photostreams". In:

During the development of this thesis most of the deep learning-based models and libraries have been developed under the Keras framework and have been made publicly available. The most relevant libraries developed during the writing of this thesis for the machine learning community are the following ones:

- MarcBS/Keras[8]: Fork of Keras library with new specific layers oriented to multimodal learning. (**GitHub stars: 224**)
- Multimodal Keras Wrapper (MWK)[9]. Wrapper for Keras with support to easy multimodal data and models loading and handling. Offers a higher level of abstraction with respect to base keras. (**GitHub stars: 26**)

[8]https://github.com/MarcBS/keras
[9]https://github.com/MarcBS/multimodal_keras_wrapper

Bibliography

Abu-El-Haija, Sami et al. (2016). "Youtube-8m: A large-scale video classification benchmark". In: *arXiv preprint arXiv:1609.08675*.

Aghaei, Maedeh, Mariella Dimiccoli, and Petia Radeva (2016). "Multi-face tracking by extended bag-of-tracklets in egocentric photo-streams". In: *Computer Vision and Image Understanding* 149, pp. 146–156.

Aguilar, Eduardo, Marc Bolanos, and Petia Radeva (2017). "Exploring food detection using CNNs". In: *International Conference on Computer Aided Systems Theory*. Springer, pp. 339–347.

Aguilar, Eduardo, Marc Bolaños, and Petia Radeva (2017). "Food recognition using fusion of classifiers based on cnns". In: *International Conference on Image Analysis and Processing*. Springer, pp. 213–224.

— (2019). "Regularized uncertainty-based multi-task learning model for food analysis". In: *Journal of Visual Communication and Image Representation*. Vol. 60, pp. 360–370.

Aguilar, Eduardo et al. (2018). "Grab, Pay, and Eat: Semantic Food Detection for Smart Restaurants". In: *IEEE Transactions on Multimedia* 20.12, pp. 3266–3275.

Aguilar, Eduardo et al. (2020). "Uncertainty Modeling and Deep Learning Applied to Food Image Analysis". In: *Proc. of the 13th Int. Joint Conf. on Biomedical Engineering Systems and Technologies - Volume 3 BIOINFORMATICS: BIOSTEC*, pp. 9–16.

Antol, Stanislaw et al. (2015). "Vqa: Visual question answering". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433.

Bolanos, Marc, Mariella Dimiccoli, and Petia Radeva (2016). "Toward storytelling from visual lifelogging: An overview". In: *IEEE Transactions on Human-Machine Systems* 47.1, pp. 77–90.

Bolaños, Marc, Aina Ferrà, and Petia Radeva (2017). "Food ingredients recognition through multi-label learning". In: *International Conference on Image Analysis and Processing*. Springer, pp. 394–402.

Bolaños, Marc, Maite Garolera, and Petia Radeva (2013). "Active labeling application applied to food-related object recognition". In: *Proceedings of the 5th international workshop on Multimedia for cooking & eating activities*. ACM, pp. 45–50.

Bolanos, Marc, Maite Garolera, and Petia Radeva (2014). "Video segmentation of lifelogging videos". In: *International Conference on Articulated Motion and Deformable Objects*. Springer, pp. 1–9.

Bolaños, Marc, Maite Garolera, and Petia Radeva (2015). "Object discovery using CNN features in egocentric videos". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 67–74.

Bolaños, Marc and Petia Radeva (2015). "Ego-object discovery". In: *arXiv preprint arXiv:1504.01639*.

Bolanos, Marc and Petia Radeva (2016). "Simultaneous food localization and recognition". In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, pp. 3140–3145.

Bolaños, Marc, Marc Valdivia, and Petia Radeva (2018). "Where and What Am I Eating? Image-Based Food Menu Recognition". In: *European Conference on Computer Vision*. Springer, pp. 590–605.

Bolanos, Marc et al. (2015). "Visual summary of egocentric photostreams by representative keyframes". In: *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, pp. 1–6.

Bolaños, Marc et al. (2017). "VIBIKNet: Visual bidirectional kernelized network for visual question answering". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 372–380.

Bolaños, Marc et al. (2018). "Egocentric video description based on temporally-linked sequences". In: *Journal of Visual Communication and Image Representation* 50, pp. 205–216.

Bossard, Lukas, Matthieu Guillaumin, and Luc Van Gool (2014). "Food-101–mining discriminative components with random forests". In: *European Conference on Computer Vision*. Springer, pp. 446–461.

Cartas, Alejandro et al. (2018). "Batch-based activity recognition from egocentric photo-streams revisited". In: *Pattern Analysis and Applications* 21.4, pp. 953–965.

Chen, Jingjing and Chong-Wah Ngo (2016). "Deep-based ingredient recognition for cooking recipe retrieval". In: *Proceedings of the 24th ACM international conference on Multimedia*, pp. 32–41.

Choudhry, Rohit and Kumkum Garg (2008). "A hybrid machine learning system for stock market forecasting". In: *World Academy of Science, Engineering and Technology* 39.3, pp. 315–318.

Ciocca, Gianluigi, Paolo Napoletano, and Raimondo Schettini (2016). "Food recognition: a new dataset, experiments, and results". In: *IEEE journal of biomedical and health informatics* 21.3, pp. 588–598.

Deng, Jia et al. (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Dimiccoli, Mariella et al. (2017). "Sr-clustering: Semantic regularized clustering for egocentric photo streams segmentation". In: *Computer Vision and Image Understanding* 155, pp. 55–69.

Doherty, Aiden R et al. (2012). "Experiences of aiding autobiographical memory using the SenseCam". In: *Human–Computer Interaction* 27.1-2, pp. 151–174.

Ege, Takumi and Keiji Yanai (2017). "Simultaneous estimation of food categories and calories with multi-task CNN". In: *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*. IEEE, pp. 198–201.

Fukushima, Kunihiko and Sei Miyake (1982). "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition". In: *Competition and cooperation in neural nets*. Springer, pp. 267–285.

Gelonch, Olga et al. (2020). "The Effects of Exposure to Recent Autobiographical Events on Declarative Memory in Amnestic Mild Cognitive Impairment: A Preliminary Pilot Study". In: *Current Alzheimer Research* 17.2, pp. 158–167.

Goel, Kratarth and Juhi Naik (2016). "DeepSeek: A video captioning tool for making videos searchable". In:

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Herruzo, Pedro, Marc Bolaños, and Petia Radeva (2016). "Can a CNN recognize Catalan diet?" In: *AIP Conference Proceedings*. Vol. 1773. 1. AIP Publishing, p. 020002.

Herruzo, Pedro et al. (2017). "Analyzing first-person stories based on socializing, eating and sedentary patterns". In: *International Conference on Image Analysis and Processing*. Springer, pp. 109–119.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Hodges, Steve et al. (2006). "SenseCam: A retrospective memory aid". In: *International conference on ubiquitous computing*. Springer, pp. 177–193.

Hsueh, Pei-Yun, Prem Melville, and Vikas Sindhwani (2009). "Data quality from crowdsourcing: a study of annotation selection criteria". In: *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pp. 27–35.

Jojic, N., A. Perina, and V. Murino (2010). "Structural epitome: a way to summarize one's visual experience". In: pp. 1027–1035.

Kendall, Alex, Yarin Gal, and Roberto Cipolla (2018). "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7482–7491.

Krasin, Ivan et al. (2016). "OpenImages: A public dataset for large-scale multi-label and multi-class image classification." In: *Dataset available from https://github.com/openimages*.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*, pp. 1097–1105.

LeCun, Yann et al. (1989). "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4, pp. 541–551.

Lee, Matthew L and Anind K Dey (2008). "Lifelogging memory appliance for people with episodic memory impairment". In: *Proceedings of the 10th international conference on Ubiquitous computing*, pp. 44–53.

Lee, YJ. and K. Grauman (2015). "Predicting Important Objects for Egocentric Video Summarization". In: *International Journal of Computer Vision* 114.1, pp. 38–55. DOI: 10.1007/s11263-014-0794-5. URL: http://dx.doi.org/10.1007/s11263-014-0794-5.

Lee, Yong Jae and Kristen Grauman (2011). "Learning the easy things first: Self-paced visual category discovery". In: *CVPR, Conference on*. IEEE, pp. 1721–1728.

Lidon, Aniol et al. (2015). "UPC-UB-STP@ MediaEval 2015 diversity task: iterative reranking of relevant images". In: *MediaEval 2015 Multimedia Benchmark Workshop*. CEUR-WS. org.

Lidon, Aniol et al. (2017). "Semantic summarization of egocentric photo stream events". In: *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*. ACM, pp. 3–11.

Lopez-Fuentes, Laura et al. (2017). "Multi-modal Deep Learning Approach for Flood Detection." In: *MediaEval* 17, pp. 13–15.

Marone, José et al. (2016). "Learning the lumen border using a convolutional neural networks classifier". In: *MICCAI CVII-STENT Workshop*.

Martinel, Niki, Gian Luca Foresti, and Christian Micheloni (2018). "Wide-slice residual networks for food recognition". In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, pp. 567–576.

Oliveira-Barra, Gabriel et al. (2017). "Serious Games Application for Memory Training Using Egocentric Images". In: *International Conference on Image Analysis and Processing*. Springer, pp. 120–130.

Oliveira-Barra, Gabriel et al. (2019). "Lifelog retrieval for memory stimulation of people with memory impairment". In: *Multimodal Behavior Analysis in the Wild*. Elsevier, pp. 135–158.

Oliveira Barra, Gabriel de et al. (2016). "Lemore: A lifelog engine for moments retrieval at the ntcir-lifelog lsat task". In: *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*.

Pennington, Jeffrey, Richard Socher, and Christopher D Manning (2014). "Glove: Global vectors for word representation". In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543.

Peris, Álvaro et al. (2016). "Video description using bidirectional recurrent neural networks". In: *International Conference on Artificial Neural Networks*. Springer, pp. 3–11.

Poleg, Y., C. Arora, and S. Peleg (2014). "Temporal segmentation of egocentric videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2537–2544.

Radeva, Petia, Marc Bolaños, and Estefania Talavera (2018). "Tutorial: Deep Learning and Applications to Activity Recognition from Egocentric Photostreams". In:

Ragusa, Francesco et al. (2016). "Food vs non-food classification". In: *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*, pp. 77–81.

Raina, Rajat, Anand Madhavan, and Andrew Y Ng (2009). "Large-scale deep unsupervised learning using graphics processors". In: *Proceedings of the 26th annual international conference on machine learning*, pp. 873–880.

Robbins, Herbert and Sutton Monro (1951). "A stochastic approximation method". In: *The annals of mathematical statistics*, pp. 400–407.

Rosenblatt, Frank (1958). "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6, p. 386.

Rozin, Paul et al. (1999). "Attitudes to food and the role of food in life in the USA, Japan, Flemish Belgium and France: Possible implications for the diet–health debate". In: *Appetite* 33.2, pp. 163–180.

Salvador, Amaia et al. (2017). "Learning cross-modal embeddings for cooking recipes and food images". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3020–3028.

Sarker, Mostafa Kamal et al. (2018). "Macnet: Multi-scale atrous convolution networks for food places classification in egocentric photo-streams". In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 0–0.

Song, Jingkuan et al. (2017). "Hierarchical LSTM with Adjusted Temporal Attention for Video Captioning". In: *arXiv preprint arXiv:1706.01231*.

Specia, Lucia et al. (2016). "A shared task on multimodal machine translation and crosslingual image description". In: *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pp. 543–553.

Szegedy, Christian et al. (2015). "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9.

Talavera, Estefania, Petia Radeva, and Nicolai Petkov (2019). "Towards Emotion Retrieval in Egocentric PhotoStream". In: *arXiv preprint arXiv:1905.04107*.

Talavera, Estefania et al. (2015). "R-clustering for egocentric video segmentation". In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, pp. 327–336.

Talavera, Estefania et al. (2020). "Topic modelling for routine discovery from egocentric photo-streams". In: *Pattern Recognition*, p. 107330.

Tang, Jian, Meng Qu, and Qiaozhu Mei (2015). "Pte: Predictive text embedding through large-scale heterogeneous text networks". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1165–1174.

Turk, Matthew A and Alex P Pentland (1991). "Face recognition using eigenfaces". In: *Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition*. IEEE Computer Society, pp. 586–587.

Wu, Wen and Jie Yang (2009). "Fast food recognition from videos of eating for calorie estimation". In: *2009 IEEE International Conference on Multimedia and Expo*. IEEE, pp. 1210–1213.

Xie, Saining et al. (2017). "Aggregated residual transformations for deep neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500.

Xu, Kelvin et al. (2015a). "Show, attend and tell: Neural image caption generation with visual attention". In: *International conference on machine learning*, pp. 2048–2057.

Xu, Ruihan et al. (2015b). "Geolocalized modeling for dish recognition". In: *IEEE transactions on multimedia* 17.8, pp. 1187–1199.

Yao, Li et al. (2015). "Describing Videos by Exploiting Temporal Structure". In: *Proceedings of the International Conference on Computer Vision*, pp. 4507–4515.

Yin, Xi and Xiaoming Liu (2017). "Multi-task convolutional neural network for pose-invariant face recognition". In: *IEEE Transactions on Image Processing* 27.2, pp. 964–975.

Zhang, Xi-Jin, Yi-Fan Lu, and Song-Hai Zhang (2016). "Multi-task learning for food identification and analysis with deep convolutional neural networks". In: *Journal of Computer Science and Technology* 31.3, pp. 489–500.