



Universitat Autònoma de Barcelona

ADVERTIMENT. L'accés als continguts d'aquesta tesi doctoral i la seva utilització ha de respectar els drets de la persona autora. Pot ser utilitzada per a consulta o estudi personal, així com en activitats o materials d'investigació i docència en els termes establerts a l'art. 32 del Text Refós de la Llei de Propietat Intel·lectual (RDL 1/1996). Per altres utilitzacions es requereix l'autorització prèvia i expressa de la persona autora. En qualsevol cas, en la utilització dels seus continguts caldrà indicar de forma clara el nom i cognoms de la persona autora i el títol de la tesi doctoral. No s'autoritza la seva reproducció o altres formes d'explotació efectuades amb finalitats de lucre ni la seva comunicació pública des d'un lloc aliè al servei TDX. Tampoc s'autoritza la presentació del seu contingut en una finestra o marc aliè a TDX (framing). Aquesta reserva de drets afecta tant als continguts de la tesi com als seus resums i índexs.

ADVERTENCIA. El acceso a los contenidos de esta tesis doctoral y su utilización debe respetar los derechos de la persona autora. Puede ser utilizada para consulta o estudio personal, así como en actividades o materiales de investigación y docencia en los términos establecidos en el art. 32 del Texto Refundido de la Ley de Propiedad Intelectual (RDL 1/1996). Para otros usos se requiere la autorización previa y expresa de la persona autora. En cualquier caso, en la utilización de sus contenidos se deberá indicar de forma clara el nombre y apellidos de la persona autora y el título de la tesis doctoral. No se autoriza su reproducción u otras formas de explotación efectuadas con fines lucrativos ni su comunicación pública desde un sitio ajeno al servicio TDR. Tampoco se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR (framing). Esta reserva de derechos afecta tanto al contenido de la tesis como a sus resúmenes e índices.

WARNING. The access to the contents of this doctoral thesis and its use must respect the rights of the author. It can be used for reference or private study, as well as research and learning activities or materials in the terms established by the 32nd article of the Spanish Consolidated Copyright Act (RDL 1/1996). Express and previous authorization of the author is required for any other uses. In any case, when using its content, full name of the author and title of the thesis must be clearly indicated. Reproduction or other forms of for profit use or public communication from outside TDX service is not allowed. Presentation of its content in a window or frame external to TDX (framing) is not authorized either. These rights affect both the content of the thesis and its abstracts and indexes.



**Universitat Autònoma
de Barcelona**

Information Extraction from Heterogeneous Handwritten Documents

A dissertation submitted by **Juan Ignacio Toledo** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor of Philosophy**.

Bellaterra, April 8, 2019

Director: **Alicia Fornés**
Autonomous University of Barcelona
ComputerScience Dept. and Computer Vision Center

Co-director: **Josep Lladós**
Autonomous University of Barcelona
ComputerScience Dept. and Computer Vision Center

Thesis Committee | **Dr. Veronique Églin**
Institut National des Sciences Appliquées de Lyon
Lyon, France
Dr. Oriol Ramos-Terrades
Dept. Ciències de la Computació
Centre de Visió per Computador
Universitat Autònoma de Barcelona
Dr. Andreas Fischer
University of Applied Sciences and Arts Western Switzerland
Delémont, Switzerland



This document was typeset by the author using L^AT_EX 2_ε.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

Copyright © MMXIX by Juan Ignacio Toledo. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the author.

ISBN “Pending”

Printed by Ediciones Gráficas Rey, S.L.

Dedicated to my family and friends

Agradecimientos

No consigo recordar exactamente dónde lo leí o lo escuché. Solo recuerdo a un empresario exitoso, muy mayor, creo que de alguna gran empresa japonesa, diciendo que a medida que se hacía mayor, se iba dando cuenta de que sus esfuerzos o sus méritos habían jugado un papel menor en sus éxitos y que lo realmente decisivo, había sido la ayuda de multitud de personas a su alrededor que le habían alcanzado la posición en la que ahora se encontraba. Estas palabras se me quedaron grabadas y ahora, tras pasar por la experiencia de hacer una tesis, creo que puedo hacerlas mías.

Realmente tengo la sensación, de que el haber llegado hasta aquí es mucho más gracias a la gente que me ha guiado y acompañado que a cualquier cosa que tenga que ver directamente conmigo. Tengo tanto que agradecer, y la cosa viene de lejos. Gracias a mis padres, por enseñarme lo divertido que puede ser aprender. A mis amigos de toda la vida, con los que tuve la oportunidad de crecer y descubrir tantas cosas. Tantísima gente de la que he podido aprender, profesores, jefes, mentores en general. Colegas con los que recorrer juntos este camino, apoyándonos mutuamente. He recibido tanto de tanta gente que casi parece injusto intentar enumerar. En cualquier caso, es incuestionable el valor de las ideas, consejos y ayuda de todo tipo que me han aportado mis directores de tesis, Josep y Alicia. También me gustaría agradecer a Jordi el darme la oportunidad de empezar la aventura de la investigación en ScytI. A todos los compañeros del departamento de Research os agradezco también el haber sabido crear un ambiente de trabajo prácticamente inmejorable y el haber compartido ideas tan diversas. También a los compañeros del CVC con los que he podido colaborar, intercambiar ideas y recorrer mundo en las varias conferencias, especialmente Pau, Albert, Sounak y muy especialmente a Marçal (ya hace 20 años de aquellas prácticas de electrónica tan divertidas!) También ha sido muy importante, para poder llegar hasta aquí el apoyo mi familia, especialmente el de Yazmina, que ha llegado incluso a colaborar en el arduo trabajo de revisar groundtruths! Y gracias también a mis hijas Lucía y Andrea, que aunque acaban de llegar también han ayudado a que este proyecto llegue a buen término haciendo algo tan sencillo y tan maravilloso como obsequiarme con su sonrisa cada vez que vuelvo a casa. Gracias a todos!

Abstract

Despite the unstoppable trend towards a fully digital paperless world, there is still an abundance of totally or partially handwritten documents that need to be automatically processed, from historical demographic records to more recent form-like documents. The most common approach to leverage current technology is by applying Document Image Analysis and Recognition techniques on the digital images of those documents acquired with scanners or digital cameras.

Over the years, as the technology advances, different approaches have been proposed to be able to access to the information contained in document images. In this thesis we explore the whole process of information extraction, with different examples of document types.

The first step towards extracting information from any document is actually understanding the document and the most common techniques required to process it. For that reason, on the first chapter of this thesis after the introduction, we provide an in depth study of electoral documents, that had not yet drawn much attention from the community. We discuss how the interpretation of document images is never as simple as it seems; even deciding if a mark is present or not in a given position can be challenging, depending on legal requirements for our system. In this chapter we also make a quick overview of some of the most common DIAR techniques that can be applied to different kind of electoral documents. In the end, electoral documents can be seen as a special kind of form documents, where the position in the form determines the semantics of each element of the document. For example the word 'John' could be the name of a candidate or the name of an officer certifying the results for a particular polling place.

If, in the most simple case, the semantic entities are determined by the spatial position, we should devote some of our efforts to the transcription, as in "reading" the information on each field. We also devote a chapter to explore two new approaches to handwriting recognition. Both assume that the handwriting can be modeled as a series of features sequentially extracted from the images. In the first approach, we use variational autoencoders to derive descriptive features from unlabeled text images and whereas in the second approach we use attribute embeddings, that allow us to derive a much more discriminative set of features by making use of the transcription information.

After learning about the factors to take into consideration when interpreting documents and the techniques required to transcribe handwritten text, we can

now extract information from highly structured types of documents, like forms, but we are interested in going one step further. A good challenge are historical handwritten birth or marriage records. This kind of documents are not exactly a form, but they share a similar structure. We can think of them as divided in records with a given set of fields whose position is not fixed in a particular location. We are interested in processing those documents as if they were a form. In order to study that problem, we elaborated a new benchmark and organized an international competition for the community. We devote another chapter to describing all the details of the dataset, required tasks and the metric.

Finally, in the last chapter, we propose a full information extraction approach, with two variants, based on a combination of Convolutional and Recurrent Neural Networks, that can deal with loosely structured documents as the ones in our proposed benchmark. This approach uses the better performing of our HTR approaches described in the chapter devoted to HTR to get the transcriptions, while deriving the semantic information directly from the word images. As a first step, we prove that classification of isolated handwritten word images into semantic classes is feasible and later we explore two alternatives to leverage contextual information to be able to use record level information to improve the accuracy of the system.

Resum

Tot i la imparable tendència cap a un món digital, abandonant el suport paper, encara són abundants els documents total o parcialment manuscrits que necessiten processar-se automàticament, des de registres demogràfics històrics fins a documents més recents tipus formulari. L'aproximació més habitual per a aprofitar la tecnologia actual és aplicar tècniques d'Anàlisi i Reconeixement d'Imatges de Documents a les imatges digitals d'aquests documents, prèviament adquirits mitjançant escàners o càmeres digitals.

Amb el pas dels anys i els avenços de la tecnologia, s'han anat proposant diferents aproximacions per a accedir a la informació continguda a imatges de documents. En aquesta tesi explorem tot el procés d'extracció d'informació amb diferents tipus de documents com a exemple.

El primer pas per a extreure informació de qualsevol tipus de document és entendre realment el document així com les tècniques més comuns que calen per processar-lo. Per aquest motiu, en el primer capítol de la tesi, realitzem un estudi en profunditat dels documents electorals, que encara no han rebut prou interès per part de la comunitat. Argumentem que la interpretació d'imatges de documents mai és tan simple com pot semblar: fins i tot decidir si hi ha una marca en una posició determinada, pot ser un repte, depenent, per exemple, dels requisits legals que tingui el nostre sistema.

En aquest primer capítol donem també unes pinzellades sobre com algunes de les tècniques més comuns de l'Anàlisi de Documents es poden aplicar a les diferents problemàtiques dels documents electorals. Al cap i a la fi, els documents electorals es poden veure com a un tipus especial de formularis, on la posició de cada element dins del formulari és el que determina el seu significat. Per exemple, la paraula "John" pot ser el nom d'un candidat o d'un membre de la mesa electoral certificant el resultat d'una elecció concreta.

Si, com hem vist, en el cas més simple, les entitats semàntiques queden determinades per la seva posició espacial, hauríem de dedicar part dels nostres esforços a la transcripció, es a dir, a "llegir" la informació de cada camp. En el segon capítol explorem dues aproximacions al reconeixement de textos manuscrits. Ambdós es basen en la idea que l'escriptura manuscrita es pot modelar com una sèrie de característiques seqüencials que es poden extreure de les imatges. En concret, per al primer mètode fem servir Variational Autoencoders per a derivar característiques descriptives des d'imatges de text manuscrit sense transcriure, mentre que

en el segon mètode fem servir una codificació basada en atributs, que ens permet derivar una serie de característiques molt mes discriminatives, gràcies a fer servir la informació de la transcripció.

Un cop sabem quins factors cal tenir en compte al interpretar diferents tipus de documents, així com les tècniques necessàries per a poder transcriure text manuscrit podem extreure informació de documents altament estructurats con formularis. No obstant, estem interessats en anar un pas més enllà. Un bon repte son els registres històrics manuscrits de naixements i matrimonis. Aquests documents, tot i no ser exactament un formulari, comparteixen amb ells una estructura similar. Estan dividits en registres, cadascun dels quals es pot veure com un conjunt de camps que no tenen una posició fixa. Ens agradaria poder processar aquest tipus de documents como si es tractés d'un formulari. Amb l'objectiu d'estudiar aquesta problemàtica, hem elaborat un benchmark i organitzat una competició internacional per a la comunitat. Dedicuem un capítol d'aquesta tesi a descriure els detalls del dataset, les tasques a realitzar així com la mètrica emprada.

Finalment, a l'últim capítol, proposem un mètode complet d'extracció d'informació de documents manuscrits, basat en una combinació de Xarxes Neuronals Recurrents i Convolucionals, que pot tractar documents amb una estructura flexible com els proposats en el nostre benchmark. Aquesta aproximació fa servir el més potent dels mètodes descrits al capítol dedicat a reconeixement de text manuscrit per a realitzar les transcripcions , mentre que deriva la informació semàntica directament de les imatges de paraules retallades del document. Com a primer pas, demostrem que es possible classificar imatges aïllades de paraules manuscrites en classes semàntiques per a després explorar dues alternatives que en permetin aprofitar la informació contextual a nivell de registre per a millorar dràsticament la precisió del nostre sistema.

Resumen

A pesar de la imparable tendencia hacia un mundo digital abandonando el soporte papel, aún abundan documentos total o parcialmente manuscritos que es necesario procesar automáticamente, desde registros demográficos históricos hasta los documentos tipo formulario más recientes. La aproximación más común para aprovechar la tecnología actual, es aplicar técnicas de Análisis y Reconocimiento de Imágenes de Documentos a las imágenes digitales de dichos documentos, previamente adquiridas a través de escáneres o cámaras digitales.

Con el paso de los años y el avance de la tecnología, se han ido proponiendo diferentes aproximaciones para acceder a la información contenida en las imágenes de documentos. En esta tesis exploramos todo el proceso de extracción de información, con diferentes tipos de documentos a modo de ejemplo.

El primer paso para extraer información de cualquier tipo de documento, es realmente entender dicho documento así como las técnicas más comunes que se necesitan para procesarlo. Por ese motivo, en el primer capítulo de la tesis, realizamos un estudio en profundidad de los documentos electorales, que aún no habían recibido el suficiente interés por parte de la comunidad. Argumentamos que la interpretación de imágenes de documentos nunca es tan simple como parece; incluso decidir si hay una marca en una posición determinada puede ser un desafío, dependiendo, por ejemplo, de los requerimientos legales que tenga nuestro sistema.

En este primer capítulo, damos también unas pinceladas sobre como algunas de las técnicas más comunes del Análisis de Documentos se pueden aplicar a las diferentes problemáticas de los documentos electorales. Al fin y al cabo, los documentos electorales se pueden ver como un tipo especial de formularios, donde la posición de cada elemento dentro del formulario es lo que determina su significado. Por ejemplo, la palabra “John” podría ser el nombre de un candidato o de un miembro de la mesa electoral certificando los resultados de una elección particular.

Si, como hemos visto, en el caso más simple, las entidades semánticas quedan determinadas por su posición espacial, deberíamos dedicar al menos parte de nuestros esfuerzos a la transcripción, es decir, a “leer” la información de cada campo. En el segundo capítulo exploramos dos aproximaciones al reconocimiento de textos manuscritos. Ambos asumen que la escritura manuscrita se puede modelar como una serie de características secuenciales que se pueden extraer de las imágenes. En concreto, para el primer método usamos Variational Autoencoders para derivar características descriptivas a partir de imágenes de texto manuscrito sin

transcribir mientras que en el segundo método usamos una codificación basada en atributos, que nos permite derivar una serie de características mucho más discriminativas, gracias al uso de la información de la transcripción.

Tras ver qué factores hay que tener en cuenta al interpretar distintos documentos así como las técnicas necesarias para transcribir texto manuscrito somos capaces de extraer información de documentos altamente estructurados como formularios. Sin embargo estamos interesados en ir un paso más allá. Un buen desafío son los registros históricos manuscritos de nacimientos o matrimonios. Estos documentos, pese a no ser exactamente un formulario, comparten con ellos una estructura similar. Están divididos en registros, cada uno de los cuales se puede ver como un conjunto de campos que no tienen una posición fija. Nos gustaría poder procesar este tipo de documentos como si fuesen un formulario.

Con el objetivo de estudiar esta problemática, hemos elaborado un benchmark y organizado una competición internacional para la comunidad. Dedicamos un capítulo de esta tesis a describir los detalles del dataset, las tareas a realizar así como la métrica empleada.

Finalmente, en el último capítulo, proponemos un método completo de extracción de información de documentos manuscritos, basado en una combinación de Redes Neuronales Recurrentes y Convolucionales, que es capaz de tratar documentos ligeramente estructurados como los propuestos en nuestro benchmark. Esta aproximación utiliza el más potente de los métodos descritos en el capítulo dedicado a reconocimiento de texto para realizar las transcripciones, mientras que deriva la información semántica directamente de las imágenes de palabras recordadas del documento. Como primer paso, demostramos que es posible la clasificación de imágenes aisladas de palabras manuscritas en clases semánticas para luego explorar dos alternativas que nos permitan aprovechar la información contextual a nivel de registro mejorando drásticamente la precisión de nuestro sistema.

Contents

1	Introduction	1
1.1	Context	1
1.2	Motivation	3
1.2.1	Electoral Documents	4
1.2.2	Historical Documents	6
1.3	Contributions	8
1.4	Outline	8
2	Electoral Documents	11
2.1	Introduction	11
2.2	Preprocessing	11
2.2.1	Ballots	13
	Mark Recognition	13
	Preferential Voting	18
2.2.2	Write-in	20
2.2.3	Tally sheets	23
2.3	Our approach for tally sheet processing	25
2.3.1	Preprocessing	25
	Orientation and skew removal	25
	Region of interest extraction and noise removal	27
2.3.2	Intelligent Character Recognition	27
2.3.3	Handwritten text recognition	29
2.4	Conclusions	29
3	Handritten Text Recognition	31
3.1	Introduction	31
3.2	Unsupervised Feature Discovery based HTR	34
3.2.1	Unsupervised Feature Learning	35
3.2.2	Variational Autoencoders	36
3.2.3	Sequence Alignment and Recognition	37
3.2.4	Experiments	37
3.3	Attribute Embedding Based HTR	38
3.3.1	Attribute Embedding	39

3.3.2	Extension to sequences	39
3.4	Network Architecture	40
3.4.1	PHOCNet	40
3.4.2	BLSTM+CTC	42
3.5	Experiments	43
3.5.1	Datasets	43
	Washington Dataset	43
	Esposalles Dataset	44
3.5.2	Experimental Setup	44
3.5.3	Results discussion	45
3.6	Conclusions	46
4	A benchmark for Information Extraction	49
4.1	Introduction	49
4.2	The "Esposalles" Marriage Records	50
4.3	The IEHHR Competition Dataset	51
4.4	Task	53
4.5	Metrics	54
4.6	Results	56
4.7	Conclusions	57
5	Information Extraction from handwritten documents	59
5.1	Introduction	59
5.2	State of the art	64
5.3	CNN Based Word Image Categorization	66
5.3.1	Convolutional Neural Networks	66
5.3.2	Fully Connected Layers	67
5.3.3	Spatial Pyramid Pooling	68
5.4	Experimental Validation	69
5.4.1	Esposalles Dataset	69
5.4.2	Experiments and Results	70
5.5	A full Information Extraction System	72
5.5.1	Semantic categorization of isolated word images	72
5.5.2	Incorporating language models	73
	Bigram inspired language model	73
	BLSTM based language model	75
5.6	Experiments	76
5.6.1	Dataset	77
5.6.2	Performance Evaluation	78
5.6.3	Experimental Details	78
	Semantic labeling	78
	Handwritten word recognition	79
	Bigram inspired language model	80
	BLSTM inspired language model	80
5.6.4	Results and Discussions	81

5.7	Conclusions and further work	83
6	Conclusions and Future Work	87
6.1	Summary and Discussion	87
6.2	Future Work	89
	Bibliography	93

List of Tables

2.1	Average digit error rate on MNIST and TallySheets datasets. . . .	28
3.1	Average character error rate and standard deviation over five different experiments for each set of hyperparameters.	38
3.2	Average number of iterations required for convergence and standard deviation over five different experiments for each set of hyperparameters.	38
3.3	Comparative with other methods CER.	46
4.1	Results of the different methods	56
4.2	Table of results iehhr.	57
5.1	Comparative with other methods.	70
5.2	Confusion Matrix for our CNN architecture, with a global accuracy 78.11%.	70
5.3	Competition Score	81
5.4	Confusion matrix for the BLSTM based model for the category label.	82
5.5	Confusion matrix for the BLSTM based model for the person label.	82

List of Figures

1.1	Samples of records from structured documents. Baptism registers from the Absdorf collection, 1853 (top). Death records from Wien, 1720 (left). Medical records from Sant Pau Hospital, Barcelona 1604 (right). Marriage records from the Barcelona Cathedral, 1619 (bottom).	7
2.1	The original ballot image acquired with a camera (left). The image thresholded with Otsu’s Method (center) and with Sauvola’s Method (right). We can see how using Otsu’s method the darker areas of the ballot become black while voting targets in the lighter areas disappear, showing the limitations of setting a global threshold.	12
2.2	An example of a ballot to be processed with OMR technology. . .	14
2.3	Different types of marks classified according to how a scanner interprets it [51].	15
2.4	Different types of marks classified according to how the law interprets it [51].	15
2.5	Voter intent is clear, according to some states this vote should be counted.	16
2.6	Three different marking styles: check, ex, and filled, and their corresponding noisy inputs generated by the voter attempting to erase a mark. Extracted from [118].	17
2.7	An example of undervote and overvote	18
2.8	Some examples from the MNIST dataset. It’s a common benchmark for isolated handwritten digit recognition consisting of 60,000 digit images from approximately 250 different writers.	19
2.9	The architecture of one column of the convolutional neural network that achieved the best scores so far in handwritten digit recognition on the MNIST dataset. The response for each neuron to the input image is also shown as an image. Extracted from [22].	19
2.10	A write in vote where the voter did not fill in the oval as prescribed.	21
2.11	The “sliding window”. Extracted from [30]	21

2.12	In Hidden Markov Models, the data is modeled as a series of observations generated by a hidden state that is only dependent on the state at the previous time step.	22
2.13	A Long Short Term Memory Cell with multiplicative input, output and forget gates. Extracted from [73]	23
2.14	Example of a ballot statement. Extracted from [2]	24
2.15	The fiducial marks in a tally sheet (highlighted in red) used to detect the orientation and skew also allow us to segment the Region of Interest for later handwriting and digit recognition steps.	26
2.16	Some examples of digits from the CVL dataset.	27
2.17	Samples for type A number seven (left) from the CVL dataset [23] and type B number seven (right) from the MNIST datasets [64] . .	28
3.1	A sequence of image patches is passed through the encoder of a Variational Autoencoder to get a latent variable representation. This representation is then fed into a bidirectional long short term memory neural network to perform the final recognition.	35
3.2	System architecture. After training a PHOCNet for word attribute embedding, we embed patches of word images into the attribute space. From these points in the attribute space we create a sequence that is passed to a two-layer BLSTM+CTC recurrent neural network that performs the transcription.	38
3.3	The architecture of PHOCNet. Best viewed in electronic format. Extracted from [102].	40
3.4	Some examples of word images in the George Washington dataset. The available images are normalized and binarized.	43
3.5	Some examples of word images in the Esposalles dataset. We see a high degree of variability both in image size and aspect ratio. . . .	43
4.1	A marriage record information can be divided in pieces of information related to a specific person(top). Also, we can identify different “semantic categories” related to a specific person that are likely to appear in most marriage records(bot).	52
4.2	An example of the IEHHR Competition Dataset:“Esposalles” at word-image level, showing word images and their corresponding, transcription (black), category label (blue) and person label (orange). 54	54
5.1	Samples of records from structured documents. Baptism registers from the Absdorf collection, 1853 (top). Death records from Wien, 1720 (left). Medical records from Sant Pau Hospital, Barcelona 1604 (right). Marriage records from the Barcelona Cathedral, 1619 (bottom).	62
5.2	Outline of our CNN architecture	66

5.3	Several examples of word images in the Esposalles Dataset. The big degree of variability both in size and aspect ratio of the images makes impractical the common approach of resizing images to a common size.	69
5.4	Bigram inspired architecture. This architecture models the relation of words by accepting two inputs: the current word image, and the predicted label from the previous word image.	74
5.5	The inner workings of an LSTM network. [73]	75
5.6	BLSTM-based architecture. This architecture models the relation among words in a record with the hidden state of a BLSTM layer. In this case we can see that the architecture has two softmax outputs because we are interested in extracting both the category and the person it relates to.	76
5.7	An example of the IEHHR Competition Dataset:“Esposalles” at word-image level, showing word images and their corresponding, transcription (black), category label (blue) and person label (orange).	77
5.8	Example of a prediction where one word was incorrectly assigned. The surname of the former husband of the bride is incorrectly assigned to her (shown in red color).	83

Chapter 1

Introduction

In this chapter we will introduce our motivation to work in information extraction. We will provide a brief overlook of the context of information extraction from handwritten documents and present the contributions of the thesis.

1.1 Context

The Document Image Analysis and Recognition (DIAR) field has the ultimate goal of achieving an understanding of document contents by the means of analyzing and recognizing its scanned or camera-captured images. One could argue that understanding a document, or in fact anything, is the process through which we are able to find a way to map pieces of information to a structured set of predefined categories. We call this general process Information Extraction. The term "Information Extraction" was originally coined by the Natural Language Processing (NLP) community, although it has been extending recently to other domains. For the document analysis domain, we see it, as stated earlier, as a general process that includes all the steps required to go from document images to a meaningful high-level representation of the information contained in it.

To be able to gain an understanding on documents contents, an obvious first step is being able to retrieve such contents. This process is known as Information Retrieval. It is the case in the DIAR community, that Information Retrieval has been, and continues to be, a challenging problem and great efforts have been made to simply be able to access document contents. There are some common techniques that are frequently used in most kind of documents to help us in this task. These tasks would fall into what we call pre-processing, whose goal is precisely to make

the processing easier.

Usually preprocessing tasks are headed towards dealing with the variability of the physical support of the document or the acquisition technique. For instance, binarization techniques to segment background and foreground or perspective removal in the case of camera-based acquisition. These techniques can range from a simple operation to a research area inside the community. The aforementioned case of binarization can range from a simple thresholding operation in an ideal scenario to Fully Convolutional Neural Networks [106] in degraded paper with the presence of show-through. In the case of handwritten text, common preprocessing steps also include some degree of text-content segmentation be it at paragraph, text-line or word level. Once we have images with only handwritten text, we can proceed to retrieve the information contained in it.

The first technique that comes into mind when attempting to retrieve the contents of a handwritten document image is handwritten text recognition (HTR). Contrary to Optical Character Recognition where one could just segment the text into characters and perform a character level classification, handwriting recognition is still today an active research field. Apparently, to be able to recognize handwritten text, one should be able to segment the individual characters, however, a good character segmentation requires recognizing those characters. In order to deal with this paradox, most of the techniques usually used in HTR try to perform both tasks at the same time. There is also the question on how much higher level information we humans use when reading handwritten text. Certainly knowing the language you are reading helps you assign *a priori* probabilities on words and characters. In a HTR system we can do so by the means of language models, assuming we have enough data in that language to get trustworthy estimations. Although great advances have been made in the area in the recent years, there is still room for improvement specially for scenarios with multiple writers, degraded documents or ancient languages.

In those kind of scenarios, where handwriting recognition is specially challenging, the DIAR community has developed Information Retrieval techniques like word spotting. In word spotting, the goal is trying to find instances of a given word in a document or collection of documents. Word-spotting approaches can be roughly divided in two great families, query-by-example, which can be seen as a special kind of image matching where we want to find parts of the document that are similar to a word image that is provided as a query and query-by-string where we try to find words that match a given transcription. Although very useful for scholars in the digital humanities, word spotting can be seen as a very limited information retrieval approach, where we can only discover the presence and location of instances of a given keyword in our corpus.

Certainly handwriting recognition is an important step in being able to access the information of handwritten documents. It supersedes word spotting in allowing a quick access to relevant keywords in the corpus and opens the possibility of processing the transcription to build higher level representations of the informa-

tion thus allowing for real Information Extraction. While handwriting recognition could also be seen as a means of Information Extraction itself, we must note that its output is just plain text, without any semantic knowledge of the document.

In this thesis we are interested in going one step beyond handwriting recognition and be able to assign meaning to these transcriptions. In the most general case, applying Natural Language Processing (NLP) techniques on top of the transcription is the only possible solution to assign a semantic value to each word. Usually these techniques rely on stemmers or lemmatizers, vocabularies or ontologies and Part-of-Speech (POS) tagging developed specifically for a language.

However, there are a big number of documents where the semantic value can be clearly derived from the document structure. The most extreme case would be a form or tabular document where the semantic value of each word is determined by its position in the document. A more interesting case is the one of documents like birth records, where we know what kind of information we should find in each record. Moreover, all records will share a similar structure that can be leveraged to extract the relevant pieces of information from each record. It is worth noting that this structure is a direct consequence of the information contained in the records, therefore, techniques that leverage the structure to assign meaning can work without modification to documents in other languages.

1.2 Motivation

There is a huge amount of information stored in totally or partially handwritten documents that needs to be accessed. Examples range from historical handwritten birth or marriage records to modern form-like documents like electoral documents or invoices. Even today, there are still some challenges to perform handwriting text recognition as we briefly described above: different writing styles, degraded documents, language models, etc. Moreover, when we do HTR, that is not usually our final goal, even though it might appear so: we want to be able to understand the contents of the documents and extract their information. This is an interesting and challenging high level application within the DIAR field that had only been shallowly explored. This is certainly the major focus of this thesis, we want to be able to process document images **extracting** relevant information that could potentially be useful. So, the material we are presenting in this thesis is meant to be both a layer above previous works that were on the border between information retrieval and extraction and future works that might rely on a semantic interpretation of documents.

When we started working on this thesis Deep Learning was just taking off. It had yet to become the major mainstream research technique that could be applied to any problem. It was certainly widely known to Computer Vision scholars as that promising technology that impressively boosted object recognition performance. But most scholars, initially doubtful of its potential, were just starting to study

its applicability in their specific domain. None of the most popular Deep Learning frameworks in use today existed (TensorFlow, Pytorch, Keras,...) neither was the web teeming with code and tutorials. We believed that these new techniques would allow us to tackle information extraction from handwritten documents. At that moment it was not a completely obvious decision, even though it might appear as so in hindsight seeing the recent widespread use of Deep Learning techniques.

Thus, we have an interesting challenge with several applications to real world problems and we have a promising set of tools that might enable us to tackle these challenge. I had to summarize the motivation of this thesis in a single sentence that would be: The motivation of this thesis is exploring different deep learning approaches and developing new information extraction techniques that can be applied to loosely structured handwritten documents. We decide to put our focus on electoral documents and historical handwritten records for several reasons that we discuss below.

1.2.1 Electoral Documents

This thesis started as an Industrial PhD with the collaboration of the electronic voting company Scytl and the Document Analysis group of the CVC. Scytl is a world leader in providing secure electronic online voting solutions based on the use of advanced cryptographic protocols and an implementation of strict computer security policies. Thus, one of the goals of the thesis was, in a broad sense, to study different Document Image Analysis and Recognition (DIAR) techniques that could be applied to electoral documents. The use of such techniques would allow Scytl the possibility to offer a wider set of solutions to its potential customers that were willing to modernize their elections tally and results consolidation without moving to electronic online voting.

Over 300 nationwide elections are held yearly. This sums up to approximately 3600 million registered voters per year, with an average spending of 5 USD per voter, elections constitute a potential market of 18 billion dollars per year. While remote or poll-site electronic voting is gaining more and more acceptance worldwide, many elections are still paper based. Be it for tradition, for its simplicity, because it leaves a physical evidence of the vote or because of a restrictive electoral law, there are several countries that are not willing to abandon paper based elections yet. However, this does not mean that they are not willing to use modern technology in elections.

Modern online voting technology has obvious advantages over traditional paper based solutions: voters can cast their vote whenever they want and wherever they are as long as they have an internet connection, formal mathematical proofs that the results have not been altered can be provided, accessibility measures for disabled voters are easier to implement, the cost of the election is reduced, etc.

However, traditional paper based voting has its strengths, which lie in its sim-

plicity. One of those advantages is the ease of use for both voters and officials. Virtually everyone grew up using pen and paper while only a portion of that people is comfortable browsing the internet. Moreover, everyone can understand all the steps of a paper based election and how they ensure voters privacy and correct results. This is not the case with online voting where, even among computer literate people, many are still expressing concerns for their privacy, results correctness, and the fact that only a few cryptographers can really understand **why** the system is secure. Finally, there are also some countries where the specific wording of the electoral law requires paper based elections.

However, this ease of use and widespread acceptance of paper based elections has its counterpart. This free interaction of humans with handwritten documents results in a considerable variance that makes it difficult to understand their contents. For instance, two 'e' letters, even from the same writer, will never be truly identical, text-lines might have different slopes, corrections marks can be made, etc. This freedom of the user transforms an apparently straightforward problem into a real challenge for any computer system.

Countries with complex electoral systems, like the US, have been exploring how to automate the tally for paper based elections for decades. Mark sense scanners, first developed for educational testing, have been used for ballot processing since the 1950's. They were based on a ballot printed with a special ink, that was invisible to the sensor, and the use of index marks to define the position of the voting targets. In the 1990's, devices using imaging technology were developed. They used fiducial marks that allowed the scanner to interpolate the voting targets and counted the number of dark pixels in each area. More recently, in 2006, a patent was granted to a device based on edge detection, which could detect empty voting targets (ovals) and filled voting targets.

We can see a trend moving from solutions requiring specific hardware to more generic hardware-independent solutions using computer vision techniques. However, there are still a lot of challenges to be able to support more complex elections. In the document analysis field, techniques have been developed to process different kind of documents. To our knowledge, the work specifically applied to electoral documents has mainly dealt with Optical Mark Recognition. We are interested in different Pattern Recognition techniques that can help us understand the contents of such document images. Optical Character Recognition (OCR), Invoice Recognition, Optical Mark Recognition (lotery ticket), Optical Music Recognition, Barcode reader, etc. are examples of wide known mature techniques that are already used in commercial software. By using this kind of techniques in combination with other to be developed during the thesis we want to allow Scytl the possibility of offering new commercial products for those customers that are not ready to move away from paper based elections to online electronic voting.

It is worth noting that in paper based elections, and due to the great diversity in electoral laws, there are several different electoral activities that produce huge amount of documents potentially suitable for automatic processing. Prior to

the election day, there are some countries that require you to register before the election. In the election day we have a spectre of voting schemes from partisan systems where all you have to do is cast a vote for a party, to preferential voting where you can rank order candidates with a preference value, or even write-in a candidate name that is not present in the ballot. There are also tally sheet documents, to perform the results consolidation once the election is over. It is thus, not surprising, that the techniques developed during this thesis, designed for handwritten documents with some level of structure, can be suitable for some sort of electoral document.

1.2.2 Historical Documents

Since the UNESCO World Heritage Convention in 1972 [110], the identification, protection and preservation of cultural and natural heritage around the world is considered to be of outstanding value to humanity. It is true that there have been huge efforts so far in preservation, but there is a long way to go. Just as an example, there are 300 billion cultural objects in Europe (e.g. books, photographs, statues, etc.), but *Europeana* [27] estimates that only the 10% of them (around 300 million resources) have been digitized [8], and from this amount, only the third part is digitally available online [44]. Moreover, in the specific case of historical documents, only a very small fraction of them are properly indexed, making the information contained really useful and accessible. Therefore, for the preservation and spread of cultural heritage, more efforts are needed not only in digitization but also in terms of indexation and access.

The first attempts to make available the contents of handwritten documents were based on handwritten text recognition and handwritten word spotting [85, 66]. Although converting a digitized document image into machine readable text is obviously a good step forward, the final goal is to extract the information contained to allow the access and search by contents. For this purpose, some level of semantic recognition and understanding is required. In fact, there is an increasing interest within the research community regarding information extraction and document understanding, with the aim to allow meaningful semantic access to the information contained in document collections [18, 24, 70, 100].

As stated before, when indexing historical documents, the extraction of their contents is of paramount relevance. Until quite recently in historical terms, most information was stored through handwritten documents. For practical reasons, the most common way to record information was using structured documents, such as the ones shown in Fig. 1.1. These examples, used in this work as application use cases, illustrate different records on individual people information (health, demography, etc.). Citizen-centered documents contain a complete, factual and reliable memory of the communities of the past. These manuscripts could either have the form of a table, where each row represents a record and each column contains a specific piece of information or just as a set of individual records in the

form of paragraphs. In both cases, each record will always contain information from a very restricted domain. For example, medical records contain information about the patient and the disease, whereas marriage records have information about husbands and wives like their names, surnames, occupations, birthplaces, their parents' names and occupations, etc. The goal of an information extraction system is to retrieve this information (i.e. named entities or proper nouns) from those historical handwritten sources, allowing to generate structured, indexable and semantically accessible databases. Thus, the knowledge is made available to scholars and citizens in general.

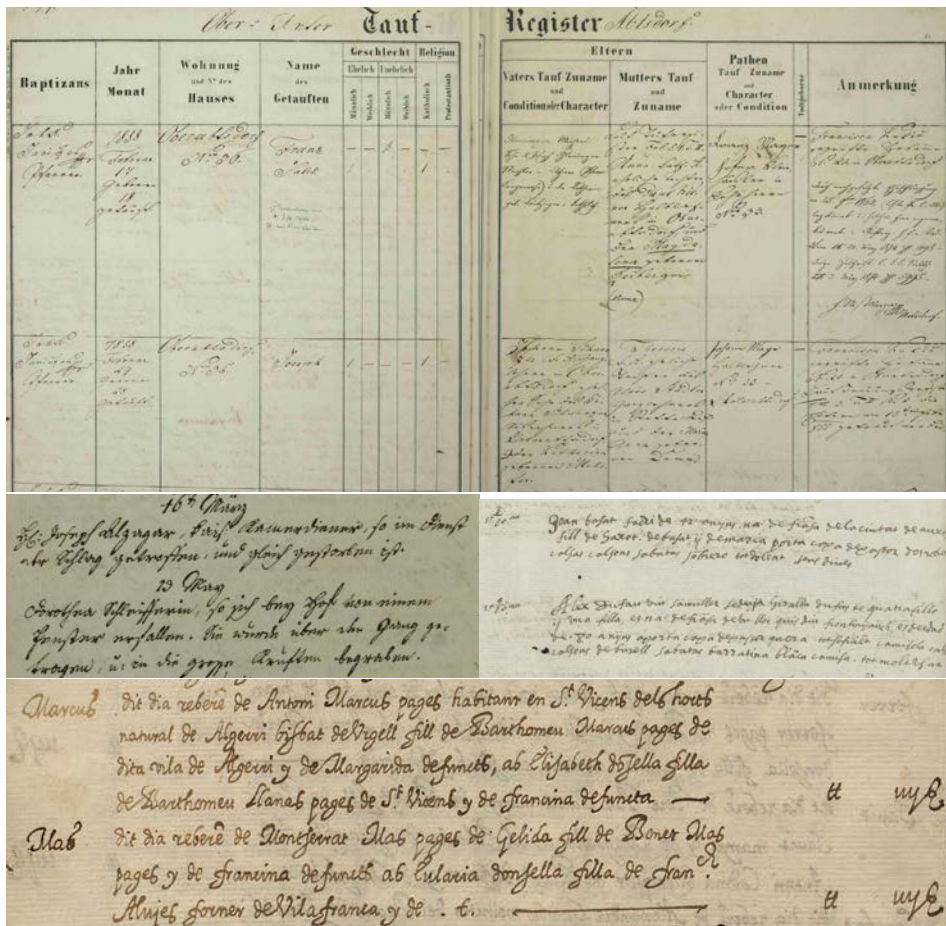


Figure 1.1: Samples of records from structured documents. Baptism registers from the Absdorf collection, 1853 (top). Death records from Wien, 1720 (left). Medical records from Sant Pau Hospital, Barcelona 1604 (right). Marriage records from the Barcelona Cathedral, 1619 (bottom).

1.3 Contributions

The main contributions of this thesis are the following:

- An in-depth case study of electoral documents, the challenges and the available options in their automatic processing with Document Image Analysis techniques.
- A system that is able to automatically process handwritten electoral tally sheets, reducing the operational costs and greatly speeding up the results consolidation process, with preprocessing, cropping, digit recognition and handwriting recognition.
- Two handwriting recognition methods. The first based on a combination of unsupervised feature learning with Variational Autoencoders and a Bidirectional Long Short Term Memory (BLSTM) recurrent neural network. And the second one based on Pyramidal Histogram of Characters (PHOC) attribute embedding and a Bidirectional Long Short Term Memory (BLSTM) recurrent neural network.
- A benchmark composed of a dataset and a set of metrics for the evaluation of Information Extraction approaches, that allowed us to organize a competition in an international conference.
- A method able to detect and semantically categorize entities from word images, without requiring any handwriting recognition system.
- Two variants of an information extraction approach for loosely structured handwritten documents with two components: a PHOC-BLSTM based handwriting recognition pipeline and a sequential word-images categorizer to detect and tag the relevant entities.

1.4 Outline

In the next chapter we present an in depth review of electoral documents. We present the most common challenges we face when trying to automatically process them and the most useful Document Analysis techniques that can be used to extract its information. We also present a system to process electoral tally sheets we developed for at the electronic voting company ScytL.

In the third chapter, we focus on Handwriting Text Recognition. We present two different approaches. The first one is based on the unsupervised training of a Variational Autoencoder to reconstruct small patches of unlabeled text. The hidden representation of the autoencoder can then serve as features to train a BLSTM+CTC network. On the second model, we adapt the attribute embedding

technique to be used in sequences by training a network to predict the PHOC representation of a word and then using it instead to predict the embeddings of small patches that are used to generate a sequence that will serve as the input to a BLSTM+CTC neural network.

In Chapter 4 we present a benchmark to evaluate Information Extraction approaches in loosely structured handwritten documents. We describe the dataset, the tasks to be done and the metric we designed to evaluate the performance of different approaches. This benchmark was also presented as an international Competition.

In Chapter 5 we present, in our opinion, the most important contributions of this thesis. First, a Convolutional Neural Network that is able to predict semantic categories of word images. Taking this CNN as a base, we propose to methods to model the contextual information at record level. The first one is based on adding the predicted label of the previous word to the current word image. The second method is a more general method that encapsulates the original CNN into a RNN that can leverage information from all the words in the current record and can predict, both a semantic label as well as the person in this record to whom that information relates to.

Finally, in the last chapter, we present the conclusions of the thesis and give an outline of possible future work related to the contributions made in this thesis.

Chapter 2

Electoral Documents

2.1 Introduction

As we stated in the introduction of this thesis, we are interested in electoral documents and forms is the most common type of election documents.

There are mainly two different approaches to electronically produce election results in paper based elections. The first approach is to directly retrieve the voters choices made to each individual ballot (See Fig. 2.2). The second one consists in automatically processing the tally sheets to retrieve ly results at polling station level. After performing the tally manually, election officers at each polling station must fill in and sign a form-like document, the tally sheet. The information contained in the tally sheet that will then be the base for the results consolidation process.

2.2 Preprocessing

In image processing, before trying to understand a document image, we can try to simplify the problem by removing some sources of variance. The same intensity value can sometimes represent a black pixel or white (background pixel) depending on the acquisition device. It is also very common to find different skews on each scan, due to small misalignments when feeding the paper sheet into the scanner. Finally the image can be noisy. We will discuss techniques to address each of these problems.

A key preprocessing step in most document analysis tasks is image binarization. That is, determining if a pixel of the image should be considered “black/foreground” or “white/background” depending on whether its darker or brighter than a certain

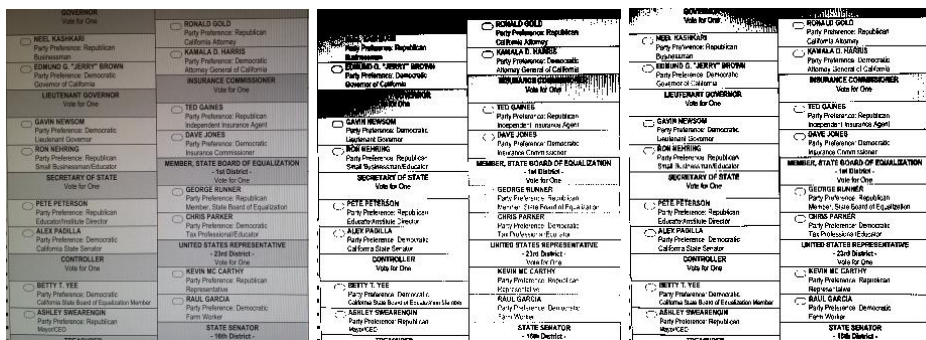


Figure 2.1: The original ballot image acquired with a camera (left). The image thresholded with Otsu’s Method (center) and with Sauvola’s Method (right). We can see how using Otsu’s method the darker areas of the ballot become black while voting targets in the lighter areas disappear, showing the limitations of setting a global threshold.

threshold value. If the image acquisition is done in a very controlled environment, a global threshold value can be predefined. This is the less flexible approach and it can result in issues if you have diversity of ink in your documents or you have to use scanners from different manufacturers with different contrast response. There are also several different methods to automatically find optimum global thresholds. One of the most widely used method is Otsu’s method [74]. This method is based on iterating through the 256 possible threshold values of a typical 8-bit gray level image finding the value that minimizes the intra-class variance (which is equivalent to maximizing the inter-class variance). This kind of methods would allow us more flexibility in the requirements of a particular scanner configuration. However this methods based on a global threshold will present problems if the scanning environment is not fully controlled and there are lighting variations throughout the page.

There are also adaptive threshold methods like Niblack [72], Bernsen [12] or Sauvola [90]. In this kind of methods, instead of selecting a single threshold value for the whole image, the threshold value is determined for each individual pixel, taking into account its neighbors in a local area of a predefined size. In the case of Sauvola, a widely used method for documents, the mean and the standard deviation in the local area are calculated. Then each pixel is classified as dark, if it is at least k times (a parameter) the standard deviation darker than the mean in that area. This kind of binarization methods are specially interesting if there are illumination changes (as for instance when the ballot images are acquired using a camera), noise in the image, or stains or folding marks in the ballot. A very interesting survey on both global and local thresholding algorithms can be found in [92, 81] showing that despite being a mature research area, there is still interest in the community for binarization techniques. See Fig. 2.1.

Another key preprocessing step is the removal of the skew; there are also several approaches to do this. One of the most common approaches [60], is based on rotating the document in all allowed skews (i.e, from -10 to 10 degrees with a precision of 1 degree), trying to find the right orientation. There are several ways to find out the correct orientation. Assuming an horizontal writing, the document will have the correct orientation when the horizontal projection histogram has a higher variance. Also, if there is a long horizontal line separating two areas, the correct orientation would be the one that produces the highest peak value for a specific line in the horizontal projection histogram. Another common approach would be to use the Hough Transform [46, 6, 94]. Using the Hough Transform we get the equation of all the lines $y = ax + b$ that we can find in a document, making it trivial to find the skew of the document. Nevertheless, mainly because of the computational cost of the Hough Transform, methods based on the horizontal projection are more commonly used.

In some cases, after thresholding and skew correction, some noise removing algorithms can be applied. For instance, the median filter can be useful to remove “salt and pepper” noise (isolated black or white pixels). Mathematical morphology operators [91] (opening, closing, erosion, etc.) can also be used in case we need to remove artifacts with a specific shape/size or connect some broken shapes.

2.2.1 Ballots

The most common election document is the ballot. Ballot design can have a high variability depending on the electoral system of each country or state. We can identify three big different scenarios: mark voting, preferential voting and write-ins. We will review each of them in the following subsections.

Mark Recognition

The ballots used in most of the elections consist of a grid where a voter selects k out of n candidates for each contest by filling in empty voting targets in predefined locations. In the most simple case, there will only be one ballot model. In this case, the recognition software will only require a mapping from a filled voting target (dark pixels in a certain area) to a candidate name.

However, in most complex elections we usually have to deal with different ballot models (in different languages, or different districts with different contests). In this scenario, the first step (after the preprocessing step) is to identify the ballot model. The most popular solutions use QR-codes or barcodes to identify each model. After reading the barcode and identifying the ballot model, the configuration for that particular model can be loaded, that is, the position of the pixels of each voting target and the candidate it is associated to and proceed to detect marks. But, sometimes detecting marks is not the same as detecting votes and we need to know

OFFICIAL BALLOT		
STATE GENERAL ELECTION BALLOT ANOKA COUNTY, MINNESOTA NOVEMBER 4, 2008		
INSTRUCTIONS TO VOTERS: To vote, completely fill in the oval(s) next to your choice(s) like this: <input checked="" type="radio"/>		
FEDERAL OFFICES	CONSTITUTIONAL AMENDMENT	CITY OFFICES CITY OF ANDOVER
PRESIDENT and VICE-PRESIDENT VOTE FOR ONE TEAM	Failure to vote on a constitutional amendment, will have the same effect as voting no for the amendment.	MAYOR VOTE FOR ONE
<input checked="" type="radio"/> JOHN MCCAIN AND SARAH PALIN <small>Republican</small> <input type="radio"/> BARACK OBAMA AND JOE BIDEN <small>Democratic-Farmer-Labor</small> <input type="radio"/> CYNTHIA MCKINNEY AND ROSA CLEMENTE <small>Green</small> <input type="radio"/> RÖGER CALERO AND ALYSON KENNEDY <small>Socialist Workers</small> <input type="radio"/> RALPH NADER AND MATT GONZALEZ <small>Independent</small> <input type="radio"/> BOB BARR AND WAYNE A. ROOT <small>Libertarian</small> <input type="radio"/> CHUCK BALDWIN AND DARRELL CASTLE <small>Constitution</small> <input type="radio"/> _____ <small>write-in, if any</small>	To vote for a proposed constitutional amendment, completely fill in the oval next to the word "YES" for that question. To vote against a proposed constitutional amendment, completely fill in the oval next to the word "NO" for that question. CLEAN WATER, WILDLIFE, CULTURAL HERITAGE, AND NATURAL AREAS. Shall the Minnesota Constitution be amended to dedicate funding to protect our drinking water sources; to protect, enhance, and restore our wetlands, prairies, forests, and fish, game, and wildlife habitat; to preserve our arts and cultural heritage; to support our parks and trails; and to protect, enhance, and restore our lakes, rivers, streams, and groundwater by increasing the sales and use tax rate beginning July 1, 2009, by three-eighths of one percent on taxable sales until the year 2034? <input checked="" type="radio"/> YES <input type="radio"/> NO	<input type="radio"/> MIKE GAMACHE <input type="radio"/> ROSELLA SONSTEBY <input type="radio"/> RICHARD EDWARD KULKEY <input type="radio"/> ERIC KOHNKE <input type="radio"/> _____ <small>write-in, if any</small> COUNCIL MEMBER VOTE FOR UP TO TWO <input type="radio"/> SHERI BUKKILA <input type="radio"/> JEREMY BOYER <input type="radio"/> GLENDA COOPER <input type="radio"/> MIKE KNIGHT <input type="radio"/> BRIAN HAUGEN <input type="radio"/> _____ <small>write-in, if any</small> <input type="radio"/> _____ <small>write-in, if any</small>
UNITED STATES SENATOR VOTE FOR ONE	COUNTY OFFICES	
<input checked="" type="radio"/> DEAN BARKLEY <small>Independence</small> <input type="radio"/> NORM COLEMAN <small>Republican</small> <input type="radio"/> AL FRANKEN <small>Democratic-Farmer-Labor</small> <input type="radio"/> CHARLES ALDRICH <small>Libertarian</small> <input type="radio"/> JAMES NIEMACKL <small>Constitution</small> <input type="radio"/> _____ <small>write-in, if any</small>	SOIL AND WATER CONSERVATION DISTRICT SUPERVISOR DISTRICT 1 VOTE FOR ONE <input type="radio"/> KARLA M. KOMEK <input type="radio"/> _____ <small>write-in, if any</small> SOIL AND WATER CONSERVATION DISTRICT SUPERVISOR DISTRICT 2 VOTE FOR ONE <input type="radio"/> JIM LINDAHL <input type="radio"/> KIM KOVICH <input type="radio"/> _____ <small>write-in, if any</small> SOIL AND WATER CONSERVATION DISTRICT SUPERVISOR DISTRICT 5 VOTE FOR ONE <input type="radio"/> VICI L. NASS <input type="radio"/> _____ <small>write-in, if any</small>	
UNITED STATES REPRESENTATIVE DISTRICT 6 VOTE FOR ONE		
<input checked="" type="radio"/> BOB ANDERSON <small>Independence</small> <input type="radio"/> MICHELE BACHMANN <small>Republican</small> <input type="radio"/> EL TINKLEBERG <small>Democratic-Farmer-Labor</small> <input type="radio"/> _____ <small>write-in, if any</small>		

Figure 2.2: An example of a ballot to be processed with OMR technology.

what is the relation between marks and votes.

According to [51], any mark near a voting target can be classified as either legal votes (if the law accepts them as indicating votes) or legally ignored (if the law considers them not to be votes). In addition, as shown on Fig. 2.3, independently of whether the mark is or is not considered a vote by the law, it may be classified

according to how the detector interprets it as reliably sensed (if every time that mark is seen it is counted as a vote), reliably ignored (if it is never seen by the scanner) and marginal marks (marks that may or may not be sensed).

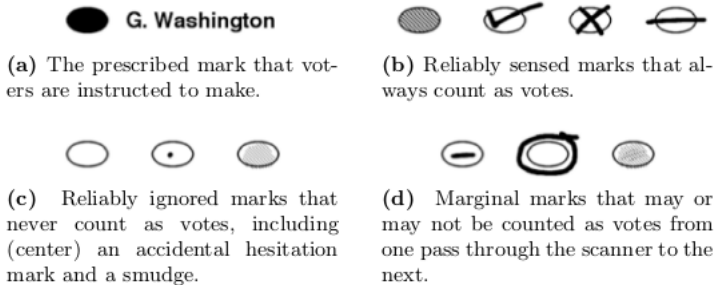


Figure 2.3: Different types of marks classified according to how a scanner interprets it [51].

There are in some states laws that enumerate the types of markings that are legal votes. For example, Michigan’s rules do not distinguish between the sensitive area and the voting target. They declare some markings to be legal votes that a scanner may miss, while declaring other marks to be legally ignored even though a scanner might count them, as illustrated in Fig. 2.4.

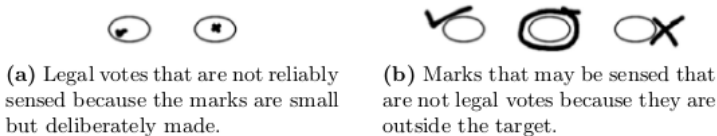


Figure 2.4: Different types of marks classified according to how the law interprets it [51].

In other states, as for example Minnesota, the definition of a vote is based on the voter intent. According to these rules, any kind of mark where the voter intent is clear, is a legal vote. On Fig. 2.5 an example of a legal vote that requires some interpretation. Voter intent is sometimes very hard to interpret, as several examples show in [1]. In any case, markings that appear to reflect a voter’s desires should not be disqualified for purely technical reasons [67]

That being said, we can safely detect most of the votes by being able to detect the most reliable marks, and we have been doing it for decades. Mark sense scanners, first developed for educational testing, have been used for ballot processing since the 1950’s. They were based on a ballot printed with a special ink, that is invisible to the sensor, and index marks to define the position of the voting targets. Infrared sensors were a common choice in these devices and voters couldn’t easily

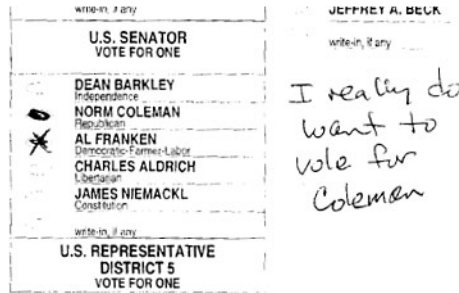


Figure 2.5: Voter intent is clear, according to some states this vote should be counted.

know if the ink of their pen would be visible to the infrared light; other devices required using a pencil. In most of these devices the sensible area does not have a sharp edge, that is, the sensor is more sensible in the center and less sensible near the edges of a target location.

In the 1990's, devices using imaging technology were developed. They used fiducial marks that allowed the scanner to interpolate the voting targets and counted the number of dark pixels in that area. A scanner based in this technology may fail to sense a mark if it is not big or dark enough. More recently, in 2006 a patent was granted to a device based on edge detection, that could detect empty voting targets (ovals) and filled voting targets. This kind of scanner would not be able to detect marks like X or checks, which are allowed in most electoral laws.

In general, Optical Mark Recognition can be considered a solved problem if we know where to look for the mark and users are required to use a prescribed mark. That is the case of educational testing. Few academic articles are being published on this subject nowadays, most of them focusing on building low cost alternatives with non-dedicated devices. For instance, in [88] they propose a threshold on the average greylevel of each target area to decide if it's filled or empty. However, since as we already discussed, most electoral laws allow different kind of marks besides the prescribed mark (like X or check marks), Optical Mark Recognition becomes more challenging.

A more recent approach to detect marks [98, 95] allow us to perform both the ballot model and mark detection at the same time, avoiding the need of barcodes. To do that, we need a template image of an empty ballot of each ballot model. The process would consist in computing the difference of the ballot and each of the templates, after preprocessing and carefully aligning them. The actual ballot model will have the smaller difference, and that difference would be the marks made by the voter. However, this difference will usually have an amount of noise due to small misalignments, dust or different scanning conditions. In order to deal with that noise, several approaches are discussed by the same author in [96, 98],



Figure 2.6: Three different marking styles: check, ex, and filled, and their corresponding noisy inputs generated by the voter attempting to erase a mark. Extracted from [118].

like using a distance transform to detect safe and unsafe zones, depending on their distance to black pixels, using Gaussian filters to smooth the images before performing the subtraction or using morphological filters. Another option to avoid false positives, would be to try to detect a grid for possible positions of marks by analyzing the geometry of the ballot [97], and restrict the mark detection to those areas.

One drawback of the approaches described above is that they mainly rely on the size of the mark. Usually, some voters do not follow exactly the instructions to completely fill the voting target area, and use marks like X or \checkmark (See Fig. 2.6). Since most electoral laws define a vote in terms of voter intent, we have to be able to detect these marks. A possibility suggested in [118], assuming that the voter makes consistent marks, is to train classifiers taking into account the style of the marks, improving mark detection.

A vote count system should also be able to detect cases of overvote (selection of more than allowed number of candidates) or undervote (selection of less than allowed candidates). Overvotes invalidate all the voter choices in a particular race, while undervotes are allowed. In some cases, a blank ballot (an special case of undervote) could be the result of the voter using an ink that the scanner fails to detect (red ink on infrared scanners for instance). Nowadays, most commercial scanners are able to detect completely blank ballots and overvotes in the moment of casting the vote, giving the voter a second chance [51]. This is easily dealt with in systems that work with a predefined set of rules. However it could require some level of understanding of the ballot document itself if we wanted to build a completely automatic system.

Finally, there is yet another kind of mark you might find on a ballot: identification marks. An identification mark is a mark made by a voter with the solely intent of making his voted ballot identifiable. To avoid coercion or vote-buying, if an identification mark is detected on a ballot, the whole ballot should be discarded. Since the definition of an identification mark is, once again, based on the intention of the voter, it is usually hard to decide if a mark or handwritten text on a ballot

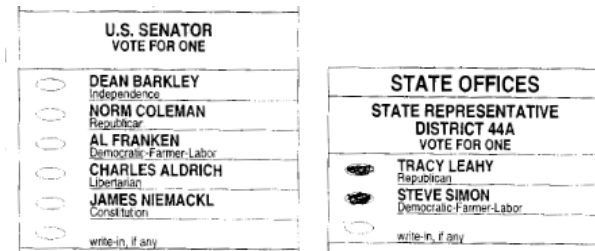


Figure 2.7: An example of undervote and overvote

is really an identification mark. Therefore, one advantage of systems based in the difference of marked ballots over ballot models over systems where we just look for a mark in a predefined area, is the ability to detect possible identification marks.

Another interesting use of OMR-like software is performing audits without requiring a predefined position of the voting targets. There has been some research on how to detect the position of the voting targets. Some authors propose detecting a grid for possible positions of marks by analyzing the geometry of the ballot [97]. Others simply require user collaboration to tag a blank voting target and then locate the rest using pattern matching techniques like Lucas-Kanade; after knowing where voting targets are, they sort them by the number of dark pixels and ask the user to select a boundary [114, 54].

Preferential Voting

In some elections the voter is allowed to perform preferential voting. In that scenario detecting a mark in a voting target is not enough. In preferential voting, the voter assigns a number to each candidate indicating their preference. In this case so we need to classify the marks we detect as belonging to a particular class (i.e. “1”, “2”, etc.).

The problem of identifying the particular class of an image among a possible set of classes is one of the classic challenges in computer vision, and specifically for handwritten numbers lots of work has been put since the 1980’s. In handwritten numbers the number of different classes is small (usually only ten different classes) and there have been free datasets available for years. The main challenge here is the huge difference in writing styles. Classifying handwritten isolated digits has been tackled by computer vision for the last three decades and there is now a wide variety of techniques that allow us to perform the recognition of individual digits with reliably [65] on the popular MNIST dataset [64]. See Fig. 2.8 for some examples.

Recently, a multicolumn convolutional deep neural network trained for weeks with several GPU has surpassed human performance in this task, achieving an error

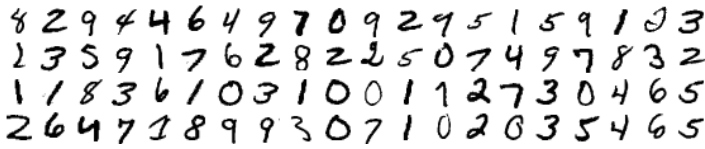


Figure 2.8: Some examples from the MNIST dataset. It's a common benchmark for isolated handwritten digit recognition consisting of 60,000 digit images from approximately 250 different writers.

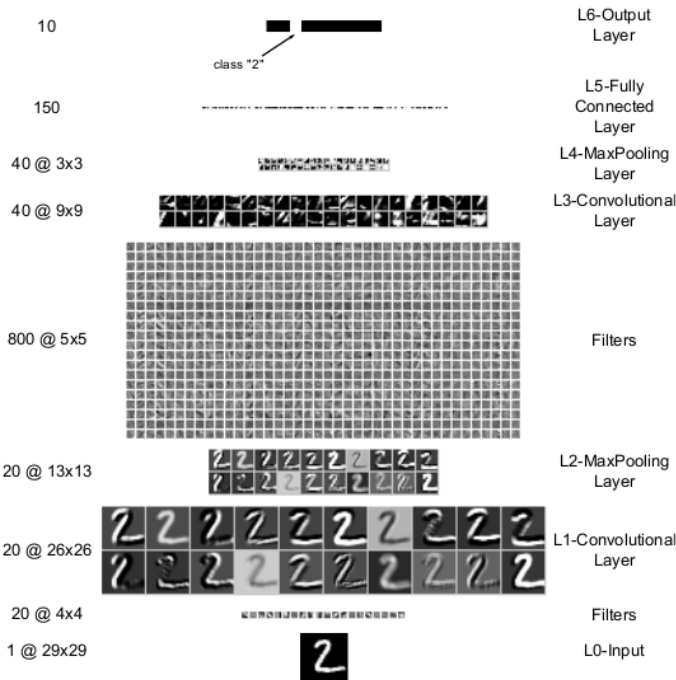


Figure 2.9: The architecture of one column of the convolutional neural network that achieved the best scores so far in handwritten digit recognition on the MNIST dataset. The response for each neuron to the input image is also shown as an image. Extracted from [22]

rate of 0.23% [22]. Convolutional neural networks combine the ability to learn low level features (convolutional layers) with the invariance to translation and scale given by max-pooling layers. Deep neural networks try to emulate the hierarchical representations of the human brain, where the first layers learn low level features, and the layers above learn higher level features (non-linear combination of the low level ones). The last layer is the actual classifier (a non-linear multiclass logistic regression) that outputs the probability of each class given that particular input image. See Fig. 2.9

In practice, these “deep learning” systems are sometimes difficult to train because they require longer training times, huge amounts of data and careful tuning of network hyperparameters. For that reason, traditional systems using hand-crafted features like Histogram of Oriented Gradients (HOG) [53] and classifiers like Support Vector Machines (SVM), which are also slightly faster in inference time, are still a very popular approach [25]. Most of these classifiers can also output the probability of the observation belonging to each specific class. This confidence level can be used to discard an ambiguous ballot and ask for a human decision if the confidence is below a certain threshold. This approach of combining Intelligent Character Recognition techniques with human inspection of dubious ballots has been used successfully in several elections in the Australian Capital Territory [3]. This is another important drawback of deep learning systems, that, as a consequence of the minimization of their loss function, they tend to produce overconfident predictions.

In preferential voting there is additional context information that can be used to further reduce the error rate. Usually a number cannot be repeated within the same contest (there cannot be two candidates with the same preference in the same contest) and usually they have to be correlative (i.e a voter cannot assign a preference “3” without previously assigning preferences “1”, and “2”). Instead of individual classifications, we are facing a problem of a set of observations with some restrictions that can help us lower our error rate even more. Finally, the number of preferences a single voter can choose is in most cases less than ten, that would reduce the number of classes (which has a great impact in error rates). For example, usually the digit 1 is mistaken by a 7, or the digit 3 with a 5 or an 8, so if we have less than 7 preferences to assign, the error rate would drastically decrease.

2.2.2 Write-in

Besides voting marks and preferential voting, we can find yet another different kind of vote in paper ballots. We are talking about the write-in areas (See Fig. 2.10). Write-in areas allow a voter to write in the name of their desired candidate, thus being able to cast a vote for candidates even if they are not listed as a voting option. A common way to implement it is in combination with a voting marks, requiring the voter to fill in a specific write-in voting target, and writing the name of the desired candidate to its right.

Recognizing the text in write-in areas is one of the most difficult problem we can find in electoral documents. Handwriting recognition can be performed with online or offline information. In online systems, the temporal sequence of the handwriting is available whereas in offline scenarios, we only have an scanned image available. While the recognition rate is better in the online scenario, we discarded its usage in our systems because: 1) it requires special hardware (a digital pen or digitizing board that records the (x,y) position of the pen tip at each timestep) and 2) it has security implications because it detaches the voter input

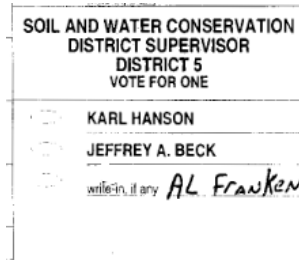


Figure 2.10: A write in vote where the voter did not fill in the oval as prescribed.

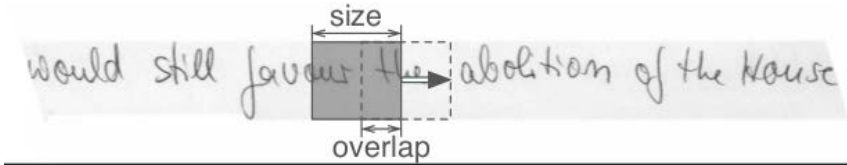


Figure 2.11: The “sliding window”. Extracted from [30]

from the ballot background, forcing to perform audits on the physical ballots to avoid ballot tampering.

Offline cursive handwriting recognition, with open vocabularies in a multi-writer scenario is still an open problem. One of the reasons why it is a much harder problem than printed text recognition (OCR) is ‘Sayre’s Paradox’. This paradox states that handwriting recognition is a “chicken-egg” problem because in order to properly segment a cursive word into characters you need to recognize the characters first, but to properly recognize the characters you first need to segment them out. A way to circumvent this problem is to use segmentation-free techniques. Another reason is the huge variability of cursive handwriting. Until recently most of the approaches include preprocessing steps trying to normalize the slant, horizontal and vertical size of the characters and, in some cases, even the stroke width.

One of the most popular approaches is to model the handwritten text line or as a temporal series of observations with a “sliding window approach”. See the example of the sliding window approach on a previously normalized handwritten text line in Fig. 2.11. That is, we focus our attention only in a column of a few pixels wide at a time and extract some representative features in that window. There are different set of features that are used in the literature, like statistical moments, the slope of the upper and lower contour, image derivatives, the number of black and white transitions, etc. Once we have the handwritten text represented as a series of features, the correct alignment with the ground truth character sequence has to be found. Since the character sequence and the feature sequence have different lengths the alignment is not trivial.

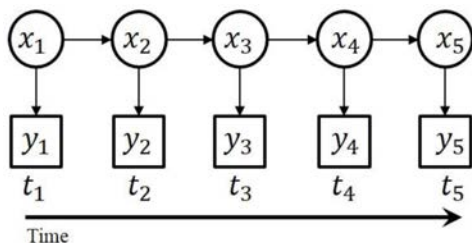


Figure 2.12: In Hidden Markov Models, the data is modeled as a series of observations generated by a hidden state that is only dependent on the state at the previous time step.

Since the 90s, technologies like Hidden Markov Models [82, 30] have been used to address this problem [78]. Hidden Markov Models are generative models that have been adapted from the speech recognition area. According to this model, each observation(x_t) in every timestep is conditionally dependent only from a latent unobserved variable (hidden state x_t), which in turn depends only on the hidden state of the previous timestep (Markov process). Given a number of states (x), a matrix T of allowed transitions among them $p(x_t|x_{t-1})$, and a parametric probability distribution P for $p(y|x)$, the Baum-Welch algorithm can be used to train the system, that is, finding the parameters for T and P that better fit our observations. A graphical representation of the HMM can be seen on Fig. 2.12

In 2009 a new algorithm was developed that allows us to use neural networks for segmentation free handwriting recognition. The algorithm, called Connectionist Temporal Classification (CTC) allows us to align two sequences of different lengths and return a differentiable error for each timestep. With the output from the CTC algorithm, and using the traditional backpropagation algorithm, it is possible to train a recurrent neural network to map the image feature representation with the character sequence. However, traditional recurrent neural networks have problems learning long sequences, because of a problem known as the vanishing gradient. After several timesteps, because the activation function of each neuron is smaller than 1, the error gradient fades into the network, making it unable to learn long range dependencies. This problem can be solved with the Long Short-Term Memory (LSTM) cells (Fig. 2.13), that incorporate input, output and forget gates, that the cell can learn to open or close depending on the input and the current state, thus allowing the network to learn arbitrarily long sequences.

The easier way to dramatically improve the recognition rate would be to change the write-in areas so that they are expected to be filled with a set of isolated capital letters. Also, in electoral documents we can assume that the content of the write-in area will be a name. We can then use a reduced vocabulary, consisting of the K most common surnames in that country, to improve the accuracy of the system in both the original connected handwriting and isolated character recognition scenarios. Finally, since the number of voters who actually use the

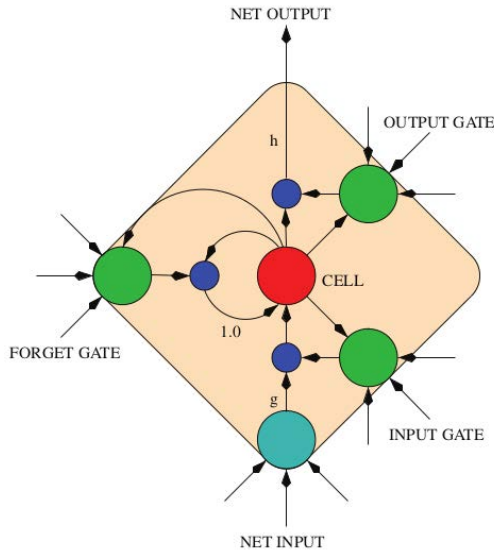


Figure 2.13: A Long Short Term Memory Cell with multiplicative input, output and forget gates. Extracted from [73]

write-ins area is usually low, there is also the option to simply detect the presence of write-in text, and mark the ballot for human inspection. This approach would still be better than current optical scan technologies, since they require the voter to fill in a mark associated to the write-in in order to process it. Requiring to fill-in that mark does not seem intuitive since, according to a study performed by Ji [50] conducted in Leon County elections with approximately 368.000 voters showed that 49% of voters who wrote something in the write-in region did not fill in the corresponding oval

2.2.3 Tally sheets

In some elections, with very simple ballot designs (e.g Partisan Ballot), processing the ballot is extremely easy, you just have to identify the party corresponding to each ballot. In that case, human tally at precinct level is feasible. After performing the tally, the electoral officials have to fill in a report, ballot statement or “tally sheet” with the election results for that precinct. We can see an example of such a document in Fig. 2.14.

These ballot statements usually contain handwritten numbers that represent the number of votes for a specific party, the number of eligible voters, etc. Intelligent Character Recognition techniques as the one described above for “Preferential Voting” in ballots can be used. In the case of ballot statements, some integrity

BALLOT STATEMENT / CERTIFICATE OF ROSTER – FILL IN ALL BLANKS WITH A NUMBER OR A ZERO
 FOLLOW DIRECTIONS FOR COMPLETING THIS FORM AS OUTLINED IN THE ELECTION MANUAL

		TOTAL
1. NUMBER OF BALLOTS RECEIVED <small>(hand count unless E&O or CV)</small>		525
2. NUMBER OF UNUSED BALLOTS <small>(hand count unless E&O)</small>		265
3. NUMBER OF VOTED BALLOTS – from ballot box		248
4. NUMBER OF VOTED BALLOTS IN PEACH PROVISIONAL ENVELOPES (from ballot box). Do not include "MAIL BALLOT WITHOUT ENVELOPE" provisionals - DO NOT OPEN ENVELOPES		10
5. NUMBER OF SPOILED BALLOTS (voter made mistake or damaged - from brown ballot carton for unused and spoiled ballots and ballot stubs)		2
6. TOTAL - ADD LINES 2, 3, 4 & 5 (The TOTAL entered on line 6 should match the TOTAL entered on line 1)		525
7. HOW MANY ROSTER PAGES ARE IN THE ROSTER ON:		
A. THE BLUE ROSTER PAGES (IF INCLUDED)	19	
B. THE WHITE ROSTER PAGE(S)	228	
C. THE PINK ROSTER PAGE(S)	1	
D. THE PEACH ROSTER PAGE(S)	11	
E. ADD LINES A, B, C, & D	259	
TOTAL SIGNATURES		259
8. ADD THE NUMBER OF VOTED BALLOTS (Total of lines 3, 4 & 5):		258
DOES THIS MATCH THE NUMBER OF SIGNATURES ON LINE 7?		
YES <input type="checkbox"/> NO <input checked="" type="checkbox"/>		
If NO, please tell us why they do not match:		
ONE PERSON SIGNED TWICE. SEE NOTE #1 ON NOTES PAGE.		
9. TOUCHSCREENS ENTER BALLOTS CAST FROM THE TOUCHSCREEN BELOW:		
A.	Touchscreen #1	No. of Ballots Cast
		0
B.	#2	0
C.	TOTAL - (A & B)	0

WE CERTIFY that the number of signatures on line 7e is the number of signatures in this roster of voters. All voters whose signatures appear in this roster voted today except where noted. This list of voters constitutes the roster of this precinct for this election. The total number of official ballots received and the number accounted for is as indicated on the ballot statement. The assisted voters list and the challenged list show a complete list of all voters assisted or challenged.

ALL BOARD MEMBERS MUST SIGN BELOW

FOR OFFICE USE ONLY	1.	
	2.	
	3.	
	4.	
	5.	
	6.	
	7.	

Figure 2.14: Example of a ballot statement. Extracted from [2]

checks could also be performed when recognizing the digits that can help to reduce even more the classification errors (or even help to detect election official errors). Spatial grammars can be defined for a ballot statement document, that is, numbers recognized in a certain area must meet some requirements. For instance, the sum of the recognized votes of all the parties and blank votes must match the number recognized as total votes cast, which in turn has to match the number recognized as number of voters, which has to be smaller than the number of eligible voters, etc.

Some ballot statements or tally sheets can also contain connected handwriting. Usually the numbers are also written in text form (like the courtesy amount in cheques). It is possible recognize this text with higher accuracy because of the very restricted vocabulary and syntax. Since recognizing the text “thirty four” and the number ‘34’ use different techniques to analyze different data, they can be considered independent probabilities, which can be easily combined to boost the confidence of the recognition.

To finish, usually, there is also an “observations” field, where the election officers can write free text to explain some anomaly during the election. As we explained above, unconstrained offline handwriting recognition is still an open problem. Since that field is usually empty, simply detecting if there are any observations, and asking a human operator for a transcription seems the best option.

2.3 Our approach for tally sheet processing

In this section, we present a system for tally sheet processing, reducing the operational costs and greatly speeding up the results consolidation process. A tally sheet is a form-like document combining printed information such as text, barcodes or ROI marks with handwritten text or digits. Several electoral commissions from different countries have shown interest in a system that can reduce the time required to process all the tally sheets that can seamlessly integrated into their traditional election processes. To our knowledge, a document analysis system specially designed to process handwritten tally sheets has not been described in the literature, so we decided to design one. Such a system should be able to deal with an extreme multi-writer scenario, given that each tally sheet will be written by a different writer we might have to deal with tens of thousands of different writers, on a country-wide election. In addition, the system should also be able to work with different scripts. We focused on documents like the one described in Fig. 2.15, where each tally sheet page is uniquely identified by a bar code, this allows us to retrieve from a database the candidates corresponding to each line of the tally sheet. From a document analysis perspective, it is only needed to extract all the lines in each tally sheet, since we already have the mapping to the candidates from the database. Finally, for each line, the goal is to recognize both the handwritten text and the digits.

2.3.1 Preprocessing


The preprocessing process consists of skew removal and the extraction of the different lines corresponding to each one of the different candidates. In order to perform the skew correction we will look at the different fiducial marks on the document (See Fig. 2.15. The location of the biggest fiducial mark allows us to tell if the page was scanned with the right orientation, while the smaller marks are used to correct rotations. In our case, we did not need to binarize the image.


Orientation and skew removal

The first thing to check is the size of the image. Since we are working with vertical tally sheets, we expect the height of our image to be larger than its width. If this is not the case (possibly due to wrong scanner configuration), the image has to be rotated by 90 degrees. Once we have a vertical image we perform a template matching with a black rectangle of a predefined size (244x64 pixels) in order to detect the biggest fiducial mark. If it is found on the first quadrant of the image we already have the correct orientation while if it is found on the fourth quadrant, we must rotate our image by 180 degrees. This step is required because images scanned upside-down are a fairly common.

The next step is finding the smaller marks in order to fix the skew. Using a

ACTA DE ESCRUTINIO


CONTROL N°. 02235183



PROVINCIA: _____ ZONA: _____
 CIRCUNSCRIPCIÓN: _____ JUNTA N°. 000 Página 1 de 3

Siendo las 17 horas 00 minutos, concluye el escrutinio de votos de la dignidad de Prefecta / Prefecto y Viceprefecta / Viceprefecto de esta Junta Receptora del Voto.

Llenar los casilleros utilizando el tipo de números, que constan a continuación: **1 2 3 4 5 6 7 8 9 0**

	VOTOS EN LETRAS	VOTOS EN NÚMEROS		
		Centena	Decena	Unidad
101 TOTAL FIRMAS Y HUELLAS DACTILARES QUE CONSTAN EN EL PADRÓN ELECTORAL (Total de Votantes)	Doscientos cincuenta y nueve	2	5	9
102 VOTOS BLANCOS (Papeletas en blanco utilizadas)	SeSENTA y tres	0	6	3
103 VOTOS NULOS (Papeletas anuladas utilizadas)	Cuarenta	0	4	0
104 TOTAL FIRMAS Y HUELLAS DACTILARES QUE CONSTAN EN EL PADRÓN ELECTORAL CON CASILLERO GRIS	Doscientos cincuenta y siete	2	5	7

VOTACIÓN OBTENIDA POR LOS CANDIDATOS

LISTAS	CANDIDATOS	VOTOS EN LETRAS	VOTOS EN NÚMEROS		
			Centena	Decena	Unidad
21 105	Candidato 1	Doce	0	1	2
23 106	Candidato 2	Ciento dieciséis	1	1	6
30-60 11 107	Candidato 3	VenTy ocho	0	2	8



Ux _____ FIRMA PRESIDENCIAL / E. JRV
_____ FIRMA SECRETARÍA / O. JRV

Figure 2.15: The fiducial marks in a tally sheet (highlighted in red) used to detect the orientation and skew also allow us to segment the Region of Interest for later handwriting and digit recognition steps.

convolution of the negated image and models of the different fiducial marks we can detect them. If we can not find all of the fiducial marks (usually due to the paper being misplaced on the scanner or partially folded) an error is raised. Using the Hough transform on the image containing only the detected fiducial marks we can find the skew and correct it if necessary.

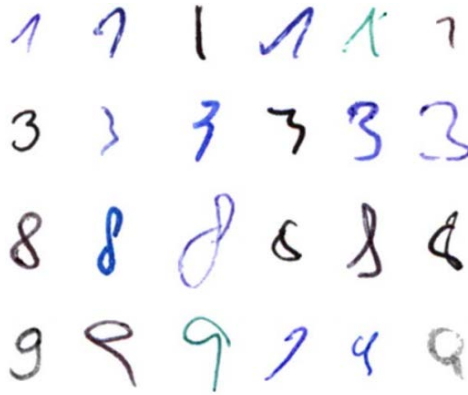


Figure 2.16: Some examples of digits from the CVL dataset.

Region of interest extraction and noise removal

Once we have corrected the orientation and the skew of the image, we can select the area delimited by the big fiducial mark and the ones on the lower part of the sheet. This area will be then divided in several lines using a predefined height parameter. And each line will be again divided into different fields. This predefined height is slightly bigger than an actual line in order to correctly capture the possible descenders that fall into the lower line.

Noise removal based on morphological operations is then applied on each one of the negated images of these regions. First we perform an opening with a 3x3 square structuring element, in order to remove noise from the image. On the result we apply a dilation with a rectangular structuring element to obtain the minimum number of connected components of 23 pixels height and width depending on the region. Finally we perform an opening with a 31x31 structuring element to remove regions that can not be considered digits or characters. Finally we will keep the regions with height bigger than a threshold $t_1=40$ pixels and a width bigger than one fifth of the region. These parameters have been determined empirically.

The Intelligent Character Recognition subsystem receives an individual character image from the cropping module and computes a description of the image based on Histogram of Oriented Gradients (HOG) features [53]. The feature description is then fed into a Support Vector Machine (SVM) based classifier that predicts the most likely digit with a confidence measure on that prediction.

2.3.2 Intelligent Character Recognition

We performed several experiments with support vector machine (SVM) on histogram of gradients (HOG) descriptor based digit recognition. First we trained

Training Data	MNIST	Tally Sheets
MNIST	1.00%	33.45%
MNIST+ CVL	1.92%	10.38%

Table 2.1: Average digit error rate on MNIST and TallySheets datasets.

with the MNIST dataset [64] using the proposed division of 50.000 digits for training and 10.000 for testing. We got an accuracy of 99% on the MNIST test set. We tested it on our internal dataset from electoral tally sheets, using the process described earlier on the paper to segment the 64596 individual digits from over 1000 different writers. On this dataset we got a 66.55% accuracy, with high percentage of the errors being on the number seven.

After some examination, we noticed that those errors where due to the existence of two very different types of number 7(See Fig. 2.17). We found an alternative, less popular but more recent dataset from an International Competition in Handwriting Digit Recognition from 2013 that had more images of type A number seven, the CVL dataset [23].We can see some samples in Fig. 2.16.

We performed then performed several experiments using all of the images from both the MNIST and CVL datasets, the performance was slightly reduced if evaluated only on the MNIST dataset getting a 98.08%, but when we tested this model on the dataset of real electoral tally sheets the accuracy went up to 89.32%. The results are summarized on Table 2.1. We also performed some experiments to determine if we could further increase the accuracy by splitting the class “number seven” into two independent classes, but the increase in performance was negligible. For all handwritten digits experiments, the size of the images was 28x28 pixels, and the descriptor had a bin size of 4.



Figure 2.17: Samples for type A number seven (left) from the CVL dataset [23] and type B number seven (right) from the MNIST datasets [64]

Even after increasing our training data with the samples from the CVL dataset, we can see that there is still a significant performance gap when compared to the MNIST results. We believe that the main reason for the difference in accuracy is probably due to the fact that the area assigned to digits in the tally sheet is usually much bigger than the digit itself, generating a variation in size and position that was not present in the original training datasets where the segmentation of the digits completely fill the image size.

We were also interested in evaluating the performance of this SVM+HOG ap-

proach on isolated handwritten characters. We decided to use the NIST Special Database 19. This database contains isolated handwritten letters both in its uppercase and lowercase form. The database is divided into three main folders. Two of them are used as train set and the other one as test, resulting in 47611 instances for training and 24684 instances to test. We evaluated two different scenarios: In the first scenario, we independently train the uppercase and lowercase characters (52 Classes), but we perform a case insensitive evaluation. For this scenario we get an average recognition rate of 86.53%. In the second scenario, we merge the uppercase and lowercase form of the letters during the training. That is, “p” and “P” are joined together as a unique class during the training. Consequently, during the test, they are considered the same. In this second scenario we got a recognition rate of 86%. For both handwritten character experiments, the size of the images was 64x64 pixels, and the descriptor had a bin size of 4.

2.3.3 Handwritten text recognition

For our tally processing system we designed a novel HTR subsystem based on automatic feature extraction and with BLSTM+CTC classifier. Since there is a special chapter in this thesis devoted to HTR, the details of this systems are discussed in the next chapter.

2.4 Conclusions

In this chapter we have done a review of different electoral documents and the challenges involved in their interpretation. We have also discussed the different processing techniques that have been used and pointed out a few simple state of the art techniques that could be use to leverage some specific characteristics of these documents. We have also described a simple system to process a specific type of document, the Tally Sheet. Electoral documents, as we already discussed, are a set of very structured documents where the semantics of each piece of information can be known from their position in the document be it a mark, a digit or handwritten text.

In electoral document processing, as in virtually all computer vision tasks, our final goal is to minimize the amount of human work required to perform a task. This automation should allow us to provide more consistent, fast and economically efficient results. However, after carefully exploring the subject, we notice that there is little room for substantial improvements in the methodologies used to process electoral documents. In modern paper-based election processing systems, human intervention is required mostly in two steps: First, defining the layout of the templates that will allow us to assign the semantics to each piece of information and second, make the final decision in some borderline cases where the interpretation by the automatic systems is doubtful.

Take the example of Optical Mark Recognition. Nowadays state of the art technologies can correctly interpret the huge majority of the ballots, but in the few borderline cases where the interpretation is uncertain, even humans have a hard time deciding and their decision has to be grounded in legal factors different depending on the specific election. Obviously some minor improvements in processing accuracy can be expected by incorporating state of the art techniques, however, this will not completely eliminate the need for human intervention but rather just slightly reduce it. Human intervention cannot be completely eliminated, in the general case, without legal changes that dictate strict rules for interpreting the ballots.

Automating the layout detection in order to build a system that could process different electoral documents without preconfiguration would be an ambitious and interesting line of research. However, this major rethinking of the whole process seems economically sound for these highly structured documents where the layout is always known a priori and the performance of traditional systems sets such a high standard.

Of course, building a real world system combines theoretical and practical issues that can make an apparently easy task challenging. Even something as simple as cropping from an strictly formatted page requires some level of tuning and tolerance, and even images scanned in a supposedly perfectly controlled scenario require denoising. There is also the degraded performance in switching from one dataset to another, even in a limited domain such as digit classification, which were due mainly to regional variations in the way to write the number 7 and the different margin areas around the digits.

After exploring the challenges in these kind of documents, we decide that in order to build Information Extraction systems from handwriting documents, it makes more sense to focus precisely in recognizing the contents of the different fields. In our case, the most challenging content we can find, from a research perspective, is handwritten text. For that reason we decided to focus on exploring new handwriting recognition techniques. Also, after studying electoral documents in depth, we decided to focus on historical handwritten text, specially marriage records. Although the change may seem huge, the reality is that the challenges are similar, since we are talking about recognizing text with a limited semantic scope and some level of structure, with the advantage that there are publicly available historical records and a bigger research community.

Chapter 3

Handwritten Text Recognition

In this chapter we review the state of the art in Handwriting recognition and present two new approaches. The first approach consists in an unsupervised training of a Variational Autoencoders, in order to extract low dimensional, descriptive features. The second approach is based instead, in training a Convolutional Neural Network to detect specific attributes of text patches. Finally, both approaches share the use of Bidirectional Long Short-Term Memory networks with Connectionist Temporal Classification loss for the sequence modelling of the different series of features.

3.1 Introduction

Offline Handwriting Text Recognition (HTR) is the task of converting a digital image of handwritten text into its textual transcription. This task has been a central challenge of the pattern recognition community for decades.

First attempts to recognize text did not made a distinction between printed, isolated characters or handwritten text. These methods were based in the segmentation of individual characters and their posterior recognition using Optical Character Recognition (OCR). In fact, the recognition of isolated handwritten digits was one of the first applications of convolutional neural networks back in the nineties [61]. But, recognizing individual characters and cursive text recognition are two completely different problems. This is because, for cursive handwriting, segmentation is a truly difficult problem, and Sayre's Paradox arises; that is, in order to be able to perform a good recognition you need to segment first, but to perform a good segmentation you need to recognize first.

In order to tackle this problem, segmentation free methods were proposed such as Hidden Markov Models (HMM) or, more recently, Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) with Connectionist Temporal Classification (CTC) loss [41, 113]. In these methods the recognition and segmentation are done at the same time, allowing to evolve from isolated character recognition to word and text line recognition. HMM-based approaches were the first ones to be successfully applied to sequences and have had substantial research efforts over the years resulting in quite reasonable performance levels [26]. Bianne *et al.* [13] built a handwriting recognizer based on HMM, decision tree and a set of expert-based questions. Bluche *et al.* [17] proposed a method of the combination of hidden Markov models (HMM) and convolutional neural networks (CNN) for handwritten word recognition. Gimenez *et al.* [37] provided a method using windowed Bernoulli mixture HMMs.

Two major problems were holding back the research in RNN. The first one was the vanishing gradient problem, that is, the fact that due to the internal working of the network cells, after several timesteps the gradient tended towards zero. This problem was produced because the activation functions used at the time, were either sigmoids in the range (0,1) or hyperbolic tangents in the range (-1,1) which caused a troubles in backpropagating the error after a few timesteps. This problem was addressed with Long Short-Term Memory networks [48] in 1997 by incorporating multiplicative input, output and forget gates, that allow the cells to ignore unimportant inputs keeping their internal state unchanged, making them specially suited for learning over long sequences.

However, there was still a second major problem, and that was the lack of a differentiable loss that could allow the training of RNNs in scenarios where the lengths of the input and the output had different lengths making it impossible to use traditional losses at each timestep. It was not until 2006 that Connectionist Temporal Classification (CTC) loss [42] was proposed to tackle this issue. In 2009, the same author proposed a model based in a combination of Bidirectional Long Short-Term Memory (BLSTM), processing the sequence forwards and backwards, and CTC loss for HTR [41] which outperformed the state-of-the-art HMM-based models. The use of LSTM with CTC became the state of the art in handwriting recognition and several different variations based on these technologies were proposed in the following years. The work of Krishnan *et al.* [58] performs word spotting and recognition by employing a Spatial Transformer Network (STN), BLSTM and CTC networks. Stuner *et al.* [101] provide a BLSTM cascade model using a lexicon verification operator and a CTC loss. Wigington *et al.* [115] perform word and line-level recognition by applying their normalization and augmentation to both training and test images using a CNN-LSTM-CTC network. One of the limitations of CTC is that it requires the output sequence cannot to have fewer time steps than the input, which is not usually not a problem in HTR tasks. Also it is a quite complex loss, difficult to parallelize for its execution in GPU. In fact it took until 2016, ten years from the original paper, until a successful GPU implementation was published by Baidu. Until then, CTC models required heavy traffic between

GPU and CPU to be able to train.

Recently, attention-based models have become of interest for the HTR community. This is in part motivated by the nuisances of CTC and their success in similar sequence related tasks like machine translation [104, 9], image captioning [119] or speech recognition [21, 10]. Some of these methods based on attention-models and LSTMs have been proposed to evolve from text line recognition to paragraph recognition, and thus performing a joint transcription and segmentation of text lines [16, 14]. With these new methods we see a trend opening towards end-to-end HTR. Exploring end-to-end HTR is of course a much harder problem, that requires much bigger and complex systems to be tackled but in return we can leverage information from a wider context, potentially resulting in a better accuracy.

As we see, handwritten Text Recognition is still today an important field of study within the DIAR community. Although impressive advances have been made in the recent years, there are some scenarios where HTR is not still able to produce satisfactory results. One of those challenging scenarios is that of Historical Documents. We have previously discussed in this thesis some of the difficulties in HTR, all of those are specially present in Historical Documents: Degraded and damaged paper with bleed-through that can make it difficult to segment the writing from the background, different handwriting styles, not just from different writers but also changing over the centuries, the scarcity of transcribed data to train systems or estimate good language models for different historical times, etc..

Given these difficulties of HTR, Word Spotting has been raised as an alternative to HTR. Word Spotting [38] is defined as the task of searching words in a document, where the query is a word image (query-by-example) or a text string (query-by-string). Thus, documents are not transcribed, but the information contained can be made accessible in retrieval scenarios.

Lately, a new family of Word Spotting methods have also shown their ability to recognize words. These methods are based on embedding the word image and its transcription into a common attribute space. In these approaches, word spotting consists in finding the nearest neighbors of a textual query in that space. This approach has been adapted to perform recognition by doing a reversed query-by-example word spotting into a given lexicon. That is, to embed the whole lexicon of words into the attribute space, and then, given a word image, embed it to find the closest word in the lexicon. Using a fixed length and low dimensional attribute representation known as PHOC as the common attribute space is a popular choice [5, 102] while a more recent work [57] proposed to learn such embeddings using a deep convolutional representation.

Recently, these embeddings have been integrated into a deep learning architecture, producing impressive results for handwritten word recognition [79]. However, these methods present several disadvantages when compared to traditional HTR. Since they are implicitly performing word classification, the main drawback is the requirement of a lexicon and their inability to deal with out of vocabulary (OOV)

words. This might seem a minor drawback for modern languages where huge lexicons are available, but it can be a problem in some scenarios. For instance, in historical documents, the amount of OOV words is usually high, and building a full lexicon might not be feasible. In addition, these methods are recognizing the word as a whole, so, by design, they depend on a good segmentation, and they cannot be extended to text lines.

In this chapter we present two contributions, relying on BLSTM+CTC networks. Since BLSTM and CTC are techniques that have shown their power in different sequence prediction tasks, we aim our efforts in the feature representation. The first contribution is a study of the use of Variational Autoencoders as a means to automatically discover features representative of handwritten text that should allow for an easier classification. One of the strengths of these approach is that unlabeled data can be used to train the feature extractor.

Our second contribution is a deep learning HTR method that adapts the attribute embedding to sequence learning. Concretely, we perform the attribute embedding of small pieces of text with a convolutional neural network (PHOC-Net) and then we construct a sequence of embeddings that are recognized by Long Short-Term Memory Recurrent Neural Networks with Connectionist Temporal Classification loss (BLSTM+CTC). Therefore, we benefit from the advantages of the attribute embedding, sequence learning and deep learning. As far as we know, this is the first work that attempts to combine both approaches by extending the attribute embedding to sequential recognition.

3.2 Unsupervised Feature Discovery based HTR

The work in [41] opened the way for neural network based approaches for HTR. In that work, a set of manually handcrafted features was extracted for each column of pixels of the text, and the resulting sequence of features was then fed to perform the sequence alignment and recognition with Bidirectional Long Short-Term Memory (BLSTM) recurrent neural network coupled with a Connectionist Temporal Classification (CTC) loss. After that the output of the recognizer is mapped to the word of the dictionary with smallest string edit distance. We believe that the Handcrafted feature extraction is the weak link in the chain because these features are usually designed for a specific alphabet and they lack a clear justification.

We propose a handwriting recognition approach inspired in the work of Graves [41] but incorporating a step of unsupervised feature discovery with variational autoencoders. A similar approach using unsupervised feature learning was recently published for alphabet independent OCR [89] with promising results. Besides the obvious differences between OCR and HTR, in their case they use Restricted Boltzman Machines, while we propose the use of Variational Autoencoders. Both perform similar feature discovery tasks but following quite different approaches.

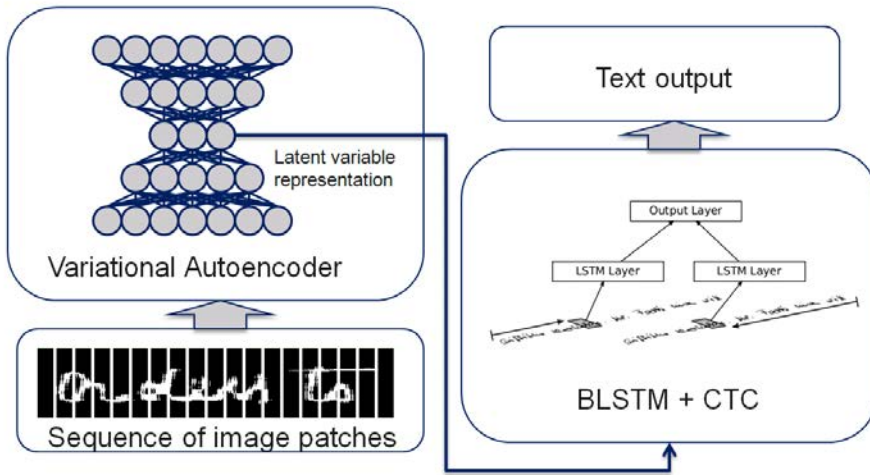


Figure 3.1: A sequence of image patches is passed through the encoder of a Variational Autoencoder to get a latent variable representation. This representation is then fed into a bidirectional long short term memory neural network to perform the final recognition.

The approach of Variational Autoencoders allows a faster and simpler training, with traditional backpropagation and has also shown to achieve lower reconstruction error. Figure 3.1 shows a schematic view of our handwriting recognition approach, which is described in more detail in the next section.

3.2.1 Unsupervised Feature Learning

Autoencoders are neural networks trained to reproduce its inputs at the output layer. In their most basic implementation, they consist of two layers, the encoder that takes us from image space into an internal representation and the decoder that does the opposite. In the most common scenario we want to learn an internal representation that is a lower dimensional representation of our data. This process can be seen as a feature extraction process [111] and has also been used in deep learning as an "unsupervised pretraining". Different architectures for autoencoders have been proposed recently, denoising autoencoders, sparse autoencoders, convolutional autoencoders, etc. One of the most promising at the moment is the Variational Autoencoder [55, 83] which has yielded impressively low reconstruction error with a really fast training times.

3.2.2 Variational Autoencoders

In order to learn about the underlying structure of our data \mathbf{x} , in Variational Autoencoders we assume that it was generated by an unobserved random variable \mathbf{z} . Since the marginal likelihood $p(\mathbf{x}) = \int p(\mathbf{z})p(\mathbf{x}|\mathbf{z})d\mathbf{z}$ is generally intractable, we can use variational inference in order to learn an approximation $q_\phi(\mathbf{z}|\mathbf{x})$ of the true posterior $p(\mathbf{z}|\mathbf{x})$.

The log-likelihood of each datapoint (example) can then be expressed as

$$\log p_\phi(\mathbf{x}) = \text{KL}(q_\mathbf{z}||p_{\mathbf{z}|\mathbf{x}}) + \mathcal{L}(\theta, \phi; \mathbf{x}),$$

where

$$\begin{aligned} \mathcal{L}(\theta, \phi; \mathbf{x}) &= \int q_\phi(\mathbf{z})(\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q_\phi(\mathbf{z}))d\mathbf{z} \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x})] \end{aligned}$$

That is, a sum of the KL divergence term between the true posterior $p(\mathbf{z}|\mathbf{x})$ and our approximation $q_\phi(\mathbf{z}|\mathbf{x})$, which is always positive, and $\mathcal{L}(\theta, \phi; \mathbf{x})$ a lower bound of the log likelihood of our data. Thus our goal will be maximizing this $\mathcal{L}(\theta, \phi; \mathbf{x})$. We can do this with standard gradient ascent algorithm using backpropagation thanks to the “reparametrization trick” proposed by the author in the original paper [55].

This “reparametrization trick” consists in modeling $q(z|x) \sim \mathcal{N}(\mu(x), \sigma(x)^2)$, and generating random perturbation $\varepsilon \sim \mathcal{N}(0, I)$. By doing so, we are able to sample from $z = \mu(x) + \sigma(x)\varepsilon$ in a way that is efficient and appropriate for differentiation with respect to our parameters.

In other words, we use Variational Autoencoders to learn a generative model of character parts or pseudo strokes. And we use the representation of those character parts in the latent variable space \mathbf{z} as our features.

From a practical point of view, we perform a height normalization on all the text lines of our training dataset, and then use a sliding window approach with a step to extract 20 pixel width, 120 pixel high, image patches from our dataset ignoring all label information. We then feed them to a Variational Autoencoder in order to find a lower dimensional latent representation. Once the autoencoder is trained, we can use the encoder weights to move from image space to this generative latent space that will constitute our features. It is worth noting that in this step, each image patch is treated as an independent sample, or datapoint, of the posterior distribution of a handwriting pseudo stroke image space.

3.2.3 Sequence Alignment and Recognition

After the unsupervised training has finished we use the same sliding window approach that we used to train the autoencoder to get an ordered sequence of image patches that represents each text line. Each of the image patches is then fed to our encoder to perform forward propagation in order to get a sequence of observations in the latent space. Each sequence, along with its transcription is now fed to a Bidirectional Long Short Term Memory (BLSTM) network with a Connectionist Temporal Classification (CTC) output layer [41] in order to get the transcription.

3.2.4 Experiments

We performed experiments with our Handwriting Recognition process on the George Washington database [117] composed of binarized and normalized text line images written in 18th century English language with two different writers splitting the dataset into train, validation and test. We decided to use the George Washington dataset because it is a standard database that allows us an easy comparison of the results with other state of the art works. The text lines were already normalized to a height of 120 pixels, we extracted individual patches of 20 pixels width with a step size of 4 pixels. These patches were used to train a Variational Autoencoder with an internal latent representation of 40 and 80 dimensions for a fixed amount of 100 iterations, which empirically showed to provide a good reconstruction error. The same patches of 120 pixels height and 20 pixels width were presented as a sequence of observations, with their labels to a standard BLSTM network with 100 cells that was trained until no improvement was observed on the validation set for 20 iterations. The experiments were repeated five times in order to reduce the impact of the random initializations of the neural networks. The network hyperparameters were selected to match those used by Fischer [29] obtaining very similar results.

The results shown in Table 3.1 were similar to the state of the art approach with Marti features using a descriptor of 40 dimensions and slightly better when using an 80 dimensions descriptor. In both cases the uncertainty due to random initializations was greatly reduced. The convergence time improved dramatically both in number of iterations and duration of the iterations, as shown in Table 3.2. The faster convergence is due to the reduction of the length of the sequences, by using one observation every 4 columns instead of each column. With regards to the convergence speed, the impact of the dimensionality of the features is negligible when compared to the length of the sequence.

Features	Avg CER	std
Marti Features	26.45%	2.12
VAE (40 dim)	26.66%	0.50
VAE (80 dim)	25.58%	0.97

Table 3.1: Average character error rate and standard deviation over five different experiments for each set of hyperparameters.

Features	Epochs	std	Epoch time
Marti Features	123.40	15.09	15 min
VAE (40 dim)	59.60	6.41	4 min
VAE (80 dim)	64.40	8.40	4.5min

Table 3.2: Average number of iterations required for convergence and standard deviation over five different experiments for each set of hyperparameters.

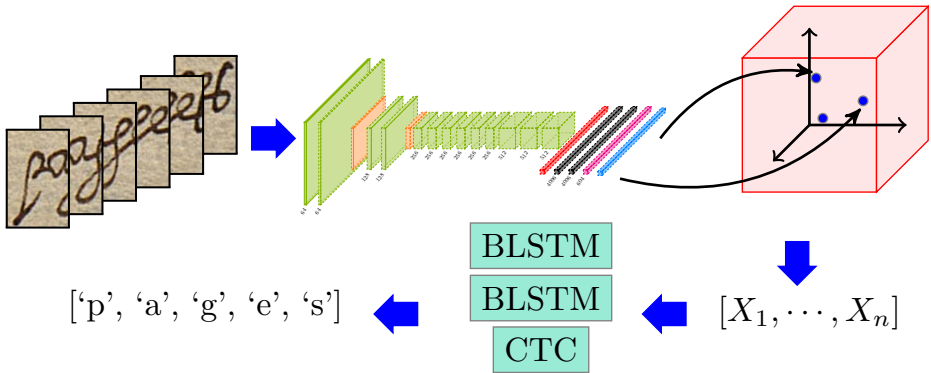


Figure 3.2: System architecture. After training a PHOCNet for word attribute embedding, we embed patches of word images into the attribute space. From these points in the attribute space we create a sequence that is passed to a two-layer BLSTM+CTC recurrent neural network that performs the transcription.

3.3 Attribute Embedding Based HTR

In this section we describe our two stage attribute based approach for handwriting word recognition. The first stage is based in attribute embedding, followed by the proper sequence transcription, which is performed over embedded text patches.

3.3.1 Attribute Embedding

The Pyramidal Histogram of Characters (PHOC) [5] is used to embed words into an attribute space. In this space, words and word images are characterized by a set of binary attributes. In the case of PHOC, each attribute represents the presence or absence of a character in a part of the word. Since the position of the characters is also important, the PHOC descriptor is built with a pyramidal structure as follows. In a scenario with n different characters, the first level of the pyramid will have $2n$ dimensions. The first n represent the presence or absence of each character in the first half of the word while the last n represent the presence or absence of a given character in the second half of the word. Each subsequent level of the pyramid will divide the word into smaller portions $1/3$, $1/4$ and $1/5$. The final level of the pyramid contains a selection of the k most common bigrams for that language. The final dimensionality of the PHOC descriptor will be $(2 + 3 + 4 + 5)n + k$.

While computing the PHOC embedding is trivial for text words, the embedding of word images requires learning. The original approach [5] consists in extracting SIFT features from the word-image, performing a Fischer Vector based clustering and finally training an individual SVM classifier for each attribute that outputs a likelihood of that word image containing a particular character in a given spatial position. Another possibility for PHOC embedding is to use PHOCNet, a deep convolutional network [102] that is trained to predict the PHOC representation of a given word image.

3.3.2 Extension to sequences

This kind of attribute embedding has shown to be a reliable representation of words. It has been effectively used for word spotting [5, 102] and recognition [5, 79] by comparing the predicted attribute representation of word images with the computed PHOC of all text words in a given known vocabulary. In our case we are interested in the evolution to sequence recognition, towards a lexicon free approach. The key observation is that a word image can be sometimes a prefix or a suffix of another word image. This means that this attribute embedding approach should also be able to correctly embed smaller patches of words. Then, if we can reliably produce attribute embeddings of arbitrary image patches, that we can extract, for instance, with a sliding window approach, we could use a sequence learning technique in order to learn to transcribe handwriting text.

To test our hypothesis we propose a method based on a modern deep neural network architecture. We start by training a PHOCNet as our attribute embedding choice for word images. Once the training is completed, we do a forward propagation of image patches in this network in order to build a sequence of PHOCs that is then fed to a two layer bidirectional LSTM recurrent neural network with CTC loss. A graphical representation of our proposed architecture can be seen in Fig. 3.2.

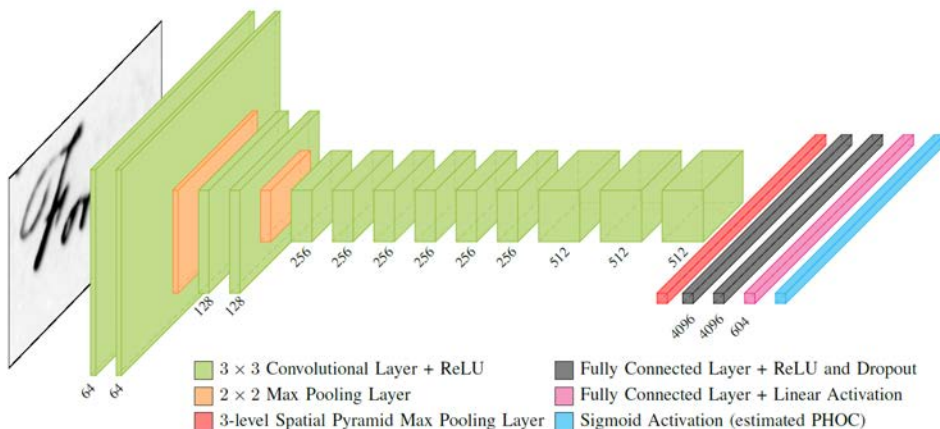


Figure 3.3: The architecture of PHOCNet. Best viewed in electronic format. Extracted from [102].

3.4 Network Architecture

In this section we describe the architecture of the two neural networks that take part in our method. We will discuss the most important characteristics of both the CNN for attribute embedding (PHOCNet) and the Bidirectional Long Short-Term Memory Recurrent Neural Networks with Connectionist Temporal Classification loss that is used for transcription (BLSTM+CTC). We will also provide facts that justify their performance.

3.4.1 PHOCNet

The PHOCNet [102] (Fig. 3.3) is a convolutional neural network architecture (CNN) used for word attribute embedding that has shown impressive results in word spotting. We use the PHOCNet not only because of its good performance but also because it is backed by a carefully thought and theoretically sound design.

Convolutional Neural Networks general layout can be split up in a convolutional and a fully connected part. The convolutional layers can be seen as a feature extractor while the fully connected layers act as a classifier.

Each one of the convolutional layers can be seen as a set of filters that are convolved with its input and followed by a non-linear activation function. These small kernels allow sharing weights for different spatial locations thus considerably reducing the number of parameters and helping in generalization [63]. Convolutional layers are combined with pooling layers in order to introduce a certain amount of translation invariance. In these layers, activations across their receptive field are pooled, and a single activation (usually the one with the maximum value) is

forwarded to the next layer [59, 93].

When stacking layers of convolutional and pooling layers, the first layers learn edge detectors that are gradually combined into more abstract features [120]. PHOCNet uses a low number of filters in the lower layers and an increasing number in the higher layers. This leads to the network learning fewer low-level features for smaller receptive fields that gradually combined into more diverse high-level abstract features. Also, all the convolutional layers in PHOCNet utilize filters of size 3x3, since they have shown to achieve better results compared to those with a bigger receptive field as they impose a regularization on the filter kernels [93].

One of the problems that arise when stacking a big amount of layers with traditional activations such as sigmoid or hyperbolic tangent functions is the Vanishing Gradient Problem [76]. This problem has been solved by using Rectified Linear Units (ReLU) as activation function [39]. This function is defined as the truncated linear function $f(x) = \max(0, x)$. By using the ReLU deep CNN architectures became effectively trainable as shown by [59].

The large number of parameters in fully connected layers make them prone to overfitting; even for larger training sets, co-adaptation is a common problem in the fully connected layers [47]. To counter this, various regularization measures have been proposed with Dropout [99] being one of the most prominent. Here, the output of each neuron has a probability (usually 0.5) to be set to 0 during training. Thus, a neuron in a given layer cannot rely on any single specific neuron activation from the preceding layer. This forces the network to learn alternative paths of activations leading to more robust representations and can be seen as an ensemble within the CNN model.

One of the key aspects of PHOCNet is the use of the Spatial Pyramid Pooling (SPP) Layer [45] over the last convolutional layer. This allows the network to process differently sized input images and output a fixed size representation avoiding the need of a potentially anisotropic rescaling or a cropping. This is crucial when working with word images where cropping is not an option, and, due to the important variability in size and aspect ratio, resizing would result in strong artificial distortions in character shapes and stroke width. In PHOCNet a 3-level Spatial Pyramid max pooling with 4x4, 2x2 and 1x1 bin sizes is used. This allows capturing meaningful features at different locations and scales within the word image.

Finally, it is worth mentioning that the network was trained for multi-label classification using the sigmoid activation function in its final layer with cross entropy loss, in contrast to the common single class classification with softmax and categorical cross entropy.

3.4.2 BLSTM+CTC

A natural way to deal with sequence learning in neural networks is with Recurrent Neural Networks. In fact, if we consider the resulting network after unfolding for a long sequence, RNNs can be seen as an extreme example of Deep Neural Network. Thus, for RNNs the vanishing gradient problem was known to be a showstopper since the early days of neural networks.

In order to deal with the vanishing gradient problem Long Short Term Memory networks [48] were designed in the late nineties incorporating multiplicative input, output and forget gates. These gates allow the cells to learn to ignore unimportant inputs while keeping their internal state unchanged, and decide when to produce an output, making them specially suited for learning over long sequences.

However, there was still the problem of sequence alignment when the input sequence and the target output were of different lengths. In the late 2000's an algorithm named Connectionist Temporal Classification [43] was invented. By introducing a *blank* "no-output" symbol and a simple algorithm to map network outputs to target sequences and vice-versa, this new algorithm allows the network to perform sequence alignment with differentiable errors. Thus, it allows the training with backpropagation for target sequences with equal or shorter length than the input sequences. Since then, this loss function has been successfully used in tasks like Speech Recognition [43] and Handwriting Recognition [41].

For a better robustness, two LSTM layers, processing the sequence forwards and backwards, are stacked forming a Bidirectional LSTM [41] layer. This kind of bidirectional layer is very useful for offline handwriting recognition where the full sequence is available from the first time-step. In our approach we stack two of these BLSTM layers, resulting in a total of 4 LSTM layers.

However, given the large number of parameters involved in the learning, RNNs are prone to overfitting. Given the success of Dropout [99] for deep neural networks, it is natural to try to use it for RNNs. But, if we use a special architecture like LSTM to keep an internal state for long time, we have to be careful when applying the dropout. One of the first successful attempts to use dropout in RNNs was done in [77] by applying dropout only to the non-recurrent weights. Recent advances in recurrent neural networks [34] allow us to use dropout in all of the connections of an RNNs in a theoretically sound by applying it to the same units at each time step, randomly dropping inputs, outputs, and recurrent connections.

Finally there are also some tricks, discovered empirically, that help to improve the training of LSTM networks [52] like initializing the bias of forget gates to 1 instead of 0 like the rest of the biases.

In this approach we rely on all these advances to design a two layer BLSTM with dropout applied to all its connections, followed by a mandatory Fully Connected layer to match the dimensionality of our output space.

Campbell, do this by If You
yourself to captain John hereof, - dispatch.
send to

Figure 3.4: Some examples of word images in the George Washington dataset. The available images are normalized and binarized.

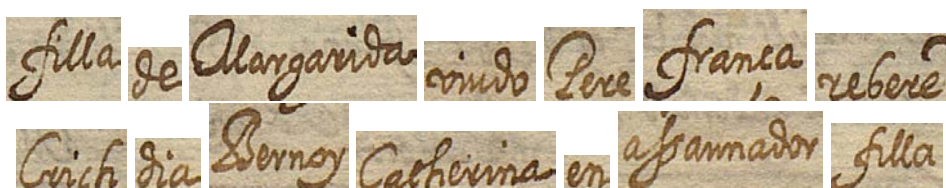


Figure 3.5: Some examples of word images in the Esposalles dataset. We see a high degree of variability both in image size and aspect ratio.

3.5 Experiments

In this section we describe the experimental validation of our proposal. We will first explain in detail the datasets used as well as some practical details relative to our training. We will then show and discuss the results.

3.5.1 Datasets

For our experiments we used two historical handwritten datasets, both in latin script. Next, we briefly describe them and show some examples below.

Washington Dataset

The George Washington (GW) dataset for handwriting recognition [31] is composed of 4894 word images written in 18th century English language with two different writers. The available word images were already normalized to a height of 120 pixels and binarized. We can see several examples of word images in Fig. 3.4. We use the first of the four different proposed partitions of the dataset which results in 2433 word images for training, 1293 for validation and 1168 for testing.

Esposalles Dataset

The Esposalles (BCN) dataset [28, 85] consists of historical handwritten marriages records stored in the archives of Barcelona cathedral. The book was written between 1617 and 1619 by a single writer in old Catalan. The data we used corresponds to 125 pages, with a total of 39527 word images and their transcriptions. The training set contains 100 pages (31501 word images) and the test set contains 25 pages (8026 word images). From the training set, we subtract the last 10 pages (3155 word images) to use them as validation. We can see several examples of word images in Fig. 5.3.

3.5.2 Experimental Setup

For each dataset, we trained the attribute embedding network PHOCNet with the hyperparameter values recommended in [102] for 30.000 iterations. For each experiment, the train, validation and test partitions are the same in all the stages (that is, the PHOCNet and BLSTM+CTC). This means that we train the PHOCNet with and only with the train samples that will later be used when training the BLSTM+CTC.

We used the network to precompute the PHOC of windows of 64 pixel width, with a step size of 8. Each word image was previously padded with 32 pixels of uniform background to the left and to the right to ensure a minimum sequence length for narrower images.

The sequence of precomputed PHOCs was then fed into a recurrent neural network consisting of two bidirectional LSTM layers with 250 neurons with Dropout probability 0.5 in all its connections, followed by a fully connected layer with the required number of neurons for each dataset ($82+blank$ for GW and $59+blank$ BCN). The loss function was CTC loss, which requires the extra *blank* symbol.

For training the recurrent neural network, we used early stopping with a tolerance of 30 epochs, and the optimization technique was done with Stochastic Gradient Descent with Nesterov momentum 0.9, a learning rate of $1e^{-4}$ and a decay factor of $1e^{-6}$.

The decoding of the network output was done by selecting the most likely prediction for each timestep and performing the CTC collapse operation, which consists in removing consecutive activations of the same character and the *blank* symbol.

For test evaluation, as well as for the early stopping in the validation set, we use the character error rate (CER) metric.

$$CER = \frac{S + D + I}{N}$$

The CER is calculated as the sum of the character substitutions, insertions and

deletions required to transform one string into the other, divided by the total number of characters of the longest string, resulting in a score between 0 and 1 for any pair of words.

3.5.3 Results discussion

Table 3.3 shows our results. We achieve a CER of 7.32% in the George Washington dataset and an impressive 0.83% in the Esposalles dataset. We observe a significant difference in the performance between both datasets. There are several factors that make these datasets different, starting by the amount of data which is crucial when training with neural networks. There is also the fact that the George Washington dataset is normalized and binarized. We hypothesize that these normalizations might remove some useful information. In fact, for humans, the images from the Esposalles Dataset (Fig. 5.3) are far more legible than the images from the George Washington dataset (Fig. 3.4). Finally we would like to remark that the Esposalles dataset consists in marriage records, which usually contain several instances of each unique word. Contrary, the George Washington dataset is a small size free text, and there are only a few instances of each unique word.

Trying to compare our method with other methods is not an straight-forward task. Both works in [5, 79] are based on attribute embeddings. From them, we chose the work from Almazan et al. [5] because it utilizes the original attribute embedding and its source code is publicly available. However the the original method and our approach have important differences, the main one is the requirement of a dictionary. For that reason we consider two different scenarios. In the first (worst case) scenario a lexicon is automatically built from the training examples, thus we will have a minimum lexicon that is always available. In the second (best case) scenario, the transcriptions from the test will also be included in the lexicon, representing a perfect lexicon, so all words can be found in the lexicon. In any case, in both scenarios our method outperforms the original approach by a big margin.

It is also worth noting that when training with the method from [5] only lowercase characters and digits are taken into consideration when building the PHOC representation. Contrary, in our method, all the characters present in the dataset are used, including uppercase, 'ç' and '#' (symbol that denotes crossed out words or characters) symbols for the Esposalles and punctuation marks for the George Washington dataset. Ignoring special characters like '.' or ',' makes the problem easier, whereas transforming all transcriptions to lowercase can help in some examples and damage in others. As a result, the comparison of these methods is even more difficult.

In order to minimize the penalization of the method described in [5] for not using special characters, we removed all punctuation marks from the ground truth, merged the ordinals (1st,2nd) with their corresponding digit and the special character for the initials GW was split into a 'G' and a 'W'. This is a strong simplification, since in the evaluation of our own method, we model and take into account

Method	Esposalles	GW
Almazan et al. [5] (Train Lexicon)*	6.18%	22.15%
Almazan et al. [5] (Full Lexicon)*	4.28%	17.40%
Fischer [29]	-	≈ 20%
Our approach	0.83%	7.32%

Table 3.3: Comparative with other methods CER.

the commas, dots, left and right parenthesis, etc. Even dealing with a much harder task, our method outperforms previous reports by a great margin.

A more direct comparison can be made with the traditional single layer 100 neuron BLSTM+CTC recurrent neural network like the one described in [41, 29]. This method does not require any kind of language model and trains the network from a sequence of a 9-dimensional handcrafted feature based on statistical and shape information. Results for word-level recognition for the George Washington dataset are reported in a plot in [29] showing a result slightly above the 20% level. In this case we are not certain if punctuation symbols were taken in consideration when calculating the CER.

In case of the Esposalles dataset, the comparison with other existing methods is even more difficult, because most of them perform the recognition at line or record level. The best approach so far is from Romero et al. [84] reporting a WER of 10.1% at line level using a lexicon and a language model. Our method achieves a 2.95% WER in Esposalles without any kind of lexicon or language model but, since we are working at word level, we can not make word insertions or deletions errors but only substitutions. These values, despite not being fully comparable might give a hint of the performance level of our approach.

3.6 Conclusions

In this chapter we presented two new handwriting recognition methods: A first method with unsupervised feature learning using variational autoencoders, and a second one based on an attribute embedding of patches of word images by a convolutional neural network. In both cases, those feature representations of handwritten text patches are then presented as a sequence to a recurrent neural network that produces the transcription. We showed that our first method could slightly improve the state of the art, held at that time by Marti Features in the character error rate, while also greatly reducing both the number of epochs needed to convergence and their duration. The uncertainty due to the random initialization of the network is also greatly reduced, therefore producing a more reliable system.

We think that there is still room for improvement in the use of autoencoders for handwriting recognition, we would like to explore different autoencoder architectures. However, for the specific case of handwriting recognition we came to the conclusion that we need to find a way to get more discriminative features. That is, autoencoders are trained with the objective of minimizing reconstruction error and can be discover very useful features for general images, but in the case of handwriting recognition, some very small differences in images can produce a different character label. For example, two images of lowercase letter “b” written in different cursive styles can have a huge difference while still representing the same character, while just a small stroke can make an “F” turn into an “E”. Therefore, we came to the conclusion that better feature discovery should be done by somehow incorporating information from the transcriptions that would force the features to become more discriminative. With our second approach to handwriting recognition we overcome the limitation of requiring a lexicon that attribute based models had, effectively moving the focus away from word-classification to a real handwriting text recognition. By incorporating information from the transcription into the training of the feature extractor we achieved very competitive results becoming the state of the art results in both of the historical handwriting datasets benchmarked. The most evident future research lines opened by this work are the extension to text lines (something that previous works based on attribute embedding where unable to do by design). In our case this should be possible if we are able to model the white space character between words either at the attribute embedding level or at the RNN sequence transcription level. A second possible improvement would be to make use of language models or lexicon information when available (but not as a requirement of the model).

By making this contributions to HTR we are now able to extract information when the semantic information is given, as it is the case, for instance, in form documents, where the location of a given word or set of words inside the form determines the meaning. However, we are interested in exploring beyond that, and going into less structured formats. We can think of some administrative handwritten documents, that have are loosely structured as a step forward from simple forms towards completely free text. For that reason in the next chapters of this thesis will be devoted to that goal. But before exploring how to derive meaning from handwritten word images, we need to have a dataset, with clearly defined tasks and metrics. In the next chapter we will discuss a new benchmark we proposed to the community for this kind of experiments, that will help us evaluate our final contribution.

Chapter 4

A benchmark for Information Extraction

In this chapter we present a new benchmark for Information Extraction from loosely structured handwritten documents. In this case, we use a marriage records dataset. We will discuss the details of the dataset, how we adapted it to be used as a benchmark and presented it in an international competition, that remains open. We justify why we think this is an interesting problem to the community, explain the different tasks we propose, the metrics we designed and show some results from the participants in the competition.

4.1 Introduction

The extraction of relevant information from historical handwritten document collections is one of the key steps in order to make these manuscripts available for access and searches. In this context, instead of handwriting recognition [33], understood as pure transcription, the objective is to move towards document understanding. Concretely, the aim is to detect the named entities and assign each of them a semantic category, such as family names, places, occupations, etc. Lately, the interest of the document image analysis community in document understanding, named entity recognition and semantic categorization is awaking, and several techniques based on Hidden Markov Models (HMMs) [84], Bidirectional Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) [4] and Convolutional Neural Networks (CNNs)[109] have been proposed.

A typical application scenario of named entity recognition in historical hand-

written documents is demographic documents, since they contain people's names, birthplaces, occupations, etc. In this scenario, the extraction of the key contents and its storage in databases allows the access to their contents and envision innovative services based in genealogical, social or demographic searches. With the aim to foster the research in this field and offer a benchmark for the research community we organized an international competition during the International Conference for Document Analysis and Recognition (ICDAR) 2017. After the conference, the competition remained and will remain open and continuous, so that researchers can upload their new results at any time.

To be able to develop this benchmark and organize a competition we based on a previously available dataset [85, 28] of handwritten marriage records that had been used for evaluating handwriting recognition. This dataset had been previously transcribed and semantically labeled, but the transcription was done to meet the needs of the humanities scholars studying these data, therefore it was not convenient for its use in machine learning problems.

Adapting the dataset to its use in a information extraction competition involved manually checking the transcriptions ensuring its consistency in both the semantic labelings and the transcriptions. Some effort was also invested in reducing the number of categories and persons labels by merging some of them and in simplifying the transcription by transforming superscript characters into regular characters. We also designed a relatively simple file structure to supply the data at both line and word level.

A key component of any competition is its metric. Our goal was to prioritize the semantic labeling while also taking into account the accuracy of the transcription. We designed our metric trying to make it fair and general enough so that it can be applied unchanged to different tracks, and both at word or line level.

In Sect. 4.2 we discuss the details of the Esposalles Dataset and in Sect. 4.3 we give some insight to the particularities of the version we published for the competition. In Sect. 4.4 we explain the different tasks that the participants in the competition are required to perform. In Sect. 4.5 we give a detailed explanation of the metric designed for the competition and in Sect. 4.6 we describe the different methods that the participants in the competition have submitted so far. Finally in Sect. 4.7 we draw some conclusions about the lessons learned during this competition.

4.2 The "Esposalles" Marriage Records

The "Esposalles" Marriage Records [85, 28] consists of 291 books stored in the archives of Barcelona Cathedral with information of approximately 600,000 unions celebrated in 250 parishes between 1451 and 1905. In addition to the marriage licenses, each book includes an index with all the husband's family names and

the page number where the marriage information appear. Each marriage record contains information about the husband’s occupation, husband’s and wife’s former marital status, socioeconomic position signaled by the fee imposed on them, and in some cases, fathers’ occupations, place of residence or geographical origin.

The structure of a marriage record tends to loosely follow a specific ”grammar“. Some anchor words (in bold) separate the different persons, as follows:

<husband> (son of) <husband’s father>**y** (and) <husband’s mother> **ab**
(with) <wife> **filla de** (daughter of) <wife’s father>**y** (and) <wife’s mother>.

In some cases, other persons may appear in the record. For example, when a widow is married again, the record may include information on the former husband. In those cases, the information of the wife’s parents usually disappears:

<husband> **fill de** (son of) <husband’s father> **y** (and) <husband’s mother>
ab (with) <wife>**viuda** (widow) <wife’s former husband>.

It must be noted that the above structures are generally loosely followed, but it is quite common that they present variations, specially omitting some information or in some cases adding extra information.

4.3 The IEHHR Competition Dataset

The Information Extraction in Historical Handwritten Records (IEHHR) Competition Dataset consists of handwritten records extracted from a volume of the Esposalles Marriage Records Database written between 1617 and 1619 by a single writer in old Catalan language. The data used for this competition corresponds to 125 pages, with a total of 1221 marriage records, with their transcriptions, semantic categories and person labels. See Fig. 5.7.

The training and test sets are composed of:

- Training set: 100 pages with a total of 968 marriage records divided in 3070 text line images or 31501 word images.
- Test set: 25 pages with a total of 253 marriage records divided in 757 text line images or 8026 word images.

For each marriage record, we provide:

- Images of segmented text lines.
- Images of segmented words.
- Text files with the corresponding transcriptions at word and line level.

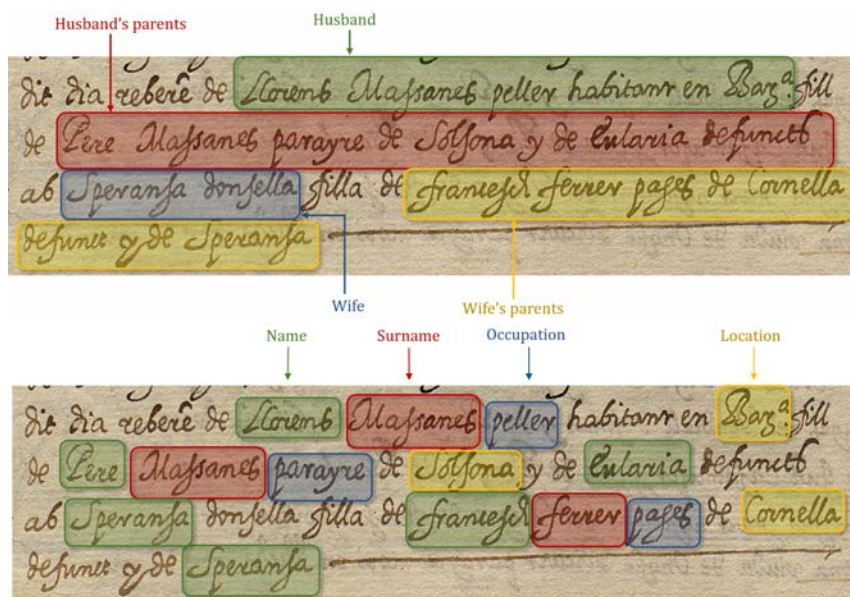


Figure 4.1: A marriage record information can be divided in pieces of information related to a specific person (top). Also, we can identify different “semantic categories” related to a specific person that are likely to appear in most marriage records (bot).

- Text files with the corresponding categories at word and line level: name, surname, occupation, location, civil state or other (for grammatical connectors and other irrelevant words).
- Text files with the corresponding person at word and line level: husband, husband’s father, husband’s mother, wife, wife’s father, wife’s mother, other-person (a different person from the ones mentioned before, for example, a former husband) and none (for grammatical connectors and other irrelevant words).
- A CSV file with the list of transcriptions, categories and associated persons of all relevant words.

Both train and test datasets are organized at record level. That is, there is one folder per record, containing the two folders with information at line and words level, and the corresponding CSV file. Since the final goal is to simulate the filling in of a knowledge database, as mentioned earlier, the CSV file only contains the relevant words of that record, i.e. the named entities. This means that only words with an associated category (e.g. names, locations, etc.) will appear in the CSV file.

Each word in the marriage record will have associated its transcription, person and category (one per word). However, take into account that some names and locations can be composed of several words, therefore in those cases, there can be more than one word per person/category. For those non-relevant words (e.g. conjunctions, prepositions, verbs, etc.) the category will be other and the person will be none. The transcription of these non-relevant words is also provided to facilitate training, but will not be taken into account during the evaluation. An example of the provided ground-truth is shown in Figure 2.

4.4 Task

The goal in the IEHHR competition is to extract information from the records. Concretely, the task is to recognize and transcribe the named entities, such as names, surnames, places, occupations, etc. In order to foster the participation in this competition, we simplified the number of semantic classes existing in the database. For example, the place of residence, geographical origin, etc. have been simplified to the same semantic class location.

As a result, relevant words can belong to five different categories: “*Name*”, “*Surname*”, “*Occupation*”, “*Location*” and “*Civil state*”. These semantic categories are associated to seven different people relationships or "person": “*Wife*”, “*Husband*”, “*Wife’s father*”, “*Wife’s mother*”, “*Husband’s father*”, “*Husband’s mother*” and “*Other person*” (usually the former husband of the wife in case she is a widow). Non-relevant words are labeled as “*Other*” and assigned to the person “*None*”.

For this benchmark, we have manually labelled the marriage records with semantic information at word level. The lines and the records in this dataset have been also manually annotated. In this way, each line is associated to its corresponding record.



Figure 4.2: An example of the IEHHR Competition Dataset: “Esposalles” at word-image level, showing word images and their corresponding, transcription (black), category label (blue) and person label (orange).

Participants are required to provide, for each test record, a CSV file following the format of the ones provided in the training set. That is, including the transcription of the relevant words (i.e. named entities) and their semantic category. Providing the person associated to each category is optional. That is, there are two different tracks for the competition that the participants can follow:

- Track Basic. The CSV should include the transcription and the semantic category (name, surname, occupation, etc.).
- Track Complete. The CSV should include the transcription, the semantic category and the person (husband, wife, wife’s father, etc.).

4.5 Metrics

The evaluation is done at marriage record level. Since the focus of the benchmark is on information extraction, the semantic label is prioritized. This means that irrelevant words are not taken into account and a perfect transcription of a word with an incorrect semantic label does not score any points. Contrary, if the semantic label is correct, then the Character Error Rate (CER) will be used to evaluate the quality of the transcription. Concretely, for each semantically labeled word:

- Track Basic. If the category is incorrect, then the score is 0. Otherwise, the score is a normalized accuracy metric on the transcription.
- Track Complete. If the category or the person are incorrect, then the score is 0. Otherwise, the score is a normalized accuracy metric on the transcription.

Note that the score at category level in both tracks is the same, so these values are directly comparable, allowing that participants in the Complete track to be automatically qualified to participate also in the Basic track.

More precisely, the evaluation procedure is the following. First, we check that the submissions are syntactically correct, that is, that one and only one CSV file is provided for each record, that it has the right number of comma separated values and that all the categories, person and record id's are valid.

We define the concept of “semantic label” as category in the basic track and the concatenation of the category and person in the complete track. Then, for each semantic label in each record, we retrieve two lists of word transcriptions: one from the submission (S) and another one from the groundtruth(T). The Character Error Rate (CER) is then calculated for each pair of submission and ground-truth words as the Levenshtein Distance between the two words, normalized by the length of the longest transcription in order to have a value between 0 and 1. Afterwards, the best word alignment for each “semantic label” is determined with Dynamic Time Warping, and the average CER for that labeling is computed. The accuracy score for each labeling is then defined as one minus that average CER (See Eq. (4.1)) Finally, we calculate the record accuracy as the average of the all the labeling accuracies found in the record. If the participant found a “semantic label” not present in the record or failed to find one that is actually present a score of 0 will be assigned to that labeling in order to compute the average.

$$\text{ACC}(S, T) = 1 - \frac{\text{DTW}(S, T)}{\max(n, m)}, \text{ where}$$

$$\text{DTW}(S, T) = \begin{cases} 0 & \text{if } S = T = \emptyset \\ 1 & \text{if } S = \emptyset \text{ or } T = \emptyset \\ \text{CER}(s_0, t_0) + \min \begin{cases} \text{DTW}([s_1 \dots s_n], T) \\ \text{DTW}(S, [t_1 \dots t_m]) \\ \text{DTW}([s_1 \dots s_n], [t_1 \dots t_m]) \end{cases} & \text{, and} \end{cases}$$

$$\text{CER}(s_i, t_j) = \frac{D(s_i, t_j)}{\max(\text{len}(s_i), \text{len}(t_j))} \quad (4.1)$$

The final score is the average of all the record accuracies. In addition to this final score, the average scores for each one of the categories are also computed and published. For better visualization, all these values are normalized between 0-100.

Method	Segmentation	Basic Score	Complete Score
Context-aware Neural Model	Word	94.62	94.02
Resnet based uni-gram	Word	94.18	91.99
CNN based Bi-gram	Word	87.58	85.74
Baseline CNN	Word	79.42	70.20
Naver Labs	Line	95.46	95.03
CITlab ARGUS (OOV)	Line	91.94	91.58
CITlab ARGUS2 (OOV)	Line	91.63	91.19
Joint HTR + NER	Line	90.59	89.40
CITlab ARGUS (no OOV)	Line	89.54	89.17
Baseline HMM	Line	80.28	63.11

Table 4.1: Results of the different methods

4.6 Results

Up to the moment of writing the thesis, the competition has received eight public submissions adding to the two baselines proposed by the organizers. Three methods work at word level and six work at line level. All of the methods opted for the Complete track, that is, providing transcription, category and person labels for each relevant word in each record. In Table 4.1 we show the competition score for the different methods. Despite solving the same problem and being evaluated with the same metric, one could argue that word and line level approaches are not directly comparable. For that reason we highlight the higher score for both image segmentation choices.

In Table 4.2 we show the average scores computed at category level, that is, instead of averaging all of the categories at record level and performing the global average, we compute a global average for each category individually.

It can be observed that the categories with fewer number of different words (i.e. vocabulary size) tend to have a higher performance. For example, the civil state has very few different words to describe it, therefore, making it easy to detect. Contrary, categories with large vocabulary, or even with many out of vocabulary words (such as surnames), tend to obtain a lower performance. The number of samples is also an important factor. For instance, names consistently show a higher performance than surnames because they are always present while surnames are quite commonly omitted when several members of the same family appear in a record.

	Method	Name	Surname	Location	Occupation	State
	Context-aware Neural Model	95.49	91.32	95.18	93.89	97.21
	Resnet based uni-gram	95.68	91.23	94.93	93.77	95.35
	CNN based Bi-gram	91.82	69.19	89.36	91.04	97.82
	Baseline CNN	83.01	65.25	66.31	86.26	97.68
	Naver Labs	97.01	92.73	95.03	96.43	96.41
	CITlab ARGUS (OOV)	95.14	85.78	88.43	93.08	97.54
	CITlab ARGUS2 (OOV)	95.09	85.84	87.32	92.96	97.19
	Joint HTR + NER	89.94	84.07	90.71	92.10	96.59
	CITlab ARGUS3 (no OOV)	94.37	76.54	87.65	92.66	97.43
	Baseline HMM	81.06	60.15	78.90	90.23	93.79

Table 4.2: Table of results iehhr.

4.7 Conclusions

We designed a benchmark for Information Extraction from historical documents, aiming to raise the interest in semantic recognition and categorization, as a first step towards the understanding of handwritten documents. This benchmark was presented to the community as an international competition co-hosted with the ICDAR conference.

We kept the competition open and continuous since then and, even after the initial period, researchers have continued to upload new results to the competition web platform. This good reception by the community proves our initial hypothesis that Information Extraction is indeed an interesting problem for the community. It seems obvious to us that, in order to produce more intelligent reading systems, able to understand and extract the information contained in handwritten documents, we need to go beyond pure transcription. Therefore we believe that this competition/benchmark goes in the right direction and has been a good contribution to the community.

As future work, we would like to expand the dataset with more pages, if possible from other tomes of the collection. Using data from another tome from one century later, for example, would be interesting. As time goes by, new occupations arise or new names and surnames appear due to migrations. These new words would increase the ratio of out-of-vocabulary words in the test set making the Information Extraction task more challenging. Of course it would also be interesting to collaborate with other groups in designing similar benchmarks from completely different collections.

Chapter 5

Information Extraction from handwritten documents

With all the tools that we have acquired so far, in this chapter, we face the challenge of extracting information from a loosely structured purely handwritten documents. We have already covered the transcription in previous chapters, so we need to assign semantic value to each word. At first, we propose a novel method based on Convolutional Neural Networks that is able to semantically classify individual word images, that is later modified to be able to capture contextual information from its neighbouring word images at record level.

5.1 Introduction

We have already previously argued that the identification, protection and preservation of cultural and natural heritage around the world is considered to be of outstanding value to humanity since the UNESCO World Heritage Convention in 1972 [110]. There have been huge efforts so far in preservation by means of digitization. However, at least in the specific case of historical documents, most of them are only available as scanned images. Only a small fraction of them are properly indexed, making the information contained therein really accessible and therefore usable. Therefore, the extraction of relevant information from historical document collections is a key steps in order to make the contents of those documents available for access and searches.

Document Image Analysis and Recognition (DIAR) is the pattern recognition research field devoted to the analysis, recognition and understanding of images of

documents. Within this field, one of the most challenging tasks is handwriting recognition [33, 41], defined as the task of converting the text contained in a document image into a machine readable format. Indeed, after decades of research, this task is still considered an open problem, specially when dealing with historical manuscripts. The main difficulties are: paper degradation, differences in the handwriting style across centuries, and old vocabulary and syntax.

Generally speaking, handwriting recognition relies on the combination of two models, the optical model and the linguistic model. The former is able to recognize the visual shape of characters or graphemes, and the second interprets them in their context based on some structural rules. The linguistic model can range from simple n-grams (probabilities of character or word sequences), to sophisticated syntactic formalisms enriched with semantic information. In this paper we focus in this last concept. Our proposed hypothesis is that in certain conditions where the text can be roughly described by a grammatical structure, the identification of named entities can boost the recognition in a parsing process. Named entity recognition is an information extraction problem consisting in detecting and classifying the text terms into pre-defined categories such as the names of people, streets, organizations, dates, etc. It can also be seen as the semantic annotation of text elements.

Converting a digitized document images into machine readable text is obviously a good step forward. For that reason, the first attempts to make those handwritten documents contents available were based on handwritten text recognition and handwritten word spotting [85, 66]. The fact that Historical Documents is one of the most challenging scenarios for handwritten text recognition, with degraded and damaged paper, bleed-through, different handwriting styles, the scarcity of transcribed data, made word Spotting [38] raise as a simpler alternative to HTR.

However, we should keep in mind that a mere transcription is not the final goal, but a means to achieve the understanding of the manuscript that allows us to extract the information contained in those documents, allowing us to access it and search by contents. For document collections in archives, museums and libraries, there is a growing interest in making the information available for accessing, searching, browsing, etc. A typical example can be demographic documents containing people's names, birthplaces, occupations, etc. In this application scenario, the extraction of the key contents and its storage in structured databases allows to envision innovative services based in genealogical, social or demographic searches.

Now that HTR is reaching a reliable level of performance in most documents, the research community is starting to show more interest in higher level tasks such as information extraction and document understanding, with the aim to allow meaningful semantic access to the information contained in document collections [18, 24, 70, 100].

A traditional approach to information extraction would be to mimic the hu-

man behaviour, i.e. to try to transcribe each record, and then to match each of the words with a set of field specific vocabularies (dictionaries of male/female names, surnames, etc), with the help of grammars or some other NLP (Natural Language Processing) techniques to detect named entities. This seemingly simple approach has several drawbacks. Moreover in historical handwritten documents, handwriting recognition the transcription itself can be problematic. While it is true that the transcription of modern printed text can be considered a solved problem this is not the case for historical handwritten text due to difficulties like variability of writing styles through centuries, specific vocabularies, paper degradation, show-through, etc. The second issue is Named entity detection [69] by the means of using vocabularies to cluster these sometimes unreliable transcriptions, because these methods usually have problems to deal with out of vocabulary words, specially if, like in our case, one wants to detect entities that do not start with a capital letter (e.g. occupations).

An alternative strategy follows the principles of cognitive reading. It consists in classifying handwritten text word images into different semantic categories (like names, surnames, locations etc.). The transcription is performed afterwards, predicting the text constrained to the word categories. This seemingly counter-intuitive approach is in fact also used by humans when trying to understand documents that are difficult to read. Indeed, it is known that the human cognitive system is able to make sense out of distorted information when context information can be used. Moreover, we humans are able to integrate information from different sources using different strategies [11] to make sure that we are correctly understanding a text. In our case, a possible example would be that, if at a certain position in a sentence it makes sense to read a male name, then its transcription is most likely "John" instead of "born", although the word shape can be indeed closer in appearance to "born".

Another option is to transcribe and detect the named entities at the same time. The method described in [87] uses Hidden Markov Models and category n-grams to transcribe and detect categories in demographic documents, obtaining a quite good accuracy. However, the method is following a handwriting recognition architecture, and thus it depends on the performance of the optical model, it needs sufficient training data, and it is unable to detect or recognize OOV words. Recently a new method based on Recurrent Neural Networks tries to following a similar approach was presented in [18].

A third alternative is to directly detect the named entities from the document image, avoiding the transcription step was recently published [4]. They use a traditional handwriting recognition approach, composed of a preprocessing step for binarization and slant normalization, and then extracting handcrafted features that are then fed into a BLSTM [48] (Bi-directional Long Short-Term Memory Blocks) neural network classifier. Afterwards, they use some post-processing heuristics to reduce false positives. For example, discarding short words or words starting by "Wh" or "Th" because they are more likely to be capitalized because they are

the first word in a sentence . The performance of the method is quite good, but its goal is only detecting named entities in uppercase and not categorizing these words. Moreover the post-processing heuristics of this method are specific for the English language.

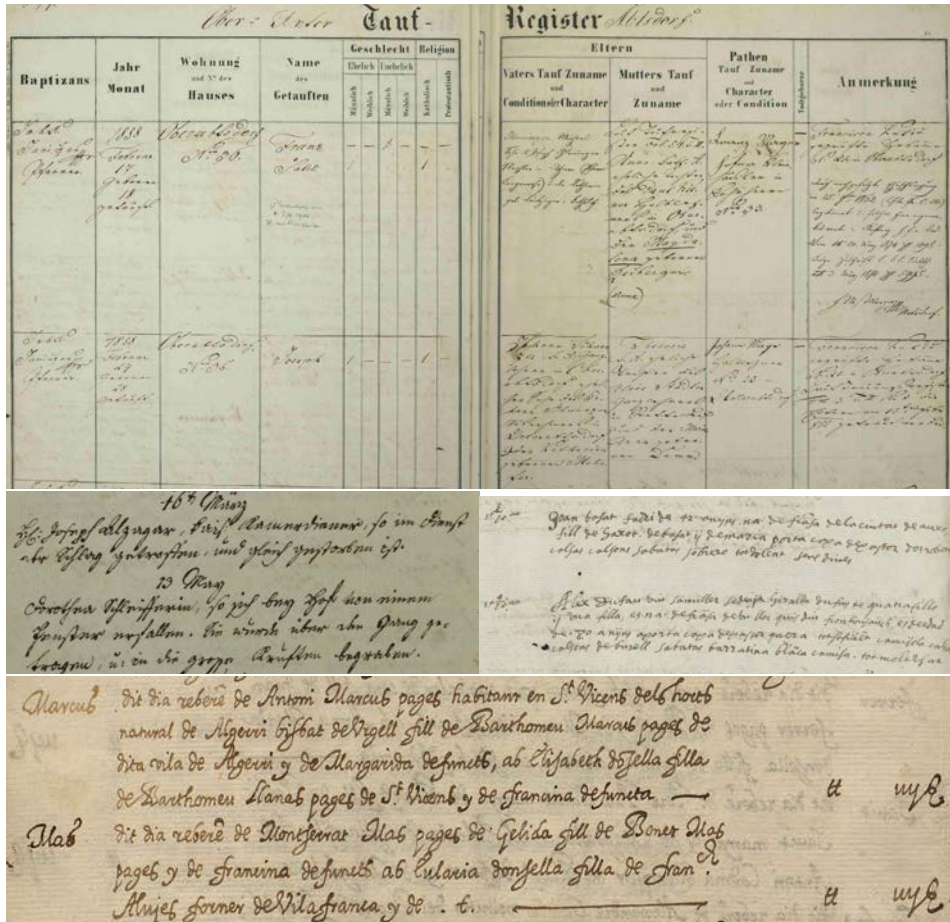


Figure 5.1: Samples of records from structured documents. Baptism registers from the Absdorf collection, 1853 (top). Death records from Wien, 1720 (left). Medical records from Sant Pau Hospital, Barcelona 1604 (right). Marriage records from the Barcelona Cathedral, 1619 (bottom).

In this chapter, we describe a new method to obtain word categories directly from non-preprocessed handwritten word images. The method can be used to directly extract information, being an alternative to the transcription. Thus it can be used as a first step in any kind of syntactical analysis. The approach is based on Convolutional Neural Networks with a Spatial Pyramid Pooling layer to deal with the different shapes of the input images. We performed the experiments

on a historical marriage record dataset, obtaining promising results.

Our approach has several advantages. First, it is able to detect entities no matter if they start with an uppercase or lowercase letters. Secondly, it can categorize these entities semantically. This means that the detected entity is also classified as belonging to a semantic category, such as name, surname, occupation, etc. The information of the semantic category of a word is a useful information in the parsing process. Third, the effort in the creation of training data is lower than the one needed for handwriting recognition (the word is not transcribed, just classified in several categories). Finally, the method does not have any problem with OOV words because it is not based on transcription or dictionaries. Even in scenarios where transcription would later be required, our method can be helpful by allowing us to use category specific models or dictionaries [87]. It can also be used to simply reduce the transcription cost by using the categorization as a way to select only relevant words to transcribe.

However, this method has a noticeable limitation, it does not leverage context information to learn to predict the semantic class of the word image. Therefore, we go further and propose the integration of syntactic context into that previous model, and also, the incorporation of a handwritten recognition module. This results in a significant increase in the classification performance, and also, it allows us to perform full information extraction directly from document images.

To solve that limitation we also propose two variants of language models to represent the context. The first variant is inspired on bigrams. The second variant is a sequential approach, able to predict the word categories and semantic relations (in our application use case, person relations) adding a *Bidirectional Long Short-Term Memory Network* (BLSTM) to our CNN model. Thus, having information about the semantic categories of the previous and upcoming words in a record, the prediction of each word category can be more accurate. The use of BLSTM as language model involves a second contribution. The model allows to infer relations between the named entities and consequently infer semantic patterns. Thus, the named entities can be assigned to individual concepts (in the context of the document topic) and therefore a knowledge database can be populated.

We also demonstrate that it is possible to skip the likely error in the step between recognition and semantic classification by using a single integrated model to identify word categories without transcription in a sequential way, so that we can benefit of context information. Thus, it contributes to reduce the semantic gap in the interpretation process of historical manuscripts. The proposed method can be used to extract information from paragraph-structured historical manuscripts with any kind of calligraphy, such as birth, marriage, death or census records. Experimentally, we have evaluated the proposed methodology with the protocol proposed in the ICDAR2017 competition on Information Extraction in Historical Handwritten Records [32], described in the previous chapter, demonstrating that our proposed methodology outperforms the state of the art methods.

The rest of the paper is organized as follows. In Sect. 5.2 we review the state of the art in this field. In Sect. 5.3 we will describe the most simple method for isolated images, with details of the architecture of the neural network we built, explaining the function of each of the different layers. In Sect. 5.4 we will explain the technical details of the dataset used, the training of our neural network and also discuss the results of the experiments. Sect. 5.5 is devoted to describe the two approaches we propose to incorporate the context information and how it was modeled. The experimental results are shown and analyzed in Sect. 5.6. Finally, in Sect. 5.7 we draw the conclusions and outline the future work.

5.2 State of the art

Information extraction from structured historical manuscripts usually includes layout analysis, handwritten text recognition (HTR) and sequence labeling. As described earlier, in cases where text is structured in paragraphs the traditional approach is to first carry out HTR, and then parse the output labeling each transcribed word with Natural Language Processing (NLP) techniques. Another option would be to directly perform semantic analysis from the visual information, leaving the transcription as the last step in order to leverage the information from the semantic structure of the paragraphs. Here we take a look at state of the art work in the different parts of the process.

The handwritten text recognition process can also be split up in sub processes like text region detection, line/word segmentation, and transcription. In each of them, the use of artificial neural networks (ANNs) has brought many improvements, but not yet achieving human level accuracy [80, 116, 86]. There have also been recent attempts to perform a fully end to end recognition in [15] with promising results but still behind the established pre-segmented text-line based recognizer. An advantage of joining sub processes in a single model that transcribes full paragraphs or full pages is that errors are not accumulated due to process concatenation. The counterpart is the "black box" effect, which makes it harder to develop better models because it is difficult to know which part of the model is causing the error.

After transcribing the handwritten text images, one can apply NLP techniques designed for computer readable textual documents (e.g. ASCII text) to perform the information extraction. An ideal information extraction model would first gather the human capabilities of finding named entities and other words considered relevant; and second, it would find relations among them based on the text context and even previous background from other sources. An example of this kind of problem is the CoNLL2003 task [107] which consists of recognizing named entities and their dependencies on a large text corpus. For this task a combination of conditional random fields (CRFs) and BLSTMs achieved state-of-the-art performance in [49]. An interesting reinforcement learning method to acquire external

evidence, which yields significant improvement for information extraction on Internet news datasets is proposed in [71]. A disadvantage of this method in our case would be that it is not easy to find external evidence of structured historical handwritten manuscripts on the web, so our method should make use as much as possible of the information in the selected manuscript database. Yet another possible approach is the one shown in [7] where dependencies are parsed into triples, allowing this way the information to be stored in short clauses containing the relationships among words in text sentences. One of the challenges in named entity recognition for textual documents is dealing with words that have never been seen in training, namely out-of-vocabulary words and there are works in the literature trying to address this specific issue like in [19].

There are also some works in the literature dealing with information extraction directly from images. For instance in [56] a combination of document physical structure recognition, linguistic rules, and text corpus knowledge is used to perform entity recognition. Also a very interesting work is presented in [75] that is able to extract information from invoices in a template free approach using BLSTM. However, these approaches are designed to work with modern clean and structured printed documents.

There are also some works trying to extract information from handwritten sources. For example, a system based on connected components detection and analysis to perform numeral recognition from chemistry documents is presented in [36]. In other cases, where there are tabular structures that will give information about the entity distributions, it is possible to label an entity based on separator distribution and text content, as shown once again for chemistry documents in [35]. These approaches look very task-specific and do not seem suitable to be applied to a paragraph-structured document in which information is stored in much more variable and complex sequence of symbols.

Finally, there are also a few works that deal with the problem of name entity recognition or information extraction from general handwritten text directly from the image. For instance, exploring the possibility of directly detect named entities from handwritten text images [4], or even classifying word images into the semantic category [109]. These, despite their obvious limitations, open a new possibility for information extraction on handwritten documents. A possibility that in our opinion was worth further research. Another interesting recent work in a related area was proposed by Gordo et al. in [40]. In this work, the authors show that it is possible to extract semantic word embeddings directly from artificially generated word images. They show that the network can even learn possible semantic categories of OOV words by reusing information from prefixes or suffixes of known words. However the training in this dataset required datasets of several millions of synthetically generated word images, that is a very different scenario from the typical handwritten dataset where the annotations are scarce.

As explained earlier, the most commonly used procedure for information extraction from handwritten documents is based on performing an HTR and then a

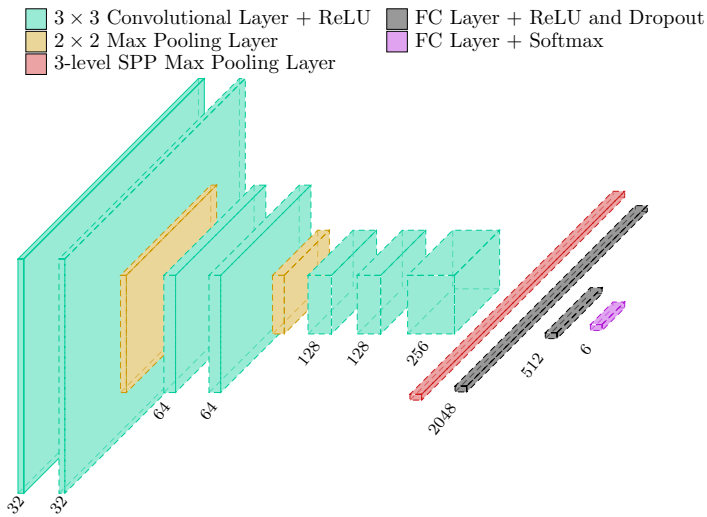


Figure 5.2: Outline of our CNN architecture

sequence labeling tasks subsequently. Following the inspiration of these last two works, we aim to develop a new method to leverage record level information to perform information extraction directly from handwritten text images. Once that we have semantic information about text images, the HTR process would be easier, because it would allow restricting the recognition to a class-specific recognizer. Also by working at image level, there is no concept of out-of-vocabulary words for the information extraction part, thus ruling out by design, one of the challenges of text based models.

5.3 CNN Based Word Image Categorization

In order to classify word images into semantic categories, we propose a CNN based method, inspired by [102](See Figure 5.2). The network is divided into three different parts: the convolutional layers, that can be seen as feature extractors; the fully connected layers that act as a classifier and the Spatial Pyramid Pooling layer that serves as a bridge between features and the classifier, by ensuring a fixed size representation. We will describe each of these different parts in this section.

5.3.1 Convolutional Neural Networks

Although Convolutional Neural Networks (CNN) were known since at least the early 1990's [63], it has only been recently that they gained major attention due to their high performance in virtually all fields of computer vision. The main building

block of these artificial neural networks are the convolutional layers. These layers can be seen as a certain amount of filters. The output of a convolutional layer is generated by a discrete convolution of these filters with the input to the layer. Furthermore, an activation function is applied to the result of the convolution in order to make the layer able to learn non-linear functions. Compared to a standard perceptron, the filters allow sharing weights for different spatial locations thus considerably reducing the number of parameters in general [63].

Convolutional layers serve as feature detectors where each individual filter learns to detect certain features of the input image. In order to introduce a certain amount of translation invariance with respect to these detected features, CNNs usually make use of so called Pooling Layers. In these layers, activations across a certain receptive field are pooled and a single activation is forwarded to the next layer. In most cases this pooling is performed by taking the maximum value seen in the receptive field [59, 93].

When stacking layers of convolutional and pooling layers, the filters in the individual convolutional layers learn edge features in the lower layers and more abstract features such as textures and object parts in the higher layers [120]. However, stacking a large amount of layers results in the so called Vanishing Gradient Problem [76] when using traditional activations such as sigmoid or hyperbolic tangent functions. Thus up until the early 2010's, neural network architectures were still fairly shallow [62]. The Vanishing Gradient Problem could first be tackled with the advent of using Rectified Linear Units (ReLU) as activation function [39]. This function is defined as the truncated linear function $f(x) = \max(0, x)$. Using the ReLU, deep CNN architectures are effectively trainable which was first successfully demonstrated in [59].

All of the convolutional layers in our architecture are a set of 3x3 Rectified Linear Units. The size of the filter was chosen to be 3x3 because they have shown to achieve better results compared to those with a bigger receptive field as they impose a regularization on the filter kernels [93]. Similar to the design presented in [93, 102], we select a low number of filters in the lower layers and an increasing number in the higher layers. This leads to the neural network learning fewer low-level features for smaller receptive fields that gradually combine into more diverse high-level abstract features.

5.3.2 Fully Connected Layers

The general layout of CNNs can be split up in a convolutional and a fully connected part. While the convolutional and max pooling layers constitute the former, the latter is a standard Multilayer Perceptron (MLP). Thus, the convolutional part can be seen as a feature extractor while the MLP serves as a classifier. The layers of the MLP are often referred to as Fully Connected Layers (FC) in this context. Just as convolutional layers, the use of ReLU as activation function has shown itself to be effective across various architectures [59, 93].

The large amount of free parameters in fully connected layers leads to the problem of the MLP learning the training set “by heart” if the amount of training samples is low. But even for larger training sets, co-adaptation is a common problem in the fully connected layers [47].

In order to counter this, various regularization measures have been proposed with Dropout [99] being one of the most prominent. Here, the output of a neuron has a probability (usually 0.5) to be set to 0 during training. A neuron in the following layer can now no longer rely on a specific neuron in the preceding layer to always be active for the same input image. Thus, the CNN has to learn multiple paths through the neural network for a single input image. This leads to more robust representations and can be seen as an ensemble within the CNN model.

The size of the different layers is a hyperparameter to tune experimentally, except for the final layer whose size has to match the number of classes we want to classify. This final layer usually uses a “softmax” activation function that outputs a probability distribution over the possible semantic categories in our experiment for each input image.

5.3.3 Spatial Pyramid Pooling

In general, the input to a CNN has to be of a fixed size (defined before training the network). For input images bigger or smaller than this defined size, the usual approach is to perform a (potentially anisotropic) rescale or crop from the image. For word images, with an important degree of variability in size and aspect ratio, cropping is of course not an option and resizing might introduce too strong artificial distortions in character shapes and stroke width. Thus it is important in our case that we allow our CNN to accept differently sized input images.

The key observation is that, while convolutional layers can deal with inputs of arbitrary shape and produce an output of variable shape, the fully connected layers demand a fixed size representation. Thus, the critical part is the connection between the convolutional and the fully connected part. In order to alleviate this problem, the authors in [45] propose a pooling strategy reminiscent of the spatial pyramid paradigm.

The pooling strategy performed over the last layer in the convolutional part is a pyramidal pooling over the entire receptive field. This way, the output of this Spatial Pyramid Pooling layer (SPP) is a representation with fixed dimension which can then serve as input for the ensuing MLP. It was also shown by the authors that this pooling strategy not only enables the CNN to accept differently sized input images, but it also increases the overall performance. In our method, we use a 3-level Spatial Pyramid max pooling with 4×4 , 2×2 and 1×1 bin sizes. This allows us to capture meaningful features at different locations and scales within the word image.

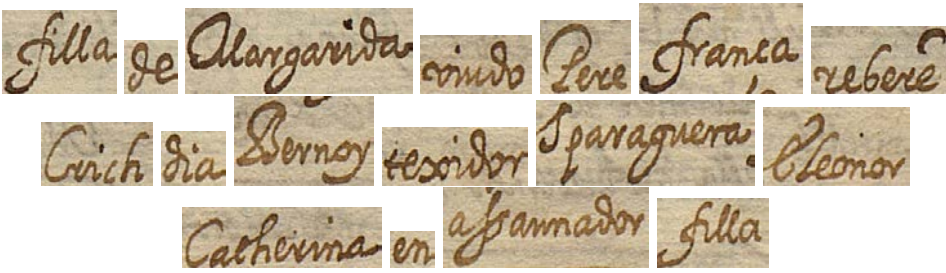


Figure 5.3: Several examples of word images in the Esposalles Dataset. The big degree of variability both in size and aspect ratio of the images makes impractical the common approach of resizing images to a common size.

5.4 Experimental Validation

In this section we will describe the experimental validation of our proposal. We will first explain in detail the dataset used as well as some practical details relative to our training. We will then show the results achieved and discuss them.

5.4.1 Esposalles Dataset

For our experiments we used the Esposalles dataset [28, 85]. This dataset consists of historical handwritten marriages records stored in the archives of Barcelona cathedral. The data we used corresponds to the volume 69, which contains 174 handwritten pages. This book was written between 1617 and 1619 by a single writer in old Catalan.

For our purpose the datasets consist of 55632 word images tagged with six different categories: “*male name*”, “*female name*”, “*surname*”, “*location*”, “*occupation*” and “*other*”. From this total we reserve 300 images of each class for testing, up to a total of 1800 images. After discarding word images smaller than 30x30pixels we end up with 53568 training examples for training and 1791 for test. In the training dataset there is a big class imbalance, with 31077 examples of the class “*other*”, 3636 “*female name*”, 4565 “*male name*”, 2854 “*surname*”, 6581 “*location*” and 4855 “*occupation*”. No normalization or preprocessing was done to word images besides remapping them to grayscale in the interval [0-1] (0: background, 1 foreground). It is worth noting that there are words with the same transcription that could potentially belong to different classes. This is specially significant for the “*surname*” class, since it is quite common for surnames to be related to a location (i.e a city name), an occupation or even a male name. We can see several examples of word images in Figure 5.3. The dataset is available upon request to the authors.

	Precision	Recall	F_1 - Measure	Classification Accuracy
Adak et al. [4]*	68.42	92.66	78.61	-
Romero et al. [87]**	69.1	69.2	69.15	-
Our approach	84.23	75.48	79.61	78.11

Table 5.1: Comparative with other methods.

Predicted Class	True Class					
	Other	Surname	Female Name	Male Name	Location	Occupation
Other	272	52	21	9	61	38
Surname	5	153	19	2	21	3
Female Name	2	34	247	9	4	4
Male Name	2	14	5	274	7	1
Location	14	33	6	4	203	4
Occupation	3	10	1	0	4	250

Table 5.2: Confusion Matrix for our CNN architecture, with a global accuracy 78.11%.

5.4.2 Experiments and Results

The network was trained using standard backpropagation with stochastic gradient descent with a learning rate of 10^{-4} Nesterov momentum 0.9 and a decay rate of 10^{-6} for 100 epochs, which proved enough to obtain a training accuracy of over 99% in all the experiments. Since we are working with images of different size, each example had to be processed individually (batch size 1), that is the reason for the low value for the decay rate.

We used the standard categorical cross entropy as loss function. In order to deal with the class imbalance problem, we introduced a “class weight” parameter in the loss function to relatively increase the impact of misclassifying the classes with less examples. The weight for each class is calculated by dividing the number of samples of the most populated class by the number of samples of each class.

$$w_i = \frac{\max(n_i)}{n_i}$$

We performed several experiments with slightly different network architectures, to empirically calibrate the hyperparameters. After having fixed the learning and decay rates and the number of epochs we noticed that even drastically changing the number of parameters in our architecture the results were similar.

The proposed network architecture, as depicted in Figure 5.2 achieved an accuracy of 78.11% in our dataset. An alternative network with the similar architecture but, halving the number of parameters of all the layers of the network (half of the channels in each of the convolutional layers and half of the neurons in each of the fully connected layers, and keeping the same 3-level pyramid pooling) produced an accuracy of 77.33%.

In Table 5.2 we see that the errors are not evenly distributed. Despite the introduced class weights, the network is still more likely to mistakenly assign the class “*other*”. We can also see that examples corresponding to the “*surname*” class are harder to classify. This may be due to several reasons since, as discussed earlier, surnames are usually derived from names, places or occupations. It is also worth noting that the “*surname*” class is the one with fewer samples, thus having seen less examples the model is more likely to overfit this particular class.

The comparison of our method with similar methods in the literature is not an easy task, because this is a relatively recent area of research and there are few publications addressing similar issues. Even the most similar methods have big differences, for instance none of the methods provides a classification accuracy metric. In [87] they address the classification as an aid to transcription, and they work with the Esposalles dataset but with a different labeling with a different number of classes. In the case of [4] the aim is a named entity detection, with a binary output. They provide results with different datasets, so we selected the best result they achieved, using the IAM dataset.

We calculated our precision/recall metrics following the approach described in [87], and defined as: “Let R be the number of relevant words contained in the document, let D be the number of relevant words that the system has detected, and let C be the number of the relevant words correctly detected by the system. Precision (π) and recall (ρ) are computed as”:

$$\pi = \frac{C}{D} \quad \rho = \frac{C}{R}$$

In order to compare our results with the state of the art we can see the class “*Other*” as non-relevant words. Then, for “relevant words” we understand words with a “True Class” other than “*Other*” and for “relevant detected word” we understand word-images assigned with a label other than “*Other*”. Finally for “correctly detected” we understand examples where the system correctly assigned a label other than “*Other*”. That means we do not consider a word as “correctly detected” unless it is also assigned to the correct category.

5.5 A full Information Extraction System

In this section we present two new architectures for extracting information in hand-written structured documents. These architectures are able not only to extract the named entities, but also to model the relation among words in a record (e.g. relations among name entities appearing in the documents). We start by reviewing our previous work on isolated words. Then, we describe how we have extended this previous method to model full information records using an integrated architecture.

5.5.1 Semantic categorization of isolated word images

The most straightforward way to perform a semantic categorization of isolated word images is to frame it as a traditional object classification framework. The supporting hypothesis is that if deep neural networks are able to grasp a concept like "dog" from images of dog races as different as a doberman and a yorkshire terrier from different viewpoints they can probably capture the variations in surnames, city names or occupations by looking at some visual clues like capital letters, some characteristic suffixes or prefixes. For example, *-ia* is a typical suffix of spanish female names (e.g. Maria, Lucia) , *-son* and *-ström* are typical suffixes of nordic surnames (e.g. Jacobson, Nyström), and *Ober-* is one of the typical prefixes of german locations (Oberhausen, Oberwesel).

In [109] we proposed a relatively shallow and simple Convolutional Neural Network for semantic classification of word images. The network is designed to start with a small number of low-level features that gradually combine into a bigger number of higher level features with some pooling layers to add some degree of scale and translation invariance. Besides the specific choice of number of layers and features it differs from the standard CNN in the fact that uses a Spatial Pyramidal Pooling layer [45] to deal with the variability in shape and aspect ratio. This SPP layer produces a fixed size pyramidal representation that further helps in developing scale invariance and also allows to bridge the variable size output of the feature map with the fixed size of fully connected layers that serve as the classifier.

However, as stated in the introduction, this approach has a very obvious limitation: the lack of context information or language model. By dealing with each word as a single example we are ignoring a very valuable source of information. It is very common that the very same word could perfectly fit into two or more semantic categories (i.e. homographs) and it is only through context that one can decide. This is specially true in structured documents, where the records loosely follow a predefined structure or grammar. For example, in most documents, labelling a capitalized word right after a name as a surname is a safe bet.

Next we will describe two different ways to leverage context information. The

two architectures jointly learn both the visual model (like the architecture described above) and the language model. This language model has to be understood as the specific language rules encoded in the structure of each collection of structured documents.

5.5.2 Incorporating language models

As we move away from the limited framework of isolated word images we can use language models to incorporate contextual information at record level. Two of the most widely used language models in the literature are n-grams [112, 121] (usually in its most cost-effective case of bigrams) or a language model estimated with a recurrent neural network [103, 68]. In our work, the language model refers to the specific context of type of structured document. Therefore, when we apply our method to, for instance a dataset of medieval birth records, we are not modeling Latin language, but the specific syntax of birth records.

Bigram inspired language model

In this first approach, we modify the original architecture described in [109] to incorporate bigram estimation. Bigrams estimate the probability of a given word given its predecessor. In our case, we are interested in the category of a word image, given its predecessor. Concretely, we propose a new architecture that accepts two inputs: the word image, and the labeling of the previous image. The word image would go through the same convolutional layers and spatial pyramid pooling, while the label was passed through three fully connected layers before being merged to the output of the spatial pyramid pooling. This combination of visual information and the label of the previous word is now fed to the usual fully connected layers that output the semantic category. In this new architecture we need a new special label to represent the starting of a record, that will be fed as the previous label of the first word in each record. The architecture is shown in Fig. 5.4.

We also incorporate label smoothing [105, 20] on target categories. That is, we modify the target distribution of the output by randomly assigning a small amount of the energy among incorrect labelings. This has proved to benefit generalization by reducing the network overconfidence in its predictions. The intuition is that, after enough training, the only way of a network to reduce the loss is by getting closer to a one-hot distribution. In sequences, this means that a wrong prediction in a given time-step is unlikely to be reconsidered in the next one, because the probability of alternative labelings is close to zero.

Formally, for each example x with correct label y our model computes the probability of each possible label $k \in \{1 \dots K\}$, where K is the number of classes. Usually we would have a normalized ground truth distribution q such that $q(k) = \delta_{k,y}$ where $\delta_{k,y}$ is the Dirac delta, $\delta_{k,y} = 1$ when $k = y$ and 0 otherwise. We

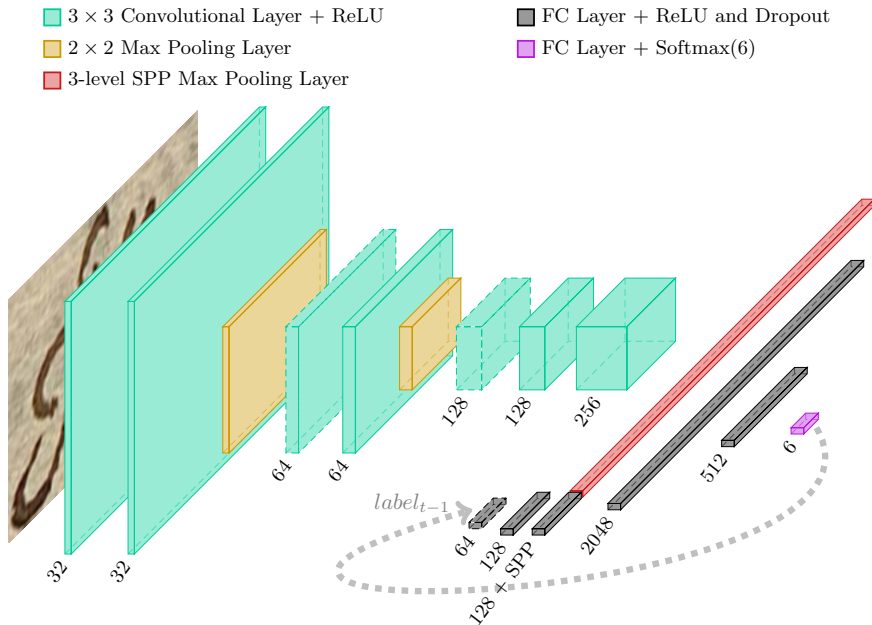


Figure 5.4: Bigram inspired architecture. This architecture models the relation of words by accepting two inputs: the current word image, and the predicted label from the previous word image.

will then use the following *smoothed label distribution* with parameters $\mu = 0.25/K$ and $\sigma = \mu/5$ instead:

$$q(k) = \delta_{k,y} \left(1 - \sum_{i \in 1, \dots, K} X_i \right) + X_k \quad (5.1)$$

where $X_i \sim p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

It must be noted that while bigrams are one of the most cost-effective language models, they have some drawbacks, like the inability to capture long range relations among words. Indeed, in our bigram-based approach, each word image is considered as an individual example. Even if it has extra information regarding the previous prediction, a relatively “simple” CNN could perform the task. Consequently, this architecture cannot remember long word dependencies, unless higher n-grams are used. But of course, this has a limit. Higher-order n-grams have the problem of sparsity, requiring larger amounts of training data. Unluckily, the availability of annotated structured historical manuscripts is very limited. Also, mainly due to that same sparsity problem, higher order n-grams do not generalize well and end up producing very small improvements in accuracy.

This inability to capture long word dependencies is one of the main reasons why language models based on Recurrent Neural Networks [103, 68] (and specially BLSTMs) have become popular, and why we propose the second approach.

BLSTM based language model

In this second approach, we capture longer relations among words in a sentence or record with variable number of words by incorporating a Recurrent Neural Network (RNN). In this case, each record represents a time-series example, and each word-image represents a time-step.

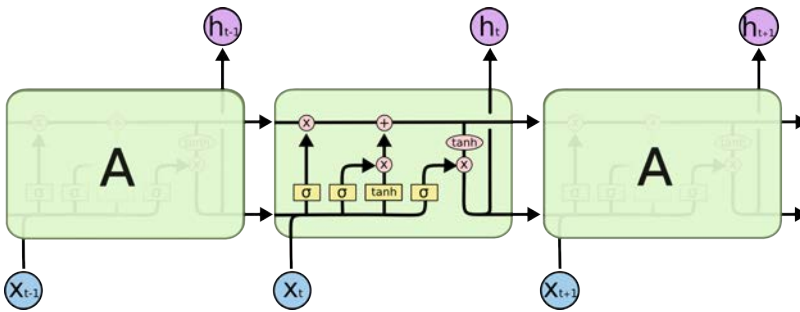


Figure 5.5: The inner workings of an LSTM network. [73]

RNN suffer the problem of the vanishing gradient, that can be alleviated by the use of Long Short-Term Memory (LSTM) units [73], shown in Figure 5.5. These units are designed to allow the neuron to keep its state value unchanged along time-steps. In order to leverage information from future and past time-steps, the sequence can be processed forwards and backwards with a Bidirectional-LSTM.

The proposed model, shown in Fig. 5.6, is composed of a convolutional part that extracts the visual features of each individual word images, combining stacks of convolutional, pooling and fully connected layers. These features are then fed to a BLSTM layer that models the relations among the words in each register. Contrary to the bigram-inspired approach, where the model integrated visual information and labels, in this model all the information is extracted from the visual features of the word images, and the relation among these visual features is represented as the hidden state of the BLSTM layer. Finally, from the output of the BLSTM softmax can be applied to reduce the output to the required number of categories in the document.

We are capturing relations among word images with the BLSTM, which means that we can now extract more information than just the semantic category. For instance, we may want to associate the category to a person or role in the record *i.e.*: husband or wife in a marriage record, father or newborn in baptism records, etc. With our integrated architecture, we can jointly output all this information

by adding additional softmax outputs. For example, in Fig. 5.6, the model has two outputs for each word image: the semantic category and the person to whom this word is referring (e.g. a certain word is the surname of the father). Finally, for each output, we minimize a separated cross-entropy cost function using SGD:

$$C(b) = -\frac{1}{n} \sum_{x \in b} [p(x) \log q(x)] \quad (5.2)$$

being $p(x)$ the output of the softmax for the example x and $q(x)$ the smoothed ground-truth label distribution described in Equation 1. And the sum is considered over the current minibatch B .

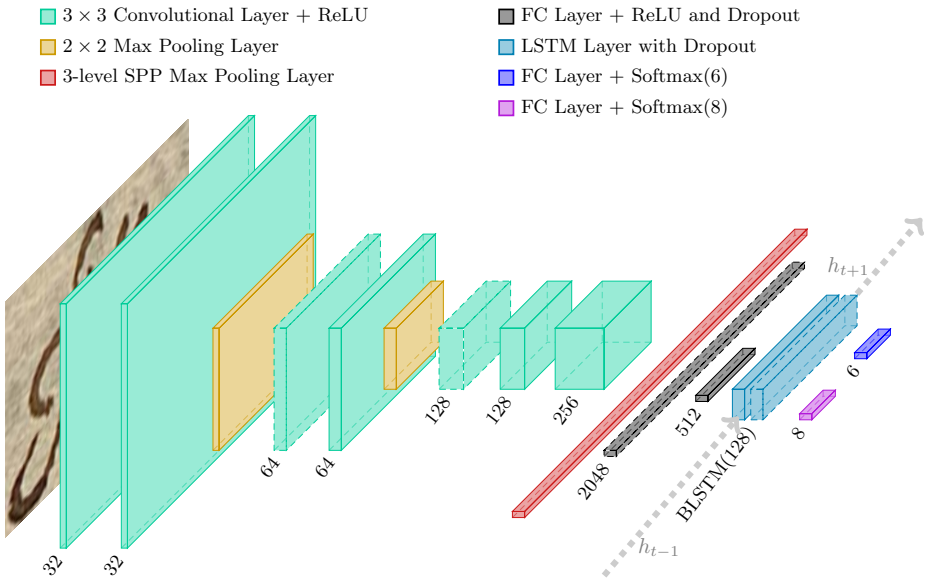


Figure 5.6: BLSTM-based architecture. This architecture models the relation among words in a record with the hidden state of a BLSTM layer. In this case we can see that the architecture has two softmax outputs because we are interested in extracting both the category and the person it relates to.

5.6 Experiments

In this section we describe the dataset, the evaluation protocol and we analyze the experimental results.

5.6.1 Dataset

Our methodology can be applied to any kind of structured document in which the information is written in sentences or paragraphs. In particular, we have tested our information extraction approaches using the dataset from the ICDAR competition on Information Extraction in Historical Handwritten Records (IEHHR) [32] because the dataset and ground-truth are publicly available and the website of the competition ¹ allows us to easily compare our work with other methods.

The dataset consists of historical handwritten marriages records [85, 28] stored in the archives of Barcelona Cathedral. The records were extracted from a book written between 1617 and 1619 by a single writer in old Catalan language. The data used in the competition corresponds to 125 pages, with a total of 1221 marriage records (39527 word images), with their transcriptions, semantic categories and person labels. See Fig. 5.7.



Figure 5.7: An example of the IEHHR Competition Dataset: “Esposalles” at word-image level, showing word images and their corresponding, transcription (black), category label (blue) and person label (orange).

The IEHHR competition consists in finding relevant words in marriage records, transcribe them and tag them with a semantic category and the person who they relate to. Relevant words can belong to five different categories: “Name”, “Surname”, “Occupation”, “Location” and “Civil state”. These semantic categories are associated to seven different people relationship: “Wife”, “Husband”, “Wife’s father”, “Wife’s mother”, “Husband’s father”, “Husband’s mother” and “Other person” (usually the former husband of the wife in case she is a widow). Non-relevant words are labeled as “Other” and assigned to the person “None”.

The provided data is divided into training and test sets. The training set contains 100 pages (968 records with 31501 word images) and the test set contains 25 pages (253 records with 8026 word images). In our work, from the proposed training set, we subtract the last 10 pages (96 records with 3155 word images) to use them as validation.

¹<http://www.cvc.uab.es/5cofm/competition/>

5.6.2 Performance Evaluation

We evaluate our approach with the metrics used in the IEHHR competition [32]. The competition had two different tracks: in the basic track, the system has to provide the transcription and the semantic category (e.g. surname, location, etc.) of the relevant words. In the complete track, the system must also provide the person relation (e.g. husband, wife, etc.).

In the evaluation, the semantic label is prioritized. This means that if the semantic category is incorrect, the transcription is not taken into account. In case the semantic category is correct, then the Character Error Rate (CER) is used to evaluate the transcription. The CER is calculated for each pair of submission and ground-truth words, and computed as the Levenshtein Distance between the two words, normalized by the length of the longest transcription in order to have a value between 0 and 1.

The accuracy score for each labeling is calculated as 1-CER, and the record accuracy is the average of the labeling accuracies of the relevant words found in the record. The final score is the average of all the record accuracies. For better visualization, all values are normalized between 0-100.

5.6.3 Experimental Details

Next we detail the experimental setup, including the semantic labeling and the description of the method that has been used for word recognition.

Semantic labeling

We performed experiments with the two proposed neural architectures. The convolutional part of both models consists of a series of convolutional and max pooling layers increasing the number of channels followed by a spatial pyramid pooling layer and fully connected layers. For more details in the architectures see Sect. 5.5 or Figures 5.4 and 5.6.

Since the BLSTM language model works at record level, we were forced by the limitations of current deep learning frameworks, to normalize each word image to have the same size. Therefore, word images are resized, preserving the aspect ratio, to fit in an image size of 80x125 (approximate average height and width of the images in the dataset). The resized image is then placed in the center of a new image of the aforementioned size, and the empty regions are filled with the average pixel intensity of each image. In order to make the comparison easier, we also used normalized image size for the bigram model. Using aspect ratio preserving image size normalization would allow us to drop the SPP layer, as it is no longer required to produce a fixed size representation. However, having an Spatial Pyramid architecture helps in dealing with size variations from the content

of the images (which were introduced by the normalization we used), and as shown in the original paper of SPP, this layer helps to improve the performance even with fixed size images [45].

The output labels of our models are trained to predict their outputs “independently”. This means that we need to remove words labeled with category “*other*” even if the predicted person is not “*none*”, and also words labeled with the person “*none*” regardless of their semantic category. This simple “post-processing” ensures that impossible labelings never occur, because, according to the dataset ground-truth, all non-relevant words are labeled with category “*other*” and person “*none*”. Indeed, these two labels are never used for any relevant word. We could have designed the models to produce a single output formed by the cartesian product of all available classes, however we chose to have independent outputs because of two main reasons: first, because we believe that our architecture can be more general in this way (the fact that some combinations of classes are not possible is just for the particular of this dataset); and secondly, because we take inspiration from the attribute representations, which allow a more meaningful use of information (i.e. some words sharing some attributes will share some activations whereas if we used a single softmax output, their activation could be completely different [5, 102]. This re-use of information also helps in speeding up the training because having more positive examples for each class compared to the unique softmax output, results in requiring less epochs to converge. Finally, there is yet another minor advantage in using models with multiple outputs, and that is that it would scale better in more complex scenarios, (i.e three outputs of 10 classes versus one output of 1000 combined classes)

It is worth mentioning that no data augmentation was used for the semantic labeling networks. In fact a study on how to produce a useful and realistic data augmentation for this task would probably be an interesting future research as we discuss on the conclusions.

Handwritten word recognition

Any information extraction approach for handwritten documents needs to perform handwriting recognition. Although the contribution of this paper is not in handwriting recognition, we will describe the method used for the sake of completeness. In both approaches we use the handwriting recognition method described in [108]. This method is based on two neural networks with two different stages of learning. The first stage uses a CNN to learn to embed word images in the PHOC (Pyramidal Histogram of Characters) space as described in [102]. Once trained, this CNN is used to embed, not full word images, but a set of handwritten patches into this PHOC space. These sets of patches are presented as a sequence of partial embeddings to a two-layered BLSTM, that produces the transcription. We chose this method because it is a lexicon free approach (and consequently, it can recognize out of vocabulary words), and because, as far as we know, it has the best

published performance on transcription at word level for this specific dataset.

Bigram inspired language model

This model was trained with a learning rate of $1e^{-3}$ with an early stopping threshold of 20 epochs without improvement on the category accuracy in the validation set. The “person” label was obtained using a simple grammar, based on finding the keywords that separate the persons in the marriage record (e.g. "son of", "daughter of"). It must be said that we also tried to model two outputs, “category” and “person”, but the person label was very unreliable, dropping the complete score to a 38%. The reason is that labeling the person requires more context than just the prediction of the previous word.

One crucial aspect for a successful training is that when feeding the category of the previous word, it must be the predicted one from the network, instead of the correct value from the groundtruth. We found out that if, during training, the network was always receiving the correct category of the previous word, then it did not pay enough attention to the current image and, consequently, the performance in test time dropped significantly. The intuition is that, by using the predicted previous category both in train and test time, this makes both training and test more similar and also allow the model to learn during training that the category of the previous word is not always reliable, and it learns it with the exact same error distribution that it will produce in test time. The size and number of fully connected layers for the branch incorporating the previous category information was determined empirically.

BLSTM inspired language model

This model was trained with a learning rate of $1e^{-2}$ with an early stopping threshold of 20 epochs without improvement on the average of category and person accuracies in the validation set. The learning rate in this case is higher than in the previous model. The reason is that the input is a full record in contrast to a single image and the label of the previous one. This fact greatly reduces the number of examples per epoch and also affects the loss value. In the bigram model, we include as input of a word i , the features of this word image, but also the predicted label of the previous word $i - 1$. Conceptually, we aimed to do something similar with a BLSTM on top, but having one softmax layer before the BLSTM only resulted into an excessive reduction of information harming the performance. Therefore, the BLSTM is now an integrated part of the network, dealing with visual features rather than a traditional language model working at semantic label level.

5.6.4 Results and Discussions

In Table 5.3 we can see a comparison of our approaches with other methods from the literature that participated in the competition [32]. The method labeled as *HMM+MGGI* is based on Hidden Markov Models and a Morphic Generator Grammatical Inference [86]. The methods labeled as *CITlab-ARGUS* are different variants of methods based on LSTMs. The method *Hitsz-ICRC-1* is a CNN based bi-gram method, whereas *Hitsz-ICRC-2* corresponds to a Resnet based uni-gram method; but both methods have a postprocessing step combined with a CRF sequence tagging method. The method labeled as *Word level CNN* corresponds to our previous work based on CNNs [109]. We included results at line level for the sake of completeness. Although not directly comparable, we can see that there is not a significant gap in performance from line level to word level approaches, so these results can help in providing a bigger picture of the difficulty of this task.

Method	Level	Basic Score	Complete Score
HMM+MGGI [86]	Line	80.28	63.11
CITlab-ARGUS-1 [32]	Line	89.53	89.16
CITlab-ARGUS-2 [32]	Line	91.93	91.56
CITlab-ARGUS-3 [32]	Line	91.61	91.17
Word Level CNN [109]	Word	79.42	70.20
Hitsz-ICRC-1 [32]	Word	87.56	85.72
Hitsz-ICRC-2 [32]	Word	94.16	91.97
Our Bigram Model	Word	87.98	79.68
Our BLSTM Model	Word	94.62	94.02

Table 5.3: Competition Score

From the results, we can observe that we outperform our previous isolated word categorization approach [109] in all scenarios, confirming the seemingly obvious hypothesis that contextual information (i.e analyzing more than one word at a time) is very important in an information extraction problem, and being able to measure this importance. We also prove that, incorporating this context, the semantic categorization of handwritten word images can be done without the need of an intermediate step of transcription.

With our bigram model, we achieve a 87.98% score on the basic track and a 79.68% score on the complete track. On the basic track, our approach would get the second position on the competition, behind the much more complex system *Hitsz-ICRC-2*, which is based on CNNs+Postprocessing+CRF. On the complete track, the score is hindered by the use of a very simple grammar.

With our BLSTM based language model, we score a 94.59% in the basic track and 94% in the complete one, outperforming the state of the art in both cases.

Predicted Category	True Category					
	Other	Surname	Name	Location	Occupation	State
Other	3754	16	7	10	15	5
Surname	8	648	22	1	15	1
Name	12	13	1278	6	1	3
Location	29	6	3	1061	13	0
Occupation	12	11	2	9	753	2
State	2	0	0	0	0	308

Table 5.4: Confusion matrix for the BLSTM based model for the category label.

Predicted Person	True Person							
	None	Other Person	Husband	Wife	Husband's Father	Husband's Mother	Wife's Mother	Wife's Father
None	3734	2	12	12	5	0	1	11
Other Person	0	126	0	2	0	0	0	0
Husband	30	0	1544	0	0	0	0	0
Wife	17	9	1	737	0	0	0	0
Husband's Father	6	0	0	1	509	0	0	0
Husband's Mother	4	0	0	0	0	154	0	0
Wife's Mother	3	0	0	6	0	0	183	0
Wife's Father	20	1	0	3	0	0	2	891

Table 5.5: Confusion matrix for the BLSTM based model for the person label.

This score, combined with the fact that the metric of the competition is designed to emphasize the category and person labelings rather than the transcription, allows us to claim that determining the semantic category (and person) of a word image can be done skipping the intermediate step of the transcription and outperform the traditional two-step pipelines (i.e. first transcription, then semantic labeling).

In Table 5.4 and Table 5.5 we can see the confusion matrices for categories and person for the BLSTM based model, the best performing of our two models. Trying to find parallelisms with the traditional two-step detection and classification approaches, we can see detection as a single classifier of the non-relevant class (“other” in the category case and “none” in the person case) versus all the other classes.

For both the person and category cases, we see that numerically, most of the mistakes are false positives followed by false negatives in the detection. Obviously the numbers are affected by the fact that the non-relevant class is the most common class and most likely by the greater inner variability of this class. Since in the dataset there was this enforcement that non-relevant words were labeled as “other”, “none”, we discard all words with either of those labelings. This “post-processing” fixes some false positives and creates new false negatives. In the end we wend up with 83 false positives and 58 false negatives out of 8026 words, which means a detection accuracy of 98.24%.

In the confusion matrix for the person label (Table 5.5) we see that the confusion among relevant classes is minimal. In relative terms the most common mistake is to attribute information relative to “*other person*” to the wife. It is a quite understandable error because, in many cases, this “*other person*” is in fact the former husband of the bride, in other words, when a widow is getting married again, the record also contains information about the deceased person. An example of this kind of mistake can be seen in Fig. 5.8.

habitac	en	Bara	ab	Antonia	viuda	de	jaume	Roger	de	Bara
Ground-Truth										
none	none	husband	none	wife	wife	none	other person	other person	none	wife
other	other	location	other	name	state	other	name	surname	other	location
Prediction										
none	none	husband	none	wife	wife	none	other person	wife	none	wife
other	other	location	other	name	state	other	name	surname	other	location

Figure 5.8: Example of a prediction where one word was incorrectly assigned. The surname of the former husband of the bride is incorrectly assigned to her (shown in red color).

The relative scarcity of examples and maybe even some natural language ambiguities may play an important role. In the category *side* (see Table 5.4), we can see other kind of errors. Note that the most “*difficult*” class is the surname. Once again, this is the category with fewer examples, because the writer usually avoided writing the same surname for all the family members. In addition, many Catalan surnames can have the exact same spelling as a name or an occupation. Some confusions with surname-occupations can be discarded by benefiting from the sentence structure. However, since the sentence structure has some degree of variability from record to record, the contextual model must be flexible, otherwise, unseen structures could never be accepted. As a consequence, confusions such as surname-occupations may still appear. In addition, extreme cases would even require expert level knowledge beyond our system (i.e. the son of a carpenter was unlikely to become a farmer in the XVII century Barcelona). In case of names-surnames confusions might be even harder in the presence of compound names and surnames. Indeed, even human annotators might make mistakes in such cases.

5.7 Conclusions and further work

This chapter, being the final chapter of the thesis, leaves quite a few of open paths for future research. We have presented a simple approach to word categorization using convolutional neural networks. The spatial pyramid pooling layer allows us to deal with the important variability in aspect ratio of word images without ar-

tificially distorting our image. The results were specially promising given that we were classifying just isolated words images with no transcription, context information or language model of any kind. Thus, it encouraged us to explore the addition of context information or simple language models, that should significantly boost the performance, specially in the mentioned case of surnames.

An interesting future work with this model would be to perform some new experiments in order to determine if the network is actually learning heuristic similar to what a human would use. For instance, names, surnames and locations usually start with a capital letter whereas occupations and other words usually do not and some word endings have a much higher likelihood on a particular class.

We have also proposed two neural architectures to extract semantic information from historical handwritten documents using context information. Contrary to traditional two-step pipelines, which first transcribe the text and then label it into semantic tags, we propose to extract the semantic categories without an intermediate transcription step. These approaches have two advantages: first, they do not have to rely on a good transcription for a correct semantic labelling, and secondly, they can naturally deal with out of vocabulary words. Moreover, our second architecture, which integrates a CNN with a BLSTM-based language model, is able to extract the semantic categories and associate them to individuals in the record (i.e. semantic relations between terms). As a result, it is able to populate a knowledge database with the document contents.

The experimental results have shown that our hypothesis and methodology is valid, while outperforming the existing approaches. We believe that the good performance of this new kind of approaches encourages further research. Recently, fully end-to-end neural methods have shown to outperform other approaches in a wide range of tasks. Moving in this direction, a logical next step would be to reformulate the task in order to be able to work at line or paragraph/record level, avoiding the need of segmenting into word images. In that sense, it would be interesting to study the possibility of adapting techniques like content based attention (often used in image captioning).

Developing useful and realistic data augmentation techniques for this kind of problems is also an interesting challenge. Probably the easiest approach would be generating new records by randomly selecting a record semantic labeling from the ground-truth and filling it with random examples of relevant word images for the corresponding semantic class. To do so we would have to find a way to deal with multi-word entities. It would also be interesting for the sake of generalization to produce new realistic semantic labelings rather than just randomly selecting them from the ground-truth. For some documents, for example tabular documents where each field is a set of paragraphs, expanding the scope beyond the paragraph level would allow us to further exploit the context and explore complex relationships among words located in different regions or cells.

Another possible future line of research is exploring how this semantic labeling

can be used to improve the transcription. One of the most obvious ones is by building category specific models for transcription, be it dictionaries or simply probabilistic analysis of common prefixes/suffixes. Yet another possible line of research is building new semantic based applications, such as semantic query-by-example word spotting, which could be used to search instances of a surname with spelling variations.

Finally, since extracting the semantic information directly from the image is significantly different from traditional transcription based methods, it can probably be expected that ensembles of both approaches can result in significant performance improvements.

Chapter 6

Conclusions and Future Work

In this last chapter of the thesis, we summarize the contributions and discuss the performance of the proposed methods as well as their limitations and, finally, we present possible lines of future work.

In this thesis we have studied different ways to access the information contained in totally or partially handwritten documents. We call this process Information Extraction, and it can require very different techniques depending on the type of document. In this thesis we explored two big types of documents. The first type are modern, highly structured form like documents, and the second type are historical handwritten documents that were commonly used before the widespread use of pre-printed forms.

6.1 Summary and Discussion

After an introduction to the topic of information extraction and the motivation behind this thesis we start by analyzing an application scenario of Information Extraction from electoral documents. Electoral documents usually combine printed and handwritten information. They are very similar to forms in the sense that we, in order to extract a specific piece of information, the easiest way to do it is just to analyze a fixed position of the document image. It is worth noting the astonishing variety of electoral documents. Most people only have the experience of voting in a single country, therefore might have a wrong impression of the general problem. The variability in electoral documents is not only due to differences in appearance but rather a deep difference in the kind of information that can be contained in those documents. Different electoral laws require a very different set of information

from voters. There are countries like Spain, where the most common elections are at Party level, and the only action from the voter is to choose from a set of fully printed ballots. The next level of complexity would be candidate level elections, where the voters are required to select k out of N candidates. Finally, we have preferential voting where voters are required to rank-order the candidates. These are just three examples to show different information that can be required from voters, but devil is in the details. Concretely, how this information is supposed to be reflected on the documents and how should the corner cases be interpreted.

It is quite common that, as researchers, we just focus on how to solve a general case, oversimplifying the problem. In the first contribution of this thesis we made an effort to describe some of the challenges that can be found when trying to process electoral documents. We tried to link those challenges with document analysis techniques that could be used to solve those issues. One of the goals of making this study of electoral documents was to understand the whole process so that we could find possible ways to improve current techniques. However, since these modern kind of documents, have been already designed to be automatically processed, huge efforts would be required to beat the traditional techniques that were taken into account when designing those documents.

Therefore, if the best way to get the semantics of a given piece of information is just to crop a predefined area of the image, we have to focus on improving our handwriting recognition abilities. Analyzing the most well known state of the art techniques at the time, we noticed that very powerful sequence processing neural networks were being trained on top of handcrafted features, specific to a given script and without a clear justification. Trying to improve the feature extraction part seemed an obvious choice. Autoencoders are a popular dimensionality reduction technique, and some of them have proven to produce meaningful hidden representations for other problems. Using the reconstruction error metric allowed us to use unlabeled data for training, which is a very good plus for handwritten text, where manually annotated data is scarce. We could successfully use Variational Autoencoders to learn a feature representation useful to perform handwriting recognition. However, the improvement over traditional handcrafted features was very modest, apparently, despite not having a clear justification to us, those handcrafted features were quite robust. After careful consideration we noticed that, autoencoders, despite producing remarkably meaningful features and discovering a lower dimensionality projection that could almost completely retain all information, were not specifically required to be discriminative for the final goal of transcription.

A second attempt at finding discriminative features for HTR came from attribute embeddings. Attribute embeddings had been very successful in word spotting, and were adapted to word recognition with dictionaries. Being able to tell one word apart from another will somehow require features able to discriminate between different characters. In the case of PHOC this was explicitly true, because the attributes were precisely, the presence or absence of a given character

in the word. By using a network trained to perform attribute embedding on full words, and sequentially applying it to patches of text, we can generate a sequence of observations that can be fed to a BLSTM+CTC. This means that we get the discriminative power of attribute embedding approaches, avoiding their limitations for recognition that were, mainly, that they are unable to deal with out-of-vocabulary words and cannot be extended beyond word level. We had very good results with this approach, however, one could argue that the complexity of having to train two components independently can be considered a drawback, specially now that the trend is to go towards end to end systems.

While working our HTR systems on marriage records datasets, we knew that the final goal was to be able to deal with those documents as if we were processing a form. We wanted to know the role of each word in a record. Therefore we adapted the dataset to this new task by simplifying some transcriptions and unifying some semantic classes to make everything more consistent. Finally we carefully thought of a metric that could be fair for a task with two different levels of complexity and that could be computed fairly for word and line level approaches. This benchmark was presented as a competition in the International Conference of Document Analysis and Recognition. To further encourage research in this interesting area, the competition was made permanent and the benchmark is available in an online platform.

In order to build an information extraction approach fro fully handwritten documents we need to transcribe and tag the sequence of words with semantic labels. If Convolutional Neural Networks can learn to identify images of classes like “dog” that have an amazing internal variability, would it be possible to classify word images into semantic classes? We designed a CNN taking into consideration the specific case of word images, and performed several experiments to tune the network hyperparameters until we were finally able to get good results. But that first system had obvious limitations, first of all, it was not leveraging any contextual information. We tried to add contextual information to the system, at first we tried to feed, not only the current word image, but also the predicted label of the previous word. This greatly improved the performance for semantic categories, but still did not allow us to capture long range dependencies among words. Finally we adapted the network to process full, record level sequences with BLSTM achieving state of the art results in the benchmark.

6.2 Future Work

We have been already highlighted several possible lines of future works throughout the thesis. The final approach of this thesis to information extraction works at word level, thus, it would require a reliable automatic word segmentation to be able to work directly at other levels of segmentation (line, paragraph/record or page). Therefore, an interesting line of research would be trying to find ways to extend

this approach beyond word level, one possible way to do it would be the uses of content based attention models, that have become quite popular recently. Another line of research would be to adapt some of the newest object detection techniques to detect handwritten words, these detectors could in fact use the predictions of our model as an extra source of information to be able to detect relevant words. It would also be very interesting to check how far this approaches can go in documents with an even less rigid structure. It would also be very interesting to find ways to leverage that semantic information to improve the transcription.

Regarding the Handwriting Recognition by attribute embedding approach, it would be interesting to actually bring them to line level, checking for the best way to model the inter word space. Two options come to mind, to model it as an attribute by training the attribute embedding network with pairs of words including the space as an extra attribute or to let the BLSTM to model it implicitly. A future line of research with autoencoders seems harder, I would like to explore how to add label information to the reconstruction loss. Of course, finding different approaches that allow for a better performance for handwriting recognition would be interesting. In particular, one approach that I think could be interesting is training triplets to decide if two handwritten words share the same transcription. Then we could embed a handwritten word into that space, and use that embedding as a seed to a BLSTM based decoder that could generate the transcription.

In the field of form documents, the most interesting and challenging future line of research, in my opinion, is towards being able to have really reliable systems that can learn to understand the form, without the need of predefining the location of the fields. Since, our final approach does precisely that, for documents that can be described as a sequence of words, a possibly interesting line of research would be to try a similar approach, adapted to work at page level, with form documents.

List of Publications

Journals

- J. Ignacio Toledo, Manuel Carbonell, Alicia Fornés and Josep Lladós. (2019) Information Extraction from Historical Handwritten Document Images with a Context-aware Neural Model, *Pattern Recognition*, (Q1)

International Conferences

- J. Ignacio Toledo, Jordi Cucurull, Jordi Puiggali, Alicia Fornés and Josep Lladós. (2015) Document Analysis Techniques for Automatic Electoral Document Processing: A Survey. In "E-voting and Identity". 5th International Conference on VoteID
- J. Ignacio Toledo, Alicia Fornés, Jordi Cucurull and Josep Lladós. (2016) Election Tally Sheets Processing System. In 12th IAPR International Workshop on Document Analysis Systems (DAS)
- J. Ignacio Toledo, Sebastian Sudholt, Alicia Fornés, Jordi Cucurull, Gernot Fink and Josep Lladós. (2016) Handwritten Word Image Categorization with Convolutional Neural Networks and Spatial Pyramid Pooling. In IAPR International Workshops on Structural and Syntactic Pattern Recognition and Statistical Techniques in Pattern Recognition (S+SSPR)
- J. Ignacio Toledo, Sounak Dey, Alicia Fornés and Josep Lladós. (2017) Handwriting Recognition by Attribute embedding and Recurrent Neural Networks. In 14th International Conference on Document Analysis and Recognition (ICDAR)
- Lei Kang, J. Ignacio Toledo, Pau Riba, Mauricio Villegas, Alicia Fornés and Marçal Rusinol. (2018) Convolve, Attend and Spell: An Attention-based Sequence-to-Sequence Model for Handwritten Word Recognition. In German Conference on Pattern Recognition (GCPR)

- A.Fornés, V.Romero, A.Baró, J.I.Toledo, J.A. Sánchez, E.Vidal, J.Lladós. (2017) Competition on Information Extraction in Historical Handwritten Records. In 14th International Conference on Document Analysis and Recognition (ICDAR)

arXiv

- Sounak Dey, Anjan Dutta, J.Ignacio Toledo, Suman Ghosh, Josep Lladós and Umapada Pal (2017) Signet: Convolutional siamese network for writer independent offline signature verification. In arXiv preprint arXiv:1707.02131

Awards

- First Prize *Digitus II* at 5th Edition of Generación de Ideas at Universitat Autònoma de Barcelona.

Bibliography

- [1] Minnesota senate recount: Challenged ballots: You be the judge. http://minnesota.publicradio.org/features/2008/11/19_challenged_ballots/, 2008.
- [2] Citizen’s oversight projects. www.copswiki.org/Cops/BallotStatements, 2009.
- [3] Elections ACT: Scanning of ballot papers. http://www.elections.act.gov.au/mbox{/elections_and_voting/scanning_of_ballot_papers}, 2015.
- [4] C. Adak, B. B. Chaudhuri, and M. Blumenstein. Named entity recognition from unstructured handwritten document images. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 375–380, April 2016.
- [5] J. Almazán, A. Gordo, A. Fornés, and E. Valveny. Word spotting and recognition with embedded attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12):2552–2566, 2014.
- [6] Adnan Amin and Stephen Fischer. A document skew detection method using the hough transform. *Pattern Analysis & Applications*, 3(3):243–253, 2000.
- [7] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. Leveraging linguistic structure for open domain information extraction. In *ACL (1)*, pages 344–354. The Association for Computer Linguistics, 2015.
- [8] Valentina Bachi, Antonella Fresa, Claudia Pierotti, and Claudio Prandoni. The digitization age: Mass culture is quality culture. challenges for cultural heritage and society. In *Digital Heritage: Progress in Cultural Heritage. Documentation, Preservation, and Protection. 5th International Conference, EuroMed 2014. Lecture Notes in Computer Science*, volume 8740, pages 786–801. Springer, 2014.
- [9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

- [10] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4945–4949, 2016.
- [11] Linda Baker and Ann L Brown. Metacognitive skills and reading. *Handbook of reading research*, 1(353):V394, 1984.
- [12] John Bernsen. Dynamic thresholding of grey-level images. In *International Conference on Pattern Recognition (ICPR)*, pages 1251–1255, 1986.
- [13] Anne-Laure Bianne-Bernard, Fares Menasri, Rami Al-Hajj Mohamad, Chafic Mokbel, Christopher Kermorvant, and Laurence Likforman-Sulem. Dynamic and contextual information in HMM modeling for handwritten word recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2066–2080, 2011.
- [14] Théodore Bluche. Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. In *Advances in Neural Information Processing Systems*, pages 838–846, 2016.
- [15] Theodore Bluche. Joint Line Segmentation and Transcription for End-to-End Handwritten Paragraph Recognition. In D D Lee, M Sugiyama, U V Luxburg, I Guyon, and R Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 838–846. Curran Associates, Inc., 2016.
- [16] Théodore Bluche, Jérôme Louradour, and Ronaldo Messina. Scan, attend and read: End-to-end handwritten paragraph recognition with mdlstm attention. *arXiv preprint arXiv:1604.03286*, 2016.
- [17] Theodore Bluche, Hermann Ney, and Christopher Kermorvant. Tandem HMM with convolutional neural network for handwritten word recognition. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2390–2394, 2013.
- [18] Manuel Carbonell, Mauricio Villegas, Alicia Fornés, and Josep Lladós. Joint recognition of handwritten text and named entities with a neural end-to-end model. In *2018 13th IAPR Workshop on Document Analysis Systems (DAS)*, March 2018.
- [19] Wei Chen, Sankaranarayanan Ananthkrishnan, Rohit Prasad, and Prem Natarajan. Variable-span out-of-vocabulary named entity detection. In *INTERSPEECH*, pages 3761–3765, 2013.
- [20] Jan Chorowski and Navdeep Jaitly. Towards better decoding and language model integration in sequence to sequence models. *arXiv preprint arXiv:1612.02695*, 2016.

- [21] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Proc. of the International Conference on Neural Information Processing Systems*, pages 577–585, 2015.
- [22] D. C. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2012*, 2012. Long preprint arXiv:1202.2745v1 [cs.CV].
- [23] Markus Diem, Stefan Fiel, Angelika Garz, Manuel Keglevic, Florian Kleber, and Robert Sablatnig. Icdar 2013 competition on handwritten digit recognition (hdrc 2013). In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 1422–1427. IEEE, 2013.
- [24] Hervé Déjean, Stéphane Clinchant, Jean-Luc Meunier, and Florian Lang, Eva Maria Kleber. Comparing machine learning approaches for table recognition in historical register books. In *2018 13th IAPR Workshop on Document Analysis Systems (DAS)*, March 2018.
- [25] Reza Ebrahimzadeh and Mahdi Jampour. Efficient handwritten digit recognition based on histogram of oriented gradients and svm. *International Journal of Computer Applications*, 104(9):10–13, October 2014.
- [26] Salvador España-Boquera, Maria Jose Castro-Bleda, Jorge Gorbe-Moya, and Francisco Zamora-Martinez. Improving offline handwritten text recognition with hybrid HMM/ANN models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):767–779, 2011.
- [27] Europeana. European digital cultural heritage platform.
- [28] David Fernández-Mota, Jon Almazán, Núria Cirera, Alicia Fornés, and Josep Lladós. Bh2m: The barcelona historical, handwritten marriages database. In *22nd International Conference on Pattern Recognition (ICPR), 2014*, pages 256–261. IEEE, 2014.
- [29] Andreas Fischer. *Handwriting recognition in historical documents*. PhD thesis, University of Bern, 2012.
- [30] Andreas Fischer, Volkmar Frinken, and Horst Bunke. Hidden markov models for off-line cursive handwriting recognition. *Handbook of Statistics: Machine Learning: Theory and Applications*, 31:421, 2013.
- [31] Andreas Fischer, Andreas Keller, Volkmar Frinken, and Horst Bunke. Lexicon-free handwritten word spotting using character hmms. *Pattern Recognition Letters*, 33(7):934–942, 2012.

- [32] Alicia Fornés, Veronica Romero, Arnau Baró, J Ignacio Toledo, Joan Andreu Sanchez, Enrique Vidal, and Josep Lladós. Competition on information extraction in historical handwritten records. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 1389–1394, 2017.
- [33] Volkmar Frinken and Horst Bunke. Continuous handwritten script recognition. In *Handbook of Document Image Processing and Recognition*, pages 391–425. Springer, 2014.
- [34] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1019–1027, 2016.
- [35] Nabil Ghanmi and Abdel Belaid. Separator and content based approach for table extraction in handwritten chemistry documents. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, pages 296–300. IEEE, 2015.
- [36] Nabil Ghanmi and Abdel Belaïd. Recognition-based approach of numeral extraction in handwritten chemistry documents using contextual knowledge. In *Document Analysis Systems (DAS), 2016 12th IAPR Workshop on*, pages 251–256. IEEE, 2016.
- [37] Adrià Giménez, Ihab Khoury, Jesús Andrés-Ferrer, and Alfons Juan. Handwriting word recognition using windowed bernoulli HMMs. *Pattern Recognition Letters*, 35:149–156, 2014.
- [38] Angelos P Giotis, Giorgos Sfikas, Basilis Gatos, and Christophoros Nikou. A survey of document image word spotting techniques. *Pattern Recognition*, 2017.
- [39] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.
- [40] Albert Gordo, Jon Almazan, Naila Murray, and Florent Perronin. Lewis: Latent embeddings for word images and their semantics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1242–1250, 2015.
- [41] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 31(5):855–868, 2009.
- [42] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. of the International Conference on Machine Learning*, pages 369–376, 2006.

- [43] Alex Graves, Santiago Fernández, and Faustino Gomez. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *In Proceedings of the International Conference on Machine Learning, ICML 2006*, pages 369–376, 2006.
- [44] Gill Hamilton and Fred Saunderson. *Open Licensing for Cultural Heritage*. Facet Publishing, 2017.
- [45] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 37(9):1904–1916, 2015.
- [46] Stuart C Hinds, James L Fisher, and Donald P D’Amato. A document skew detection method using run-length encoding and the hough transform. In *10th International Conference on Pattern Recognition (ICPR), 1990*, volume 1, pages 464–468. IEEE, 1990.
- [47] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
- [48] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [49] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991, 2015.
- [50] Theron Ji, Eric Kim, Raji Srikantan, Alan Tsai, Arel Cordero, and David Wagner. An analysis of write-in marks on optical scan ballots. In *Proceedings of the 2011 Conference on Electronic Voting Technology/Workshop on Trustworthy Elections, EVT/WOTE’11*, Berkeley, CA, USA, 2011. USENIX Association.
- [51] Douglas W. Jones. On optical mark-sense scanning. In David Chaum, Markus Jakobsson, Ronald L. Rivest, Peter Y. A. Ryan, Josh Benaloh, Mirosław Kutylowski, and Ben Adida, editors, *Towards Trustworthy Elections*, volume 6000 of *Lecture Notes in Computer Science*, pages 175–190. Springer, 2010.
- [52] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. *Journal of Machine Learning Research*, 2015.
- [53] Daniel Keysers, Christian Gollan, and Hermann Ney. Local context in non-linear deformation models for handwritten character recognition. In *17th International Conference on Pattern Recognition (ICPR), 2004.*, volume 4, pages 511–514. IEEE, 2004.

- [54] Eric Kim, Nicholas Carlini, Andrew Chang, George Yiu, Kai Wang, and David Wagner. Improved support for machine-assisted ballot-level audits. In *Presented as part of the 2013 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections*, Berkeley, CA, 2013. USENIX.
- [55] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR), 2014*, page 1, 2014.
- [56] N. Kooli and A. Belaïd. Inexact graph matching for entity recognition in ocred documents. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 4071–4076, Dec 2016.
- [57] Praveen Krishnan, Kartik Dutta, and CV Jawahar. Deep feature embedding for accurate recognition and retrieval of handwritten text. In *Proc. of the International Conference on Frontiers in Handwriting Recognition*, pages 289–294, 2016.
- [58] Praveen Krishnan, Kartik Dutta, and CV Jawahar. Word spotting and recognition using deep embedding. In *Proc. of the IAPR International Workshop on Document Analysis*, 2018.
- [59] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F Pereira, C J C Burges, L Bottou, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012.
- [60] Daniel S Le, George R Thoma, and Harry Wechsler. Automated page orientation and skew angle detection for binary document images. *Pattern Recognition*, 27(10):1325–1344, 1994.
- [61] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [62] Y LeCun, L Bottou, Y Bengio, and P Haffner. Gradient Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [63] Yann LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Handwritten Digit Recognition with a Back-Propagation Network. *NIPS*, pages 396–404, 1990.
- [64] Yann Lecun and Corinna Cortes. The MNIST database of handwritten digits.
- [65] Cheng-Lin Liu, Kazuki Nakashima, Hiroshi Sako, and Hiromichi Fujisawa. Handwritten digit recognition: benchmarking of state-of-the-art techniques. *Pattern Recognition*, 36(10):2271–2285, 2003.

- [66] Josep Lladós, Marçal Rusinol, Alicia Fornés, David Fernández, and Anjan Dutta. On the influence of word representations for handwritten word spotting in historical documents. *International journal of pattern recognition and artificial intelligence*, 26(05):1263002, 2012.
- [67] Daniel P. Lopresti, George Nagy, and Elisa H. Barney Smith. Document analysis issues in reading optical scan ballots. In David S. Doermann, Venu Govindaraju, Daniel P. Lopresti, and Premkumar Natarajan, editors, *Document Analysis Systems*, ACM International Conference Proceeding Series, pages 105–112. ACM, 2010.
- [68] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, pages 1045–1048, 2010.
- [69] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.
- [70] George Nagy and David Embley. Green interaction for extracting family information from ocr’d books. In *2018 13th IAPR Workshop on Document Analysis Systems (DAS)*, March 2018.
- [71] Karthik Narasimhan, Adam Yala, and Regina Barzilay. Improving information extraction by acquiring external evidence with reinforcement learning. *CoRR*, abs/1603.07954, 2016.
- [72] Wayne Niblack. *An introduction to digital image processing*. Strandberg Publishing Company, 1985.
- [73] Christopher Olah. Understanding lstm networks. <http://colah.github.io/posts/2015-08-Understanding-LSTMs>, 2015.
- [74] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
- [75] Rasmus Berg Palm, Ole Winther, and Florian Laws. Cloudscan - a configuration-free invoice analysis system using recurrent neural networks. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 407–414, 2017.
- [76] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, number 2, pages 1310–1318, 2013.
- [77] Vu Pham, Théodore Bluche, Christopher Kermorvant, and Jérôme Louradour. Dropout improves recurrent neural networks for handwriting recognition. In *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2014.

- [78] Thomas Plötz and Gernot A. Fink. Markov Models for Offline Handwriting Recognition: A Survey. *Int. Journal on Document Analysis and Recognition*, 12(4):269–298, 2009.
- [79] A. Poznanski and L. Wolf. Cnn-n-gram for handwritingword recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2305–2314, 2016.
- [80] I. Pratikakis, K. Zagoris, B. Gatos, J. Puigcerver, A. H. Toselli, and E. Vidal. Icfhr2016 handwritten keyword spotting competition (h-kws 2016). In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 613–618, Oct 2016.
- [81] Ioannis Pratikakis, Basilis Gatos, and Konstantinos Ntirogiannis. Icdar 2013 document image binarization contest (dibco 2013). In *12th International Conference on Document Analysis and Recognition (ICDAR), 2013*, pages 1471–1476. IEEE, 2013.
- [82] Lawrence Rabiner and Biing-Hwang Juang. An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4–16, 1986.
- [83] Danilo J Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic back-propagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning (ICML), 2014*, pages 1278–1286, 2014.
- [84] V. Romero, A. Fornés, E. Vidal, and J. A. Sánchez. Using the mggi methodology for category-based language modeling in handwritten marriage licenses books. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 331–336, Oct 2016.
- [85] Verónica Romero, Alicia Fornés, Nicolás Serrano, Joan Andreu Sánchez, Alejandro H Toselli, Volkmar Frinken, Enrique Vidal, and Josep Lladós. The esposalles database: An ancient marriage license corpus for off-line handwriting recognition. *Pattern Recognition*, 46(6):1658–1669, 2013.
- [86] Verónica Romero, Alicia Fornés, Enrique Vidal, and Joan Andreu Sánchez. Information Extraction in Handwritten Marriage Licenses Books Using the MGGI Methodology. In *Pattern Recognition and Image Analysis: 8th Iberian Conference, IbPRIA 2017, Faro, Portugal, June 20-23, 2017, Proceedings*, pages 287–294, Cham, 2017. Springer International Publishing.
- [87] Verónica Romero and Joan Andreu Sánchez. Category-based language models for handwriting recognition of marriage license books. In *12th International Conference on Document Analysis and Recognition (ICDAR), 2013*, pages 788–792. IEEE, 2013.
- [88] Rakesh S., Kailash Atal, and Ashish Arora. Cost effective optical mark reader. 3(2):44–49, Jun 2013.

- [89] Devendra Sahu and Jawahar C. V. Unsupervised feature learning for optical character recognition. In *13th International Conference on Document Analysis and Recognition (ICDAR), 2015*, pages 1041–1045. IEEE, 2015.
- [90] Jaakko Sauvola and Matti Pietikäinen. Adaptive document image binarization. *Pattern recognition*, 33(2):225–236, 2000.
- [91] Jean Serra. Introduction to mathematical morphology. *Comput. Vision Graph. Image Process.*, 35(3):283–305, September 1986.
- [92] Mehmet Sezgin et al. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1):146–168, 2004.
- [93] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [94] Chandan Singh, Nitin Bhatia, and Amandeep Kaur. Hough transform based fast skew detection and accurate skew correction methods. *Pattern Recognition*, 41(12):3528–3546, 2008.
- [95] Elisa H. Barney Smith, Shatakshi Goyal, Robbie Scott, and Daniel P. Lopresti. Evaluation of voting with form dropout techniques for ballot vote counting. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 473–477. IEEE, 2011.
- [96] Elisa H. Barney Smith, Daniel P. Lopresti, and George Nagy. Ballot mark detection. In *ICPR*, pages 1–4. IEEE, 2008.
- [97] Elisa H. Barney Smith, Daniel P. Lopresti, George Nagy, and Ziyang Wu. Towards improved paper-based election technology. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 1255–1259. IEEE, 2011.
- [98] Elisa H. Barney Smith, George Nagy, and Daniel P. Lopresti. Mark detection from scanned ballots. In Kathrin Berkner and Laurence Likforman-Sulem, editors, *DRR*, volume 7247 of *SPIE Proceedings*, pages 1–10. SPIE, 2009.
- [99] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.
- [100] Tobias Strauß, Max Weidemann, Johannes Michael, Gundram Leifert, Tobias Grüning, and Roger Labahn. System description of citlab’s recognition & retrieval engine for icdar2017 competition on information extraction in historical handwritten records. *arXiv preprint arXiv:1804.09943*, 2018.

- [101] Bruno Stuner, Clément Chatelain, and Thierry Paquet. Handwriting recognition using cohort of LSTM and lexicon verification with extremely large lexicon. *CoRR*, vol. *abs/1612.07528*, 2016.
- [102] Sebastian Sudholt and Gernot A Fink. Phocnet: A deep convolutional neural network for word spotting in handwritten documents. In *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*, pages 277–282. IEEE, 2016.
- [103] Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. Lstm neural networks for language modeling. In *Thirteenth Annual Conference of the International Speech Communication Association*, pages 194–197, 2012.
- [104] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Proc. of the International Conference on Neural Information Processing Systems*, pages 3104–3112, 2014.
- [105] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.
- [106] Chris Tensmeyer and Tony Martinez. Document image binarization with fully convolutional neural networks. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 99–104. IEEE, 2017.
- [107] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada, 2003.
- [108] J Ignacio Toledo, Sounak Dey, Alicia Fornés, and Josep Lladós. Handwriting recognition by attribute embedding and recurrent neural networks. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 1038–1043, 2017.
- [109] J Ignacio Toledo, Sebastian Sudholt, Alicia Fornés, Jordi Cucurull, Gernot A Fink, and Josep Lladós. Handwritten word image categorization with convolutional neural networks and spatial pyramid pooling. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 543–552. Springer, 2016.
- [110] UNESCO. Convention concerning the protection of the world cultural and natural heritage, 1972.

- [111] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.
- [112] Alessandro Vinciarelli, Samy Bengio, and Horst Bunke. Offline recognition of unconstrained handwritten texts using hmms and statistical language models. *IEEE transactions on Pattern analysis and Machine intelligence*, 26(6):709–720, 2004.
- [113] Paul Voigtlaender, Patrick Doetsch, and Hermann Ney. Handwriting recognition with large multidimensional long short-term memory recurrent neural networks. In *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*, pages 228–233. IEEE, 2016.
- [114] Kai Wang, Eric Kim, Nicholas Carlini, Ivan Motyashov, Daniel Nguyen, and David Wagner. Operator-assisted tabulation of optical scan ballots. In *Presented as part of the 2012 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections*, Berkeley, CA, 2012. USENIX.
- [115] Curtis Wigington, Seth Stewart, Brian Davis, Bill Barrett, Brian Price, and Scott Cohen. Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network. In *Proc. of the IAPR International Conference on Document Analysis and Recognition*, pages 639–645, 2017.
- [116] Tomas Wilkinson and Anders Brun. Semantic and Verbatim Word Spotting using Deep Neural Networks. *Handwriting Recognition (ICFHR)*, 2016.
- [117] Safwan Wshah, Girish Kumar, and Vengatesan Govindaraju. Script independent word spotting in offline handwritten documents based on hidden markov models. In *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, pages 14–19. IEEE, 2012.
- [118] Pingping Xiu, Daniel P. Lopresti, Henry S. Baird, George Nagy, and Elisa H. Barney Smith. Style-based ballot mark recognition. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 216–220. IEEE, 2009.
- [119] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proc. of the International Conference on Machine Learning*, pages 2048–2057, 2015.
- [120] Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. *ECCV 2014*, 8689:818–833, 2014.
- [121] Matthias Zimmermann and Horst Bunke. N-gram language models for offline handwritten text recognition. In *Frontiers in Handwriting Recognition, 2004*.

IWFHR-9 2004. Ninth International Workshop on, pages 203–208. IEEE, 2004.

This work has been partially supported by the Spanish project TIN2015-70924-C2-2-R, the European project ERC- 2010-AdG-20100407-269796, the grants 2013-DI-067 and 2016-DI-095 from the Secretaria dUniversitats i Recerca del Departament dEconomia i Coneixement de la Generalitat de Catalunya, the Ramon y Cajal Fellowship RYC-2014-16831, the project RecerCaixa (XARXES, 2016ACUP-00008), a research program from Obra Social “La Caixa” with the collaboration of the ACUP, and the CERCA Programme/Generalitat de Catalunya. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

