UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Context-aware home monitoring system for Parkinson's disease patients ambient and wearable sensing for freezing of gait detection

Boris Takač

Context-aware Home Monitoring System for Parkinson's Disease Patients

Ambient and Wearable Sensing for Freezing of Gait Detection

Boris Takač

A catalogue record is available from the Eindhoven University of Technology
ISBN 978-90-386-3752-5

Proefschrift Technische Universiteit Eindhoven
Keywords: Parkinson's disease / Freezing of Gait / Context-aware system /
Health monitoring / Video tracking / Indoor localization /
Person orientation / Person re-identification

Typeset with LaTeX
Cover design by Šareni Artikl - studio za vizualne komunikacije, Zagreb, Croatia
Printed by Ipskamp Drukkers, Enschede, The Netherlands

Context-aware Home Monitoring System for Parkinson's Disease Patients

Ambient and Wearable Sensing for Freezing of Gait Detection

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties, in het openbaar te verdedigen
op maandag 15 december 2014 om 16:00 uur

door

Boris Takač

geboren te Sisak, Kroatië

Dit proefschrift is goedgekeurd door de promotoren en samenstelling van de promotiecommissie is als volgt:

voorzitter:	prof.dr.ir. A.C. Brombacher
1e promotor:	prof.dr. A. Català (Universitat Politècnica de Catalunya)
2e promotor:	prof.dr. M. Rauterberg
co-promotor:	dr. W. Chen
leden:	prof.dr.ir. L.M.G. Feijs
	prof.dr. J-M. Moreno Aróstegui (Universitat Politècnica de Catalunya)
	dr. A. Rodríguez-Molinero, M.D. (National University of Ireland)
	dr. Nico van der Aa (Noldus Information Technology)

This dissertation was produced under Erasmus Mundus Joint Doctorate Program in Interactive and Cognitive Environments. The research was conducted towards a joint double PhD degree between the following partner universities:

TECHNISCHE UNIVERSITEIT EINDHOVEN

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Acknowledgements

This PhD Thesis has been developed in the framework of, and according to, the rules of the Erasmus Mundus Joint Doctorate on Interactive and Cognitive Environments EMJD ICE [FPA n° 2010-0012] with the cooperation of the following Universities:

Alpen-Adria-Universität Klagenfurt - AAU

Queen Mary, University of London - QMUL

Technische Universiteit Eindhoven - TU/e

Università degli Studi di Genova - UNIGE

Universitat Politècnica de Catalunya - UPC

According to ICE regulations, the Italian PhD title has also been awarded by the Università degli Studi di Genova.

Mojim roditeljima i sestri,
za vaše strpljenje, podršku i ljubav
koju ste mi davali kroz sve ove godine

Context-aware Home Monitoring System for Parkinson's Disease Patients
Ambient and Wearable Sensing for Freezing of Gait Detection

Summary

Freezing of gait (FOG) is a disabling symptom commonly occurring in later stages of Parkinson's disease (PD). It is characterized by brief episodes of inability to step, or by extremely short steps that typically occur on gait initiation or on turning while walking. The consequences of FOG are aggravated mobility and higher affinity to falls, which have a direct effect on the quality of life of the individual. There does not exist completely effective pharmacological treatment for the FOG phenomena. However, external stimuli, such as lines on the floor or rhythmic sounds, can focus the attention of a person who experiences a FOG episode and help her initiate gait. The optimal effectiveness in such approach, known as cueing, is achieved through timely activation of a cueing device upon the accurate detection of a FOG episode. Therefore, a robust and accurate FOG detection is the main problem that needs to be solved when developing a suitable assistive technology solution for this specific user group.

This thesis proposes the use of activity and spatial context of a person as the means to improve the detection of FOG episodes during monitoring at home. The thesis describes design, algorithm implementation and evaluation of a distributed home system for FOG detection based on multiple cameras and a single inertial gait sensor worn at the waist of the patient.

Through detailed observation of collected home data of 17 PD patients, we realized that a novel solution for FOG detection can be achieved by using contextual information of the patient's position, orientation, basic posture and movement on a semantically annotated two-dimensional (2D) map of the indoor environment. We envisioned the future context-aware system as a network of Microsoft Kinect cameras placed in the patient's home, that interacts with a wearable inertial sensor on the patient (smartphone). Since the hardware platform of the system constitutes from the commercial of-the-shelf hardware, the majority of the system development efforts involved the production of software modules (for position tracking, orientation tracking, activity recognition) that run on top of the middle-ware operating system in the home gateway server.

The main component of the system that had to be developed is the Kinect application for tracking the position and height of multiple people, based on the input in the form of 3D point cloud data. Besides position tracking, this software module also provides mapping and semantic annotation of FOG specific zones on the scene in front of the Kinect.

One instance of vision tracking application is supposed to run for every Kinect sensor in the system, yielding potentially high number of simultaneous tracks. At any moment, the system has to track one specific person - the patient. To enable tracking of the patient between different non-overlapped cameras in the distributed system, a new re-identification approach based on appearance model learning with one-class Support Vector Machine (SVM) was developed. Evaluation of the re-identification method was conducted on a 16 people dataset in a laboratory

environment.

Since the patient orientation in the indoor space was recognized as an important part of the context, the system necessitated the ability to estimate the orientation of the person, expressed in the frame of the 2D scene on which the patient is tracked by the camera. We devised a method to fuse position tracking information from the vision system and inertial data from the smartphone in order to obtain patient's 2D pose estimation on the scene map. Additionally, a method for the estimation of the position of the smartphone on the waist of the patient was proposed. Position and orientation estimation accuracy were evaluated on a 12 people dataset. Finally, having available positional, orientation and height information, a new seven-class activity classification was realized using a hierarchical classifier that combines height-based posture classifier with translational and rotational SVM movement classifiers. Each of the SVM movement classifiers and the joint hierarchical classifier were evaluated in the laboratory experiment with 8 healthy persons.

The final context-based FOG detection algorithm uses activity information and spatial context information in order to confirm or disprove FOG detected by the current state-of-the-art FOG detection algorithm (which uses only wearable sensor data). A dataset with home data of 3 PD patients was produced using two Kinect cameras and a smartphone in synchronized recording. The new context-based FOG detection algorithm and the wearable-only FOG detection algorithm, were both evaluated with the home dataset and their results were compared. The context-based algorithm very positively influences the reduction of false positive detections, which is expressed through achieved higher specificity. In some cases, context-based algorithm also eliminates true positive detections, reducing sensitivity to the lesser extent. The final comparison of the two algorithms on the basis of their sensitivity and specificity, shows the improvement in the overall FOG detection achieved with the new context-aware home system.

Contents

List of Figures

Abbreviations

2D - Two-dimensional
3D - Three-dimensional
ANN - Artificial Neural Network
AOE - Absolute Orientation Estimation
APA - Anticipatory Postural Adjustment
ARMT - Advanced Remote Monitoring and Treatment
BAG -Bootstrap Aggregation
BN - Bayesian Network
BSN - Body Sensor Network
BAN - Body Area Network
CFT - Confusion Table
CMOS - Complementary Metal-Oxide Semiconductor
COMT - Catechol-O-methyl transferase
DECB - Depth-Extended Codebook
DNN - Dynamic Neural Network
DT - Decision Tree
DWT - Discrete Wavelet Transform
ECG - Electrocardiograph
EEG - Electroencephalograph
EM - Electromagnetic
EMG - Electromyograph
EPDA - European Parkinson's Disease Association
FI - Freezing Index
FFT - Fast Fourier Transformation
FN - False Negative
fNIR - Functional Near-Infrared Spectroscopy
FOG - Freezing of Gait
FOG-Q - Freezing of Gait Questionnaire
FOV - Field of View
FP - False Positive
FThr - Freeze Threshold

FSI - Freezing State Interpreter
FSM - Finite State Machine
GB - Gigabyte
GMM - Gaussian Mixture Model
GROE - Gravity-Relative Orientation Estimation
GPS - Global Positioning Satellite
GSR - Galvanic Skin Response
GTL - Ground Truth Label
GUI - Graphical User Interface
HD - High Definition
H&Y - Hoehn and Yahr (scale)
IEEE - Institute of Electrical and Electronics Engineers
IIR - Infinite Impulse Response
IMU - Inertial Measurement Unit
IR - Infra-Red
KNN - k-Nearest Neighbour
MAO-B - Monoamine oxidase type-B
MAP - Maximum a posteriori
MARG - Magnetic, Angular rate, Gravity
MCD - Multiple Component Dissimilarity
MCM - Multiple Component Matching
MLP - Multilayer Perceptron
NB - Naive Bayes
NPV - Negative Predictive Value
PCL - Point Cloud Library
PD - Parkinson's disease
PF - Particle filter
PHS - Personal Health System
PI - Power Index
PPV - Positive Predictive Value
PThr - Power Threshold
QVGA - Quarter-VGA (320×240 pixels)
QQVGA - Quarter-QVGA (160×120 pixels)
OCSVM - Once-Class Support Vector Machine
RBF - Radial Basis Function
RAC - Rhythmic Auditory Cueing
RAM - Random Access Memory
RANSAC - RANdom SAmple Consensus
RF - Random Forest
RFID - Radio-Frequency Identification

RGB - Red Green Blue
RGB-D - Red Green Blue Depth
RMSE - Root Mean Square Error
RMT - Remote Monitoring and Treatment
ROS - Robotic Operating System
RT - Random Tree
SD - Secure Digital (memory card format)
SQL - Structured Query Language
SVM - Support Vector Machine
TN - True Negative
TP - True Positive
TRAP - Tremor, Rigidity, Akinesia, Postural instability
TUG - Timed Up and Go
QoL - Quality of Life
UML - Unified Modelling Language
UPDRS - Unified Parkinson's Disease Rating Scale
Wi-Fi - Wireless Local Area Network
WLAN - Wireless Local Area Network
VGA - Video Graphics Array (640×480 pixels)

1 Introduction

Thanks to the expected improvements of life standard in the future, more and more people will have a chance to experience the joy of a long life. And although the promise of longer living by itself is very good news for each of us, we must not forget difficulties that the old age can bring. Some of the biggest threats to one's quality of life (QoL), which increase in likelihood as the person is getting older, are various chronic health conditions such as diabetes, Alzheimer's and Parkinson's disease. Chronic conditions are usually progressive by nature, gradually worsening the physical and cognitive state of the individual, until one ultimately comes to the stage where constant attention of a caregiver is necessary. Future population growth projections, along with already high numbers of elder citizens in developed countries, require changes in the way in which existing public health systems deal with chronic conditions. Catering to needs of those with chronic condition is a labour intensive errand that none of the countries will be able to afford it to its citizens. What is needed is a sensible way to economize on human labour, the most costly element in healthcare. As the answer to this need arose the concept of Personal Health System (PHS).

Technical and scientific advances over the last two decades made computers to evolve rapidly, becoming affordable, powerful and omnipresent in our lives. Nowadays, computing devices have the ability to collect, transmit and store human generated content and, when a sufficient miniaturization is achieved, human body signals. The described state of information technology enables design and implementation of systems able to automatically gather data necessary for construction of the knowledge about the health state of each individual. Using comprehensive knowledge of individual's health state to "assist in the provision of continuous, quality controlled and personalised health services to empowered individuals regardless of their loca-

tion" (Abadie et al., 2011) is the core function of PHS. Foresight (2013) consortium defined PHS as a system consisting of:

- Ambient, wearable and/or in-body devices, which acquire, monitor and communicate physiological and other health-related data → Perception
- Intelligent processing of the acquired information (data analytics), able to couple the acquired information with expert biomedical knowledge, and in some cases, knowledge of social circumstances and living conditions → Cognition
- Action based on the processing of acquired information, either applied to the individuals being monitored, or to health practice more generally, concerning information provision and/or more active engagement in anything from disease and disability prevention to diagnosis, treatment and rehabilitation. → Action

We should note that PHS is not strictly limited to dealing with the care for elders and to difficult cases such as chronic conditions. There is a whole spectrum of possible PHS applications ranging from life-style management (involving well-being, fitness, prevention and early detection) to the most demanding cases necessitating the independent living support. Evidently, the potential applications of PHS correlate with the possible health conditions of different target groups, where the groups are ranging from healthy fit people to the ones with severe chronic cases. Also, it is natural to expect that various health conditions will require different levels of technical complexity in the implementation of supporting PHS. Abadie et al. (2011) proposed a hybrid typology for PHS taking those two factors (health condition and technical sophistication) into account. This typology is presented in Figure 1.1, along with the short descriptions for each type of PHS.

In the examination of Figure 1.1, we turn our attention towards the right side; where the past, the present and the future of PHS for management of chronic diseases is displayed. The concept of Remote Monitoring and Treatment (RMT) was recognized as the key aspect in improving the care for people with chronic conditions. In the first generation of RMT, only the monitoring part was automatized using a single disease-specific sensor. The available amount of data about the patient's condition improved, but that data was still used only for the treatment by on-demand human intervention. In the next phase, there has been present the integration of the remote monitoring directly into a traditional disease management process. Management systems have been designed to fully include medical professionals in the treatment and help them through availability of improved analytical possibilities provided by the knowledge that was extracted from the collected patient data. Concurrently, the independence of the patients has been improved through integrated telecare solutions. After improving the integration, the next step for RMT systems that is occurring right now, is the development of Advanced RMT (ARMT) systems.

ARMT solutions are the ones possessing the capability to permanently monitor one or even several diseases at the same time, and to provide an immediate closed-loop treatment via actuation. These solutions are fuelled with the technological progress reflected in the form of sensor

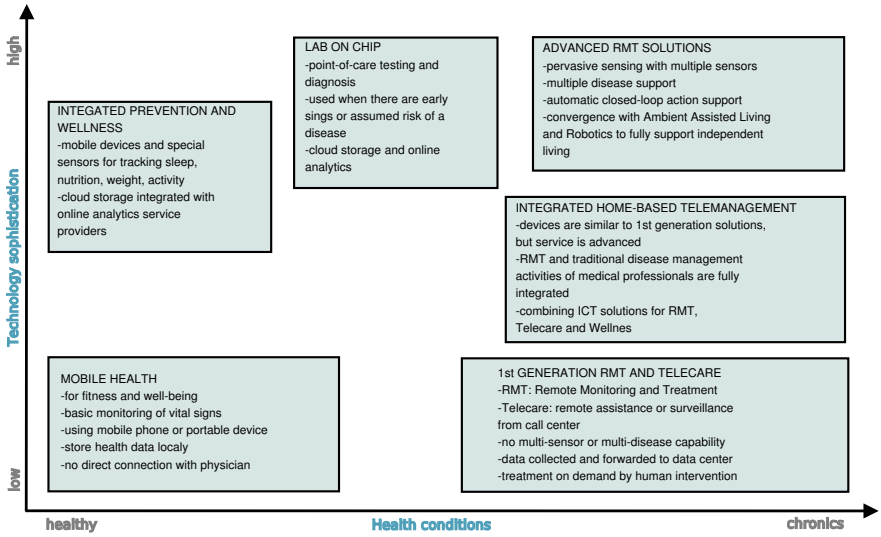


Figure 1.1: Hybrid typology for PHS according to Abadie et al. (2011)

miniaturization, processing power increase, energy consumption decrease, artificial intelligence advancement and omnipresent wireless communication networks. To get the complete picture of the patient's health state, ARMTs not only need to pick-up the vital signals from his body, but also need to capture the context in which those signals have been acquired. ARMT systems thrive on the achievements of context-aware computing and converge with the efforts in the fields of Ambient Intelligence, Ambient Assisted Living and even Robotics.

This thesis presents a contribution within Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments to the body of work in Advanced Remote Monitoring and Treatment systems for chronic condition patients. The chronic condition in the focus of the thesis is Parkinson's disease (PD), or to be more specific, the most peculiar symptom of Parkinson's disease known as *Freezing of Gait* (FOG). The work presented here is a continuation of the line of research on assistive technologies for Parkinson's disease patients started at the Technical Research Centre for Autonomous Living and Dependency Care at Technical University of Catalonia, and it is a continuation of the commitment to healthcare systems design of the Department of Industrial Design at Eindhoven University of Technology.

1.1 Background

1.1.1 Parkinson's Disease

Parkinson's disease is a progressive neurological disorder that results from degeneration of neurons in a region of the brain that controls movement. This degeneration creates a shortage of the brain signalling chemical (neurotransmitter) known as dopamine, causing the movement

impairments that characterize the disease. Parkinson's disease was first described in "An Essay on the Shaking Palsy," published in 1817 by a London physician James Parkinson (Parkinson, 2002). According to the estimations of Parkinson's Disease Fundation* there is between seven and ten million of people worldwide who suffer from the disease, with one million of them living in the United States of America. The European Parkinson's Disease Association† (EPDA) represents more than 1.2 million of people with Parkinson's in Europe, with the projections of this number being double by 2030 due to ageing population and the fact that incidence of Parkinson's increases with age. Economic impact of the disease is already enormous with the current EPDA estimate of annual European cost at 13.9 billion euros.

PD has a great impact on the everyday life of each individual suffering from it. The ability of the brain to generate and coordinate body movements is disrupted in a person stricken with PD, and this disruption produces characteristic signs and motor symptoms which complicate every day living. There are four key motor symptoms, whose occurrence is unmistakable indication of the onset of PD. These cardinal symptoms can be grouped under the acronym TRAP (Jankovic, 2008):

Tremor
: Involuntary rhythmic shaking of the limbs, head or parts of the face, which comes in two types. The first type is the *rest* tremor, which is probably the most common and easily recognizable symptom of PD present in around 75% of cases (Hughes et al., 1993). It happens when the muscles are at rest and are not used. The second type is the *postural* tremor which is phenomenologically identical to essential tremor that appears during action of affected muscles.

Rigidity
: Increased resistance that is present during passive movement of a limb in the joint, due to an increase in the muscle tone. There are two types of rigidity related with PD, *lead-pipe* and *cogwheel* rigidity. The names of the types are based on the description of the feeling experienced by a person who is trying to bend a limb of someone with the symptom. In the case of the *lead-pipe* rigidity, the person applying external force to the affected limb would feel like bending a lead-pipe. During the same examination in the case of the *cogwheel* rigidity, the person would feel a jerky or a ratchet-like movement.

Akinesia (or bradykinesia)
: Hypo-kinetic disorders due to lack of dopamine. Akinesia is defined as the inability to initiate movement resulting in complete stand-still at moments, while bradykinesia describes a slowness in the execution of a movement. Bradykinesia is the most characteristic clinical feature of PD as it encompasses difficulties with planning, initiating and executing movement, and with performing sequential and simultaneous tasks (Berardelli et al., 2001).

*http://www.pdf.org

†http://www.epda.eu.com

Postural instability
: A balance disorder that happens due to the loss of postural reflexes. The most effective way to asses it is the *pull test*, in which the patient is quickly pulled backward by the shoulders. If he takes more than two steps backward, or if there is no postural response at all, this is the indicator of an abnormal postural response. The postural instability (along with FOG) is the most common cause of falls and contributes significantly to the risk of hip fractures (Williams et al., 2006).

There are three general stages in the PD development: *a*) early; *b*) moderate; and *c*) advanced stage, that appertain both to the severity of motor symptoms and to the impact that the disease has on a person's daily living activities. It is important to note that, besides the motor symptoms, the non-motor symptoms of PD have been shown to have as equal influence on the QoL. The list of the possible non-motor symptoms includes: mood changes, cognitive decline, pain, autonomic dysfunction, olfactory problems, dribbling saliva, constipation, sleep disorders, depression, apathy, hallucinations and more (Chaudhuri et al., 2005; Chaudhuri and Schapira, 2009). Progression of PD varies among different individuals, so the stages of PD are better explained by describing their characteristic sets of symptoms, than giving the exact time frames:

Early stage
: Motor symptoms (inner tremor, light tremor) occur on one side of the body. These symptoms may be inconvenient, but do not affect daily activities. People around the patient may notice changes in the person's posture, walking ability, facial expression and voice. All these body changes can cause anxiety, and in the case of receiving a positive PD diagnosis, also depression and apathy might occur. The effectiveness of Parkinson's medications in suppressing movement symptoms at this stage is high.

Moderate stage
: Motor symptoms occur on both sides of the body. The cardinal motor symptoms of tremor, rigidity and bradykinesia are now fully present. Trouble with balance and coordination may develop, resulting in stooped posture and postural instability. This is when *freezing of gait*, described by the patients as "the feeling of having your feet glued to the ground" (Giladi and Nieuwboer, 2008), may occur. The effectiveness of Parkinson's medications is weaker. Weaker effectiveness can cause *wearing-off* effect in which the symptoms re-imerge between the doses. It can also cause involuntary movements (called *dyskinesia*) at the beginning of a dose, when the medication concentration is too high (Marconi et al., 1994).

Advanced stage
: Motor symptoms become so heavy that there is a great difficulty in walking. A patient gets tied to a wheelchair and falls into bed for most of the day. This means that the assistance is needed with all the daily activities. Different combinations of all previously

mentioned non-motor symptoms are possible. The effectiveness of Parkinson's medications is low, which causes the balancing of the benefits of medications with their side effects to be very challenging.

The most common way to asses the stage of PD in a patient is by using rating scales. Two scales are used most often. *Hoehn and Yahr* (H&Y) scale (Hoehn and Yahr, 1967), takes into account only motor symptoms and rates them from 1 to 5. On this scale, depending on mobility difficulties that the patient experiences, 1 and 2 correspond to early stage, 2 and 3 to moderate stage, and 4 and 5 to advanced stage PD. For its practicality, H&Y scale is used in practical research and in patient care setting. The Unified Parkinson's Disease Rating Scale (UPDRS) is the most commonly used scale in clinical research (Mitchell et al., 2000). It consists of four sections. Three sections evaluate the main areas of disability (mental and cognitive state, activities of daily living, motor function), while the fourth section evaluates treatment complications. Very often the UPDRS scale is accompanied with H&Y scale and *Schwab and England Activities of Daily Living* scale (McRae et al., 2002).

The response of the patient to medications for treating symptoms of Parkinson's is as an important feature of each stage of the progression of the disease as are its original symptoms. The medication treatment in PD is done with several types of drugs, such as levodopa, dopamine agonists and inhibitors. The most potent and effective medication for PD is levodopa, developed in the late 1960s (Barbeau, 1969). Levodopa medication is based on L-dopa, a chemical that is a precursor to dopamine. Neurons in the brain have the ability to convert L-dopa into dopamine, which directly nullifies dopamine deficiency responsible for the majority of the symptoms. This is why levodopa has the broadest antiparkinsonian effects compared to any other medication. Levodopa causes nausea and vomiting if converted into dopamine while in the peripheral nervous system, so it is usually combined with carbidopa which prevents this conversion to occur before the medication reaches the brain. Carbidopa cancels the nausea side effects, making carbidopa/levodopa mix the basis of the PD treatment.

The treatment with levodopa also has its downsides. In the early PD, side effects include dry mouth, nausea and dizziness. With the progression of the disease the effectiveness of levodopa decreases, requiring increased dosage and causing patients to experience dyskinesias and *ON-OFF* periods. The *ON-OFF* periods are the occasions when the medication will suddenly and unpredictably start or stop working (Nutt et al., 1984). The best description of the clinical picture of the *ON-OFF* phenomenon might had been given in an extract from a letter written by a patient who had been taking levodopa (Lees, 1989):

> *It is in fact difficult now to stick to the 2-hour regime because of this apparent unreliability. If for instance I find myself over, suffering from so-called involuntary movements, my limbs behaving as if controlled by a drunken marionette master, I am reluctant to take a pill in the midst of these side-effects. So I postpone it. And then before I know where I am I am OFF. ON is quite simply normal; I can survive a dinner party, drive a car, write a fair, round hand, my voice is*

> *normal. I can fall asleep rather easily unless I am trying not to. OFF on the other hand is very unpleasant. I lose almost all motor power in my legs; and this paralysis increasingly now spreads to my arms. Sometimes odd pains and cramps move round the body. There is no position in which I am comfortable. I can't write, I can't type, my speech is slurred and low powered. The OFF comes on with increasingly little warning.*

In the *ON* state the majority of the PD symptoms are suppressed by levodopa, enabling mobility level that is almost comparable to a healthy person. In the *OFF* state all the characteristic symptoms of PD return. This makes prolongation of *ON* periods and avoidance of overdoses to be the main goal for clinicians in the PD management. Other PD medications are usually combined with levodopa to improve its effectiveness. For example, dopamine agonists are drugs that mimic the activity of dopamine in order to stimulate dopamine deficient parts of the brain. If a person takes dopamine agonists, they need less levodopa, which can avoid overdose and reduce dyskinesias. On the other hand, different inhibitors prolong the *ON* state. Monoamine oxidase type-B (MAO-B) inhibitors prevent breakdown of existing dopamine in the brain by blocking the enzymes responsible for the process. Catechol-O-methyl transferase (COMT) inhibitors do the similar. These inhibitors modestly suppress symptoms of PD and are usually used to avoid problems of *wearing-off*. Anticholinergics may alleviate tremor and may help with symptoms associated with the *wearing-off* or the *peak-dose* effect. Similarly, amantadine has been found useful in helping with tremor and reducing dyskinesias.

The pharmacological treatment tailored specifically for each patient can be successful in alleviating majority of symptoms of PD until very advanced stage of the disease progression. The main problem is the dosage in order to avoid side-effects, and how to combine the available medications for the optimal effect. So far, the normal practice for assigning complex medication has included periodic visits to the neurologist. The number and the complexity of Parkinson's disease motor symptoms, along with their variability over time, make the optimal prescription assignment difficult for the therapist.

The systems for remote monitoring, evaluation and management of PD patients (e.g. PERFORM (2008), HOME (2008)) have been recognized as a potential solution to this problem. These systems are able to recognize, monitor and objectively asses patient's motor status and support physicians in taking therapy-modification decisions. The most advanced future solutions, such as REMPARK (2011), are expected to adapt to patient's symptoms on-the-go and dispense drug into the the bloodstream of the patient automatically. In the systems with closed-loop medication dispensation, the accurate recognition of each specific symptom of PD is very important in order to achieve the complete picture of the patient's PD state, and in that way minimize the possibility of an incorrect medication dose.

1.1.2 Freezing of Gait

Freezing of gait (FOG) is a temporary, involuntary inability to initiate or continue movement experienced by approximately 50% of patients with advanced Parkinson's disease (Macht et al., 2007). Giladi and Nieuwboer (2008) define it as "an episodic inability (lasting seconds) to generate effective stepping in the absence of any known cause other than parkinsonism or high-level gait disorders." Most commonly, FOG lasts a couple of seconds, but episodes can occasionally exceed 30 seconds. FOG usually depends on the walking situation. It often occurs at turns, start of walking, upon reaching the destination and in open spaces (Schaafsma et al., 2003). It can also occur when people approach narrow spaces, such as doors, and when people are in crowded places (Giladi et al., 1992). In the home environment, freezing episodes are usually reported by patients to occur at the same location every day.

The apprehension of FOG as an episodic phenomenon is important. Unlike the continuous gait disorders, where the slow progression of gait disturbances allows patients to adapt slowly to the alterations in their walking, with episodic gait disorders it is very hard to make those adjustments. FOG is unpredictable in nature, which influences the QoL of the patient in two ways. The more obvious impact is seen in the reduced mobility and affinity to falls due to balance problems (Bloem et al., 2004; Kerr et al., 2010) which lead to the direct loss of independence. Falls in the elder age are a great cause of hip fractures, resulting in high mortality in PD patients (Coughlin and Templeton, 1980) or admission in the care-taking institution. The less obvious impact is recognized in the fear of future falls and the overall sense of loosing control and feeling of helplessness (Wallhagen and Brod, 1997). Helplessness, and potential embarrassment and frustration that come with it, can be a cause of decreased socialization and lead towards depression (Giladi and Hausdorff, 2006). The significant impact of FOG on QoL in PD patients that goes beyond its effect on gait and mobility has been clearly demonstrated in the study by Moore et al. (2007).

What seems as a simple event described by the terms "lack of movement" or "inability to step" in reality is a very complex phenomenon. The complexity of FOG is reflected in its dependence on the interplay of patient's gait abnormality, patient's internal emotive and cognitive state and his environment. Nutt et al. (2011) consider that FOG might be not one single phenomenon, but possibly a set of several different syndromes in which each syndrome has their own underlying mechanism. The difficulty in providing universal explanation for FOG is visible from completely distinctive ways in which FOG can be manifested. Three types of manifestation, originally introduced by Thompson and Marsden (2000) and later confirmed by Schaafsma et al. (2003), are distinguished:

Shuffling
: Very small steps during which there is practically no lifting of the feet from the ground. It results in a minimal forward movement.

Leg trembling
: Legs tremble with slight movements in the knees, while feet are fully on the ground or

with slightly raised heels. The asymmetrical case, where only one leg is trembling, is also possible.

Complete akinesia
: Person stays immobile, rigid and totally still with no observable motion of the legs. Typically occurring at the movement initiation.

Whenever FOG manifests, what we actually see is the resolution of the process previously triggered by the internal pathophysiology of the motor system and/or external factors in the environment. In other words, the likelihood for FOG to occur depends heavily on the situation that the patient is experiencing. Five types of specific situations are recognized in the literature (Fahn, 1995; Schaafsma et al., 2003):

Start hesitation
: When the person wants to start walking. Frequently it is preceded with a postural change from sitting or lying to standing posture.

Turn hesitation
: When turning during walking. Often happens as a response to the (movable or immovable) obstacles on the path.

Tight quarter hesitation
: When approaching and walking in narrow zones, such as passages or doorways.

Destination hesitation
: When reaching a final target, such as a chair or a sofa.

Open space hesitation
: When there is no obvious reason in the environment for causing the episode.

We can divide the described situations into two groups, depending on whether the patient was already walking when the episode started. The first group, which considers the non-locomotive state as the starting point for the freezing episode, includes only the FOG by start hesitation. The second group encompasses the cases of freezing in which the episodes start while the person is walking, and in this group belong all other types of the listed situations. Nieuwboer and Giladi (2013) used a similar division, using the terms *akinetic freezing* and *motor freezing* in their discussion about potential mechanisms behind the episodic nature of FOG.

Starting hesitation is manifested as a complete akinesia or leg tremor. During the starting hesitations there is no external environment trigger for the onset of the episode. In this case the start and the end of the episodes are linked with the internal desire to execute a motor task. This kind of FOG has been related with the problems in the preparatory phase of the step initiation (known as anticipatory postural adjustment; APA), when the patient is trying to make a first step and move his center of mass forward (Jacobs et al., 2009). The experiments in which the forces under the feet of the patients were measured during such type of FOG

episodes, recorded low amplitude complex oscillations in the range of 3-6 Hz (Hausdorff et al., 2003a). This oscillations were brought into relation with the inability to couple normal APA to the stepping motor pattern, which would enable the instigation of gait.

On the other hand, the evolution of a FOG episode is different when it occurs due to one of the precipitating situations from the second group. Such episode, that is initiated during gait, usually is characterized by one or more of the several following features:

- Incremental decrease in the patient's step length;
- Decrease in joint ranges in a hip, a knee and an ankle;
- Lost temporal control of the gait cycle; and
- Appearance of trembling leg movements at a frequency between 3 Hz and 8 Hz.

These listed episodic gait abnormalities that are preceding and accompanying FOG episodes, are the reflection of the overall set of gait impairments and continuous gait abnormalities that has been found in patients with FOG. Five features of the continuous gait have been observed to be under the negative influence: *a*) bilateral step coordination (Plotnik et al., 2008); *b*) step length (Chee et al., 2009); *c*) gait symmetry (Plotnik et al., 2005); *d*) gait rhythmicity (Hausdorff et al., 2003b); and *e*) dynamic postural control (Jacobs et al., 2009). According to the theory of Plotnik et al. (2012), these continuous gait impairments can start to influence negatively one another, until the point in which the breakdown of the automatic locomotion program becomes inevitable. This idea was supported by several studies that tested spatio-temporal properties of gait. For example, the experiments in which high cadence was imposed on the patients with FOG resulted in frequent FOG episodes (Moreau et al., 2008). Similarly, the experiments in which very short stride lengths were imposed resulted in a provocation of the *sequence effect* (step-to-step reduction in amplitude) which lead to a shuffling FOG (Chee et al., 2009).

The aforementioned theoretical concept of Plotnik et al. (2012) predicts two kinds of potential triggers causing the appearance of episodic gait abnormalities that lead to FOG. The first type of trigger is related to transitions between different types of walking. An excellent example of such trigger is the change in the trajectory type, between the straight line walking and turning. During turning each leg (inner and outer) has its separate gait control program. Changing between the two types of trajectories challenges the locomotion control by requiring asymmetric step lengths and good bilateral step coordination. Since these two gait properties deviate from the normal in FOG patients, the induction of FOG is plausible due to the mutual negative influence of the said gait parameters. Additionally, the demand of the step size reduction on the inner leg can lead to the appearance of the *sequence effect*. One more example of a situation in which a walking type transition occurs is when approaching narrow spaces. When PD patients perceive the space as too narrow for the dimensions of their body, adaptive postural changes involving shoulder rotations may be needed during locomotion to achieve a

collision-free passage. If the upper body rotations are limited due to the faulty dynamic postural control, there will be a large reduction in the speed of movement (Higuchi et al., 2006). This speed reduction can then lead to shorter steps and the *sequence effect*.

The second type of gait breaking trigger are the attention shifts. In cognitive psychology, attentional set-shifting is defined as the ability to move back and forth between tasks, operations, or mental sets in response to changing internal goals or the changes in the environment perceived through senses (Miyake et al., 2000). According to Naismith et al. (2010), the ability to keep different motor and cognitive tasks active at the same time is reduced in the persons with FOG. Passing through a doorway (Cowie et al., 2012), negotiating obstacles (Almeida and Lebold, 2010), turning (Spildooren et al., 2010), reaching destinations - all off these are the types of environment-related situations that require adaptation of the gait pattern, along with the elevated level of attention. The best example of the interference between the required elevated level of attention and a motor control task is visible from the behaviour during performance of a dual-task (Yogev et al., 2005). One dual-task that is often present in daily living situations is walking and carrying a tray with a glass full of liquid. This task becomes even more difficult if making a turn is required, instead of straight walking. Spildooren et al. (2010) have found that turning for 360°, in the combination with a dual-task, is the most important trigger for freezing.

Except by the five types of hesitation situations of Schaafasma that involve motor control and cognitive aspects, FOG behaviour may be caused by the strain in the emotional and mental state. Known as such internal triggers are: stress, anxiety, depression and fatigue (Giladi and Hausdorff, 2006; Moreau et al., 2008). Susceptibility to such conditions may explain why in daily life FOG often happens in crowded areas, when trying to reach a ringing telephone, when trying to enter the elevator or cross a street at the green signal light. Experimental studies on influence of internal mental states on FOG onset are very difficult to conduct, because it is not easy to objectively measure such long term qualities as depression and fatigue. Short-term emotional states like stressful events or momentary anxiety might be easier to sense. One good example of a possible approach to their sensing is the study by Maidan et al. (2010) in which FOG was associated with the increased heart rate dynamics.

Levodopa is mostly beneficial, but occasionally also happens that FOG gets worse under levodopa influence (Ambani and Van Woert, 1973; Giladi et al., 1992). Other types of medication are likewise not fully effective. It was shown that dopamine agonists are able to decrease the *OFF* time in advanced PD patients, which should hypothetically also reduce the amount of *OFF*-FOG. Confusingly, some studies on dopamine agonists reported that FOG episodes are actually more frequent in the patients receiving this type of medication (Jankovic, 1985). MAO-B inhibitors have been associated with a decreased likelihood of developing FOG, but they rarely reduce FOG once it has developed (Giladi et al., 2001a). One study revealed that patients receiving amantadine are less likely to develop FOG (Giladi et al., 2001b), while the other study came to a less favourable conclusion, associating the amantadine treatment with the higher frequency of FOG (Macht et al., 2007).

Fortunately, there exists an additional way to deal with FOG, besides using medications. In the past, it was observed that some of the patients developed by themselves various techniques for solving the start hesitation, such as lateral swaying, stepping over someone's foot, stepping over lines on the floor or moving in a rhythm of music. Observations of these techniques led to the development of the *sensory cueing* as a feasible therapeutic option. As a consequence, rehabilitation approaches based on the *sensory cueing* received a lot of attention during the last decade (Nieuwboer, 2008).

Sensory cueing is defined as "the use of external temporal or spatial stimuli to facilitate movement, gait initiation and continuation" (Nieuwboer et al., 2007). There are three main modalities of cueing: *a*) visual cueing; *b*) auditory cueing; and *c*) tactile cueing. The fourth possible modality is the cueing as a mix of the previous three. Examples of devices used for visual cueing include perpendicular stripes on the floor (Bagley et al., 1991; Morris et al., 1994; Azulay et al., 1999), walking sticks (Dietz et al., 1990), rhythmic flashing light fixed on glass frames (van Wegen et al., 2006) and a laser beam mounted on a chest (Lewis et al., 2000). The most used device for rhythmic auditory cueing (RAC) is metronome (Freedland et al., 2002; Ledger et al., 2008). Tactile cueing via tapping on patient's shoulder or using a combination of audio (metronome) and video cues (bright coloured lines) (Suteerawattananon et al., 2004) has also been used. The theory behind the *sensory cueing* predicts that improvements in walking speed, stride length and cadence should be observed when external stimuli are applied during the gait of a PD patient. However, not all sensory modalities act equally on all gait parameters. Visual cues, which are spatial in nature, help more to enlarge the stride length and generate sufficient amplitude movement (Bagley et al., 1991; Azulay et al., 1999), while rhythmical auditory (temporal) cues target to stabilize the gait timing (Freedland et al., 2002).

The effectiveness of the *sensory cueing* in improving gait has been established for general PD patients, but the evidence is limited for cueing used to mitigate FOG (Morris et al., 1994; Nieuwboer, 2008). Some of the single session studies that specifically explored the relation between the effects of cueing on FOG during walking, showed no change in the number of FOG episodes and walking time, regardless whether the visual (Kompoliti et al., 2000) or the auditory (Cubo et al., 2004) modality had been used. On the other hand, there was a single session study demonstrating that walking over parallel lines is capable to reduce the number of FOG episodes (Dietz et al., 1990), and two longer studies lasting 6 and 12 weeks which indicated the positive influence of the cueing by the assessment with FOG questionnaire scores (Brichetto et al., 2006; Nieuwboer et al., 2007).

Interesting relations were found between the cueing and its influence on starting hesitation, turning and attention. It seems that the movement initiation is more successful when stimulated by visual spatial cues (Jiang and Norman, 2006). Rhythmical temporal cues help to maintain the gait during turns, by forcing patients to apply the wide-arc turning strategy which involves multiple, more evenly, timed steps (Willems et al., 2007). Influence of the cueing to attention related FOG is somewhat surprising. Although it has been demonstrated that walking becomes attention-demanding and worsens when the secondary tasks are performed

(Yogev et al., 2005), the outcome of the study by Rochester et al. (2007) showed improvements in patients' gaits during dual-tasks. This results were explained by the theory that the cues can reduce attentional demands by making it easier to allocate attention.

The preliminary findings in the studies of RAC with FOG patients reveal that the metronome frequency set to be 10 % lesser than the person's self-selected walking speed can improve the stride length (Willems et al., 2006). In the conclusion about the future perspectives of the cueing in FOG, the world renown expert Nieuwboer (2008) considers as crucial factors both the personalized parametrization and the proper use of the cueing modalities:

> *To address the motor control deficits leading to rhythm and amplitude interference, it makes sense to address both aspects in any therapeutic intervention by providing a stabilizing cueing (baseline) frequency combined with appropriate attentional strategies and instructions to alleviate scaling deficits. Where possible, visual cues to maintain and trigger amplitude generation in the very context in which FOG takes place may be of use.*

Her conclusion suggests that the context-aware *sensory cueing* adapted to the current situation of the patient, has the most chances to overcome freezing and re-initiate gait. The challenge in such approach is to reliably and in real-time detect FOG episodes by using the currently existing sensor technology.

1.1.3 Context-aware Monitoring

A completely robust, clinically proven solution for the automatic detection of FOG still does not exist, but there are several research groups that have been making the advancements towards the final goal (Moore et al., 2008; Bächlin et al., 2009a; Zhao et al., 2012). While studying the most prominent systems and methods for FOG detection, we witnessed to a variety of approaches differing in the number of sensors, detection accuracy and detection speed. Still, there are two properties that all the state-of-the-art solutions have in common. First, they have all been based on inertial sensors, usually attached to the middle or the lower part of the patient's body. And second, none of them featured sensor data collected in the patient's home environment.

The detection algorithms based on the inertial sensors (Moore et al., 2008; Mazilu et al., 2012; Zhao et al., 2012) are known to achieve very good accuracies in highly controlled clinical or laboratory tests. The significant problem with sensory data collected in such environments is that during data collection experiments a large percentage of the PD patients do not have FOG episodes as often, or in the same manner, as they would have at their home (Nieuwboer et al., 1998). Therefore, when deployed in free-living conditions, the algorithms that were optimized with the laboratory data might get subjected to unpredicted everyday life situations (Moore et al., 2008; Bächlin et al., 2012). The usual movements and activities (e.g. rhythmically moving legs while sitting or brushing teeth) may produce inertial signal patterns that are unexpectedly similar to those during FOG episodes. Therefore, such situations will most probably result

in false positive detections. The false positive detections may threaten the user-acceptance of a system by annoying patients, or in a worse case, even cause dangerous situations if the *sensory cueing* is engaged in an unfavourable moment. Consequently, it is important to minimize the number of false detections and achieve the maximum possible clinical efficacy in a home environment.

Due to the dependency of FOG on different internal and external triggers, relating patient's movement data with his/her broader contextual image has a potential to significantly improve the FOG detection. Bächlin et al. (2009b) defined four types of context situation aspects that have a potential for a FOG detection improvement and related these aspects with the appropriate sensor modalities:

Situational aspects
: These are the specific situations causing hesitation according to Schaafsma et al. (2003), such as turns, start of walking, walking in narrow spaces and reaching destinations. Possible sensors for sensing this type of situations include gyroscopes for detecting turns and proximity sensors for detecting obstacles.

Local aspects
: FOG often happens at the same location in the patient's everyday environment. Tracking the patient's current location and his location history, and relating those with the previous instances of FOG, could be a good predictor of the next episode. To incorporate the information about the patient's location in the home, it is necessary to have an indoor localization system. So far, general purpose localization systems have been implemented with a variety of sensor technologies (e.g. camera, Wi-Fi, ultrasound) (Teixeira et al., 2010).

Cognitive-affective aspects
: This aspect includes internally oriented freezing factors, such as the attention shifts, cognitive load, stress, anxiety and depression. The appropriate perceptional input for the assessment of the cognitive-affective state can be achieved with sensors for physiological signals (e.g. sensor for galvanic skin response (GSR), electrocardiograph (ECG)).

Physiological aspects
: This aspect is related with the direct manifestation of FOG through physiological parameters like changes in gait and heart rate. This is the most utilized contextual aspect for FOG detection, since the inertial sensing of gait abnormalities is the basis of all the existing ambulatory monitoring systems.

Several recent studies have investigated the potentials of multi-modal sensing. Mainly they have been focused at the additional benefits from the inclusion of physiological and cognitive-affective contextual aspects into FOG prediction. So far, the most elaborate study in terms of the number of sensors and the variety in sensor modalities, included foot pressure sensors,

ECG, GSR and a sensor for the measurement of brain activity (fNIR), along with multiple inertial measurement units (IMUs) (Mazilu et al., 2013a). Since this study was a data collection study, only the visual inspection of results was performed, indicating possible benefits from the GSR and magnetometers signals. A study by Maidan et al. (2010) in which ECG was used, reported that there is an increase in the heart rate several seconds prior to a FOG episode, and that such event could be used as a potential episode indicator. The collection of GSR or ECG data under the ambulatory conditions is difficult because there is currently no sensor technology that would enable long-term robustness in the sensor placement (Mazilu et al., 2013a). Handojoseno et al. (2012) presented results of the laboratory study that used electroencephalography (EEG) signals for the early detection of FOG in the PD patients. The utility of this approach has not yet been tested in daily life situations.

Hitherto, there has not existed any system that explicitly uses the situational and/or local aspects of context to improve the FOG detection. To enable more accurate and robust FOG detection algorithms, and to provide a contribution to the state-of-the-art in the previously non-explored direction, we decided to investigate the properties of these two untapped contextual aspects. Such effort demanded development of a new multi-modal context-aware monitoring system, that will be presented in this thesis.

1.2 Research Objectives

Since the start of the project, we used the term *spatial context* to refer to the union of the situational and local contextual aspects. The development of the system and the method that use the spatial context for improving the detection of FOG was set as the final goal of the thesis. This main goal has been carried out trough the following three research objectives:

Objective 1: *Design of the distributed multi-modal system for home monitoring of FOG*

The starting point towards the final goal of improved detection is the development of the system capable to capture, store and process (in real-time) contextual data of the patient. The system has to be designed to undertake double role, both as a research platform, and a final in-the-home deployment platform. The design of the system must include the best features of the previously developed FOG monitoring systems in terms of usability and robustness, and offer the space for the sensory and algorithmic upgrade. The complexity of the specific FOG situations that we want to recognize in relation to the *spatial context*, requires high density of information and implies the necessity for multiple sensors. The everyday ease-of-use of the system is considered as the top priority, meaning that an endeavour has to be taken to minimize the number of wearable sensors.

Contribution

We present a distributed system consisting of one wearable inertial sensor device worn by the patient and an arbitrary number of additional camera sensors in the patient's environment. The system is designed to support usability, fast setup, maximum spatial

coverage via modularity and multi-functionality. The main contribution accomplished during this objective is the establishment of the knowledge base about the types and onset locations of the FOG episodes. This knowledge base was obtained through a detailed observation of home collected patient videos, and it was a very valuable input for the analysis of the system requirements.

Objective 2: *Development of algorithms for extraction of spatial context information*

Once we are set on the system architecture that provides necessary sensory inputs, one of the most important tasks is to design the algorithms for extraction of the *spatial context* information from the incoming sensory data. In the proposed concept, the extraction of the location aspect of the context is expected through the localization of the patient within a camera network, while the situational aspect should be assessed through the correlation of kinematic parameters of the patient (e.g. walking velocity, height difference) with the inertial data from the worn sensor. In both cases, the tracking and the identification of the patient in the network of cameras is of critical importance.

Contribution

We describe the practical implementation and evaluation of algorithms for video tracking of persons, person re-identification and basic activity classification. The attention in selection and implementation of the algorithms was given to achieving robustness when dealing with potentially noisy sensor data in home environments. The fulfilment of the Objective 2 also resulted with several contributions to the existing state of-the-art in multiple fields:

- We developed a new method for estimation of the absolute orientation of the patient in indoor spaces based on the fusion of gyroscopic data from the wearable IMU device with image-based features from a camera;
- We developed a new method for patient re-identification in a camera network based on machine learning; and
- We developed an algorithm that fuses wearable IMU data with the trajectory data from video tracking to detect the basic human postures and activities (e.g. standing, sitting down, walking backwards).

Objective 3: *Contextualization of freezing of gait using spatial context information*

Contextualization is expected to improve the FOG detection, especially in terms of the specificity rate. Having the perception system that is able to localize the patient in the known environment and recognize his basic movements, we are left with the task of finding the way to optimally use this additional information.

Contribution

We propose and implement the method which uses explicit information about the *spatial context* to support or reject the primary detection of FOG. The primary detection

is based on the current state-of-the art algorithm, which uses solely the accelerometer signal from the wearable sensor. The new context-based detection method is evaluated on data that was collected in the homes of PD patients using the prototype of the new distributed monitoring system. The results for the context-based FOG detection algorithm are numerically compared with the results of the wearable-only detection algorithm. The comparison shows the final contribution of the new approach in the form of the achieved higher specificity.

1.3 Organization of the Thesis

Chapter 2 provides a chronological and critical review of the state-of-the-art in ambulatory systems for detection of FOG. The attention in the critical review is given to the analysis of the common properties, strengths and shortcomings of such systems.

Following the review of the state-of-the-art, in Chapter 3 we make our own exploration about FOG by conducting a comprehensive analysis of the FOG episode situations. For the analysis, we use a set of home videos from the PD patient database of REMPARK project. The newly gathered knowledge about the usual behaviour of patients in their homes, and the recognition of critical situations and locations that form the *spatial context* is used to set the clinical requirements for the new system.

In Chapter 4 we describe the design process and the final concept of the distributed monitoring system. We present the selection of hardware and software platforms, and we end the chapter with the presentation of the general architecture of the home monitoring system

Since the hardware of the system is based on commercially available sensor devices, our work in this thesis was mainly oriented towards the software development. From Chapter 5 to Chapter 8, we describe the design and the implementation of the system software modules. Chapter 5 brings the description of the application for multiple people position tracking using one camera.

In Chapter 6 we introduce and solve the problem of tracking the absolute orientation of the patient in reference to the observing camera. Towards the end of this chapter, we conduct the evaluation of both position tracking and orientation tracking accuracy on a 12 people dataset.

In Chapter 7 we develop a new re-identification method that enables tracking of the patient between the cameras in a home network. The evaluation of the new re-identification method conducted with 16 people in a laboratory environment is presented.

In Chapter 8 we turn our attention to the recognition of the elements of the situational aspect of FOG context. The knowledge of the postures and basic movements is expected to help in the recognition of the situations that are characteristic for FOG, such as the starting hesitation or the turning hesitation. We implement a vision-based classifier to assess the patient's posture state (e.g. sitting, standing) and combine it into a hierarchical classifier for the recognition of elementary human movements (e.g. walking forward, bending, turning) based on fusion of inertial and video data. We report the results for the evaluation of the posture and

activity classifiers done with 8 healthy people and 4 PD patients.

A new FOG detection algorithm that uses the *spatial context* is presented and evaluated in Chapter 9. The algorithm is tested on home data of three PD patients, collected with the prototype of the monitoring system. The characteristics of the patients' home environments are analysed and the system is evaluated for each patient case separately. The final results of the thesis are reported.

In Chapter 10 we offer a general discussion of our research findings and outline the directions for the possible future work.

2

State of the Art

The assessment of FOG is a difficult task due to the variability of its manifestation in each patient. An additional aggravating circumstance is that FOG happens more often during daily life at the home and much less often during observations at the doctor's office or in a research laboratory (Nieuwboer et al., 1998). The data that the medical specialists get from the PD patients when assessing their physical state via clinical examination is not a reliable representative of their state throughout a period of days. The only validated tool that has been available to doctors for assessing FOG in daily living of their patients is the Freezing of Gait Questionnaire (FOG-Q) (Giladi et al., 2000, 2009), which is susceptible to subjective impressions and memories of the patient. Due to the above reasons, the engineering and scientific community recognized the need for the development of more objective methods for FOG assessment, that will be based on quantitative long-term data.

The active monitoring technology has a potential to objectively assess FOG on a long-term scale and to alleviate the episodes that happen in daily living, by using a timely detection and the context-aware sensory cueing. The usual approach to the assessment of motor related symptoms is to use wearable inertial sensors in order to measure the kinematic parameters of the movements of body segments. Since various gait alterations, like short shuffling steps and festinations are characteristic for FOG, the analysis of the gait parameters has been recognized as a good indicator of the patient's FOG state. In this chapter, we are providing a review of the most prominent FOG detection methods, along with our critical insight on their common characteristics, their deficiencies and the possibilities for improvement.

2.1 Chronological Review

The first who tried to detect FOG episodes using on-body inertial sensors were Han et al. (2003). They used one accelerometer at each ankle and with the help of the frequency analysis came to a conclusion that the characteristic signal of a freezing episode contains frequency components between 6 Hz to 8 Hz, setting it apart from the normal walking dominated by the 2 Hz frequency component. Several years later, Moore et al. (2008) used accelerometers mounted on the left ankle of 11 advanced PD patients to measure their vertical leg movement. Power analysis revealed that during freezing episodes there are frequency components in the 3 Hz to 8 Hz frequency band, which are not present during the normal walking or the volitional standing. Their offline detection algorithm used the *freezing index* (FI), the ratio of power in the *freeze* band (3 Hz to 8 Hz) and the *locomotor* band (0.5 Hz to 3 Hz). The *freezing index* was compared against the *freezing threshold*, a threshold value optimized based on the available patient data. All calculation windows in which the *freezing index* was higher than the *freezing threshold* were designated as FOG. The FI algorithm used sliding window of 6 s length for the calculation of power spectrum. Such approach resulted in a considerable detection latency (> 2 s).

The work following the direction set by Moore et al. (2008) has been continued by the group of researchers lead by Bächlin within the DAPHNET (2006) project. They put the emphasis on enabling automatic online detection of the symptom and making the necessary improvements so that the system based on the FI method can be used outside of the laboratory environment. Efforts were done in the direction of lowering the FI algorithm latency and enhancing its specificity. (Bächlin et al., 2009a, 2010a,b). The latency of the FI algorithm dominated by the window length needed for the Fast Fourier Transformation (FFT), was reduced by using the 4 s analytic window with the 0.5 s window step. Their acquisition system consisted of a portable computer and three accelerometer sensors placed on the trunk, a thigh and a shank. To help with the elimination of the false positives that occur in daily situations in which patients are in a static posture state, like sitting or standing, the DAPHNET research group expanded the original algorithm with the second threshold (*power threshold*). This second threshold was used to drive the output of the *freezing threshold* comparison to *False* state in the case when the lack of sufficient power in the total observed frequency spectra (0.5 Hz to 8 Hz) indicated that the person was not moving.

There were also other systems inspired by the algorithm of Moore et al. (2008). Jovanov et al. (2009) developed deFOG, a system consisting of a small sensor module with a 3-axial accelerometer and a 2-axial gyroscope that had an online processing capability and could connect to a wireless headset for RAC via Bluetooth. The sensor module was worn on a foot. Their algorithm used correlation with the total power in the FFT calculation window to eliminate false detections produced by the FI-based thresholding. The algorithm was designed to have a minimum latency in the FOG episode detection thanks to the usage of a very short (320 ms) window for spectral processing.

MiMed-Pants by Niazmand et al. (2011) is a textile integrated measurement device that con-

Table 2.1: Overview of ambulatory systems for FOG detection.

Method	Base Algorithm	Online	Sensors	Participants	Experiment	# FOG events (duration)	Evaluation
Han et al. (2003)	Wavelet comparison	No	2 accelerometers (ankles)	2 patients 5 control	Laboratory (walk, turn)	-	-
Moore et al. (2008)	FFT with FI (6 sec window)	No	1 accelerometer (left ankle)	11 patients	Laboratory (walk, turn, stand, doorway, obstacle)	46 (2-128 sec)	78% FOG events detected correct 20% stand events incorrect
Zabaleta et al. (2008)	STFT analysis	No	6 accelerometers (6 points on leg and hip)	3 patients	Laboratory (walk, turn, sit-stand, obstacle, dual-task)	-	-
Jovanov et al. (2009)	FFT with FI (0.32 sec window)	Yes	1 IMU module (foot)	1 patient 4 control	Laboratory (walk, sit-stand)	-	-
Bächlin et al. (2009a, 2010a,b)	FFT with FI (4 sec window)	Yes	3 accelerometers (trunk, thigh, shank)	10 patients	Laboratory (straight walk, obstacle, dual-task)	237 (0.5-40.5 sec)	73.1% (88.6%)[a] sensitivity 81.6% (92.4%) specificity
Djurić-Jovičić et al. (2010)	DT with ANN (1 sec window)	No	6 IMU modules (feet, shanks, thighs)	4 patients	Laboratory (walk, sit-stand, doorway, turn)	-	-
Niazmand et al. (2011)	DT with FI (1.5 sec window)	No	5 accelerometers (shanks, thighs, belly)	6 patients	Laboratory (walk, turn)	-	88.3% sensitivity 85.3% specificity
Cole et al. (2011)	DNN with upright detection (2 sec window)	No	3 accelerometers (shank, thigh, forearm), 1 EMG (shank)	10 patients 2 controls	Laboratory	87 (> 1 sec)	83.0% sensitivity 97.0% specificity
Zhao et al. (2012)	DT with FI (1.5 sec window)	Yes	5 accelerometers (shanks, thighs, belly)	8 patients	Laboratory (walk, turn)	-	81.7% sensitivity
Mazilu et al. (2012)	RT, RF, DT, BN, KNN, MLP, Adaboost, BAG (1 or 4 sec window)	Yes	3 accelerometers (trunk, thigh, shank)	10 patients	Laboratory (straight walk, obstacle, dual-task)	237 (0.5-40.5 sec)	62.1% (99.7%) sensitivity 95.2% (99.9%) specificity
Tripoliti et al. (2013)	NB, RF, RT, DT (1 sec window)	No	4 accelerometers (wrists, shanks), 2 IMU modules (waist, chest)	5 patients 11 controls	Laboratory (walk, stand-up, open door, drink)	93 (2–20 sec)	81.94% sensitivity 98.74% specificity
Moore et al. (2013)	FFT with FI (2.5, 5, 7.5, 10 sec window)	No	7 accelerometers (back, thighs, shanks, feet)	25 patients	Laboratory (walk, stand-up, turn)	293	84.3% sensitivity 78.4% specificity

[a] with optimal user specific parameters

sists of ordinary pants with sewn-in five tri-axial acceleration sensors (on thighs, shanks and waist). The sensor data from MiMed-Pants can be saved on the µSD-card enclosed in the central processing unit, or it can be sent by wireless connection to a PC. Niazmand et al. (2011) performed offline FOG detection using a hybrid approach in which they analysed the inertial signals both in time and frequency domains in order to reduce the detection delay and minimize the processing requirements. Time domain analysis was used to detect the "normal" walking pattern, which is the pattern that is very regular and periodic in comparison to the "freezing" pattern. A shorter analysis window of 1.5 s was used to detect a suspicious non-rhythmic walking which triggers the frequency analysis on a 4 s window. The final decision was made using the well established FI method.

Recently, Moore et al. (2013) extended their initial work on the FI method by searching for the optimal configuration of the sensor placement and the signal processing parameters. They used seven sensors attached to the lumbar back, thighs, shanks and feet, and tested their signals with different sizes of the analytic window (2.5, 5, 7.5 and 10 s). Binary waveforms of classification output were obtained for each accelerometer sensor by applying the FI threshold. The majority vote that requests at least 4 out of 7 sensors to register FOG at the same moment was used to obtain the unified output of the multi-sensor system. Based on the measures of performance, Moore et al. (2013) found this complex seven-sensor configuration suitable for demanding research applications, but overly complicated for everyday use. A single sensor at the middle of the back was recommended for use in ambulatory monitoring applications.

The complete chronological development of ambulatory systems for detection of FOG is given in Table 2.1. Except the frequency domain based approach to FOG detection which was the most popular at the end of the last decade, we can see a new type of approach that has started to be more prominent in the recent years. This new approach involves the use of various machine learning techniques. In the attempt to classify between different types of gait disturbances Djurić-Jovičić et al. (2010) used 6 inertial measurements units (IMUs) placed laterally on each leg segment of both legs. Their algorithm combined an artificial neural network (ANN) with a simple signal processing and the rule-based classification to distinguish between the normal (walking, standing) and the pathological (small steps, shuffling, akinesia, festination) states. Due to its ability to recognize patterns, ANN was used primary to identify regular strides, while the heuristically set thresholds on the quantities such as energy and direction were used to discriminate between the rest of the classes in the decision tree (DT).

Instead of a static neural network, Cole et al. (2011) applied a dynamic neural network (DNN) (Sinha et al., 2000). Their two-stage FOG detection algorithm consisted of a linear classifier for detecting when the subject is upright (i.e. standing or walking), and a DNN designed to detect FOG given the first classifier decided that the subject is upright. The expected advantage of using a DNN was its ability to learn how the features of FOG change over time, and in that way better capture the time-varying nature of FOG, in comparison with a normal static ANN. The eleven node input layer of the DNN accepted features extracted from the signals of the three accelerometers (shank, thigh and forearm) and the electromyography (EMG) sensor on a

shank. All the features were calculated using a 2 s analytic window.

In an attempt to improve the FOG-detection performance of the wearable system previously developed within the DAPHNET project, Mazilu et al. (2012) started from the presumption that "FOG can be seen as a specific activity in the context of activity recognition" and applied on the problem the majority of the supervised machine learning approaches used for activity recognition. Their list of applied algorithms included: Random Trees (RT), Random Forests (RF), Decision Trees (DT), Naive Bayes (NB), Bayes Nets (BN), k-Nearest Neighbour (KNN), Multilayer Perceptron (MLP), boosting (AdaBoost) (Freund and Schapire, 1996) and bootstrap aggregating (BAG) (Breiman, 1996). Besides the attempt to improve the FOG detection rate, the authors also strived to improve the wearable system in terms of the technology, economics and unobtrusiveness by using a smartphone as the main online processing unit. The FOG detection classifiers were trained online and then serialized and downloaded onto a smartphone. Their new Android application utilized de-serialized classifiers with online sensor data to detect FOG events in a real-time. The extensive evaluation of the system used the DAPHNET dataset (Bächlin et al., 2010b) with 10 patients and the accelerometers fixed at three positions (lower back, thigh and shank).

The most recent attempt of using machine learning for automatic detection of FOG was presented by Tripoliti et al. (2013). Their new method takes signals received from six accelerometers, placed on the right and the left leg, the right and the left wrist, the chest and the waist and two gyroscopes placed on the chest and the waist. The entropy of signal for each axis of each sensor is calculated in the sliding window and taken to be a part of the feature vector used in the classification. Four different classifiers were tested: Naive Bayes, Random Forests, Random Trees and Decision Trees. The best results were achieved for the Random Forests classifier. The Decision Tree algorithm came close second in accuracy, while being much simpler and offering a lesser complexity in the decision making process.

2.2 Critical Insight

In the following part, we present the most significant aspects of the reviewed systems and methods for the ambulatory FOG detection, along with our insight into their strengths, shortcomings and future trends:

Threshold based vs. machine learning approach

> The majority of early solutions used signal processing in frequency domain to obtain the spectral components of signals coming from inertial sensors, and then based their decision on the comparison of this newly obtained frequency spectrum with the frequency spectrum characteristic for the normal gait. The comparison that leads to the decision about the FOG, can be done either directly between the separate spectral components (Han et al., 2003; Zabaleta et al., 2008), or implicitly using the *freezing index* approach (Moore et al., 2008; Jovanov et al., 2009; Bächlin et al., 2009a; Niazmand et al., 2011;

Moore et al., 2013). Regardless, in both cases it is necessary to obtain the threshold values upon which the comparison is based. For best results user-dependent thresholds are required, due to different types of manifestation of FOG between the patients. Since usually only a few threshold comparisons are used in the decision process, the optimal threshold values are easily found with the parameter sweep method.

In contrast, the machine learning approaches deal with data of higher dimensionality and have the ability to automatically and optimally set decision boundaries for a specific dataset. It was shown that different supervised machine learning algorithms can outperform the threshold-based approaches in terms of detection accuracy, if given enough training data for the specific patient (Mazilu et al., 2012; Tripoliti et al., 2013). A recent study by (Mazilu et al., 2013b) has shown that by using the unsupervised feature learning with the accelerometer data, it is possible to identify the occurrence of patterns even before the FOG episodes start. This approach could lead towards a very effective FOG prediction, and it will probably be one of the main directions for the research in the future.

Online processing and minimal latency

The ability to recognize FOG in (nearly) real-time is a necessary prerequisite for a timely actuation of the *sensory cueing*. Previously to the advent of the "age of the smartphone", the ability to process data online in the ambulatory system required the development of special wearable hardware devices and systems (Jovanov et al., 2009; Bächlin et al., 2009a; Zhao et al., 2012; Rodríguez-Martín et al., 2013). Mazilu et al. (2012) have shown that modern smartphones have a serious potential to be used not just as the processing and communication hub of a body sensor network (BSN), but also as a single necessary sensing device. In any case, the commitment to online classification requires optimized algorithms with a small memory imprint and minimal processing requirements, which both help to preserve the battery life.

The size of the analytic window used to extract the features for classification, is the key parameter for the faster detection of FOG. The mean latency of the FOG detection increases linearly with the size of the window. The size of the analytic window also influences the discriminative power of the algorithm. The exact influence depends on the type of the used approach. For instance, when using machine learning algorithms Mazilu et al. (2012) observed that shorter windows have a lesser discriminative power due to the noise. On the other hand, when using the *freezing index* method, Moore et al. (2013) noticed that shorter windows (towards 1 s) result in a higher sensitivity, and longer windows (towards 10 s) result in a higher specificity. Hence, the state-of-the-art FI based methods use a 4 s to 6 s window length, while the machine learning approaches use shorter 1 s to 2 s windows.

Type, position and number of sensors

Accelerometer was used as the primary sensor in all the reviewed systems for the am-

bulatory FOG detection. All of these systems, except MiMed-Pants, used at least one accelerometer placed at the low part of the leg (either shank, ankle or foot). Additional accelerometers in the systems that use multiple sensors were fixed at the positions higher on the body, such as the thigh, the waist or the chest. The lower leg is used as the primary sensing location because it is the closest to the point of impact of the foot on the ground, which is the main source of accelerations during walking. The potential problem with placing accelerometers higher on the body is that the inertial forces produced by the foot impact will be attenuated. Several studies (Bächlin et al., 2009a; Mazilu et al., 2012; Tripoliti et al., 2013; Moore et al., 2013) have examined the accuracy of FOG detection algorithms that use solely accelerometers fixed close to the middle of the body (thigh, waist). It was shown that in this case the negative effect of the acceleration attenuation lowered the detection accuracy for only 1% to 2%, compared to the case when the same algorithms used the lower leg sensor placement. The best results were, of course, achieved when the algorithms used all available sensors, both on lower and upper legs. Having one sensor on a convenient location seems like the best way to achieve a good online wearable solution, since it minimizes the energy consumption, communication problems and the system setup time in the everyday use.

Except with the superior sensor positioning and the higher number of sensors, the improvement in the detection accuracy has also been sought with the introduction of the additional sensor types, such as a gyroscope and an electromyograph. Their role was to enrich the basic movement information sensed by the accelerometers by providing new features such as the energy of the limb rotation (Djurić-Jovičić et al., 2010) or the energy of muscular activity (Cole et al., 2011). Using new types of sensors for sensing physiological changes could be a promising path, not just for detection, but for the prediction of the actual FOG episodes. For example, the study of Maidan et al. (2010) suggests that there is an increase in the heart rate several seconds prior to the freezing episode, which is a favourable condition for the potential use of electrocardiograph (ECG). Even better example is the study by Handojoseno et al. (2012) in which an electroencephalogram (EEG) was used to predict the transition from walking to freezing with around 75% specificity and sensitivity. The main problem that these physiology based approaches will need to solve in the future, is to ensure the quality of the necessary physiological signal acquired outside of the experimental laboratory environment (Mazilu et al., 2013a).

Scripted laboratory datasets and spatial context

An important fact that was noticed is that all the datasets used in the studies were produced in a clinical or a laboratory environment. When the researchers who study FOG do clinical or laboratory data collection with the patients, they need to come up with different ways to induce freezing episodes. Through the years, more and more potentially triggering situations were identified. The latest datasets contain the combination of the walking along predefined trajectories and the simulation of several daily living activities, that are all intertwined with the potential FOG triggering situations. The most often

captured situations in the datasets were: sitting, sit-to-stand transitions, straight walking, walking around obstacles, 180° and 360° turning, opening door, passing through doorway, making a stop and executing dual-task. There does not yet exist any standardized experimental protocol for FOG research, which would provide the exactly defined combination and order of the FOG triggering situations. Instead, each group of researchers have hitherto used their own protocol.

One major deficiency of the data collection, as it has been done so far, has to do with FOG being heavily dependent upon the environment around the patient. A clinical or a laboratory environment are not people's natural environment, and their freezing episodes do not happen on the artificial polygons in the same way as they would happen at the home. That especially has to do with the specific locations at the home where FOG happens every day due to unknown parameters (e.g. discomfort, fear or just plain habit), which we are not able to recreate in a laboratory setting. The relation between the FOG patient and the space surrounding it (that we call *spatial context*), has so far not been thoroughly studied, especially not in the home environment. Recently, there has been some intention to record FOG episodes at a home, mainly under the patronage of two big European projects, REMPARK (2011) and CuPID (2011). Even though the newest datasets from these European projects have been recording data at patients' homes, they are still limited to the acquisition of the physiological parameters and do not have the intention to capitalize on the *spatial context* as a factor in the FOG detection.

Evaluation measures

In clinical decision making it has been generally accepted that the results of diagnostic tests are reported in terms of sensitivity and specificity (Simon and Boring, 1990; Lalkhen and McCluskey, 2008). In the FOG detection these statistical values are defined based on the relation of the classification output with the ground truth value set by the expert medical personnel. In the case when a FOG is correctly classified by having a positive classifier output and a positive ground truth label, a *true positive* (TP) detection is obtained. If a classifier detects a FOG, but the ground truth label claims the opposite, it is the case of a *false positive* (FP) detection. When a FOG is not correctly detected during an actual FOG episode, it is considered to be a *false negative* (FN) detection. All other instances when the classifier predicts that there is no FOG, while there is truly no FOG are *true negative* (TN) cases. Sensitivity* is defined as the ability of the system to identify the episode correctly. It is calculated as the proportion of the number of truly positive (TP) classifications to the total number of classifications in which FOG should have been found positive (TP + FN). Specificity† is related with the ability of the classifier to exclude a FOG episode correctly. As such, it is obtained as the ratio of the number of truly negative classifications (TN) and the number of instances in which the

*sensitivity = $\frac{TP}{TP+FN}$

†specificity = $\frac{TN}{TN+FP}$

FOG should have not been detected (TN + FP).

The last column in the Table 2.1 gives the overview of the values of average sensitivity and specificity that were achieved by the reviewed systems while using user-independent data in the evaluation. The values in parentheses are the results with the user-dependent data, where the same data of one patient was used for the classifier training (or manual threshold setting) and for the subsequent classifier evaluation. Although the majority of the systems used sensitivity and specificity as the measures in evaluation, it is hard to make straightforward comparison of the presented results. One of the reasons is that sensitivity and specificity were not always calculated in the exact same manner in all the studies. When sliding windows are applied in the FOG classification, there are two main ways of evaluation applicable, the *window-based* and the *episode-based* evaluation. The *window-based* evaluation (Mazilu et al., 2012; Tripoliti et al., 2013) compares classification output of each window with the ground truth at the moment when the classification is executed. In the *episode-based* evaluation (Bächlin et al., 2009a; Niazmand et al., 2011; Cole et al., 2011; Moore et al., 2013), a classification is considered to be TP if the classifier detects FOG at least in one window during the time of lasting of the actual episode. Furthermore, a short delay (2 s) between the classification output and the start of the episode is usually tolerated. The total number of classification values participating in the calculation of sensitivity and specificity for the *window-based* evaluation is equal to the total number of windows, while in the episode-based evaluation it is equal to the total number of FOG episodes. This means that in the former case the statistical calculation is done with tens to hundreds of times more values than in the latter case.

Length and manifestation type of FOG episode

Another reason why it is difficult to mutually compare the reported results of the reviewed studies, is the minimal length of the FOG episodes used for evaluation. The lengths of the FOG episodes in each study are given by the values in the parentheses in the *FOG events* column of Table 1.1. In this table, it can be notices that among the studies that reported the information about the length of the episodes, two studies (Bächlin et al., 2009a; Mazilu et al., 2012) worked with the minimal length of episodes of only 0.5 s, one study (Cole et al., 2011) with the 1 s episode length, and two studies (Moore et al., 2008; Tripoliti et al., 2013) with the 2 s episode length. The shorter episodes that last under a second contain very fast changes in the stepping pattern that are sometimes difficult to spot, even by an experienced observer. On the other hand, the freezing pattern is more recognizable (both for a human observer and a pattern based detection algorithm) in the episodes that last several seconds. This implies that a more sensitive algorithm is required for capturing very short FOG episodes, and that the accuracy of the algorithms tested with a longer minimal episode length would probably not be as high, if these algorithms had been applied to the shorter (0.5 s) episodes.

Another manifestation-related factor influencing FOG detection accuracy are akinetic

episodes. Due to the total lack of movement, it is impossible to detect truly akinetic episodes using solely inertial measurements (Niazmand et al., 2011). Sometimes even the experienced human observers can have problems with recognizing akinesia in FOG, especially if they base their decision solely on the video, and are not aware of the patient's context and intentions. The additional sensor modalities like ECG, or the additional information like the location based behaviour patterns, have to be used in the systems that have the goal of detecting the akinetic FOG. The automatic detection of the akinetic FOG is still a completely open area of research.

2.3 Summary

We have provided a chronological and a critical review of the state-of-the-art in the ambulatory systems for FOG detection. The chronological review reported on the most relevant systems and studies in the ten year period between the years 2003 and 2013. The state-of-the-art presented in this chapter is an updated version of the initial state-of-the-art made when the work on this thesis started at the beginning of 2011. Since the year 2011 we have witnessed several novelties bringing the improvement to the field, such as the introduction of machine learning algorithms, the use of smartphones and the experimentation with the additional modalities like GSR and ECG.

The observed main approach for the implementation of the ambulatory system for FOG detection is to use several inertial sensors (accelerometers) on the lower extremities. Although some studies have used a fairly high number of accelerometers (5 or more), several comparison studies (Mazilu et al., 2012; Samà et al., 2013) have proved that good detection results can be achieved already with one sensor placed in the waist region. In the terms of the preferred algorithmic approach, the *freezing index* calculation is the de-facto basic method that was gradually expanded, improved and tested by several different research groups throughout the years. The recent use of various machine learning algorithms showed their potential to become a new state-of-the-art approach in the future. Furthermore, we observed the trend towards the minimization of the latency in detection, and even working towards the pre-emptive behaviour and the development of a system that would be able to predict in advance the future upcoming episodes.

The literature review has shown that there does not yet exist a mature home tested monitoring system (available on the market), although a few research groups might be close to the required solution. Currently, it is hard to directly compare the results between the groups and pick the best approach due to the difference in the datasets (episode length, manifestation types) and the measurements they used in evaluation. Generally, the detection accuracy of the existing systems based on the clinical and the laboratory data obtained through scripted lab experiments is around 80% to 90% in terms of sensitivity and specificity. Until we have a way to collect long-term PD patient data in their home environment, it will be unknown how much exactly the activities and fluctuations of other PD symptoms in daily living influence the

detection of FOG.

The knowledge of the context of the patient was recognized as a good way for improving the FOG detection. Some of the newest systems added physiological sensors for tracking the cognitive aspects (Handojoseno et al., 2012; Mazilu et al., 2013a). However, we have not witnessed any systems or studies that explicitly in their detection algorithms use the dependency of FOG on the properties of the home environment. There is a knowledge gap about the required characteristics and the possible benefits of location based context-aware home monitoring systems for FOG. We start the exploration of this gap in the next chapter, with a detailed analysis of the behaviour of the PD patients in their homes.

3

Characterization of Freezing of Gait on REMPARK Database

REMPARK database (Samà et al., 2013) is a heterogeneous database for movement knowledge extraction consisting of inertial signals, ground truth videos and questionnaires collected from 90 PD patients within the REMPARK project (Cabestany et al., 2013). The database has been built with the collaboration of 4 different hospitals: Maccabi Healthcare Centre (Israel), Fundazione Santa Lucia (Italy), National University of Galway (Ireland) and Centro Médico Teknon (Spain). Movement signals from PD patients have been collected in uncontrolled home or outdoor environments in order to be used in training and evaluation of learning algorithms for detection of all major symptoms in PD: tremor, FOG, gait bradykinesia, dyskinesia, as well as the recognition of the *ON* and *OFF* motor states.

D2FOG subset of the REMPARK database is the part of the database that contains signals necessary for the development and the evaluation of the FOG detection algorithms. The inertial signals and the videos for the D2FOG subset were collected from the PD patients with at least 2 points on H&Y scale (Hoehn and Yahr, 1967) and a reported history of FOG. For this thesis, we analysed inertial and video data of 17 PD patients (12 men, 5 women) from the D2FOG subset. The analysed patients had an average age [$\mu = 72.2, \sigma = 5.21$] years and an average H&Y score [$\mu = 2.91, \sigma = 0.40$].

The goal of this chapter is to present the results of a systematic and objective analysis of FOG in the home environment. To achieve this goal we use the D2FOG subset of the REMPARK database. We start the chapter with the descriptions of the tests performed by the patients in

the experiments for collection of the FOG-related movement data. To achieve an objective assessment of FOG, we use a state-of-the-art FOG detection algorithm. The outputs of the algorithm are compared with the ground truth observations by human observers. For each misdetection of the algorithm, we record the description of the context in which it occurred. At the end of the chapter, we present the results in the form of the assignment of certain types of the algorithm misdetections to the specific categories of the contextual influences in the environment.

3.1 Collection of FOG-related Movement Data

Data collection for the D2FOG subset was done by research teams consisting of at least 3 people. Patients were in their homes wearing the inertial sensor device called *9x2* (Rodríguez-Martín et al., 2013) on the elastic belt around their waist. The sensor was fixed on the left side of the body above the hip. Acquired inertial data was stored on a μSD card with 200 Hz sampling rate*. During data collection sessions one member of the research team stayed close to the patient at all times to prevent possible falls, one performed video recording using a mobile phone video camera of HD quality (Google Nexus S), and one clinical expert executed on-site observation of the FOG episodes (and other PD symptoms) using specially designed annotation software for a tablet computer. To keep the experiment objective, and to avoid deconcentration of the patient, the patient was not asked to give any subjective reports of the FOG instances during the tests.

The experimental protocol for D2FOG subset of REMPARK database consisted of four types of FOG-related short controlled tests:

Indoor walking test
: The patient starts while seated on a chair in the living room. The patient stands up and shows his house to the researchers, as if he was trying to sell the house. He shows each room and explains what is the room for. After the tour of the house, the patient returns to the chair in the living room and sits down again. Since during the test the patient is making his own decisions about what path to take inside the house, the test is useful for capturing the natural walking behaviour and non-deliberately provoked FOG episodes. Estimated duration of the test is between 5 and 10 minutes.

Gait test
: Gait test is done after the indoor walking test. The test is executed by having the patient walk on a clear straight path of the length of approximatively 20 meters. Due to spatial constraints the test is usually done outdoors. The main purpose of this test is to measure the speed and the cadence of the patient and not to explicitly provoke a FOG. A video camera records the patient, not only during the test walk, but also while he is exiting

*Detailed description of all devices and tools for data collection is provided by Samà et al. (2013)

and returning to the building where he lives. This allows us to observe potential FOG episodes under natural circumstances. The estimated duration of this test is 5 minutes.

FOG provocation test
: This test aims to capture several FOG episodes in the short period of time. The patient walks through the door, a passage or a narrow place repeatedly, up to 10 times. This zone is placed at the middle of the straight 5 meter path. The test is quite similar to the ordinary *Up an Go Test* (Mathias et al., 1986). Each trial of the test starts with the patient sitting on a chair. The patient gets up and goes through a narrow FOG provoking zone. After passing the zone and reaching the end of the path, a 180° turn is taken and the patient goes back through the FOG provoking zone, reaches the chair and sits down. Estimated duration of the test is 5 minutes.

Tremor and FOG false positives test
: In this test the patient is invited to perform the following activities (depending on the availability of the necessary props in the home): to brush their hair, to shake a deodorant bottle, to erase something with a rubber, to type on the researcher's computer (or in their own computer if available), to wipe the glasses or the furniture and to wash a glass. All these activities simulate upper body movements that have the frequencies corresponding to tremor. Consequently, since the inertial unit for detecting FOG is placed around the waist, the measured upper body movement frequencies are similar to the frequencies characteristic for leg trembling in FOG, which can potentially result in false positive detections. Estimated duration of the test is 5 minutes.

The "Indoor walking test" and the "Gait test" were performed twice, both in the *ON* and the *OFF* state. However, the "FOG provocation test" usually was performed only once, in the *OFF* state. Similarly, the "FOG false positives test" was performed only in the *ON* state.

3.2 Video and Inertial Signal Assessment

The assessment of the signal database was a four stage process. The first stage dealt with establishing the ground truth for the FOG episodes. This stage was the necessary precursor for the second stage, which was to run the current state-of-the-art FOG detection algorithm and obtain its optimal classification labels for the dataset. Comparing the algorithm's output labels with initially established ground truth labels in the third stage, we found the exact time instances in which the algorithm gave false positive and false negative FOG detections. In the final stage we assessed the context in which the falsely detected FOG episodes happened. We observed the contextual causes for the false detections produced by the algorithm and we searched for the characteristic types of situations that indicate the higher probability for misdetections.

3.2.1 Moore-Bächlin Algorithm

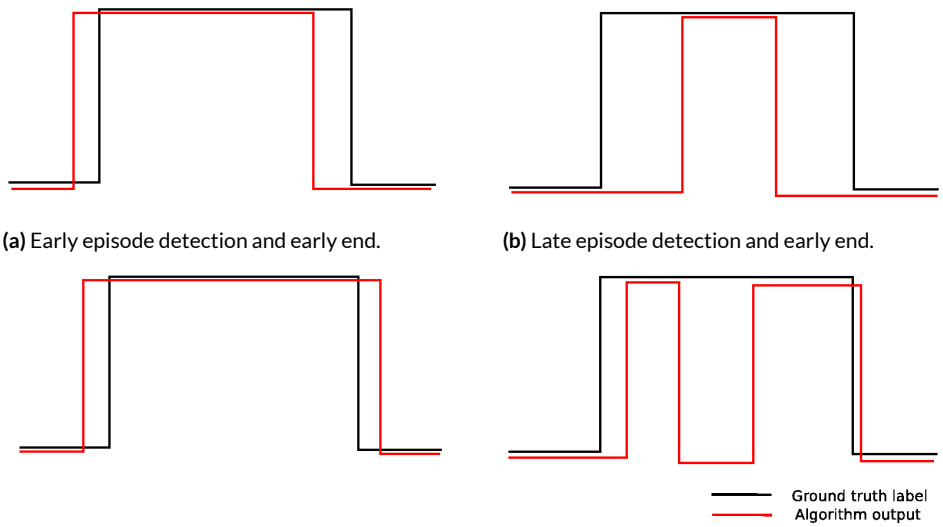
Under the name Moore-Bächlin (Moore-Bächlin) algorithm we refer to the FOG detection algorithm that is based on the use of two thresholds, the *freeze threshold* (FThr) introduced by Moore et al. (2008), and the *power threshold* (PThr) introduced by Bächlin et al. (2009a). For a long time the Moore-Bächlin algorithm has been considered as the state-of-the-art FOG detection method, due to its ease of implementation, low algorithm complexity, and good detection results proven on a publicly available dataset. For these reasons we used the Moore-Bächlin algorithm as the main algorithm in the analysis of the FOG characteristics on the D2FOG dataset.

The algorithm was already briefly mentioned earlier in the Section 2.1, and now we give a more detailed description. The principle described by Moore et al. (2008) requires the calculation of the *freezing index* (FI) from the FFT signal by dividing the spectral power in the 3 Hz to 8 Hz *freeze band* with the spectral power in 0.5 Hz to 3 Hz *locomotor band*. The FI values above the *freeze threshold* are identified as a FOG. The FI has very high values when the patient is in a static situation, such as sitting or standing in the place. This happens because both the locomotor and the freeze band have a very small power when there is no movement. Having a very small value in the divisor results in a high ratio value, regardless of the value of the dividend. The *power index* (PI) calculated as the sum of powers in both bands (FB and LB) takes care of this problem (Bächlin et al., 2009a). A person is considered to be moving only when the PI value exceeds the *power threshold*. This condition is the prerequisite for the subsequent comparison of the FI value with the *freeze threshold* which forms the final detection output. If the value of the PI is under the *power threshold*, then the Moore-Bächlin algorithm output automatically declares *False* FOG state. The Moore-Bächlin algorithm can be defined as a piecewise function by the following equation:

$$FOG(FI, PI, FThr, PThr) = \begin{cases} True : FI > FThr \wedge\ PI > PThr \\ False : otherwise \end{cases} \tag{3.1}$$

The choice of the *freeze threshold* and the *power threshold* parameters has the direct influence on the sensitivity and the specificity of the Moore-Bächlin algorithm. With the *freeze threshold* set too low there will be too many false detections, while the *freeze threshold* that is set too high will result in additional FOG events being missed. The behaviour of the algorithm output in relation to the *power threshold* follows a similar trend. A low *power threshold* acts towards a higher sensitivity and a lower specificity, because it accepts a higher percentage of analytic windows as the ones with a potential FOG episode. Oppositely, a high *power threshold* designates a major portion of analytic windows as a part of some non-FOG static activity which raises the specificity and lowers the sensitivity of the FOG detection. The main task in the application of the Moore-Bächlin algorithm is to properly optimize these two thresholds. The optimization can be conducted to get one common pair of values of the threshold parameters for all patients in the dataset, or it can be done to get a pair of values for each patient separately.

The recorded accelerometer signal of each patient was used as the input in the MATLAB



(a) Early episode detection and early end.

(b) Late episode detection and early end.

(c) Early episode detection and late end.

(d) Multiple episode detections during one labelled episode.

Figure 3.1: Event-based detection algorithm evaluation.

implementation of the Moore-Bächlin algorithm. Raw data was resampled at 40 Hz, and the detection windows of the length 2.56 s were applied. For each combination of the patient and his FOG state, we used optimal values of the *freezing threshold* FThr and the *power threshold* PThr. The values of these thresholds, along with all other statistical data about the used part of the REMPARK dataset are given in Appendix A (Table A.1).

3.2.2 Comparison with Ground Truth

The classification algorithm output from the Moore-Bächlin detection algorithm was synchronized with the ground truth labels (GTL) and the video captured with a HD camera. We used an evaluation method based on events to assess the conformity between the GTL and the output from the classification algorithm. This specific event based evaluation method relates each *True* period in the GTL signal with a *True* period in the detection algorithm output signal. Figure 3.1 depicts our approach. It illustrates different cases for which we considered to have a *true positive* (TP) matching between the GTL and the algorithm output. Since the synchronous classification output is available every 1.25 s, it is impossible to achieve the perfect parallelism of the two signals. Thus, we allowed a delay to exist between the GTL and the algorithm output. We accepted to have the rising edge of the algorithm output up to half a second earlier than the start of GTL for the *true positive* case, such as shown in Figures 3.1a and 3.1c. The same held for the falling edge of the algorithm output, where half a second delay of the algorithm out-

put after the GTL was acceptable (Figures 3.1c and 3.1d). If the algorithm output had a rising and a falling edge (or several of them) inside the time when the GTL was *True*, this was also considered as a *true positive* (Figure 3.1d). When at the same time the GTL was *True* and the algorithm output *False*, the FOG was declared as a *false positive* (FP). Similarly, for a negative GTL and a positive algorithm output, a *false negative* (FN) was set.

3.2.3 Context of FOG Episodes

The episodes that were incorrectly classified by the Moore-Bächlin algorithm (FP and FN) were carefully inspected in the video. We noted down the following factors that describe the situation of the patient:

General situation
: The misdetected FOG episodes were described with a free vocabulary. We noted the posture, location and the detailed behaviour of patient's legs (e.g. whether one or both legs are trembling, which leg is used for pivoting during turns). Also, we recorded information about the intention of the patient and whether the patient was helped by a person or a walking instrument. The most characteristic traits of each episode were noted. Some examples of the produced descriptions are:

  - *Timed up and go / Turning 180° before sitting / Small steps while pivoting around the standing leg.*
  - *Stepping in place / Starting hesitation after getting up from a chair.*
  - *Sudden stop during straight walk using a walker / Light trembling in the left leg / Leaning forward on a walker.*
  - *Sitting in a chair with some dyskinesia movements.*

Duration
: The exact duration of the event in seconds. If a FP was detected, we took the duration of the episode on the output of Moore-Bächlin algorithm. For a FN, we took the undetected episode duration according to the GTL.

Location / Spatial relation
: This factor was specified as a combination of the coarse location in the home (e.g. kitchen, living room) and the relation between the patient and the closest contextual influence (e.g. after a doorway, in front of a chair). Some examples of the produced descriptions are:

  - *Bedroom / In the passage between a wardrobe and a wall / Oriented towards a doorway.*
  - *Living room / Passing* 20 cm *distance from a chair, but still there is a lot of space in front and right of the patient.*

- *Kitchen / Tight passage between the kitchen table and the fridge (* 0.5 m *width).*

Activity / Posture
: A description of the ongoing activity (e.g. walking, turning, starting walk) or the static posture occurring just before and during the misclassified event (e.g. standing, sitting). For example, if there was a starting hesitation FOG that occurred after getting up from a chair, the description such as "Sitting / Standing" was used.

Exit strategy
: A description of how the patient exited from a FOG situation in his home environment. We noted whether the patient was helped by someone and what tool, or part of the environment he used. The example situation descriptions are:

  - *Helped by other / Given a chair to sit.*
  - *Helped by other / Held by hand and given a foot to step over.*
  - *Alone / Hand on an armchair close to the patient.*

3.3 Results

3.3.1 False Positives

We found 158 instances of FP detections that had total duration of 782 seconds. We aimed to find the characteristic situations under which these FP detections (re)occurred. The previously collected descriptions of the general situations during which FP detections occurred, along with the descriptions of the related locations, activities and postures, provided us with a wealth of qualitative data. These qualitative data can be considered as a result of the initial coding process. In qualitative inquiry a *code* is defined by Saldana (2009) as:

> *...most often a word or short phrase that symbolically assigns a summative, salient, essence-capturing, and/or evocative attribute for a portion of language-based or visual data. Coding enables the organization and grouping of similarly coded data into categories or "families" because they share some characteristic – the beginning of a pattern.*

The results of the initial coding were taken as a data input for the categorization during the second coding cycle. Similar word formulations in our descriptions were reoccurring and our coding patterns started becoming recognizable. Clusters of similar situations that all can be described by the same category name emerged one by one. A set of 11 categories was ultimately defined, that allowed each of the 158 FP situations to be assigned to at least one of the categories. Table 3.1 displays the names of the categories and the number of the observed FP detections for each category. The complete table with the distribution of categories per each patient can be found in Appendix A (Table A.2).

Table 3.1: Categorization of false positive (FP) detections of Moore-Bächlin algorithm on D2FOG dataset.

Category	Total
Turning	48
Small steps	28
Start walking	9
Stop walking	7
Conditioned walking	3
Backward/Lateral steps	13
Normal walking	18
Standing with legs moving	7
Standing with upper body moving	12
Posture change	2
Sitting	11

The values in Table 3.1 show that the FP detection error most often happened during *turning* (48 instances), followed in frequency by situations in which patients produced *steps of small length* (28 instances). In the large majority of the observed cases of turning, patients were also making smaller steps. This observation is in accordance with the previously recorded impairment of the gait parameters (such as the step length and the gait time variability) in PD patients during turns (Willems et al., 2007). The majority of the FP detections involving turns (30 of 48) and small steps (20 of 28) were recorded for patients in their *OFF* state. The amount of time that a PD patient spends in the double support phase during gait cycle in the *OFF* state increases for 5-10% compared to the *ON* state, causing the patient to produce slower and shorter steps, and to have a reduced ground clearance of the feet (Morris et al., 2001a). A reduced ground clearance and a sporadic contact of a foot with the surface can cause the appearance of higher harmonics in the measured accelerometer signal. These higher harmonics influence the power of the calculated *freeze band*, potentially increasing the estimated value of the *freezing index* above the *freezing threshold* and generating a *false positive*. A similar reasoning about the cause of the *false positives* can be applied to some of the other categories that we extracted; especially to the categories of *walk initiation* and sudden *stop of walking*. Due to the manner in which the patients performed these actions (with more upper body movement or accentuated first/last step), and due to the presence of the change from a static to a dynamic accelerometer signal, the proportion of higher harmonics in the *freezing index* ratio could have easily become sufficient to exceed the *freezing threshold*.

Into the category *conditioned walk*, we classified the situations where false positive detections were caused due to the use of a walking aid. For instance, in one case a walker device got stuck under the carpet and the patient tried to get it unstuck by performing rhythmical up-and-down movements with the whole body. This is an unexpected behaviour that was hard to imagine

prior to seeing it in the home videos.

We did not expect to find a high number of FP detections in the category *normal walking*. Under the term *normal walking*, we considered the behaviour in which the patient produced steps of a seemingly normal length and frequency. However, with 18 instances, normal walking is the third category by incidence on our list. The category *backward/lateral steps* is on the fourth place of the list with 13 instances. Backward and lateral steps usually occur at a home, but they are rarely tested in a laboratory. During the REMPARK home data collection, these types of steps transpired when patients had to open a door, or when they had to sit down on a chair behind a table. The Moore-Bächlin algorithm resulted in *false positives* because the acceleration signal acquired from patients during non-forward stepping deviated considerably from the nominal conditions for which the algorithm was originally designed (i.e. forward walking).

A similar deviation of the accelerometer signal from the nominal conditions happens also during *posture changes*. The major observed types of posture changes in the videos were from standing to sitting, and vice-versa. Once the patient is in the sitting posture, *false positives* are also possible due to some rhythmic activity of hands or legs. Examples of such activities are cutting food while eating, or making (unconscious) rhythmical leg movements under the table. For this reason, *sitting* as a static posture was given its own category, independently from the dynamic posture changes.

Another category that involves static posture is *standing*. We made a distinction between *standing with legs moving*, and *standing with upper body moving* categories. The description "standing at the spot while there is movement of the legs" sounds quite similar to the usual description of FOG. However, the leg movements during *standing with legs moving* were of a different type than in any of the usual FOG manifestations. They mainly consisted of balance adjustments from one leg to another, during which a minimal feet displacement was present. In the category *standing with upper body moving* are gathered the *false positives* which originate from the "FOG false positives test" (see Section 3.1).

3.3.2 False Negatives

We found 127 instances of false negative detections that had the total duration of 739 seconds. We described the situations in which these false negatives occurred, similarly to how was done for the false positive detections. In this case, it was not necessary to define new categories for characterization of FOG episodes. Instead, we used the three manifestation and the five hesitation types introduced by Schaafsma et al. (2003), that were already described in detail in Section 1.1.2. The aggregated results of this categorisation are presented in Table 3.2, while detailed data for each patient is given in Table A.2 in Appendix A.

We can notice in Table 3.2 that the incidence of FOG episodes in terms of manifestation types has an almost even distribution, with *shuffling*, *trembling* and *akinetic* FOG episodes, all having around 40 recorded instances. The episodes involving *shuffling* and *trembling* FOG that were not correctly captured, were the ones that lasted too short, or had an insufficient

Table 3.2: Categorization of false negative (FN) detections of Moore-Bächlin algorithm on D2FOG dataset.

	Category	Total
Manifestation	Shuffling	45
	Trembling	44
	Akinetic	38
Freezing Type	Starting	34
	Turning	38
	Narrow space	27
	Destination	3
	Open space	25

level of leg movement. It has to be noted that due to its dependency on dynamic signals, the Moore-Bächlin algorithm had no capability to capture an *akinetic* FOG. Consequently, the set of 38 false negative detections of the akinetic FOG manifestation includes every akinetic FOG episode that was observed during the experiments. Taking into account the number of correctly detected FOG episodes (221 instances) and the non-akinetic FN detections (89 instances), we come to the conclusion that the *akinetic* FOG episodes in the dataset were represented in around 8% of the cases. This percentage gives us an orientation about the possible improvement in the FOG detection sensitivity, if we are successful in finding a way to detect an *akinetic* FOG. The results of the categorization according to the freezing type reflect the nature of the tests performed during the data collection. Each patient repeated *Up and Go* test several times, with the goal of provoking a FOG. Therefore, it is not surprising that the main components of the test, such as starting and turning, caused the highest number of the detected episodes. The activity of approaching a chair at the end of an *Up and Go* test has a potential to cause a destination hesitation. A chair approach was often preceded by a 180° turn prior to sitting, which seemed to be a more direct FOG trigger.

Since the available home data allowed us to give a special attention to the environment of the patient, in addition to the manifestation and the freezing type, we observed the locations and objects in the patients' homes that seemed to have influence on the FOG episodes. We will refer to the locations in question as FOG *micro-locations*, contrary to *macro-locations* that is the term that we use for the environment at the level of a room or bigger. We analysed the FOG-related micro-locations for all the observed instances of FOG in the dataset, which includes both the true positive and the false negative detections.

Table 3.3 presents a distribution of FOG episodes per micro-location for the whole dataset. The *Total* percentage in the last row of the table was calculated as the ratio of the FOG episodes at the specific location type (TP+FN) over the total number of the observed FOG episodes (348 instances). A detailed distribution of the micro-locations per patient (on which Table 3.3 is based) is given in Appendix A (Table A.4 for the *true positives* and Table A.5 for the *false negatives*). Out of all the locations, doorways had the most obvious influence (26%). This

Table 3.3: Distribution of FOG per micro location on D2FOG dataset.

Condition	Open space	Narrow space	Doorway	Chair	Bed	Sofa / Couch	Table	Lift door	Other
TP	81	19	56	41	1	5	4	11	3
FN	44	18	33	16	1	1	8	0	5
TP + FN	125	37	89	57	2	6	12	11	8
Total [%]	36	11	26	16	1	2	3	3	2

influence is well known, and has even been assessed in a specific laboratory study (Cowie et al., 2012). Since the patients usually started each new test from the sitting posture, and due to a high number of *Up and Go* tests, "in front of a chair" and "near a chair" were the second most used spatial relations (16%). Chairs were triggers for both starting and destination hesitations. Since patients can also sit at sofas and beds, it is safe to presume that these furniture have the same kind of influence on triggering FOG, as chairs.

The influence of the environment is the most evident in narrow spaces (11%). We noted the configurations of the environment elements that created such critical situations (Appendix A, Table A.3). Under the term *Open space* (36%) we classified all the episodes that we could not bring into any relation with the surrounding environment. The category *Other* includes locations that were mentioned rarely, such as a kitchen sink or a bottom of the stairway.

A secondary analysis of the places where a FOG happened was done on the macroscopic level. Appendix A (Table A.6 for TP and Table A.7 for FN) offers detailed analyses per patient, while their summary is presented in Table 3.4, in the same way as it was for micro-locations.

Table 3.4: Distribution of FOG per macro location on D2FOG dataset.

Condition	Living room	Kitchen	Hall	Bedroom	Bathroom	Other indoor	Terrace	Building	Outside
TP	104	16	39	7	0	5	14	25	11
FN	59	13	26	3	5	0	5	11	5
TP+FN	163	29	65	10	5	5	19	36	16
Total	47	8	19	3	1	1	5	10	5

In the macro-location analysis we formed the main location classes according to the main rooms that make a living space (living room, kitchen, hall, bedroom). The category *Other indoor* includes spaces such as a dedicated office, a dining room or a stairway. Into the category *Building*, we classified FOG episodes that happened inside the building, but outside of the apartment, such as getting stuck in front of the lift door or on the building stairways.

3.4 Summary

Schaafsma et al. (2003) offered the first objective and systematic analysis of FOG by the means of video recordings. In this chapter, we tried to repeat that process, not just by analysing FOG episodes, but by additionally analysing the situations in which FOG was falsely detected by

the state-of-the-art detection algorithm. After the detailed observation of home videos of 17 patients, we introduced 11 categories of situations in which we found the Moore-Bächlin algorithm to give false positives. We also observed all the false negative detections and marked the locations and the interesting obstacle configurations in home environments that induced FOG episodes. The behavioural observation process executed in this chapter was important, since the knowledge gathered here directed the choice regarding what elements of the contextual information should be recognized and used in the new context-aware monitoring system.

4

Context-aware Distributed Home Monitoring System

The main objective of our research is to discover how spatial context can effectively be used to improve automatic detection of FOG in the home of a patient. The fulfilment of this objective requires design and development of a technical system that is able to: *a*) collect sensory data about the patient and his/her indoor environment; and *b*) match the perceived sensory stimulus with the FOG-related context.

The first important step towards the realization of such a system is to fully understand the requirements and to obtain a clear idea about the qualities that the future monitoring system should contain. When specifying the requirements, it is necessary to bear in mind the desired functionality of the system, but also to think about other important factors, such as the system's environment, the user-centric and the economical aspects. In the system design phase we explore how to optimally fulfil the given requirements. We start with a preliminary design of a system concept, then we choose the appropriate technologies (hardware and software) and in the end we present a detailed system design. We define the architecture in a form of system modules with strictly assigned functionalities, inputs, outputs and execution domains.

In this chapter, we present the requirements analysis and the design process for the new context-aware monitoring system. First, based on the conclusions of the research on the state-of-the-art and the observations of the home patient videos, we investigate how different context

Parts of this chapter appear in Takač et al. (2012a), Takač et al. (2012b) and Takač et al. (2013)

data types (e.g. location, orientation, activity) could be used to spot a FOG episode. The chosen types of context data define the kind of the perception system that has to be built. Knowing the potential context data types, we explore the related work on existing sensor types and systems which are able to capture them. Having the idea about the basic functionality that will be requested from the system, we form the functional and user requirements. Towards the end of the chapter, we present the system concept and our choice of the technologies for the system implementation. We finish the chapter with the overview of the general system architecture.

4.1 Context Information, Data and Required Functionality

Initial definitions of *context* were based on the enumeration of different types of information. Schilit and Theimer (1994) thought of *context* as location, identities of nearby people, identities of nearby objects and changes to those objects. Brown et al. (1997) expanded this definition of *context* by addition of new contextual aspects such as time of the day, season and temperature. Dey (1998) went even further, and enumerated *context* as the user's emotional state, focus of attention, location and orientation, date and time, objects, and people in the user's environment. Abowd et al. (1999) considered that these definitions were all too specific and that it is not possible to enumerate all important aspects of all situations in advance, since these aspects change from application to application. Finally they gave a widely accepted definition of context as:

> *...any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*

This definition leaves it to the application developer to enumerate the context for his particular application scenario. Under this definition any piece of information that is significant enough to describe some situation of interest can be considered as *context*. In the home video analysis in the previous chapter we selected the context information types that are interesting for our FOG-related context application. These types are *location* and *orientation* for minimization of false negative detections, and *activity* for minimization of false positive detections. In the following sections, we analyse the possible ways for using each of the specified context information types in the FOG monitoring system.

4.1.1 Location

In their home, PD patients are likely to encounter narrow passages such as doorways or dynamically changing spaces created by the presence of other people and movable objects such as chairs. When PD patients perceive the space as too narrow for the dimensions of their body, adaptive postural changes during locomotion may be needed to achieve a collision-free passage (Higuchi et al., 2006). Laboratory experiments with the PD patients showed that there might be a direct correlation between the width of the narrow space and the tendency for a FOG

episode (Almeida and Lebold, 2010). In an extensive study, Cowie et al. (2012) successfully provoked freeze-like events near a doorway and concluded that the prevalence of such events significantly increased with the narrowness of the doorway. Furthermore, they showed that a decreasing door width caused progression velocity to drop for approximately 20% in the region preceding the doorway, or immediately after it.

Our observations of the PD patients' home video-recordings (presented in Section 3.3.2) are in line with the findings above, since we witnessed numerous FOG episodes that happened in doorways (26%) and other narrow spaces (11%). The design of a new perceptive-cognitive system starts with an established representation model for humans. The representation model should be informative enough to allow the description of the targeted human behaviour, and simple enough to be easily computable. In the case of FOG detection, we are primarily interested in the locomotion behaviour of the patient. Already a two-dimensional (2D) point (x, y) can be used as a sufficient representation to track humans, depict human walking trajectories, detect behaviour anomalies, and to offer an effective navigational assistance (Fajen et al., 2003; Tastan and Sukthankar, 2011).

We developed the concept for a location-aware approach which uses 2D positional data with location semantics. For reasoning about the FOG, the future system could use probabilistic inference (Steinhauer et al., 2012; Hong and Nugent, 2013) in which evidence components of the patient's gait and location are fused together. If the patient is inside a marked FOG zone, the location dependent evidence of FOG would be modified accordingly to the type of the zone (*location semantics*) and the geometrical distances towards the objects dominating the specific zone. For example, for the object *door*, the probability of a FOG episode could be modelled to have the highest value approximatively 0.3 m before the middle of the doorway.

Another possible approach can be the *historical location-based* reasoning. This approach is related with the local aspect of the FOG explained in Section 1.1.2, which claims that some specific places in the patient's environment can often cause an episode. The observation of the historical positional data over a long-term period can provide the knowledge about the exact places in the home where this claim is actually true. To expand the *location semantics* approach that uses geometrical and gait evidence components, a probability value related to the history of FOG on the location could be taken as an additional evidence component.

The system that can provide a 2D position of a patient is a necessary requirement for any of the proposed location based approaches. In the *semantic* and the *historical* location approach, we can recognize the accuracy of location as a very important factor. The use of micro-locations requires the capability to measure in centimetres the distances between the patient and the obstacles around him/her. Such requirement makes it difficult to find an already existing system or technology for localization that could be directly applied.

4.1.2 Orientation

Many PD patients are experiencing FOG during turns. Snijders et al. (2008) note that wider turns are easier to perform than axial turns on the spot, and slow turns are easier to perform

than rapid turns. A rapid turn on the spot is hard to track using only a 2D point as a representation of the tracking target. The observation of on-the-spot turns can be achieved only by precisely estimating the angular velocity of the person, which requires an additional tracking of the heading angle.

Furthermore, there was an additional realization about the possible role of the orientation in the FOG detection by using the location-based principles explained above. Human locomotion through an obstacle environment is influenced by visual input and visual field limitations (Jansen et al., 2011). For instance, the influence of a FOG zone on a PD patient is not exclusively limited to the space next to the obstacle, but instead its influence starts a few meters prior, in the moment when the obstacle comes into the focus of the patient's visual field and causes a switch in attention (Riess, 1998). Hence, to assign an influence to a FOG zone, it is necessary that the obstacle takes the main part of the visual field. In some situations the patient can be located inside the FOG zone, but also visually oriented away from the obstacle. Let us take, once again, as an example, the situation when the patient passes through a door, that is in this case represented as a symmetric FOG zone. After the patient passes the middle of the doorway, he is still in the door FOG zone, but now he faces an open space. If the reasoning about the influence of the door is based solely on the position, then due to symmetry, the same FOG probability would be given at the same distance from the middle of the doorway, regardless of whether the person is in front or behind the doorway. On the other hand, if the information about the orientation of the patient is added into reasoning, it would be possible to assign more meaningful probabilities.

For the reasons above, we consider that the instant orientation of the patient, measured by a continuous tracking of the patient's heading angle (ϑ) on a 2D floor plane, is an important contextual information type for our context-aware application.

4.1.3 Activity

Before starting a deeper discussion about *activity* as a contextual information type, we need to develop the appropriate taxonomy of human behaviours and to understand the relation between *action*, *posture* and *activity*.

The terms *action* and *activity* can sometimes be used interchangeably, and sometimes they can be used to distinguish between behaviours of different complexity and time duration. We connect the term *action* with shorter lasting behaviours, that usually describe some simple movements or ambulatory behaviour (Moeslund et al., 2006). Appropriate examples for *action* are the movements such as making a step, making a turn, putting a glass on a table or picking up something from the floor. Compared to *action*, *activity* is considered to be a more complex behaviour that typically lasts longer, and consists out of a sequence of actions. It is hard to find a clear boundary between the two terms and the distinction is pretty much by instinct, but if the relation of sets is applied on the two concepts, it can be said that *action* is a subset of *activity*.

In a more exact interpretation, postures such as standing, sitting and lying are called *static postures*. Between these main three static postures, there are *dynamic postures*, or more precisely *dynamic posture transitions* (i.e. sit-to-stand, stand-to-sit, sit-to-lie). The definition of *action* includes basic movements of a relatively short duration. Thus, the postural transitions can also be classified as actions. On the other hand, *static postures* usually are executed over a prolonged time duration, i.e. person can be sitting or standing on the place for several minutes or tenths of minutes. Therefore, *static posture* can be considered to be the same as *activity*. This equality of *static posture* and *activity* is often visible in the literature. There are many examples where postures are considered as a class in the standard activity recognition (Frank et al., 2010). In the rest of the thesis we will use the simple categorization that we have just introduced. The term *activity* will be used to refer to simple actions, dynamic postures, static postures and complex activities. The specific terms for *action*, *dynamic posture* and *static posture* will be used if an additional distinction is necessary.

In Table 3.1 in Chapter 3, we presented 11 categories of activity related situations. The recognition of these categories would help into minimizing FP detections. The majority of these categories are actually *actions* with a relatively short duration, lasting from under 1 second to few seconds. This insight suggests that the one of the requirements of the future system will be to recognize the actions that have a very short duration (< 1 second).

4.2 Related Work

In this section we give a short overview of some of the existing solutions and the general approaches for human localization, orientation tracking and activity recognition. We are mainly focused on the advantages and shortcoming of the most often used sensor types for each of these tasks.

4.2.1 Localization Systems

The origins of today's context-aware computing can be found in the location-aware computing for mobile applications initiated in the 90-ies of the last century (Hull et al., 1997; Abowd et al., 1998). An inquiry about localization systems showed that many different technologies are used for this task. Radio frequency, sonic waves, inertial systems and photonic energy, have been used to solve the problem of a precise indoor localization. Each of these systems has its own set of limitations (Torres-Solis et al., 2010). Examples of hardware systems based on Radio Frequency (RF) technology include WLAN (Xiang and al, 2004; Yim et al., 2008), Ultra-Wideband (Gentile et al., 2008) and RFID (Ni et al., 2003; Tesoriero et al., 2008). Usually, by using RF technology it is possible to achieve the positioning accuracy of a few meters, but the main limitation is the impact that a physical environment can have on the quality of a measurement (e.g. radio-interference, EM noise). The environmental factors like ambient noise, echoes, air temperature and co-interference, also pose limitations for the technology based on

sonic waves, such as ultrasound sensors and microphones (Kleine-Cosack et al., 2010). In contrast, the limitations of inertial sensors are internal and not external. Inertial sensors have internal drift associated with thermal changes, which can add up to a great positional error due to a double integration of acceleration necessary for the calculation of displacement. Inertial sensors yield only relative positional information, and therefore should be combined with some absolute reference to provide an absolute location estimate (Evennou and Marx, 2006; Retscher, 2007).

Use of cameras and color image processing is the most popular technique for people tracking and indoor localization. Spatial coverage of such systems can range from one part of a room, when using a single ceiling-mounted camera (McKenna and Charif, 2004) or an over-head camera (Ribeiro and Santos-victor, 2005), to the whole room coverage in the case of overlapping multiple-cameras (Petrushin et al., 2006). Camera based solutions provide a sub-meter localization accuracy, but this is paid with a high cost in terms of computational efforts. Further limitation of RGB camera systems is the sensitivity to time-varying light conditions, shadows and occlusions. Regardless of the above disadvantages, cameras still seem like the best solution for our indoor localization problem.

4.2.2 Activity Recognition

An accurate activity recognition depends on the quality of information that the sensors in the system can collect. Hence, the activity recognition is directly dependent on the type (and number) of used sensors. One of the main divisions according to these principles is between *vision-based* and *sensor-based* activity recognition (Chen et al., 2012).

A *vision-based* activity recognition uses video cameras to observe a user's behaviour and changes in the environment. The cameras generate fast sequence of 2D images that are processed with different computer vision techniques such as feature extraction, movement segmentation, structural modelling and pattern recognition. A general processing pipeline for a *vision-based* activity recognition consist of the following steps: 1) human model initialization, 2) person tracking, 3) extraction of low level features 4) inference of the current activity using the extracted features with the previously obtained referent activity model. The field of *vision-based* activity recognition has been in the intense research focus for a long-time, and there are already plenty of existing solutions for each stage of the pipeline.

There are several surveys that give a nice overview of the state-of-the-art in the *vision-based* activity recognition domain. Cédras and Shah (1995) provide a review of the computer vision based motion recognition (i.e. walking, skipping, running), and focus on the two most important steps in motion recognition: a) motion information extraction and motion information models building; and b) matching unknown inputs with the constructed model. Aggarwal and Cai (1999) discuss body structure analysis, tracking and recognition. Gavrila (1999) is interested in the recognition of whole-body motion and hand motion, and is focused on the various methodologies for human model representation (stick-figure based, volumetric, statistical). Poppe (2010) reviews the techniques for human action recognition that focus only on the full

body movement, and excludes the work on gesture recognition. He takes into account only the activities that are not depending on the context. Moeslund et al. (2006) offer an extensive survey of over 300 papers between years 2000 and 2006. The survey focuses on all the important stages in the *vision-based* activity recognition, from model initialization, over tracking and pose estimation to high-level behaviour recognition. Yilmaz et al. (2006) present a comprehensive survey of the efforts in the past couple of decades to address the problems of representation, recognition, and learning of human activities from video. In their latest effort Aggarwal and Ryoo (2011) provide a detailed overview of various state-of-the-art research papers on human activity recognition. The authors discuss both the methodologies developed for simple human actions and those for high-level activities.

Sensor-based activity recognition involves the use of a wide range of different types of non-vision sensors such as accelerometers, RFID (Fishkin et al., 2005; Patterson et al., 2005), motion sensors (Wilson and Atkeson, 2005; Wren and Tapia, 2006), pressure sensors (Orr and Abowd, 2000), microphones (Chen et al., 2005), etc. These sensors can sense an activity either by being deployed on the person which leads to *wearable sensing*, or by being deployed in the environment which is called *dense sensing* (Tapia et al., 2004; Chen et al., 2012). The most often used sensor for wearable activity recognition is the accelerometer (Bao and Intille, 2004a; Maurer et al., 2006; Yang et al., 2007; Cho et al., 2008). The use of accelerometers for activity recognition in terms of the number of sensors, position on human body and employed machine learning algorithms is very similar to how they are used in the FOG detection, which was previously described in Section 2.1. Accelerometers are usually used with supervised learning methods and are able to classify with high accuracy between simple activities like walking, running, sitting, standing, and climbing stairs (Anguita et al., 2012). Wearable sensor-based systems have no data association problem and also have less data to process, compared with the vision-based systems.

The obvious problem of the vision-based activity recognition under real-world circumstances is the dependency on an elaborate human model that requires a lot of visual data and permanently good subject visibility in front of the camera. Compared to vision-based systems, wearable sensor-based systems process less data, but they also need several wearable sensors placed at potentially obtrusive body locations when the system needs to recognize certain complicated activities. However, there is also a third, hybrid approach to activity recognition based on the mix of vision and wearable inertial sensor data (Zhu and Sheng, 2011). The advantages of this approach are the need for a minimum number of wearable sensors worn by the user which reduces encumbrance, the use of a simpler human model which requires less visual data and lowers the visibility demands, and the opportunity to maintain the classification accuracy by using independent data modalities. The hybrid approach has already been used for activity recognition and in the healthcare domain (Pansiot et al., 2007; ElHelw et al., 2009; ElSayed et al., 2010).

4.3 Requirements

Based on the analysis of existing technologies for different types of context (Section 4.2), we concluded that it is more preferable to use video cameras for a precise indoor localization, while hybrid systems based on vision and wearable sensors are well suited for a recognition of activities. A hybrid system, due to the placement of its physical components, can also be referred to as a *distributed sensor system*.

The design of a distributed sensor system for healthcare is a collaborative and multidisciplinary process that involves engineers, medical personnel (geriatric specialists and clinical rehabilitation specialists) and the end-users (PD patients in our case). Two parallel imperatives should be met with such system - the solution should meet the clinical (or research) requirements and it should be appropriate and acceptable for the end users (Dishongh and McGrath, 2009).

4.3.1 Research Requirements

We, the researchers, aim to fill the knowledge gap about the relation of FOG and spatial context. To fulfil this goal, we need to develop an instrument that provides a solid technical framework for collecting, storing and analysing the set of sensor signals from which the necessary contextual information can be extracted. The main functional requirements for the new context-aware system, were defined in the analysis of the context information types that were presented in Section 4.1. We need a system that can:

- Track human position in real-time with a sufficient accuracy (acceptable error < 10 cm);
- Track human orientation in in real-time with a sufficient angle (allowed error < 20°);
- Recognize a specific set of patient's activities; and
- Be used in a multiple people environment.

For a FOG detection based on location, it is of a great importance to achieve a sufficient accuracy when measuring the distance between the patient and an obstacle. When the system needs to observe that the patient is passing through a door frame, a necessary accuracy of location sensing is in the range of several decimetres. The same is true when the patient is standing next to an object, such as chair. Proximity to an object in a congested space can easily be inferred when the person is standing at a very short distance (< 0.4 m to 0.5 m). To set the criteria for a sufficient accuracy, we can use the literature about the minimal distance from objects that was observed for people during locomotion behaviour. According to Weidmann (1993), a person walking in a corridor keeps on average a minimal distance of 0.25 m to a wall made of concrete and 0.20 m to a wall made of metal. Obstacles in a general environment need to be avoided with a gap of at least 0.10 m.

The heading of the patient should be observed with the goal of inferring if he is facing any specific landmark on the map. When observing the patient's relation with the landmark, such as having the intention of going through a door or facing a kitchen sink, the heading error of 15-20 degrees left or right from the true angle is acceptable, because such an error cannot change the perception about the patient being generally directed towards the object.

A context-aware system usually contains sensors that have to be installed in the environment. Thus, it is a custom to build a test-bed for the prototyping stage, or to develop a piece of software that will simulate context information inputs (Oh et al., 2007). This approach holds for the majority of context-aware applications that are targeting "normal" users and non-medical context. In the development of the system that targets the context of FOG that is not the case. and it is necessary to do more than build just one prototyping test bed. There is a variety in the symptoms between different patients. If we have a test-bed permanently installed at one location, we will not be able to capture the influence of the natural environment on the FOG episodes of the patient. This importance of being able to obtain the sensor data from the patient's natural environment, was already explained earlier in Section 2.2. These reasons impose an additional research requirement - a *physical portability* of the system prototype. A portable system prototype would allow us to go into the homes of PD patients, do a fast system setup and collect data about their natural behaviour. A fast system setup requires a *rapid mounting* of the vision sensors in an environment and a *fast setup* of the software application for localization. We need to localize people in relation to objects, landmarks and zones in an indoor space. This imposes a requirement to have an appropriate tool for *making maps* of the observed environment.

Two of the main functional requirements demand the system to operate in real-time. By the definition of IEEE (1990), *real-time* pertains to:

> *...a system or a mode of operation in which computation is performed during the actual time that an external process occurs, in order that the computation results can be used to respond in a timely manner to the external process.*

We aim to enable iterative development of real-time algorithms during the prototyping stage. Therefore, we need to *record* raw data from all the sensors, and also to be able to *replay* the recorded data in real-time. We expect to have sensors of different modalities, which means that each of those sensors might acquire data using its own data acquisition frequency. For example, cameras usually operate with 30 Hz, while accelerometers can operate with anything between 20-200 Hz. To enable real-time operation of the system, a mechanism for *synchronization* of sensor streams with different data frequencies has to be provided.

The last research requirement is related with the transition of the system from the prototype stage into the permanent monitoring solution stage. This transition can happen when the necessary (context) algorithms have been implemented and evaluated, and it is easier to perform if we take care to *limit the costs* and to handle the *system complexity* during the system design stage. The costs can be limited by using affordable sensors, and by upholding the principle of *scalability*. In terms of a system that is focused on localization, scalability implies that the costs

would increase linearly with the indoor area monitored. *System complexity* can be handled by the appropriate software design that supports decentralized computing. A distributed system should calculate context information on sensor locations and only communicate message packets with smaller data size between its nodes. With *decentralized* sensor data processing, we avoid the possibility of having a system "bottleneck". In a real-time sensor system, a bottleneck can appear if there is a single processing unit that has to take care of all the low level sensor data, and its processing capabilities become insufficient to uphold the real-time operation requirement as the number of sensors in the system increases.

4.3.2 User Requirements

Elderly individuals are often very sensitive to small changes in environment (Steele et al., 2009; Beringer et al., 2011). Home monitoring systems require sensors on the users or in their environment and this "sacrifice" is accepted by the users only if the system is effective and they perceive that its usefulness clearly improves their quality of life (Rahimpour et al., 2008). Hence, the main functional requirement concerning the user of our system is the ability to detect FOG with the maximum accuracy/robustness and minimum latency. This requirement has been the main motivation for the research described in this thesis.

Sensor acceptance by the patient is pivotal for the success. Wearable and ambient sensors have different requirements in relation to the user acceptance. The main issue with using wearable sensors is adherence and getting the user to wear necessary sensors each day (Steele et al., 2009). Minimizing the number of the sensors that will be worn on the body, and optimizing their position so that they minimally influence the user's freedom of movement, is definitely a step in the right direction. The other step is achieving a sufficient power efficiency and working autonomy, so that a wearable sensor can last at least a day on one charge. An important factor is also the *ease of use* of wearable sensors. The *ease of use* involves simple interactions for powering on/off and recharging the sensors, a simple way to fix them on the body and avoidance of an additional sensor management training. Lastly, the physical form of the wearable sensors and their placement should avoid to offer a space for any social stigma (Parette and Scherer, 2004).

Ambient sensors should ideally be invisible or have a familiar form. If it is too difficult to embed the sensors in the environment, they should be installed in the home in such a way that they cause the least amount of infrastructural work (e.g. drilling, cabling). In a great majority of cases, ambient sensors have to be retrofitted, so the advantage should go to wireless sensor solutions that can avoid the need for additional wiring.

One of the main concerns related to ambient sensors, especially cameras, is privacy and protection of sensitive information (Friedewald et al., 2007). Loss of privacy and constant video monitoring is often seen by older adults as obtrusive and a violation of privacy in one's own home (Demiris et al., 2004, 2009). The participant study by Coughlin et al. (2007) showed that the constant video monitoring is perceived as acceptable only in the case when the individuals are extremely frail, or when the only other alternative may be nursing care or living with an adult child. Participant studies also show that "anonymizing" captured images, by introducing

shadows, silhouettes and other distorting features makes the cameras more acceptable (Demiris et al., 2004, 2009). An example of such approach are some of the fall detection systems based on video with enhanced privacy (Edgcomb and Vahid, 2012; Zhang et al., 2012). Another way to to ensure the privacy is that the raw data is not recorded or transmitted outside of the home.

4.4 System Concept

In our concept, a wearable assistive system is used to monitor gait and to treat the patient's FOG via a cueing device at any time or place (both indoor and outdoor) during the day. The sensing capacity and the detection capabilities of the wearable assistive system are expanded with the contextual information that is produced by a network of vision sensors installed in the patient's home environment. Vision sensors are placed in the areas of the home where the patient spends the most of his/her time every day, such as living room, kitchen, and hall.

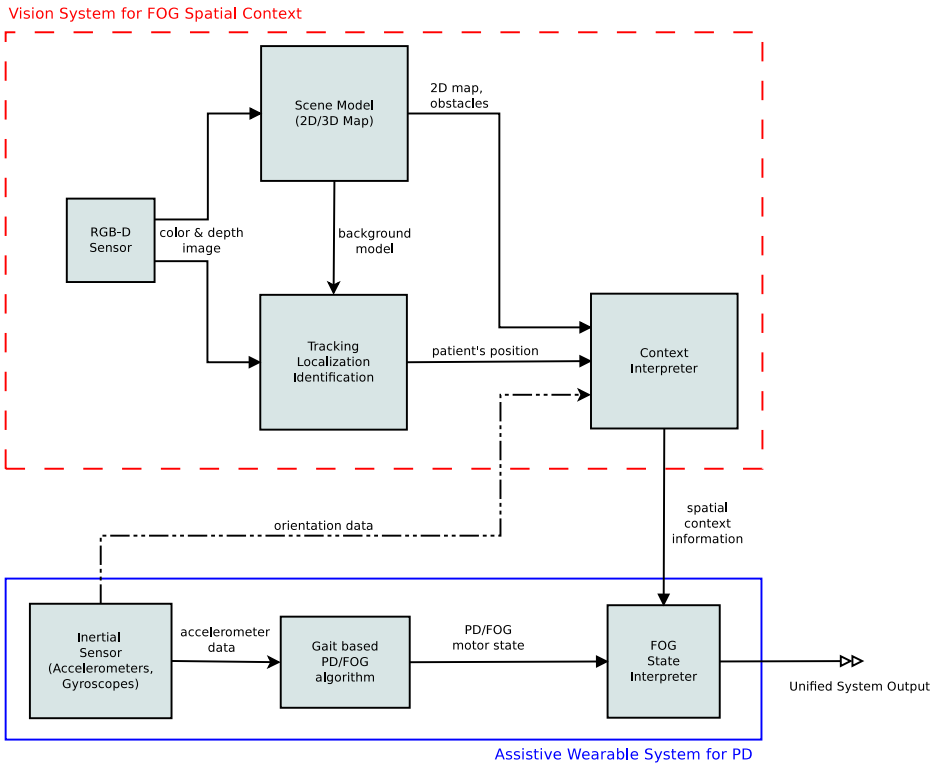


Figure 4.1: Block diagram for the concept of the distributed monitoring system. The wearable system independently detects FOG based on inertial data (blue rectangle). Gait-based detection is complemented by the user's spatial context from the vision sensor system (red rectangle) in the areas of the home where such system is present.

The distributed vision system provides patient localization, environment mapping and context inference. The concept and the main components of the monitoring system are presented in the block diagram in Figure 4.1.

A wearable assistive system (Figure 4.2) uses one inertial sensor device that is worn on the waist. This device collects the egocentric data about the patient's gait. Instead of developing a dedicated hardware, we propose to use a smartphone as a relatively affordable alternative that has good sensing capabilities and high computational power. Sensing capabilities of a smartphone arise from a set of cheap and powerful embedded sensors: accelerometer, digital compass, gyroscope, GPS, microphone, and camera (Lane et al., 2010). Besides as a sensor platform, a smartphone can also be used as a communication hub. It is able to connect to a Body Area Network (BAN) sensors via Bluetooth, or to communicate with a home ambient system via Wi-Fi. Concerning the user requirements, smartphone can potentially bridge the problem of technology acceptance. It is a familiar device that has already penetrated into peoples lives. Economical and social merits of its use should also not be forgotten. Using smartphone produced in big series is cheaper than developing dedicated hardware, and they are already socially accepted devices that will not draw attention and bring social stigma.

The concept of cueing for the prevention or termination of the FOG state has been previously presented in the introductory chapter in Section 1.1.2. A wearable assistive system needs to have a cueing device that will use the cueing modality that is optimal for the patient. Similarly to the concepts for cueing in the REMPARK (Cabestany et al., 2013) project, we predict a possibility to use either earphone for audio modality cueing, or a functional electro stimulation

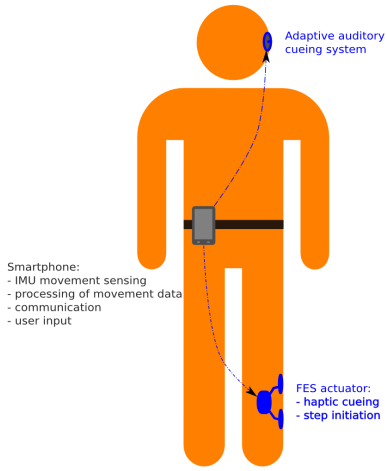


Figure 4.2: Wearable assistive system for FOG. Smartphone is used for sensing and communication. In the case of FOG detection cueing can be executed by using audio or haptic modality. Only one cueing devices is worn, depending on the patient's preference.

device worn on a leg for the haptic modality cueing. The effectiveness of a cueing modality and the exact cueing methods are not in the focus of this thesis. The only requirement for the use of the specific cueing device is that it needs to have the communication capabilities for inclusion into BAN, such as a Bluetooth connection support.

As vision sensors in our system, we propose to use cameras which have the capability to sense both the color and the depth (RGB-D). These cameras, also known as active 3D range cameras are able to overcome the illumination caused drawbacks of color vision systems, which is very favourable for improving the background subtraction. Furthermore, one depth sensor is enough to retrieve the precise spatial information about the 3D environment, compared to multiple color cameras required for the same task. The spatial 3D information can be used for an environment mapping, while reduced number of cameras in the system minimizes the complexity and simplifies the installation. A 3D sensor-based in-home monitoring was already considered as a wide-range solution suitable in several assisted living scenarios (Leone et al., 2013).

The workflow diagram of the system is given in Figure 4.3. The diagram shows how one RGB-D camera is paired with a wearable sensor in order to achieve improved FOG detection, and consequently improved cueing actuation. This process can be executed for each RGB-D camera. Independent elements of the process include a 2D position tracking and a 2D scene map calculation using RGB-D image, a 3D orientation calculation using inertial data from the wearable sensor, and a gait-based detection of FOG from inertial signals. These elements have to work independently, so that the FOG detection can be achieved by the wearable sensor, even when the patient is not in front of the camera. The main prerequisite for position tracking is the background subtraction in each frame. The background subtraction is heavily based on depth image processing. The background model for subtraction is set by periodic updates of the 3D point cloud of the whole observed scene. These periodic updates are done every few minutes on occasions when no tracked objects are present in the field of view. Furthermore, this background model is used to build the 2D map of the scene, which is used as one of the inputs for spatial context inference.

The foreground image obtained after background subtraction is used to build point clouds for updating the positions of the persons being tracked, and to detect any new persons in front of the camera. After the detection of new persons, positions of all tracked persons are updated. We are only interested in the position of the patient. If the track of the patient is not identified, the process of matching all known track histories against inertial sensor data is executed. If the match is successful and the patient's track is known, the position of the matched track is used in the calculation of the patient's pose. If none of the tracks in front of the camera are identified as the patient, the camera data is excluded from the FOG detection. The calculation of 2D pose involves a combination of the position obtained from the vision tracker and the 2D heading obtained from the wearable sensor. The estimated 2D pose is combined with the 2D map information and the history of FOG detections to infer contextual probability of a FOG episode. This probability is published over a wireless network and read by the FOG State

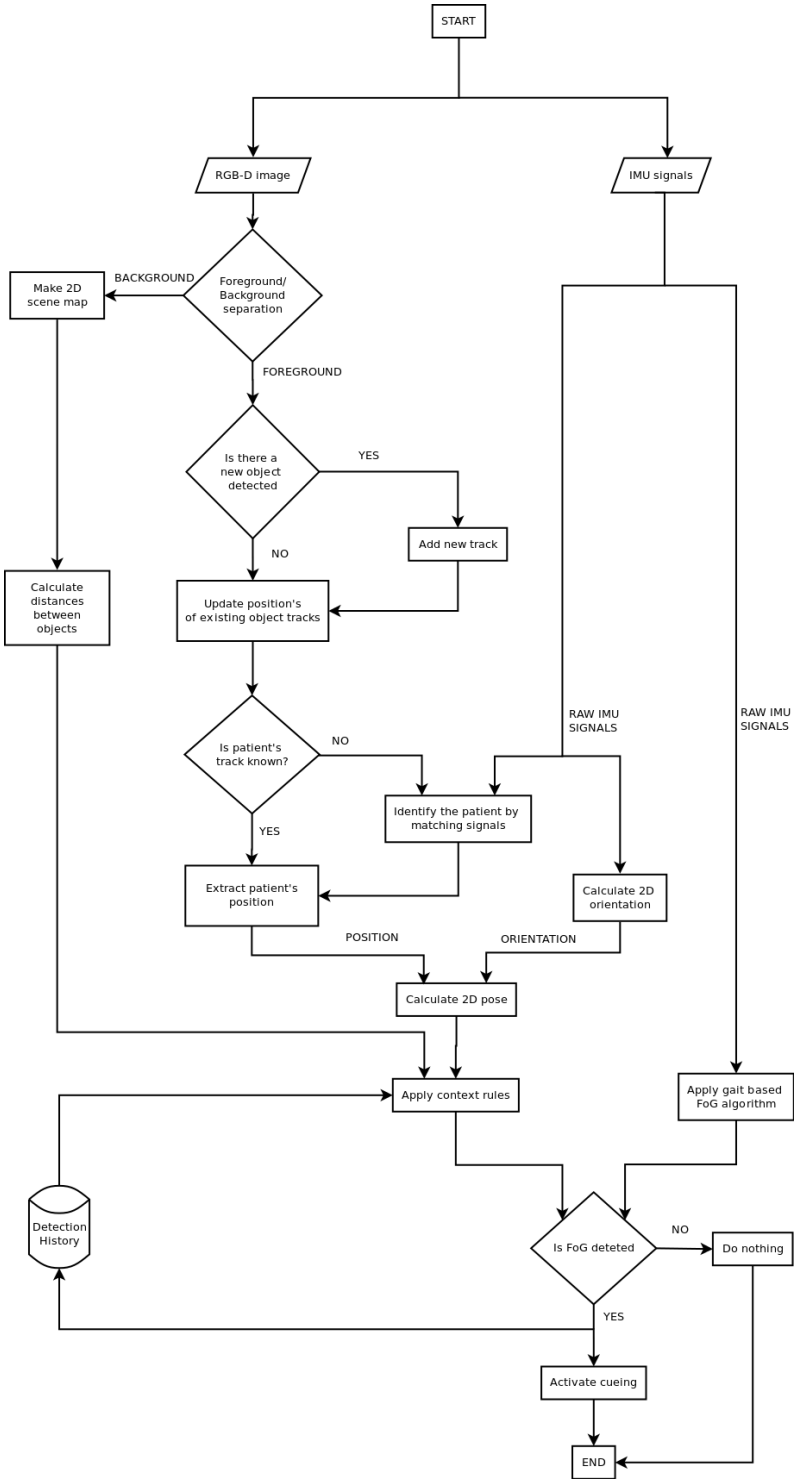


Figure 4.3: Workflow diagram for FOG detection using the distributed sensor system.

Interpreter (FSI) module that runs on a smartphone device. The FSI module conducts a high level fusion of the (spatial) context information and the gait classifier output to produce the final system output. A Boolean *True* or *False* at the output of FSI is what activates the cueing device.

4.5 Technology Selection

After completing exploratory steps and establishing the functional requirements of the system, our next step was to choose the most appropriate hardware and software that will enable the implementation of the system. During the requirement analysis and the preliminary design, we already chose a smartphone and RGB-D cameras as the appropriate types of sensors. Both smartphones and RGB-D cameras are commercially available devices that are offered by different manufacturers and with different characteristics. When using such standard of-the-shelf components for the project, it is important to chose the exact device models that will fully meet the requirements. Our device model selections for the camera and the smartphone are presented in Section 4.5.1 and 4.5.2 respectively.

Equally important as the selection of the sensor devices is the choice of the appropriate software platform. When there is a necessity to develop a software application that involves handling inputs/outputs and lots of interprocess communication, as is the case with a distributed sensor system, a computer software known as *middleware* can be used to provide these services (Hadim and Mohamed, 2006). When using middleware the application developers can focus on the specific purpose of their application instead of writing the code to solve generic problems. Section 4.5.3 presents our choice for the middleware, while in Section 4.5.4 we present other important software that was used in the development process (e.g. development environment, data processing libraries).

4.5.1 Ambient Sensor: Microsoft Kinect

Microsoft Kinect gives us an open, easily programmable, well supported and economical sensor platform with satisfying technical properties. The main Kinect sensor modality that we want to use is the depth sensor based on PrimeSense *LightCoding* technology. The basic principle behind the Kinect depth sensor is to emit the infra-red (IR) light, and then to utilize a standard off-the-shelf CMOS image sensor fitted with an IR-pass filter to read the IR light back from the scene. The IR light that is emitted has a special pattern known by the Kinect. When the image processor of the Kinect reads the returned light pattern, it can calculate the depth displacement at each pixel position in the image (Batlle et al., 1998). The depth measured is an estimate of the distance from the object to the plane formed by the IR camera and laser, rather than the actual distance from the object to the IR camera opening. In this way, the depth sensor is basically a device that returns the (x, y, z)-coordinates of 3D objects. The main nominal specifications of Kinect are given in Table 4.1.

Table 4.1: Kinect hardware specifications.

Property	Value
Angular Field of View	57° horz., 43° vert.
Framerate	30 Hz
Depth image resolution	640x480 (VGA)
Nominal spatial resolution (at 2m distance)	3 mm
Nominal depth range	0.8 m - 3.5 m
Nominal depth resolution (at 2m distance)	1 cm
Extended depth range	0.5 m - 9.7 m

Kinect sensor was primary intended to be used indoor for gaming purposes. Since its release in November 2010, it was popularized by the enthusiasts in the field of computer vision as a tool for people tracking (Shotton et al., 2011; Xia et al., 2011), as a motion capture system (Dutta, 2012) and for building indoor maps (Stoyanov et al., 2013). Although the additional 3D information that it provides opens a whole spectrum of new possibilities and gives it the edge over RGB systems, especially in background subtraction and object recognition, there are still some limitations. One of the limitations is unavoidably related to characteristics of the viewed object. Due to the Kinect's dependency on reflected IR light, there are obvious problems with very reflective, mirror-like surfaces that are unable to reflect the light back. The other problematic type of surface is the one that is not reflective enough, such as very dark pieces of cloth. An additional complication can also be an interference with other IR light sources (e.g. sun or an open fire). The interference with the sun can limit the usefulness of Kinect in the areas close to windows, where strong sun rays are possible. Except with the external factors, Kinect is also characterized with the internally related factors. A few studies (Khoshelham, 2011; Andersen et al., 2012) measured the internal technical properties of the depth sensor, such as linearity, depth resolution, depth accuracy and precision, spatial precision and structural noise. More about how those internal aspects influence decisions concerning our tracking algorithm will be presented in Chapter 5.

4.5.2 Wearable Sensor: Smartphone

We used the Samsung Galaxy Nexus (GT-I9250) smartphone for the mobile sensing and processing. Relevant technical specifications are given in Table 4.2.

4.5.3 Middleware: Robotic Operating System

After the investigation of the available middleware systems for intelligent environments, we chose an open source, community-supported middleware from the robotics domain to develop our distributed sensor system. Robot Operating System (ROS) (Quigley et al., 2009)

Table 4.2: Smartphone hardware specifications.

Property	Value
Dimensions	135.5×67.9×8.9 mm
Weight	135 g
Processor	Dual-core 1.2 GHz Cortex-A9
Memory	1 GB RAM
Operating System	Android 4.0
Accelerometer	Bosch BMA220, 3-axis, 100 Hz, ±16 g max. range,
Gyroscope	Invesense MPU-3050, 3-axis, 100Hz, ±2000 $^\circ/s$ max. range
Magnetometer	Yamaha YAS530, 3-axis, 100Hz, ±800 µT max. range

is a meta-operating system that runs on top of the "real" operating system (e.g. Linux, Windows). ROS provides the services that would be expected from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers. ROS is not a hard real-time framework, though it is possible to integrate ROS with real-time code.

The fundamental concepts of the ROS implementation are nodes, messages, topics, and services. Nodes are the processes that perform computation, similar to an independent software module. Nodes communicate with each other by passing messages. Node sends a message by publishing it to a given topic. A node that is interested in a certain kind of data will subscribe to the appropriate topic. There may be multiple concurrent publishers and subscribers for a single topic. Publishers and subscribers are not aware of each other's existence. Publishing on topics is asynchronous communication. Synchronous transactions are supported by the concept of services, where messages are passed on the request/reply principle. ROS introduces the concepts of *packages* and *stacks* for easier distribution and reconfigurability. Nodes are grouped into packages, and a collection of packages makes a stack. One package usually solves one functionality, like camera calibration or face detection, while a stack covers the whole field of application, like computer vision or navigation.

Potentials of the ROS middleware in the context of the use in an intelligent environment were explored by Roalter et al. (2010). A great advantage of ROS are the stacks that provide automatic hardware support (openni-kinect) and access to various open source processing libraries.

4.5.4 Software Development Platform

The basic operating system for which the system was developed is Linux Ubuntu 12.04 LTS, with installed ROS (version Fuerte). Eclipse (version 3.7.2) integrated development environment was used for code development and debugging. Graphical user interface was developed

with QT 4.8 library. For 3D point cloud processing, the Point Cloud Library v.1.5 (PCL) was used, while image processing was done in OpenCV 2.0. The code was written in the C++ programming language.

4.6 Software Architecture

Figure 4.4 presents the software architecture of the system. The architecture is presented in the way that is used for the presentation of a graph of the distributed system in the ROS middleware. The oval objects represent ROS nodes. Each node is a process that is dedicated to the execution of the specific functionality (specified with a node name). As mentioned before, ROS enables interprocess communication by passing messages between nodes. The main topics in the system are marked by arrow lines. The node that is the source of an arrow line is the *publisher* for the topic. The nodes where the arrow line sinks are the *subscribers* for the topic. Topics are usually named by the notation that involves the name of the source node and the type of data that is published on them.

A smartphone and an unspecified number of Kinect sensors are the possible sources of sensor data. Each Kinect has its own dedicated node that processes its color and depth image data. This node is called the `Vision` node. The main task of the `Vision` node is to track all the persons that are in front of the Kinect and to provide their locations for each image frame. The secondary task of the `Vision` node is to collect information about the appearances of the tracked persons. The `Vision` node also contains graphical user interface (GUI) that is used by a researcher for setting up environment maps and context zones. Environment maps are obtained from the dedicated `Map_Server` node. This node is always active as it waits a request from any `Vision` node to produce a map. When the request comes, `Map_Server` node takes the 3D point cloud of the background in front of the Kinect as the input, and returns the 2D projection of the cloud in the form of a bitmap. The details of the `Vision` node implementation and its interaction with the `Map_Server` node are described in Chapter 5.

In a distributed system with many Kinect cameras, many person tracks may exist at the same time. In our application we are interested to know which of those tracks belong to the patient. The identification manager (`ID_Manager`) node handles the assignment of the identity for all tracks. It supports two modes of operation: 1) a mode in which it learns the identity from the appearance of the persons in the system; and 2) a mode in which it uses the learned appearance models to identify persons. When the `ID_Manager` identifies the patient, it forwards the patient's location (and other context data) to the rest of the distributed system. The algorithm for the extraction of appearance features and learning identification models will be presented in Chapter 7.

To use an Android smartphone inside the ROS middleware, the smartphone needs to have installed a special node, named `ROS_Android`. This node enables the smarthphone to publish and subscribe on ROS topics in the same network. The `ROS_Android` node acquires raw signals from the inertial sensors and distributes them to the two processing nodes. The `IMU_Orientation`

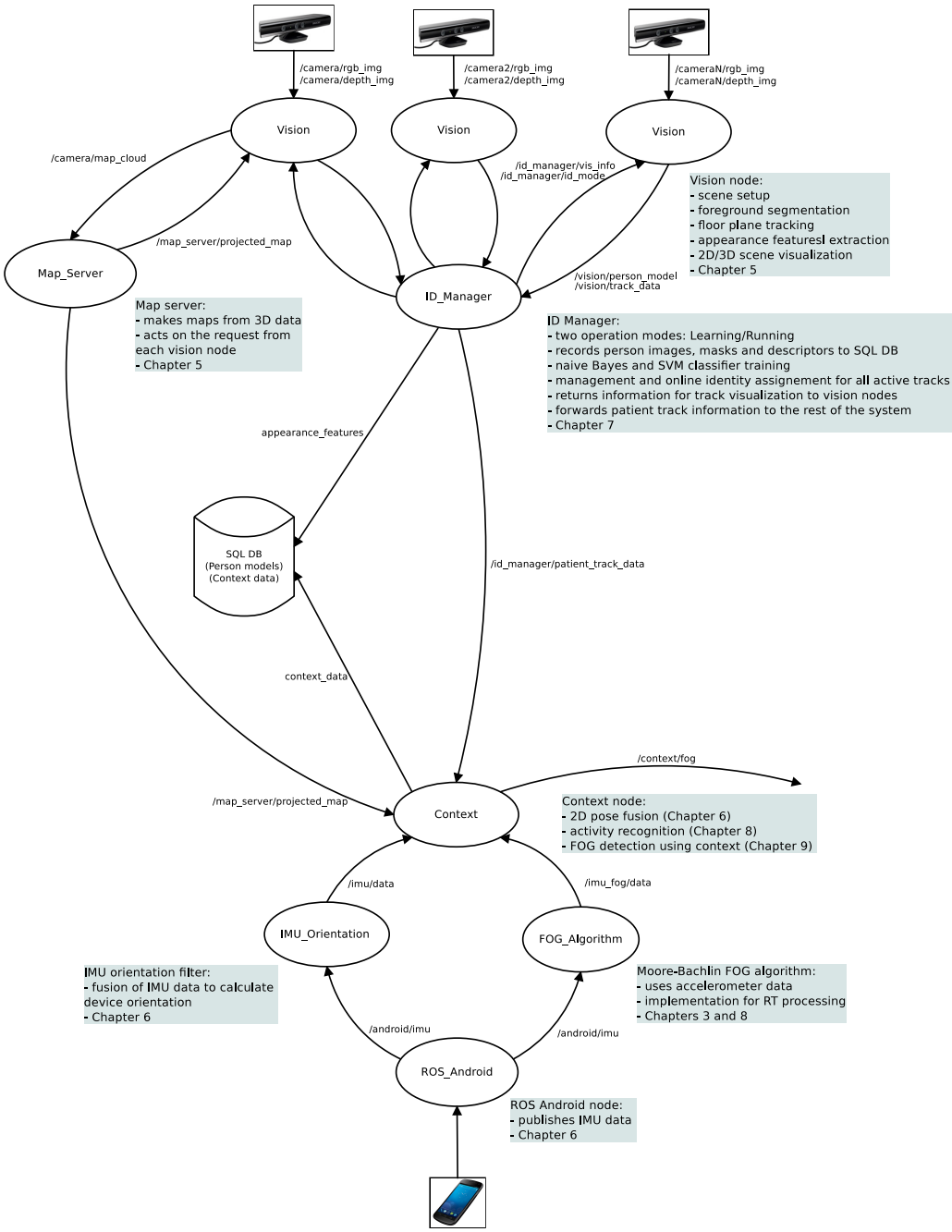


Figure 4.4: System architecture overview.

node calculates the absolute 3D orientation of the smartphone device based on the fusion of the accelerometer, gyroscope and magnetometer sensor channels. The `FOG_Algorithm` node uses data from 3 accelerometer channels to calculate the *FI* using the Moore-Bächlin algorithm. Results from the both nodes are then forwarded towards the `Context` node, where the final decision regarding FOG is made. The use of a smartphone for tracking the patient's orientation and calculation of his/her 2D pose is explained in Chapter 6. Two other main tasks of the `Context` node, activity recognition and contextually enhanced FOG detection, will be presented in Chapter 8 and Chapter 9, respectively.

5

Multiple Person Tracking and Localization

The concept of our monitoring system requires the placement of multiple Kinect cameras in a home. These cameras form a home camera network whose main purpose is to localize the PD patient. The location of the patient has to be determined on a map of the environment. Thus, the camera network needs to ensure two functionalities: 1) a way to continuously provide accurate location of the patient (expressed in the coordinates of the appropriate coordinate system); and 2) a way to obtain the appropriate map of the environment.

Video tracking is the process which uses a camera to estimate the location of one or more objects over time (Maggio and Cavallaro, 2011). The actual object of interest (tracking target) can be anything, depending on the specific application. In our medical application the object of interest is a human target. Video tracking is an established field that already offers plenty of potential methods and algorithms that can be used with our hardware setup in order to localize PD patients.

We start the chapter with an overview of the main elements of the tracking process, and we list the existing difficulties that are present when tracking humans with a color camera in indoor environments. We recognize that in our system, a possibility to eliminate some of the tracking errors and achieve a robust indoor tracking, exist in the correct use of the Kinect's depth sensor. One of the most common approaches to make use of the depth data is the method known as *background subtraction*. We conduct an exploration of some of the existing background subtraction methods, and search for a people tracking method that is able to benefit from the addi-

tional depth information. We conclude the first part of the chapter with a detailed theoretical description of one such tracking method, that we have chosen to implement. In the second part of the chapter, we describe the practical implementation of the chosen tracking algorithm, along with the implementation of the additional algorithms that are necessary for the extraction of context data. The tracking method and the extraction algorithms are implemented in the `Vision` node. This node is a complete software solution that, besides tracking, also tries to handle the mapping of the environment and to provide the setup utility to the researchers. After reading this chapter, the importance of the `Vision` node, its inner workings and the way in which its elements interact with the other nodes in the distributed system should be clarified.

5.1 People Tracking

The main logical components of the video tracking algorithm, also known as the *video-tracking pipeline*, include:

- Initial detection of the target object;
- Extraction of relevant information (features) from the tracking target and encoding this information into a suitable representation inside a computing system;
- Propagation of the state of the target via an update with new features; and
- Managing the target by eliminating it when it is evident that it has left the scene.

The video tracking process is hampered due to the loss of information by the projection of the 3D world on a 2D image. The challenges in this process are related to the similarity of appearances between the target and other objects in the scene, and the variations of the appearance of the target (Maggio and Cavallaro, 2011). Appearance variations can happen due to:

Angle change
: Except the completely spheric objects, all other types of objects vary in appearance when seen from different angles.

Translation
: The distance from the camera directly influences the size of the object; those further from the camera appear smaller than those near to the camera.

Deformation
: Some objects (e.g. a car) are completely rigid, while the other (e.g. a human) are deformable and can assume shapes that are hard to predict and model in advance.

Illumination changes
: The appearance of the object may vary due to the properties of the ambient light, such as direction, intensity and color temperature. Change in the position of the light source or the object movement can also influence the amount of light that falls on the object.

Shadows and reflections
: When a shadow of the object is cast on the ground it appears in the shape of the object that has produced it. For the trackers that do not use color and use just motion or shape, the shadow practically represents a duplicate of the object. The reflections of objects on smooth surfaces have the same effect as shadows.

Occlusions
: Occlusion occurs when there is an object that prevents full visibility between the camera and the tracking target. A target object can get occluded by another moving object, by its own moving part, or by moving itself behind a static object that forms the scene background. Furthermore, an occlusion can be a *partial occlusion*, where only a part of the target is not visible, or a *full occlusion* in which the target completely disappears.

One of the most fundamental problems in the design of video tracking is finding the appropriate description, known as the *object model*, for the object that we want to track. The *object model* has to be specific enough to allow the target to be clearly distinguished from other similar areas on the image. However, it has to be general enough so that the target can be linked to its previous instances in time. The *object model* usually includes the information about the shape and/or the appearance of the target (Yilmaz et al., 2006). Object representations that are based solely on shape encompass: points (Veenman et al., 2001; Shafique and Shah, 2003), primitive geometric forms such as rectangle or ellipse (Comaniciu et al., 2003), object silhouette and contour (Yilmaz et al., 2004) and articulated models (Thome et al., 2008; Sundaresan and Chellappa, 2009). Examples of the strictly appearance-based representations are probability density estimates (e.g. Gaussian distribution (Han and Davis, 2005), histograms (Pérez et al., 2002)) and templates (Jurie and Dhome, 2002). Active appearance models are generated by simultaneously modelling the shape and the appearance of the object (Cootes et al., 1998).

The choice of the object model for tracking is dependent on the particular application. For tracking locations of small (Jaqaman et al., 2008) or distant (Shafique and Shah, 2003) objects already a point representation can be sufficient. On the completely other part of the spectrum are detailed human motion recognition applications in which articulated 2D or 3D models are used. More about these complex models can be found in surveys on human motion recognition (Moeslund et al., 2006; Poppe, 2007).

5.1.1 Background Subtraction and Depth Data

Smith (2007) proposes an alternative approach for modelling objects in a video by modelling everything that is not the object itself. This technique is known as background subtraction or foreground segmentation. This approach aims to detect moving objects within a video stream from the difference between the current frame and a reference frame, called the *background image*, or the *background model* (Piccardi, 2004). The simplest method for background modelling is the *static frame difference* where one image is taken at the beginning of the tracking and is used for the difference calculation with each new frame. This static model has no way to

deal with the various dynamic changes of the scene such as illumination changes or shadows. The *frame difference* approach uses the previous frame as the background model for the current frame, which successfully eliminates changes in the background, but also fails if the moving object suddenly becomes static. A more robust and complex solution is to obtain a continuously adaptive background model from the temporal sequence of frames. Features such as color and texture (Zhang and Xu, 2006; Jian et al., 2008) or edges (Jain et al., 2007) may be used for this task.

One of the most often used background modelling methods are the statistical methods. Wren et al. (1996) modelled the background independently at each pixel location by fitting a Gaussian probability density function. This model copes well with gradual illumination changes, but fails for non-static backgrounds, such as moving leaves of a tree. To improve the algorithm behaviour in such situations, Stauffer and Grimson (1999) proposed to model each pixel color as a sum of weighted Gaussian distributions, known as Gaussian Mixture Models (GMMs). The GMM has a good performance in the analysis of outdoor scenes, and has become a very popular background subtraction algorithm due to its ability to handle low illumination variations (Sobral and Vacavant, 2014). Rapid variations of illumination and shadows are still problematic and many authors studied ways how to improve this method (KaewTraKulPong and Bowden, 2002; Zivkovic, 2004; Tuzel et al., 2005).

One of the possible solutions for improving the GMM-based background subtraction is to add depth data. The potentials of using depth data in a tracking system based on background subtraction were nicely summarized by Harville (2004). The author advocates the use of depth in a person tracking system and summarizes its advantages:

- It is a powerful cue for foreground segmentation,
- It provides shape and metric size information that can be used to distinguish people from other foreground objects,
- It allows occlusions of people by each other or by background objects to be detected and handled more explicitly,
- It permits the quick computation of new types of features for matching person descriptions across time,
- It provides a third, disambiguating dimension of prediction in tracking.

One recently introduced approach to the background subtraction that achieves excellent results is the algorithm called *Depth-Extended Codebook* (DECB) proposed by Fernandez-Sanchez et al. (2013). The DECB builds up on the well known *Codebook* background subtraction algorithm (Kim et al., 2005), by fusing depth and color information to segment foreground regions. The basic color *Codebook* method samples values of the background at each pixel and quantizes/clusters them into a compressed representation of the background model convenient for

a long-term observation. For each pixel location, the image color values are grouped into clusters called *codewords*. After some time there will be several codewords built for each pixel. This set of codewords is called the *codebook*. The criteria for assignment into a codeword cluster is the color distortion metric with brightness bounds. Although the color-based algorithm handles well moving backgrounds or illumination variations, there is a problem with the robustness of the algorithm to shadows, highlighted regions and sudden lighting changes. An additional depth information can be helpful in improving these shortcomings.

When a shadow appears or the illumination changes suddenly, the depth information of the pixel stays unchanged. The easiest way to exploit the depth would be to base background subtraction only on the depth and disregard the color information. In that case, the objects that have a similar depth as the background would miss from the foreground, because of non-sufficient depth sensitivity. Hence, the DECB approach consists of modifying the condition around the color distortion to consider the depth only when the color distortion is between two specific thresholds ε_1 and ε_2. The threshold ε_1 is used in the original color-based *Codebook* algorithm. If the color distortion measures less than ε_1, the pixel is definitively considered to be a part of the background. The addition of the second threshold ε_2, where $\varepsilon_2 > \varepsilon_1$, defines the area of uncertainty stemming from illumination changes. This is the case when a pixel would be declared as the foreground, but is still close enough to the ε_1 threshold that it might be a part of the background. In such case, pixel depth value is taken into account.

The comparison of the DECB and the *Codebook* algorithm by Fernandez-Sanchez et al. (2013) demonstrated clear benefits of using depth data for the background subtraction. Besides improving the background subtraction, depth data offers additional advantages for people tracking, such as better handling of occlusions and image noise. This benefits are possible if we use a special data representation, the *plan-view*, which we will explain in the next section.

5.1.2 Plan-view Representation and Maps

When using the Kinect in the overhead position, a 3D point cloud can be constructed from the depth data. The created point cloud does not have the same depth resolution at all the distances from the camera. The depth resolution decreases quadratically with increasing the distance from the sensor. The point spacing in the depth direction (along the optical axis of the sensor) is around 7 cm at the range of 5 meters, compared to 1 cm at 2 m distance (Khoshelham, 2011). Also, the random error of depth measurements increases quadratically with increasing distance from the sensor and reaches 4 cm at the range of 5 meters (Khoshelham and Elberink, 2012). These sensor characteristics make it difficult to apply typical image analysis and tracking methods to depth data with the same confidence on all the distances from the camera. To deal with the sensor related problems and equalize their influence over the whole scene, it is more favourable to analyse the depth data statistics rather than working with the raw depth values (Harville, 2004).

Plan-view projection is one of the six possible multi-view orthographic projections (Carlbom and Paciorek, 1978) of a body in which the projecting plane is horizontal (parallel to the floor)

and the object is seen from the top. The separation between the people who are close together is better on *plan-view* projection, then in the image produced by the camera positioned under the ceiling with a downward angle of around 45° from the horizontal (which we designate as *camera-view*). Projected depth data can then be combined with the statistical approach to avoid the influence of occlusion, depth sensor non-linearity and noise on the tracking. Our main arguments to use a *plan-view* based person tracker are: 1) it is easy to produce the *plan-view* representation by orthographically projecting 3D point cloud of the scene foreground on the floor plane; and 2) the projected depth data can be combined with the statistical approach to avoid the influence of occlusion, sensor resolution non-linearity and noise on the tracking process.

Figure 5.1 presents the main geometric principles, coordinate frames and frame transformations necessary to obtain a *plan-view* projection from an overhead camera view of a 3D point cloud. The two main coordinate systems are *world* coordinate system $X_w - Y_w - Z_w$ and *camera_image* coordinate system $X_{ci} - Y_{ci} - Z_{ci}$. The original point cloud built from Kinect data has the $X_{ci} - Y_{ci} - Z_{ci}$ frame as its reference, and its points are expressed in homogeneous coordinates as $p_{ci} = (x_{ci}, y_{ci}, z_{ci}, 1)$. The *world* frame $X_w - Y_w - Z_w$ has two of its axes parallel to the floor plane, while the Z coordinate axis is orthogonal to the floor and aligned with the vertical axis of the world. We can expect that people will be aligned with this vertical axis during locomotion and the majority of other activities that do not involve lying down. Hence,

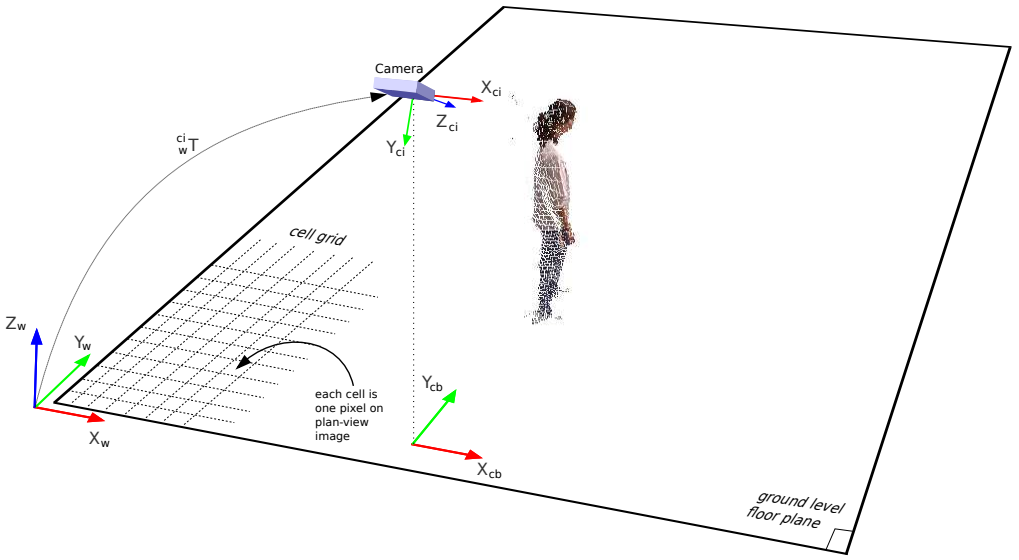


Figure 5.1: Three-dimensional reconstruction of the scene showing the reference coordinate frames for plan-view mapping.

for tracking purposes it is preferable to translate the foreground point cloud in the *world* frame coordinates. To obtain a point $p_w = (x_w, y_w, z_w, 1)$ in the *world* frame, we need to know the position and orientation of the camera towards the floor plane and to establish a homogeneous transformation ${}^{ci}_{w}T$. In Figure 5.1 we can also observe *camera_base* frame $X_{cb} - Y_{cb}$. This an auxiliary frame which is formed by the projections of the axes X_{ci} and Z_{ci} onto the floor plane. This frame is often used as a frame for representing the camera on the 2D plan-view image.

From Kinect we obtain a large amount of 3D data which makes it possible to have tracking directly on the point cloud. However, this approach can be computationally expensive. Instead, it is possible to find 2D projections of the 3D data that reduce the original amount of information, but still preserve meaningful spatial properties useful for tracking. The solution for dimensionality reduction is to divide the world into vertical bins parallel to the axis Z_w at which base on the floor is a grid of 2D cells and then project each point cloud point p_w on the grid. Such discrete grid can easily be seen as a 2D image where one cell equals one pixel. A value of each pixel value can be calculated as some statistic of the 3D points within the corresponding vertical bin. When a statistic is calculated, we call the obtained data representation *plan-view* map. Three kinds of maps have been used so far:

Occupancy map
: First introduced by Beymer (2000). Instead of simply counting the number of points in the vertical grid cell, we calculate how much space they occupy. In this way the map displays weighted counts of the points in each bin, which compensates for the smaller appearance of more distant objects in the *camera-view* image. The usual weighting equation is Z_{ci}^2/f^2, where the measured depth value in the *camera_image* frame coordinates is weighted by the focal length f of camera. Depth value weighted with the focal length approximates the physical surface area covered by the related image point. This representation omits almost all object shape information in the vertical dimension.

Height map
: The height above the ground-level plane of the highest point within each vertical bin. Height maps preserve as much 3D shape information as possible in a 2D image, and therefore seem better suited than occupancy maps for distinguishing people from each other and from other objects (Harville, 2004). A more pronounced shape of projections provides better features for accurately tracking people during close interactions and partial occlusions. Also, from the overhead angle the head and the upper body of a person are usually visible even when another person is passing very close by and producing an occlusion. Hence, height map contains data that is more robust to partial occlusions.

Color map
: First deployed by Harville (2005), this type of map registers color information of the scene. Different color statistics can be used, such as the color of the highest point in each vertical bin, or the mean color of all points in the bin. Color maps add important

appearance information that can be used for making both short and long term person appearance models.

5.1.3 Plan-view Tracking

Several multiple person tracking approaches with *plan-view* maps have been applied so far. Beymer (2000) modelled people with Gaussians that were applied to occupancy maps and used the Kalman filter maintained through position and velocity updates. Darrell et al. (2001) used dynamic programming with occupancy map to solve a batch trajectory estimation problem for each person. However, their solution was still impractical for real-time use. A method that allowed for online, real-time people detection and tracking was developed by Harville (2004). He used a Kalman filter with occupancy maps, height maps and adaptive templates. In the continuation of the work with adaptive templates, Harville and Li (2004) replaced the Kalman filter with a new probabilistic technique based on the maximum a posteriori (MAP) method. This method was further adapted to incorporate *plan-view* color maps and long-term appearance models (Harville, 2005).

The first to use particle filters for multiple people tracking with *plan-view* occupancy maps was Hayashi et al. (2004). Unlike Kalman filter, particle filters can deal naturally with systems where both the posterior density and the observation density are nonlinear and non-Gaussian. A particle filter provides a robust tracking framework, since it models uncertainty and considers multiple state hypotheses simultaneously, which helps when dealing with short occlusions (Nummiaro et al., 2002).

Muñoz-Salinas (2008) presented a low-error stereo camera tracking method that can deal with partial and total occlusions. The method uses multiple particle filters and three types of *plan-view* maps. We evaluated that this method could meet the requirements of our application, and used it to implement the core people tracking functionality of the `Vision` node. In this subsection, we summarize the main principles of the method presented by Muñoz-Salinas (2008) using the author's original notation.

Space discretization

The first task is to discretize properly the space in front of camera and to produce maps. A cell grid on the floor plane has $n \times m$ rectangular cells with fixed a size of side δ. The origin of the *plan-view*, the cell $(0, 0)$, is set at the position $(0, 0, Z_w)$ in the *world* frame (Figure 5.1). Cell coordinates (x^i, y^i) can be obtained from the 3D point p_w^i by the following calculations:

$$x^i = \frac{X_w^i}{\delta},\ y^i = \frac{Y_w^i}{\delta}. \tag{5.1}$$

Only the 3D points in the specified height range are allowed for tracking. This is achieved by using height thresholds. Maximum height limit h_{max} avoids including points from the ceiling, while the minimum height limit h_{min} avoids inclusion of floor or very low

objects. A set of points in the vertical bin projecting into a cell (x, y) is defined as:

$$P_{(x,y)} = \{i \mid x^i = x \wedge y^i = y \wedge Z_w^i \in [h_{min}, h_{max}]\} \tag{5.2}$$

Making plan-view maps

The method uses all three types of *plan-view* maps presented in the previous section; occupancy map $\mathcal{O}$, height map $\mathcal{H}$ and color map $\mathcal{C}$. The value in each cell of the occupancy map is calculated using *camera_image* pixels by the formula:

$$\mathcal{O}_{(x,y)} = \sum_{j \in P_{(x,y)}} \frac{(Z_{ci}^j)^2}{f} \tag{5.3}$$

Each cell of the height map takes the maximum height among all the points in its vertical bin according to the equation:

$$\mathcal{H}_{(x,y)} = max(Z_w^j \mid j \in P_{(x,y)}) \tag{5.4}$$

Each color map cell $\mathcal{C}_{(x,y)}$ contains a color histogram in HSV space of the points from $P_{(x,y)}$. The histogram is composed out of $m = n_h \times n_s + n_v$ bins, where the subscripts h, s and v designate hue, saturation and luminance respectively. A 2D histogram of $n_h \times n_s$ bins contains chromatic information part, while the luminance information in n_v bins accounts for credibility of color information when it is too bright or too dark.

Extraction of measurements

An unoccluded person will project to an area on the map proportional to its real dimensions. A major part of a person in an upright posture will fit in a rectangular region whose size of the side, ζ_R, varies from 0.4 to 0.6 m (Muñoz-Salinas, 2008). People detection and tracking is based on the analysis of rectangular regions from which the three measures, each based on one type of map, are extracted. Before extracting measures, the person region has to be defined in the coordinates of the *plan-view* map. For this purpose, we introduce the parameter ζ_M that represents the size of a person's projection ζ_R expressed in the number of grid cells. Using the parameter ζ_M, a rectangular person region can be defined as a set of cells centred around (x, y) that satisfy the equation:

$$\mathcal{R}(x, y) = \{i \mid max(|x^i - x|, |y^i - y|) < \zeta_M/2\} \tag{5.5}$$

The first measure, defined as:

$$\mathcal{O}_{\mathcal{R}(x,y)} = \sum_{i \in \mathcal{R}(x,y)} \mathcal{O}_{(x^i, y^i)} \tag{5.6}$$

provides information about the total surface area that an object in the region occupies.

The second measure, defined as:

$$\mathcal{H}_{\mathcal{R}(x,y)} = max(\mathcal{H}_{(x^i,y^i)}), \forall i \in \mathcal{R}(x,y) \tag{5.7}$$

represents the height information of a region by recording the maximum height among all the points projected in the region. The third measure $\mathcal{C}_{\mathcal{R}(x,y)}$ is the color histogram of a region. The region histogram has the same number of bins m as the color map. Each bin u is calculated by aggregating color information of the cells in the region by applying:

$$\mathcal{C}_{\mathcal{R}(x,y)}(u) = 1/|\mathcal{R}(x,y)| \sum_{i \in \mathcal{R}(x,y)} \mathcal{C}_{(x^i,y^i)}(u) \tag{5.8}$$

The division by the total number of cells $|\mathcal{R}(x,y)|$ is done in order to obtain a normalized histogram.

People detection

The method presumes the availability of a foreground point cloud in which there can be several people (or other movable objects). Regions in which people project on *plan-view* are expected to have height $\mathcal{H}_{\mathcal{R}(x,y)}$ and occupancy $\mathcal{O}_{\mathcal{R}(x,y)}$ that fall into some normally distributed ranges defined by Gaussian distribution parameters (μ_h, σ_h) and (μ_o, σ_o). The two measures can be combined into a likelihood of the cell(x,y) to be a centre of the person by the equation:

$$\mathcal{L}_{new}(x,y) = \frac{exp\left(-\left((\mathcal{O}_{\mathcal{R}(x,y)} - \mu_o)^2/\sigma_o^2 + \mathcal{H}_{\mathcal{R}(x,y)} - \mu_h)^2/\sigma_h^2\right)\right)}{2\pi\sigma_o\sigma_h} \tag{5.9}$$

The above equation is applied to every cell in the *plan-view* in order to create a likelihood map $\mathcal{L}_{new}$. The presence of people in the scene causes regions of high likelihood in $\mathcal{L}_{new}$. People are detected by searching for peaks in this new type of map. For each new frame a new $\mathcal{L}_{new}$ is calculated and it can potentially contain the information about the people that are already being tracked and about the new persons that just entered in the scene and are not tracked. New persons are detected only after the positions of already tracked people are determined and erased from the $\mathcal{L}_{new}$. After the deletion, the cell (x,y) with the maximum likelihood is selected as the candidate for a new person. After being added to the list of potential candidates, the likelihood in all the cells of region $\mathcal{R}(x,y)$ is set to zero to avoid considering the region again as a candidate. The search process is repeated for other potential new candidate regions, until the cell with maximum likelihood is below a certain threshold. The error of omitting some region as a person candidate, is still possible due to noise and occlusions. Therefore, the principle of temporal consistence is applied. The same candidate region has to be detected for a minimum number of consecutive times, before it gets promoted into a person track. When the promotion to a track occurs, the person's color model $\mathcal{C}_p$ is created using color

histogram of the candidate region $\mathcal{C}_{\mathcal{R}(x,y)}$.

Particle filter

The state of the object tracked by PF at the time t is described by the vector X_t, while the vector Z_t denotes all observations $\{\vec{z}_1, ..., \vec{z}_t\}$ up to time t. The particle filter uses a weighted sample set $S_t = \{(\vec{s}_t^{(n)}, \pi_t^{(n)}) | n = 1...N\}$ (where n indicates a particular sample in the set of N samples) to approximate the posterior density $p(X_t|Z_t)$ at a discrete time t. Each particle $\vec{s}_t^{(n)}$ represents a possible state of the object that can be true with some sampling probability $\pi_t^{(n)}$, conditioned by $\sum_{n=1}^{N} \pi_t^{(n)} = 1$. The particle filter incorporates a cyclical repetition between the three main actions: *state prediction*, *state correction* and *particle resampling*. During *state prediction*, the sample set is propagated according to the dynamic system model $\vec{s}_t = \mathcal{A} \cdot \vec{s}_{t-1} + \vec{w}_{t-1}$, where t designates the current iteration of the discrete time system, $t-1$ designates the previous iteration, matrix $\mathcal{A}$ defines the deterministic component of the model, and $\vec{w}_{t-1}$ is a multivariate Gaussian random variable. Usually, to describe human movement on a plane, it is sufficient to use a matrix $\mathcal{A}$ that describes a first order dynamic system with a constant velocity $(\dot{x}, \dot{y})$. During *state correction* each sample $\vec{s}_t^{(n)}$ in the set $S_t = \{(\vec{s}_t^{(1)}, ..., \vec{s}_t^{(N)}\}$ is weighted accordingly to its observation density $p(\vec{z}_t|X_t^{(n)})$ to find its measured probability $\pi_t^{(n)} = p(\vec{z}_t|X_t^{(n)})$. The *particle resampling* process generates a new set of particles with the probability of each sample n from the old set to be repeated equal to $\pi^{(n)}$. After each *state correction* step, a mean state of the tracked object ε_t is estimated by calculating $\sum_{n=1}^{N} \pi_t^{(n)} \cdot \vec{s}_t^{(n)}$.

State and observation model

With the assumption that there are P people tracked, the state of the specific particle filter for the j-th tracked person can be defined in terms of variables X_t^j and Z_t. State vector $X_t^j = (x_t^j, y_t^j, \dot{x}_t^j, \dot{y}_t^j)$ represents position and velocity on a 2D floor plane at time t, while Z_t denotes concurrent world observations in the form of the *plan-view* maps. The observation model combines occupancy O_t^j, color C_t^j and height information H_t^j from the *plan-view* maps to calculate the observation density, defined as:

$$p(Z_t|X_t^j) = p_o(O_t^j|X_t^j) \cdot p_h(H_t^j|X_t^j) \cdot p_c(C_t^j|X_t^j) \tag{5.10}$$

The likelihood of the specific particle n at time t in filter j is obtained from the general observation by applying:

$$\pi_t^{j,(n)} = p(Z_t|X_t^{j,(n)}) \cdot p_h(H_t^{j,(n)}|X_t^{j,(n)}) \cdot p_c(C_t^{j,(n)}|X_t^{j,(n)}) \tag{5.11}$$

The variable $\mathcal{O}_t^j = \mathcal{O}_{\mathcal{R}(x,y)} \sim \mathcal{N}(\mu_o, \sigma_o^2)$ represents the occupancy level of region $\mathcal{R}(x, y)$ centred at the particle n. Similarly, the maximum height in the region $\mathcal{R}(x, y)$

is taken for the calculation of variable $\mathcal{H}_t^j = \mathcal{H}_{\mathcal{R}(x,y)} \sim \mathcal{N}(\mu_h, \sigma_h^2)$. Probability distributions $p_o(O_t^j|X_t^j)$ and $p_h(H_t^j|X_t^j)$ are obtained by comparing the level of divergence between the variable values and expected mean values μ_o and μ_h. The color distribution $p_c(C_t^j|X_t^j)$ is defined based on a Bhattacharyya distance C_t^j between two color distributions; the histogram color model sustained for the person C_p, and the histogram $\mathcal{C}_{\mathcal{R}(x,y)}$ of the region on which the particle is centred.

Multiple trackers and occlusion handling

Alterations in the original CONDENSATION algorithm were made to improve total occlusions handling and preserve multimodality. An occlusion is recognized when the observation likelihood for estimated person track is below a threshold. Multimodality is preserved by defining an interaction factor that models interactions between persons (Khan et al., 2005). The interaction factor $\mathcal{I}_{i,t}^j$ prevents particles from the j-th tracker to end up tracking another target with a similar observation model. The interaction factor goes towards a zero value, when particles of the tracker are near positions of other people with similarly colored clothes. When particles are far from other people, or near people with clothes of different color, the interaction factor goes towards value 1. The calculation of the interaction factor requires a prior knowledge of the distances between the mean position estimation ε_t^j for tracker j and position estimations $\varepsilon_t^i, \forall i \neq j$ for all other active trackers. In this summary we just gave the intuition about the purpose of the interaction factor $\mathcal{I}_{i,t}^j$ and presented the necessary inputs for its calculation. For a more detailed explanation and exact equations, it is recommended to consult the original work of Muñoz-Salinas (2008).

5.2 Vision Node

`Vision` node is the basic node in our system, which tracks people and their locations. Besides the primary tracking functionality, this node has several support functions that were listed in its description in Figure 4.4 in Chapter 4. The `Vision` node has to provide the data for learning appearances of persons in `ID_Manager` node, it has to provide the data for mapping in `Map_Server` node and, it serves as the user interface for the whole distributed system. In this section, we describe in detail the implementation of the `Vision` node, its interaction with the other nodes in the distributed system that depend on it, and the interaction with the system administrator.

5.2.1 Overview

Each Kinect in the network has its own instance of `Vision` node. The context in which one `Vision` node operates inside the distributed system is illustrated in Figure 5.2. The Kinects in the system are using the *camera* + *ordinal number* naming convention (e.g. *camera2*), with the

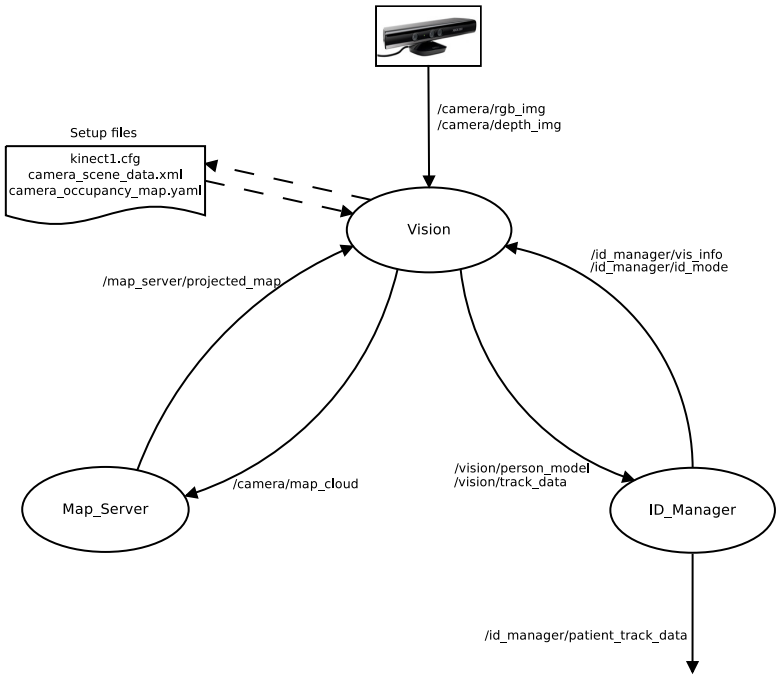


Figure 5.2: `Vision` node in a distributed system.

first Kinect simply named *camera*. The `Vision` node takes two inputs from its Kinect: color image and depth image. For example the Kinect named *camera* (Figure 5.2) publishes its images on the topics */camera/rgb_img* and */camera/depth_img*, respectively. Color image is the standard 24-bit image with 3 channels (RGB), where each channel has 8 bits. The 11-bit depth image is obtained from the raw disparity measurements that are normalized and quantized between 0 and 2,047 (Martinez and Stiefelhagen, 2013). When using the ROS middleware with OpenNI drivers for Kinect, color and depth images can be streamed in one of the three different resolutions: a) 640x480 (VGA); b) 320x240 (QVGA); and c) 160x120 (QQVGA). Also, Kinect can be set to stream images with either 15 Hz or 30 Hz frame rate. The choices of streaming parameters, along with the other data related to the specific camera, such as the camera calibration data or the maximum depth to be covered are recorded in the camera configuration file (*kinect1.cfg*).

New Kinect image data has to be processed every time before it is passed to the `Vision` node. The principal mechanism of sensor data processing in the ROS setting is callback function. Whenever a new message is available on the topic, ROS calls the callback function registered with the topic and passes the new message. The data inside the new message is read in the function and a desired routine is performed. After the routine is finished, the result for new message data is usually published on the output topic.

The integration of the depth and color data into a 3D color point cloud requires use of color and depth images in the same callback function. Color and depth images are transmitted from

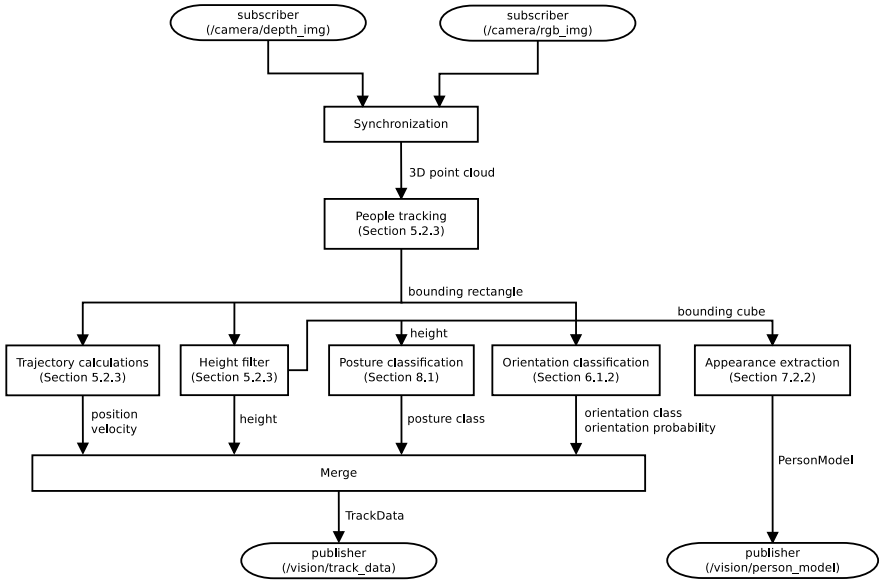


Figure 5.3: Main functionalities and data types in *rgbdCallback* function.

Kinect via two independent streams and their frames have a slightly different timestamps of origin. The image streams need to be matched so that the frames that originated the closest in time are paired together. A ROS message filter mechanism can take in messages and may output those messages at a later time, based on the conditions that are prescribed for the filter. The *Synchronizer* filter in ROS setting synchronizes incoming channels by their timestamps, and outputs them in the form of a single callback (ROS.org, 2014)*. The callback function for processing the Kinect data in the `Vision` node is called *rgbdCallback*. A simplified block diagram showing functional components and data types in *rgbdCallback* is shown in Figure 5.3.

The most critical and important functionality in the *rgbdCallback* function is people tracking (see Section 5.2.2). People are tracked in the two dimensions (x, y) of the floor plane, and the output of the tracking for each person is a bounding rectangle around the person of the constant width w. Since the tracking algorithm uses as the input 3D point cloud data, the height h of the point cloud at the current 2D position of the person's track can be easily retrieved. Under the assumption of the person's vertical orientation, a bounding cube (x, y, h) is obtained around the person's point cloud.

Kinect depth data can be very noisy at large distances which introduces errors into the tracking process (Khoshelham and Elberink, 2012). Even for the perfectly still target there will be variations in the 2D position estimation output. Hence, it is useful to employ filters on the position data from the tracker (see Section 5.2.3). Filtered 2D position data can be used to cal-

*A complete online documentation for ROS middleware is available at: *http://wiki.ros.org/*

culate its derivative, the velocity, that will be used for activity recognition based on trajectory properties (presented in Chapter 9). Except from the trajectory data, activity recognition also benefits from the knowledge of the person's static posture. The classification of static posture uses the filtered height data as the main input in the finite state machine explained in Chapter 8 (see Section 8.1).

The bounding cube representation simplifies the extraction of the necessary visual features from the tracked person, since the points of the bounded point cloud can easily be projected onto the scene floor plane or the camera image plane. By projecting the cloud points onto the floor plane, it is possible to get a 2D image with patterns that are useful for the classification of a person's orientation. Although this visual orientation classifier is implemented as the part of `Vision` node, the exact role of the visual orientation classification in the person orientation tracking process is explained in Chapter 6 (Section 6.1.2). Similarly, in Chapter 7 (Section 7.2.2) we will explain the algorithm for the extraction of appearance features from the projection of the bounded points on the camera image plane and its importance in the process of people identification.

The final output of the `Vision` node towards the other nodes in the system is published using two custom message types, *TrackData* and *PersonModel* (see Appendix E for both). *TrackData* messages are published on the end of each call to rgbdCallback. One *TrackData* message is made for each person track that is active, and the message contains all the trajectory related data, along with the results of both classifications. *PersonModel* messages contain the information about the appearance of each track. Their scheduling is dependent on the internal timers that are controlled via messages from the *IDManager* node.

The `Vision` node supports real-time visualization of the tracking process output. Supported modes of visualization are: 1) bounding rectangle around the person on a 2D map (seen from above); 2) bounding rectangle around the person in the original RGB image (seen from overhead perspective); and 3) bounding cube around the 3D point cloud of the person. The ability to directly visualize results of the tracking is very useful during the final phase of the system setup process, when we want to confirm that the Kinect was set in the optimal position to cover the desired surveillance area. Prior to the final verification phase, there is the offline phase of setup that consist of: 1) finding the plane of the floor; 2) recovering 3D pose of the camera; 3) mapping the scene and setting the boundaries of the tracking area; and 4) adding contextual zones on the map. The elements of the user interface for visualization and system setup will be presented in Section 5.2.4.

5.2.2 Tracker Implementation

This subsection displays the specifics of the implementation and additional interventions that adapt the people tracking method of Muñoz-Salinas (2008) for our specific purpose. We give the overview of the implemented processing stages and their outcomes in Figure 5.4.

Color and depth images are synchronized in the *rgbdCallback* function (Section 5.2). After the synchronization, both images are used in the background subtraction. For the back-

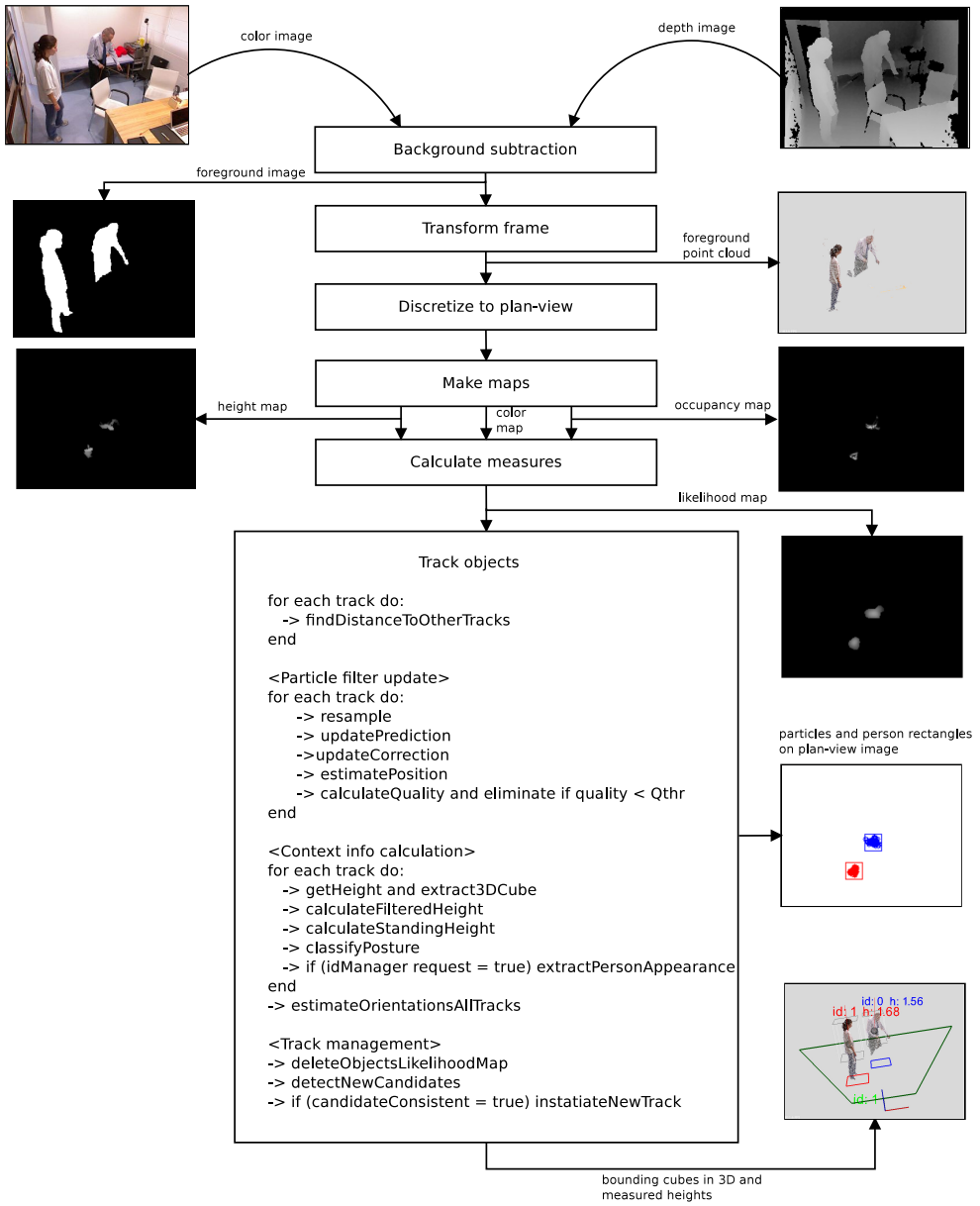


Figure 5.4: Main functions in multiple people tracking implementation.

ground subtraction we used the DECB algorithm that was previously explained in Section 5.1.1. Our implementation of the DECB algorithm uses the same parameters that were proposed by Fernandez-Sanchez et al. (2013): $\alpha = 0.75, \beta = 1.3, \varepsilon_1 = 10, \varepsilon_2 = 1.6\varepsilon_1, \alpha_d = 0.75, \beta_d = 1.25, t_{train} = 50$. The background subtraction process results in the foreground mask. The foreground mask is used on depth and color images for building a foreground point cloud in *camera_image* frame (${}^{ci}\mathcal{P}_{FG}$).

Next, the ${}^{ci}P_{FG}$ cloud gets transformed into the *camera_base* frame by applying the transformation ${}^{ci}_{cb}T$ obtained during the system setup (${}^{cb}\mathcal{P}_{FG} = {}^{ci}_{cb}T \cdot {}^{ci}\mathcal{P}_{FG}$). In our system, the *camera_base* coordinate frame replaces the *world* frame, which was introduced earlier during the general description of *plan-view* approach (see Figure 5.1). Compared to the *world* coordinate frame, the *camera_base* frame keeps the same orientation of the axes in relation to the grid cells, but its origin is translated directly under the origin of the *camera_image* frame (see Figure 5.5, next page). Thus, each camera has its own referent frame on the floor plane, where positions are tracked in the local coordinates. Also, by having many *camera_base* frames, the *world* frame can be given its real meaning in the context of multiple camera network, and proclaimed as the single referent frame for the whole system. It is then easy to change local position coordinates from different *camera_base* frames into the *world* coordinate system by applying a transformation ${}^{cb}_{w}T^{(j)}$ for each camera j. This transformation is a 2D transformation on the floor plane, which means that it is easily obtainable by measuring one metric distance and one rotation angle between the two frames.

The cloud ${}^{cb}\mathcal{P}_{FG}$ is subjected to the space discretization and 3D to *plan-view* conversion as described in Section 5.1.3. Prior to the discretization the cell grid has to be initialized. The parameters of cell grid define the size and resolution of the *plan-view* image. The cell size $\partial = 3$ cm recommended by Muñoz-Salinas (2008) is used. The width n and the height m of the grid are not defined as constant values, but they are calculated from metric values obtained by projecting the Kinect camera frustum on the floor plane. The frustum is defined with the distances of *close clip* (${}^{ci}Z_{close}$) and *far clip* (${}^{ci}Z_{far}$) planes and the field of view (FOV) of the Kinect camera determined by its horizontal and vertical viewing angle parameters. The frustum projection defines four coordinates ${}^{cb}X_{min}$, ${}^{cb}X_{max}$, ${}^{cb}Y_{min}$ and ${}^{cb}Y_{max}$, in *camera_base* frame that are used as the rectangular area limits (see Figure 5.5). With these coordinates the *plan-view* image is minimized to contain only the area of the scene in which the range sensor is actively used. Such parametric approach enables us to easily and efficiently use other types of depth sensors (that have a different FOV geometry) with the `Vision` node. That would only require a change of the horizontal and the vertical FOV angles in the camera model. The position of the Kinect and its rotation affect the coverage of the floor area and the image size. For example, we usually used a downward camera angle between 20° and 25° and the clipping plane distances ${}^{ci}Z_{close} = 0.7$ m and ${}^{cb}Z_{close} = 5.5$ m. With these parameters the area limit coordinates are: ${}^{cb}X_{min} = -3.3$ m, ${}^{cb}X_{max} = 3.3$ m, ${}^{cb}Y_{min} = 0.5$ m and ${}^{cb}Y_{max} = 5.5$ m. This set of coordinates with a 3 cm cell size transforms the observed floor area into a grid of dimensions $n \times m = 220 \times 167$.

We expanded multiple particle filters person tracking of Muñoz-Salinas (2008) with algo-

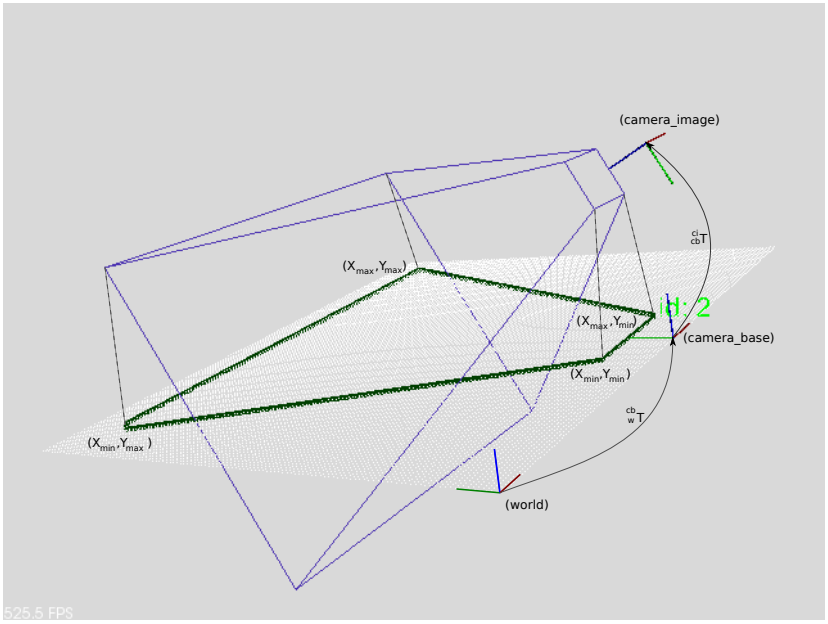
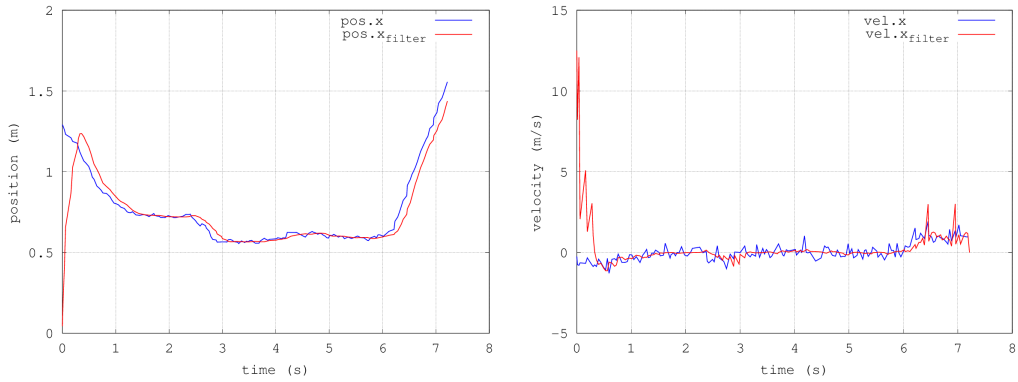


Figure 5.5: Camera frustum and active tracking area on the floor.

rithms that extract contextual and appearance data from each tracker. Person tracking starts with the update of the existing particle filters when new observation data is available. To achieve an update, the loop over the existing trackers needs to be executed twice. In the first pass, the distances between each tracker and all other trackers are calculated in order to get the necessary interaction factors $\mathcal{I}_{i,t}^{j}$. In the second pass, particle filters are updated through resampling, prediction and correction to obtain the mean position estimations $\epsilon_t^j, \forall j = 1, ..., P$. We used the number of the particles in the filter $N = 100$ and the rectangular person region of size $\zeta_R = 0.6$ m, the same as in the original work. The second image from the bottom on the right side on the Figure 5.4 visualizes the particles and bounding rectangles in *plan-view* for two particle filters that simultaneously track two persons. After the positions are estimated for all active trackers, we obtain the set of tracks with sufficient support in the observation data Z_t. This set of tracks has to be traversed once more in order to extract *TrackData* messages with updated position values for each track. Tracking and contextual information processing for a synchronized pair of input images is finalized with the PF management stage. During the management stage, new person candidates are detected and assigned to a PF that will track them. The condition for the addition of a new PF is that the appearance of the candidate has been sufficiently consistent for at least 3 frames in a row (based on histogram comparison).



(a) Position filtering.

(b) Velocity filtering.

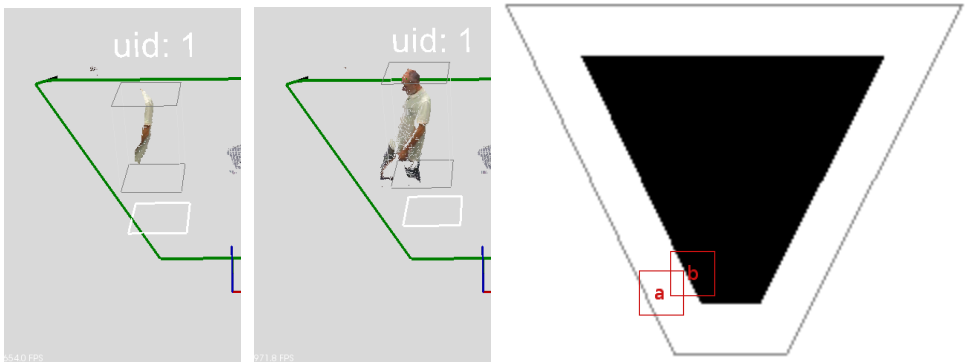
Figure 5.6: Influence of the 2^{nd} order lowpass Butherworth filter on position and velocity data.

5.2.3 Track Data Extraction

After a new position estimation ε_t is obtained for a person track, its 2D position coordinates ${}^{pv}x_t$ and ${}^{pv}y_t$ expressed as pixel values on the *plan-view* image are transformed into metric values of the *camera_base* frame $x_t = {}^{cb}x_t$ and $y_t = {}^{cb}y_t$. The height of the person h_t is read directly in meters from the measure $\mathcal{H}_{\mathcal{R}({}^{pv}x,\,{}^{pv}y)}$. The newly obtained position and height data require filtering before they can be applied in the subsequent algorithms. For that purpose we use a 2^{nd} order low-pass infinite impulse response (IIR) Butterworth filter. An appealing property of the Butterworth filter is the maximally flat frequency response in the passband - a range of frequencies that can pass through a filter without being attenuated. The cut-off frequency f_c defines the passband, as on that frequency the output starts to get attenuated with the slope of $20 \cdot n \cdot dB/decade$ on the logarithmic scale, where n is the order of the filter. The digital filter of the 2^{nd} order can be implemented with the following equation:

$$q_t = b_0 p_t + b_1 p_{t-1} + b_2 p_{t-2} - a_1 q_{t-1} - a_2 q_{t-2} \tag{5.12}$$

where q_t is the current output sample that we want to obtain, p_t the current input sample, and p_{t-1}, p_{t-2}, q_{t-1} the inputs and outputs from the previous iterations. Since it is necessary to wait 3 samples to determine the value of the current output sample, a small delay in the signal is obtained (ca. 100 ms). This delay is an acceptable trade-off for getting a filtered output signal with reduced noise as shown in Figure 5.6a. The filter coefficients b_0, b_1, b_2, a_0, a_1 are determined in dependency of the base sampling frequency f_s of image data and the cut-off frequency f_c. These parameters, along with the order of the filter, also implicitly define how long it will last the transient period at the beginning of the filter operation. For the low-pass filter with the parameters $fs = 30$ Hz and $fc = 2$ Hz which were used in our algorithms for position and height data filtering, the transient period t_{tr} takes around 500 ms. Figure 5.6a shows posi-



(a) Partial person point cloud.

(b) Complete person point cloud.

(c) Plan-view image with highlighted height update zone and rectangles for cases a) and b).

Figure 5.7: Height update.

tion measured along the coordinate axis ${}^{cb}X$ for one of the tracks. The transient that happens at the start of the tracking when a person enters the camera FOV is visible in the 1^{st} second. During the transient the filter produces a substantial error that influences the accuracy of the location dependent calculations in the system. To avoid this error, during the first 500 ms after the tracker initialization the original, unfiltered position and height data is copied at the output of the low-pass filter. After the filter transient has passed until track termination, the actual filtered data is used.

To calculate the derivative of the discrete position signal, we use the simple first order difference $\dot{x}_t = (x_t - x_{t-1})/\Delta t$. Figure 5.6b shows the influence the low-pass filtering of position has on the output of velocity calculation by first-order differencing. If we disregard the signal during the transient period of the filter, it is visible that the velocity calculated from the filtered position gives more accurate information about the movement of the person. For example, in the period between 2.5 s and 3.0 s in Figure 5.6a there is a change of position in the X coordinate from 0.75 m to 0.55 m. The velocity calculation that uses the low-pass filtered position data, manages to capture the general direction of the human movement and registers a negative velocity in the ${}^{cb}X$ coordinate, whereas during the same period the differentiation of the raw position signal produces a noisy velocity that has both positive and negative values.

Besides the estimation of the current height, the second type of the height-related calculations is done in order to estimate a person's standing height h_t^{st}. Standing height h_t^{st} is an important input for the height-based posture classification process (see Section 8.1). A simple approach towards the estimation of h_t^{st} is to calculate the average of the height values during the time when the person is standing. For the average value calculation, we use a computationally efficient implementation of the linear average filter based on the recurrence formulas introduced by Welford (1962). To provide the correct standing height h_t^{st}, the averaging filter needs to be initialized and updated with correct data. We assume that people will usually enter

the scene in the upright, standing posture. However, when people enter or exit the scene, their point clouds may be cut-off since they are crossing over the border of the depth sensor FOV. In this situation, the tracker will measure only a part of the real standing height (see Figure 5.7a). Thus, the calculation of the standing height might take in a wrong value. To prevent the update of the *standing height* averaging filter with incorrect height values, we built in two mechanisms in the algorithm:

Averaging sum control
: The averaging filter is updated in a different way depending on if there is an increase, or a decrease in the averaging sum. Assuming that the initial height during the first few frames after entering the scene still might be lower than the real height of the person, the algorithm allows the addition to the average sum of every new height value h_t bigger than the current standing height h_t^{st}. To prevent the change of sum when sitting down, the algorithm prevents adding heights that are smaller than the threshold $\vartheta_{hmin} = h_t^{st} - \delta h$, with δh set at 0.15 m.

Height update zone
: To demonstrate the concept, we will use Figure 5.7c. Gray lines mark the limits of the active tracking area of the camera. The black area in the middle is the *height update* zone in which the averaging filter updates are allowed due to the availability of complete point clouds. The zone between *height update* zone and scene limits is the *border* zone where averaging filter updates are not allowed. Red rectangles in Figure 5.7c show the approximations of track positions that correspond to the situations depicted in Figures 5.7a and 5.7b. In both cases the referent point for inclusion into filter update process is the center of the bounding rectangle.

5.2.4 Setup and Scene Visualization Requirements

The requirement of physical portability of the system depends on the ability to conduct fast sensor mounting. We acquired special hardware for temporary physical placement of cameras - extensible tripods up to 3 m of height with a spherical bearing on the top, on which a Kinect could be mounted. This specific hardware allows for setting all possible camera heights and viewing angles that might be needed in experiments.

The idea of a portable system allows a researcher to arrive in a clinic or a home of a patient, and need to spend only several minutes to set the physical position, tracker parameters and scene properties for each camera. After choosing areas in the patient's environment that will be covered by Kinects and setting optimal camera viewing angles, a researcher does the setup of the tracking software. Therefore, GUI application for vision tracking setup was designed and implemented as a part of the `Vision` node. An example of the application GUI is given in Figure 5.8.

For each stage during the setup, there is a corresponding tab on which the appropriate information can be visualized. When physically setting up a camera, live feeds with the color,

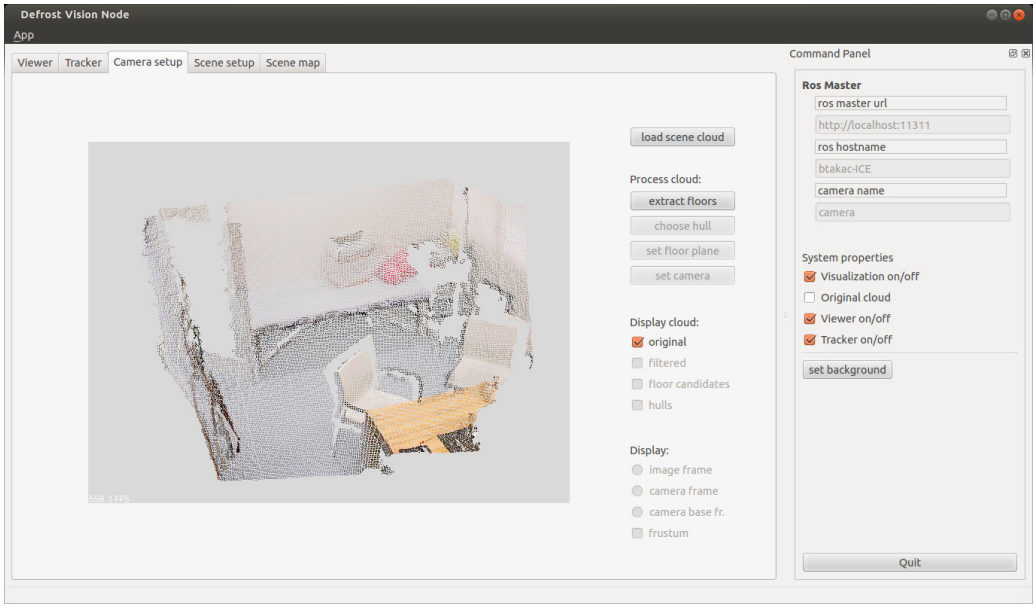


Figure 5.8: An example of GUI for camera setup in the `Vision` node application.

depth and foreground images, along with the foreground and background point clouds, can be observed on the *Viewer* tab. The *Tracker* tab visualizes the data produced by the trackers, such as bounding rectangles in images or bounding cubes in point clouds. On the same tab, the system operator can also observe in real-time the intermediate outputs of the tracking process, such as height and likelihood maps. The last three tabs are designed for work with camera in the offline mode. Upon saving the point cloud of the camera scene, the first step is to find the floor plane on the scene and recover the 3D pose of the camera in relation to the floor. The floor detection process is largely automated, requiring minimal human intervention (Section 5.2.5). The control over the stages of the floor detection process and its outcomes are visualized on the *Camera setup* tab (Figure 5.8). Afterwards, the background of the scene has to be mapped into a 2D map on which 2D contextual FOG zones can be added by manual editing (Section 5.2.6). This is done using tools on tabs *Scene setup* and *Scene map* tabs, respectively.

5.2.5 Floor Plane Detection and Camera Setup

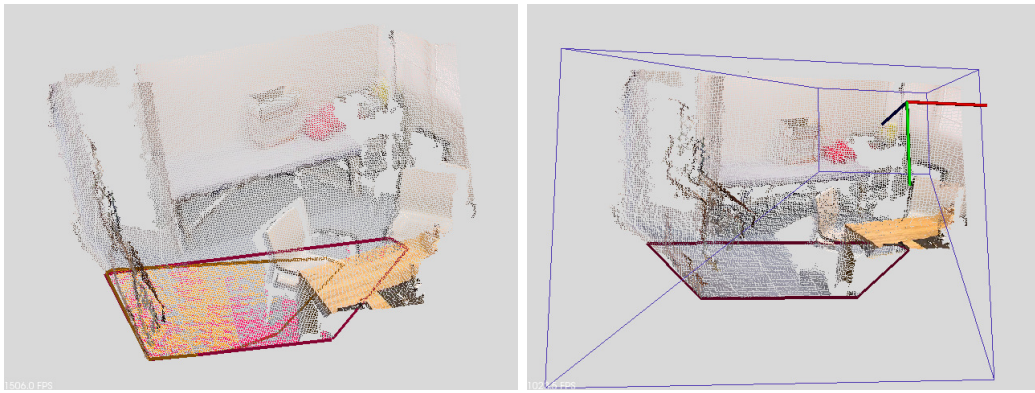
The process of camera setup allows defining the position of the camera in the space where it is installed, expressed as the transformation ${}^{ci}_{cb}T$ between the *camera_image* frame and the *camera_base* frame (see Figure 5.5). The input data is acquired in the *camera_image* frame, in the form of the point cloud of the scene background. Since the *camera_base* frame is supposed to span the floor plane with two of its axes (${}^{cb}X$ and ${}^{cb}Y$), the first task towards getting the desired frame transformation is to find the equation of the floor plane in the *camera_image* frame.

When the floor plane equation is defined, we can use it to have the origin of the *camera_image* coordinate frame and unit vectors along the axes ${}^{ci}X$ and ${}^{ci}Z$ orthogonally projected on it. The projected origin with each of the projected points makes a directional vector in the floor plane. These two directional vectors are the base of the new *camera_base* coordinate frame. The vector ${}^{ci}X$ projects into the vector ${}^{cb}X$ on the floor plane, while ${}^{ci}Z$ projects into ${}^{cb}Y$. The third vector of the frame has to be orthogonal ${}^{cb}Z$, so it is found by the vector product of the other two. When the three vectors of the new frame are defined, it is easy to obtain the transformation ${}^{cb}_{ci}T$. This transformation can then be inverted to find the final solution, the transformation ${}^{ci}_{cb}T$.

Finding the correct equation of the floor plane is the critical part of establishing the *camera_base* coordinate system. To find the equation of the plane from the set of 3D points, we use *RANdom SAmple Consensus* (RANSAC) method (Fischler and Bolles, 1981). The RANSAC method takes as input a point cloud, the parametric model of the sought geometric object and acceptable confidence measures. The method then tries to guess the model parameters that best fit the point cloud data.

The search for the floor plane in the indoor point cloud usually results not just with one, but with several appropriate plane models. Some of those models might correspond to the actual floor surface, but also there will be plane models that will correspond to any other flat surface on the camera scene, such as a wall or the top of a table. Hence, to speed up the floor detection process, it is necessary to minimize the number of potential floor plane solutions when using RANSAC method. To do so, we introduce additional constraints on the plane equation in the form of the distance of the plane from the *camera_image* frame origin and the minimum number of points that the plane should have. These constraints can be used only if two particular assumptions about the positioning of the Kinect are satisfied. The first assumption is about the height of the camera above the floor. We predict that Kinects will be mounted at the height between 2.1 m and 2.5 m, which means that these two heights are the minimum and the maximum distance from the potential floor plane. The second assumption is that Kinects will always observe the scene from the overhead position with a downwards angle. Such orientation angle should ensure that the floor has a plane model with the highest number of points, out of all plane models in the point cloud.

Since the spacing between points and the random error of measurements in Kinect become larger with the increased depth, the points of the floor close to the camera are very dense while those distant from the sensor become dispersed. With the increased distance from Kinect, the spacing between points and the random error of depth measurements become larger. Thus, the points of the floor close to the Kinect are very dense, while the points that are distant from it become dispersed. Even with the applied constraints, the RANSAC method might still give as output a few potential floor plane models, instead of only one. Potential plane models will all fit the same part of the point cloud that corresponds to the floor, but with a slight difference between their model coefficients. This exact fit depends on the distance threshold (d_{thr}) parameter in the RANSAC method. The d_{thr} is used to decide whether a certain point is an inlier or an outlier in relation to the plane model. With a very small d_{thr}, only points lying very close



(a) Two potential floor planes.

(b) Chosen floor plane and the corresponding camera frustum.

Figure 5.9: Floor detection and the setup process.

to the ideal mathematical plane will be considered inliers. Since the relevancy of a proposed model is evaluated based on the number of the points that is fit to it, in the case of a too small d_{thr} the number of points can be insufficient and the model deemed irrelevant. On the other hand, with a very big d_{thr} many points far from the ideal mathematical plane will be accepted. In this case, an incorrect plane model that visibly deviates from the actual floor surface can still have sufficient support in the observation data, and become accepted as the final solution.

Usually, when a moderate $d_{thr} = 0.01$ m is used together with the previously introduced constraints, two or three plane equations will be fit onto the floor point cloud data. This situation is visualized in Figure 5.9a, where we can see two hulls for two potential floor planes. The `Vision` node GUI enables the system operator to cycle through the plane hulls and choose the most appropriate plane. The camera setup steps that the system user should perform are visible at the example of GUI in Figure 5.8. First, the user loads the background point cloud (*load scene cloud*). The cloud can be inspected in the interactive 3D point cloud visualizer. Then, floor detection process is started with the *detect floors* button. The button *choose hull* cycles through the detected floor planes hulls, while the *set floor plane* button chooses the plane model for the currently active (highlighted) hull, and saves its coefficients into the setup file. Ultimately, the user selects the *set camera* button that calculates the camera frustum limits, finds the frame transformation ${}^{cb}_{ci}T$, and defines the active tracking zone (see Figure 5.9b).

5.2.6 2D Mapping and Scene Setup

After setting the camera, we need to model the scene in front of the camera in the format that is appropriate for the usage within the computer system. In Section 4.1.1 we introduced the idea of semantic maps that use different types of FOG zones to relate the patient's location with the FOG probability.

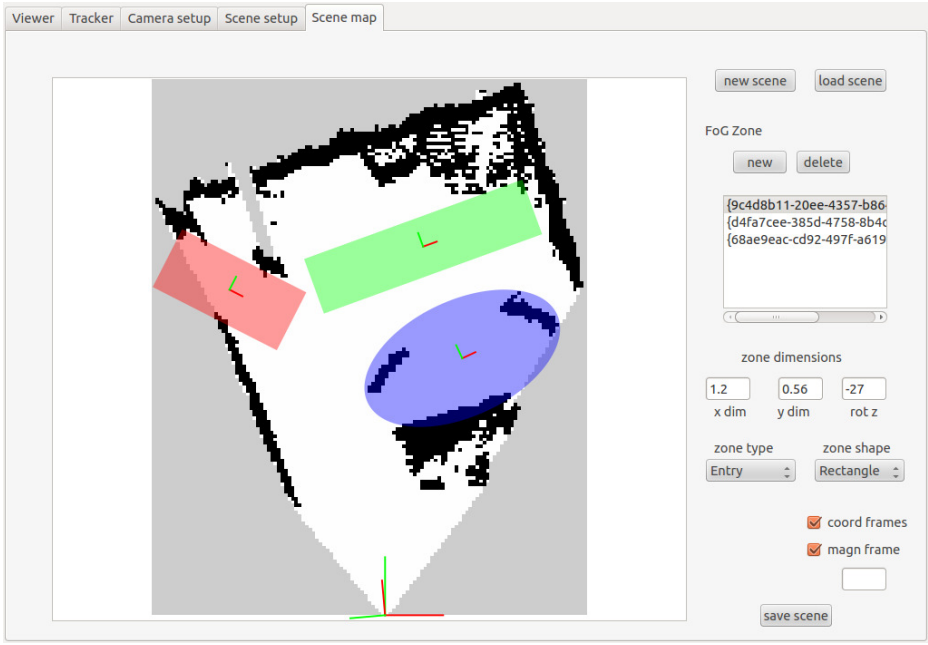


Figure 5.10: An example of scene editing in the `Vision` node Qt application.

To obtain semantic maps during the prototype and development stage, the easiest and fastest way is to produce them manually. There are two steps in obtaining the semantic map: 1) getting a *base map* which shows empty and taken space in front of the camera in the form of the 2D bitmap; and 2) imposing 2D FOG zones of different types, sizes and orientations on the new bitmap. During this last step, we should have in mind the exact metric relations between the real world and the map.

The 2D bitmap is obtained by using the *occupancy grid*. As a mapping approach, the *occupancy grid* was first introduced in the robotic community during the 1980-ies for mobile robot perception and navigation (Moravec, 1988; Elfes, 1989). When the *occupancy grid* is used for the mobile robot navigation, each grid cell (x, y) in the map has an occupancy value which measures the subjective belief whether, or not, the center of the robot can be moved to the center of that cell (Thrun and Bü, 1996). The *occupancy grid* based map is built on the probabilistic principles from several consecutive observations over time. By doing so it takes into account sensor noise and potential robot movement. In our application, the use of a static camera eliminates the need to account for potential movement, but the depth sensor uncertainties still persist. We use the advantages of working with the ROS framework, to easily get high quality *occupancy grid* maps. The ROS *octomap_server* package (Hornung et al., 2013) allows a volumetric 3D *occupancy grid* map to be incrementally built from the incoming range data (formatted as 3D point clouds). Occupancy map in 2D is obtained from the 3D volumetric map by a simple down-projection. For each `Vision` node there is one `Map_Server`

node. The nodes communicate with each other via input topic */camera/map_cloud* and output topic */map_server/projected_map*. Input data for mapping is the point cloud of the scene background transformed from the *camera_image* frame into the *camera_base* frame. Additionally, we apply a height threshold filter on the transformed cloud, so that it contains only the points higher than 0.2 m from the floor. Two of the main parameters for `Map_Server` are *grid resolution* and *maximum depth range*. Usually, these parameters are set at 3 cm and 5.5 m, respectively.

Figure 5.10 shows a simple editor that is used to set FOG zones. The button *new scene* loads the 2D background bitmap obtained from the map server. Multiple FOG zones can be added, deleted and changed via editor. The basic properties of a FOG zone are its geometric shape (*rectangle*, *ellipse*, *polygon*) and its type (*Entry*, *FoG_Hesitation*, *FoG_Cluter*, *FoG_Turning*). The orientation of a zone towards the *camera_base* frame can be manually set by inputting the angle in degrees. The example camera scene displayed in Figure 5.10 has one *entry* zone at the door (red), one *clutter* zone between two chairs and the table (blue), and one *starting hesitation* zone next to the bed (green).

5.3 Summary

In this chapter we described how to solve the problem of people localization in a distributed monitoring system which uses Kinect sensors. We built a GUI application, called the `Vision` node in order to satisfy the following functional requirements:

- Multiple people tracking;
- Extraction of the relevant contextual data from person tracks;
- Easy setup of camera hardware;
- Facilitation of environment mapping; and
- Real-time operation.

The development process started with the research about the main theoretical concepts and state-of-the-art methods for people tracking. The goal was to find an already existing algorithm that can use the depth data for reliable people tracking. The bulk of work presented in this chapter was software system engineering, that involved the implementation of the chosen algorithm for multiple person tracking, and the additional algorithms and tools for data filtering, data recording, data visualization and system setup manipulation.

We used a well known *plan-view* representation and occupancy-based grid maps to provide the basis for the implementation of the tracking and mapping functionality. The motivation for using the *plan-view* approach was to annulate the deficiencies that Kinect's depth sensor has at the larger distances (over 3.5 m from the camera), and to use its depth data with the same confidence on the whole camera scene. The position tracking accuracy of the implemented

multiple-person tracker that uses *plan-view* approach on Kinect data will be examined in the Chapter 6.

The original person tracking algorithm was expanded with the functions for extracting contextual data necessary for orientation estimation and activity recognition. Contextual data shared with the other nodes in the distributed monitoring system includes person's position, velocity, height, orientation and posture. There is an existing long history of using occupancy grids for tasks in robotics. Since we use ROS framework, scene mapping was simplified with the usage of a proven state-of-the-art robotic mapping algorithm (Hornung et al., 2013). This reduced the need for implementation of extensive code in the `Vision` node. We only had to implement the code that provides the correct background point cloud to `Map_Server` node and saves returned bitmaps. Our experience of the easy and fast system setup, gained while collecting the PD patient data from different clinical and home settings, showed that the effort put into the design and the implementation of the visualization and setup algorithms paid off.

6

Orientation Tracking and Two-dimensional Pose Estimation

In the conceptual stage of the design of our monitoring system, 2D pose of the patient was recognized as the necessary data to be delivered. In the previous chapter (Chapter 5) we presented how to track positions of multiple people in front of one camera. In this chapter, we will focus on achieving the solution for tracking the orientation of one, specific person. According to the concept of the system, the necessary data for the estimation of the orientation is provided by a smartphone that is supposed to be worn by the patient as the main gait sensor. The combination of the accelerometer, gyroscope and magnetometer signals in modern smartphones already allows the calculation of what is called the *absolute* 3D orientation. The *absolute* 3D orientation of the device is the one which uses the angle measurements in the coordinates of the axes of the global Earth coordinate system. As the main referent axes in this system there are the axes along the direction of gravity field and in direction of the magnetic North. The third axis that completes the system is then, naturally, defined by the Cartesian product of the first two axes.

The global absolute system is very good for usage in the smartphone when the device is used for navigation purposes and people can use the display, and in that way by themselves relate the orientation of the smartphone in global coordinates to the space surrounding them. However, for the use in an autonomous monitoring system it is not possible to rely on human intuition

Parts of this chapter appear in (Takač et al., 2013)

for such process. There are two reasons why we have to use a different method for orientation estimation:

Local coordinate system
: The estimation of the patient's orientation is needed in the distributed system only when the user is viewed by any of RGB-D cameras. Each camera in the system has its own coordinate system. Therefore, the orientation of the user at the given moment needs to be expressed as the angle in the coordinate system of the camera which performs the tracking, instead of being expressed in the global magnetic-North referenced world frame.

Person vs. device orientation
: It is necessary to strictly differentiate between the orientation of the inertial device (smartphone) and the orientation of the user, since they cannot be considered equal. When the inertial device estimating orientation in reference to the global frame is fixed on the body of the user, its 3D pose in reference to the user's body must be exactly known for the system to correctly calculate the user's orientation in the global frame of reference. In a real-world, everyday scenario, the autonomous system has no means to exactly know how and where the smartphone is fixed on the user. It is just presumed by convention that the smartphone will be fixed on the user at the expected, previously set, position and orientation. Still, the uncertainty about the current pose of the sensor will always exist. A smartphone is usually fixed as a sensing device on the patient by being placed in a horizontal belt case or an elastic strap around the waist. Even if the device was ideally positioned at the beginning of the day, due to postural changes and other trunk movements, it is possible that its position will change during the day by rotating for some angle in the plane around the waist.

The focus of our work in people orientation estimation that will be explained in this chapter is not on the development of new fusion algorithms for inertial devices, but it is on the development of methods that enable the existing inertial fusion orientation algorithms to be used for sensing the orientation of the people in reference to the coordinate frames in our distributed system. We present two methods that were developed and evaluated.

6.1 Two Methods for Orientation Tracking

We have developed two methods for transforming the orientation of the inertial device into the 2D heading of the user expressed in the referent camera coordinate system. In our methods, we use a very good and proven device orientation estimation algorithm introduced by Madgwick et al. (2011). The algorithm uses numerical integration of the orientation data in the quaternion representation. There are two versions of the algorithm depending on the number and the type

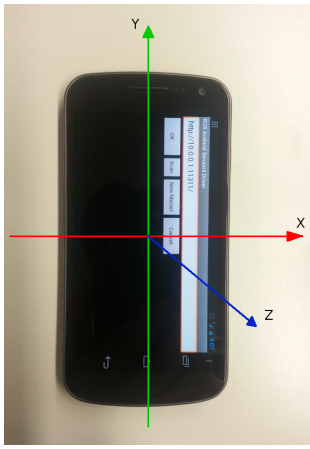
of sensors available in the inertial sensor system where it is applied. The basic version of the algorithm is suitable for IMU devices consisting only of gyroscopes and accelerometers, enabling the tracking of rotational and translational movement. This basic version of the algorithm uses gradient descent optimization, which makes it possible to obtain the relative orientation of the device towards the gravity field based on accelerometer input. When referring to this version of the algorithm in the rest of the paper, we will use the name Gravity Relative Orientation Estimation (GROE) algorithm. This basic algorithm is not able to give absolute 3D orientation, since there is no absolute reference in the plane perpendicular to the gravity vector. To achieve complete measurement of 3D orientation in the gravity field - Earth's magnetic North reference system, it is necessary to have the ability to sense the Earth's magnetic field. The MARG (Magnetic, Angular rate and Gravity) sensor is an extension of IMU which also incorporates a tri-axis magnetometer. An extended version of the algorithm that can be applied on MARG sensory platform computes its result by numerically integrating changes of orientation measured by gyroscopes, and then correcting gyroscopic measurement errors using a compensation component obtained from the combination of accelerometer and magnetometer measurements. The gradient descent algorithm that uses the combination of accelerometer and magnetometer data takes care of achieving absolute 3D orientation in several iterations after the algorithm initialization. We will refer to this version of the algorithm as the Absolute Orientation Estimation (AOE) algorithm. Both versions of the algorithm are stable, computationally inexpensive and effective at low sampling rates.

6.1.1 Methodi: Using Solely Wearable Inertial Sensor Data

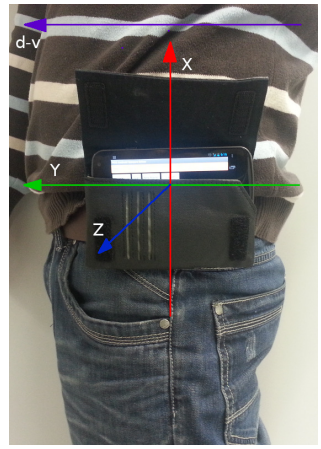
The first method we developed for person orientation estimation uses data only from wearable inertial sensor. The method employs AOE algorithm to obtain absolute 3D orientation of the device and relies on the following three assumptions: 1) the sensor device is worn in the predetermined orientation and at the predetermined position relative to the body of the user, 2) the heading is estimated only when the user is standing, and 3) the angle between the magnetic North frame and ground camera frame is known in advance.

We defined the user's orientation as a vector along his dorsoventral axis with the direction from the dorsal to the ventral side of the body. As the predetermined position for placing the smartphone, we chose the left hip. As the reference coordinate system orientation for the smartphone, we set the X-axis facing upward along the anteroposterior axis of the body, the Y-axis parallel to dorsoventral axis, and the Z-axis facing left from the body along the left-right axis. Expected smartphone positioning is depicted at Figure 6.1b.

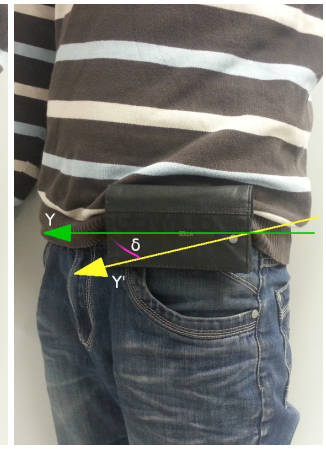
When the smartphone is in the expected ideal position and orientation on the user's body, the vector of gravity will be along its negative X-axis, while Y-axis and Z-axis define the plane parallel with the floor (see Figure 6.1b). Thus, we can obtain the 2D heading of the device in the floor plane by measuring the angle between the Y-axis of the smartphone and the axis of magnetic North (α) with AOE algorithm. Since there is no difference between the presumed



(a) Smartphone reference axes.



(b) Smartphone in the correct predetermined orientation at the expected position and orientation on the waist.



(c) Smartphone in the non-expected position and orientation on the waist. There is an angle of error in the transverse body plane between the device's real (green arrow) and expected (yellow arrow) orientation.

Figure 6.1: Frame definitions for orientation estimation.

direction of the Y-axis of the smartphone and the user's heading vector ($\partial = 0$), angle α also gives the heading of the user in reference to the magnetic North, as shown in Figure 6.2.

Our final goal is to obtain the heading of the user in the camera frame (ϑ). Two corrections with known static angle values are necessary. To get the user's heading ϑ, first the measurement of the smartphone (α) is corrected for angle (ψ) between the Y_c-axis of the camera coordinate system and the Y_m-axis of pointing to magnetic North. This gives angle φ, which defines the user's heading in reference to the Y_c-axis of the camera coordinate frame. Since user's heading ϑ is always expressed as the angle towards X_c-axis, a final correction is executed by adding 90° to angle φ.

6.1.2 Method2: Combining Wearable Inertial Sensor and Vision Tracking Data

Our second person orientation estimation method uses wearable inertial sensor data in combination with the classification of the person's orientation conducted in the vision tracking system. The goal of the method is to eliminate the set of assumptions used in the first method, making it more robust and applicable for use in uncontrolled home environments. The method uses the previously-introduced GROE algorithm, which estimates the 3D orientation of the device relative only to gravity. As the algorithm can align just two of the inertial device's axes

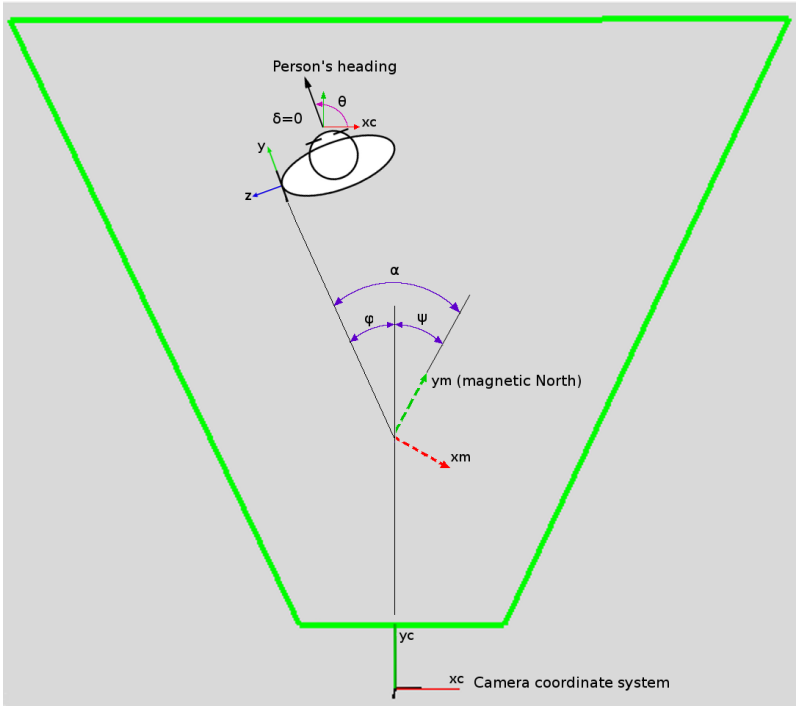


Figure 6.2: Overhead view of relations between the frames in the system.

with the plane perpendicular to the gravity (presumed floor plane), this leaves the final angle of the device unknown. To calculate the device's heading in the floor plane an external reference angle is needed. If, instead of the gravity-magnetic North, we use as the referent frame for the external reference angle the frame in which the camera is currently tracking the user's position, we can eliminate the need for finding the angle between the camera tracking frame and the gravity-magnetic North frame. Furthermore, the assumption of having the wearable sensor in the predetermined position can be eliminated if the external heading reference angle given to the inertial sensor contains information about the true heading of the user expressed in the common frame of reference. Providing the necessary external heading reference is therefore the task of the vision tracking system, because of its ability to observe the user directly in the camera reference system.

The implemented vision-based orientation classifier was inspired by the work of Harville and Li (2004), where the person's plan-view height templates are used to classify eight different headings in the range between 0° and 360° with a 45° resolution for humans standing upright (see Figure 6.3). Our neural network classification algorithm was trained with the features of 4 persons of different heights. To achieve uniformity of the visual orientation detection in the whole area covered by one camera, training data was collected from people standing at different distances and positions in relation to the camera. The positions for data collection were set

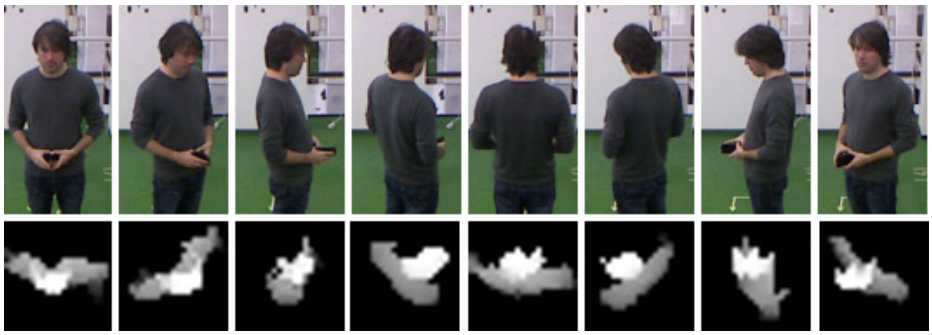
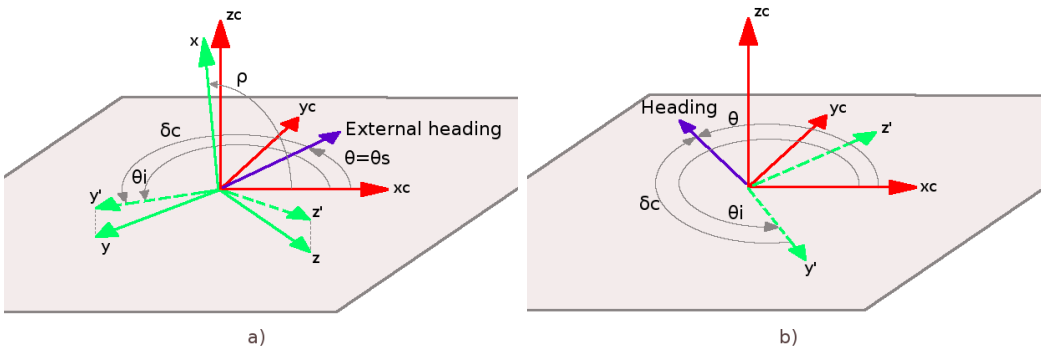


Figure 6.3: Patterns for neural network training. The top row shows eight headings for one person at the same position in reference to the camera. The bottom row contains examples of related height templates used in orientation classification with neural network.

using a grid of 0.5 × 0.5 meter rectangles on the floor. People were asked to move horizontally, vertically and diagonally on the grid, akin to pieces in chess, and to stop in the middle of each rectangle of the grid for one second. During post-processing, a total of 6022 height templates for 4 persons were extracted and labeled with their pertaining classes. The feature vector for classification consists of 443 attributes, the first 441 being normalized pixel values coming from the 21x21 pixel height image template, and the last two being height normalization constant and the number of non-zero elements in the template image. The neural network has an input layer with 443 neurons, a hidden layer with 25 neurons and an output layer with 8 neurons. Classic back-propagation training algorithm with symmetric sigmoid activation function was utilized.

The classification accuracy test on 100 height templates gave 92% correct classifications. During testing under real-world circumstances errors were noticed in classification between opposite directions, and also in classification of body poses which differ too much from upright standing. This has been noted as the source of the possible error in the heading reference.

When the classifier proposes the orientation reference for wearable system, its accuracy needs to be ensured. A high confidence level for the heading reference can be achieved with the use of two additional sources of information: the quality score of the classification result, and the position history of the person. The quality score of the classification result is calculated using values at output neurons. A neural network for 8 classes has eight output neurons, and the rule is that the output class of the whole classifier is assigned to the neuron with the maximum probability. The output neuron with the maximum probability has a high value when the user's height template is similar to a training template. This probability number can be used as the quality indicator for classification. A high confidence level using the classification quality score is achieved through the temporal process, n which the classifier output is tracked for consistency to be above a certain threshold during several consecutive frames. When this consistency holds, the orientation angle represented by the class can be taken as the person's heading proposition. We call this angle *static heading*. To further strengthen the heading proposition and minimize the probability of assigning the opposite direction, the kinematic properties of



(a) The moment in time when there exists external heading reference.

(b) Using the calculated correction angle to get the person's heading during periods in which only inertial orientation estimation is available.

Figure 6.4: Coordinate frames in the process of fusion of vision and inertial information for orientation estimation.

the person's track are used. Using position history, the velocity vector for the tracked 2D point is calculated. This vector in relation to X_c-axis of the referent coordinate frame gives the angle called *dynamic heading*. Ultimately, when the angular difference between the *static heading* and *dynamic heading* is inside a specified error boundary (i.e. +/- 15°) for 3 consecutive image frames, the static heading is confirmed to be the external heading reference for inertial system.

When the person is upright and wears the smartphone in the belt case, one of the axes of the device points approximately along the gravity vector, while the other two axes span the plane which is almost parallel with the floor. This can be observed on Figure 6.1b, where the X-axis of the smartphone is pointing upwards and axes Y and Z are forming the specified "almost parallel" plane. Since the GROE algorithm estimates the angle of orientation of the smartphone towards the gravity, it measures how much the plane formed by Y and Z axes is deviating from being fully parallel with the floor plane. This angle can be used to calculate the projection of Y and Z axes on the floor plane. Axes Y' and Z' shown in Figure 6.4-a are the result of such projection.

The external heading reference angle ϑ_s is not always available, but only when the vision tracker has a heading proposition of sufficient quality. When the external heading reference angle ϑ_s is known, it is possible to calculate the value of the correction angle δ_c between the external heading reference vector and the referent orientation axis of the inertial sensor system. In Figures 6.1b and 6.1c, the Y-axis is set closer to the user's dorsoventral axis, so we choose its projection Y' to be the referent orientation axis for the fusion. Figure 6.4a shows the relations between the $X - Y - Z$ coordinate frame of the smartphone, the $X_c - Y_c - Z_c$ coordinate frame of the camera and the linking $Z_c - Y' - Z'$ frame used for the fusion at the moment in time when the static heading is known. Correction angle δ_c is calculated as the difference

between the Y' axis angle ϑ_i and the external heading reference angle ϑ_s, which at that moment also represents the person's true orientation ϑ.

In the subsequent frames when no external heading reference is available and there is the dependency only on the inertial system orientation estimation, angle δ_c is subtracted from the observed angle ϑ_i to get the person's true heading ϑ. This is demonstrated in Figure 6.4b

6.2 Orientation and Position Tracking Evaluation

The purpose of the experiment was to confirm the functionality of the position and orientation tracking system for different users, and to collect sufficient data for the statistical analysis of the system accuracy. Additionally, we wanted to show that the user's position can constantly be estimated within certain statistical error limits irrespective of his distance from the camera and his orientation. We chose the approach with the known static ground truths for position and orientation to enable an evaluation based on comparing with known referent values. The smartphone position on the waist of the participant was taken as a parameter in this experiment with the objective of assessing how each of the two heading estimation methods adapts to a change in the sensor attachment position.

6.2.1 Experiment

The experiment had 12 participants (9 male, 3 female), who were recruited from among the staff and graduate students of the Industrial Design Department of Eindhoven University of Technology. The average height of the participants was $[\mu = 174.2, \sigma = 8.8]$ cm. None of the participants had gait problems. The area used for walking had dimensions 8 x 5 meters, and it was covered with a green carpet which had a visible grid of squares of size 0.5 $\times$ 0.5 m. Two Kinect devices were put at a height of 2.25 m facing downwards with a pitch angle of approximately 25°. The devices were placed to cover the walking area in a non-overlapping manner. A unique world frame for the experiment was set at the corner of the walking area, with its orientation equal to the base frame orientation of Kinect1. To confirm the uniformity of the magnetic field in the walking area, we executed control measurements of its quality at approximate waist height ($h = 1.0$ m) before and after the experiment.

On the green carpet surface, markers were placed to indicate points on the floor where the participants are supposed to stop in predefined orientations (see Figure 6.5). For each designated pose, two parallel lines of 0.5 m length were put on the floor at the mutual distance of 0.25 m. As the reference for measuring the marker position, the center point between two lines was taken.

The experimental condition was the sensor attachment position with two possibilities, Position1 with the smartphone fixed at the iliac crest on the left hip (see Figure 6.1-b) and Position2 with the smartphone rotated between 50° and 60° around the waist and put on the frontal left side under the belly (see Figure 6.1c. Position1 is the expected sensor position for the method

using the AOE algorithm, while Position2 is substantially deviating from the expected position for the same method. The second method using the GROE algorithm and video orientation classifier has no expected sensor position. The test for each sensor position was split into two walks, one walk with predominantly left turns (see AppendixB, Figure B.1) and the other one with predominantly right turns (see AppendixB, Figure B.2). Walks were designed with multiple consecutive turns in the same direction in order to induce possible orientation bias. Participants were instructed to walk to each marked position, where they were told to stand still for 3 seconds before continuing towards the next marked point. The procedure was repeated for each subsequent point. Each test walk lasted around one minute. Each participant first did two walks for Position1, followed by two walks for Position2.

During the experiment, color images and depth data of each Kinect were recorded along with the data from the smartphone which encompassed raw accelerations, orientation, magnetometer measurements and calculated orientations for GROE and AOE algorithms. Estimations of the positions obtained from the video tracking algorithm along with the absolute heading estimation angle for the two orientation estimation methods were stored in a SQL database. Post-processing consisted of annotation of frames when participants were standing still on the marked floor positions and calculation of the average position coordinates and heading angles from sensor data. A video segment of around one second was extracted each time a

Figure 6.5: The experiment venue. Markers on the floor indicate start and end points and numbered reference points for standing in a predefined orientation. Additional markers also show which part of the area is covered by which Kinect device.

participant stood still at a reference point.

The vision-based position tracking algorithm gives a new estimation of the position for each frame. With a 30 Hz frame rate, approximately 30 position estimations were available to calculate the average value of the X and Y coordinates during a one second video. Average values with a sufficiently small standard deviation ($\sigma < 0.04$) were taken as the measured position coordinates. In total, 288 pairs of position coordinates were obtained (12 participants $\times$ 12 reference points $\times$ 2 sensor attachment positions). The average value of the heading angle was calculated using temporal alignment of inertial signals with video segments. For the first method using AOE approximately 80-100 orientation estimation values were extracted for each 1 second video segment to calculate average angle value. In total, 288 average angle values were calculated. For the second method using GROE, the combination of vision-based orientation classification information and smartphone inertial information was collected at the smallest common denominator update rate, which is the rate of the video tracking algorithm. Around 30 orientation estimates were produced each time a person stood on a reference point. The total of 288 average angle values was expected, but orientation was not registered due to an algorithm failure, in 11 out of 288 cases.

6.2.2 Results

Calculated position values from all test walks were aggregated on a per point basis to enable comparison with reference values. Statistical results (see Table 6.1) include average value, average error and root-mean-square error (RMSE) for each of the two position coordinates at each stopping point. Under the presumption of normal distribution, the average error value is an indicator of the presence of a bias in the measurement. In our experiments, the overall randomness of the error values does not point to any significant positive or negative bias, or bias in any of the coordinates. The RMSE, which is a good measure of accuracy, indicates that the estimated position was on average in the majority of points 0.16 m or less from the true position.

The results of the estimation of person orientation closest to the ground truth were expected for tests with the sensor in Position1 when all assumptions needed to get the correct result were satisfied. The results for Method1 (AOE algorithm) with the smartphone in Position1 are reported in Table 6.2. The average angle value for a stopping point (each row in Table 6.2) was calculated from the set of direction angles estimated for each of the 12 participants. The average angle was compared with the point's reference angle value to give the average error and RMSE. We also observed the maximal error, by extracting the angle value for the case when the participant's orientation was furthest away from the ground truth. The average error values do not point to the existence of any specific bias in angle measurement. We took the highest observed value of the RMSE as the reference for error. Statistically, an average error of 17° can be expected if the initially assumed conditions about smartphone placement and upright walking posture hold.

Table 6.1: Statistical results for position measurements of reference points.

Point ID	Coordinate	Ref. value [m]	Avg. value [m]	Mean error [m]	RMSE [m]
1	x	2.25	2.21	-0.04	0.07
	y	1.75	1.74	-0.01	0.06
2	x	3.25	3.14	-0.11	0.16
	y	1.75	1.66	-0.09	0.13
3	x	5.75	5.65	-0.10	0.15
	y	3.00	3.02	0.02	0.05
4	x	3.25	3.23	-0.02	0.09
	y	4.25	4.19	-0.06	0.10
5	x	1.25	1.22	-0.03	0.06
	y	4.25	4.31	0.06	0.10
6	x	1.75	1.64	-0.11	0.16
	y	2.75	2.77	0.02	0.06
7	x	6.25	6.20	-0.05	0.07
	y	1.75	1.89	0.14	0.20
8	x	4.75	4.79	0.04	0.08
	y	1.75	1.93	0.18	0.25
9	x	1.75	1.73	-0.02	0.07
	y	2.75	2.72	-0.03	0.06
10	x	1.25	1.14	-0.11	0.16
	y	4.25	4.21	-0.04	0.08
11	x	3.25	3.17	-0.08	0.13
	y	4.25	4.19	-0.06	0.10
12	x	6.75	6.71	-0.04	0.08
	y	2.25	2.27	0.02	0.06

Ref. - Referent, *Avg.* - Average

Table 6.2: Statistical results aggregated per marker point for person orientation estimation method using AOE algorithm (Method1) with the smartphone on the hip (Position1).

Point ID	Ref. angle [°]	Avg. angle [°]	Avg. error [°]	RMSE [°]	Max. error [°]
1	270	278	8	11	24
2	0	-2	-2	8	20
3	30	37	7	13	24
4	180	181	1	7	13
5	225	231	6	10	19
6	330	333	3	7	13
7	270	269	-1	10	26
8	180	181	1	9	16
9	150	150	0	9	17
10	45	41	-4	12	22
11	0	-8	-8	13	23
12	330	331	1	12	22

The results of the estimation of person orientation closest to the ground truth were expected for tests with the sensor in Position1 when all assumptions needed to get the correct result were satisfied. The results for Method1 (AOE algorithm) with the smartphone in Position1 are reported in Table 6.2. The average angle value for a stopping point (each row in Table 6.2) was calculated from the set of direction angles estimated for each of the 12 participants. The average angle was compared with the point's reference angle value to give the average error and RMSE. We also observed the maximal error, by extracting the angle value for the case when the participant's orientation was furthest away from the ground truth. The average error values do not point to the existence of any specific bias in angle measurement. We took the highest observed value of the RMSE as the reference for error. Statistically, an average error of 17° can be expected if the initially assumed conditions about smartphone placement and upright walking posture hold.

Table 6.3 provides the data for the comparison of the two different smartphone attachment positions when Method1 (AOE algorithm) was used. The data in the table was obtained by aggregating on a per participant basis. This means that to get the data of one row in the table statistics were based on a set of 12 different orientations calculated for the stops of one person. The most notable observation is the uniformly negative angle of the average orientation error obtained for Position2. This negative angle is anticipated considering the orientation change of the smartphone performed for the tests with Position2. The average error values in each row of Table 6.3 indicate how much the smartphone was rotated around the anteroposterior axis for each participant. Negative angle values of the average error for Position2 closely match values of the RMSE.

Evaluation results of the person orientation using Method2 (see Table 6.4) are similar to

Table 6.3: Statistical results aggregated per participant for the orientation estimation method using AOE algorithm (Method1) with two sensor attachment positions.

	Position1		Position2	
Participant	Avg. error [°]	RMSE [°]	Avg. error [°]	RMSE [°]
1	-8	9	-66	66
2	-7	13	-41	42
3	3	8	-60	62
4	3	5	-60	60
5	3	5	-43	43
6	7	8	-55	55
7	-8	8	-62	63
8	-6	14	-57	57
9	8	8	-50	50
10	11	11	-47	47
11	13	15	-58	58
12	5	7	-39	40

Table 6.4: Statistical results aggregated per marker point for orientation estimation using vision based classification and the GROE algorithm (Method2) with the smartphone on the hip (Position1).

Point ID	Ref. angle [°]	Avg. angle [°]	Avg. error [°]	RMSE [°]	Max. error [°]
1	270	276	6	15	47
2	0	2	2	15	44
3	30	50	20	21	32
4	180	188	8	10	15
5	225	236	11	17	37
6	330	334	4	13	33
7	270	272	2	14	27
8	180	187	7	16	35
9	150	143	-7	24	32
10	45	40	-5	17	32
11	0	-6	-6	13	22
12	330	313	-17	18	28

Table 6.5: Statistical results aggregated per participant for the orientation estimation method using AOE algorithm (Method1) with two sensor attachment positions.

	Position1		Position2	
Participant	Avg. error [°]	RMSE [°]	Avg. error [°]	RMSE [°]
1	11	28	5	13
2	-2	19	3	14
3	-4	20	4	21
4	10	14	13	22
5	-3	15	4	14
6	4	17	6	16
7	0	13	1	14
8	4	12	11	17
9	-3	12	0	13
10	0	13	-6	18
11	0	14	13	12
12	8	12	9	16

those achieved with Method1 (see Table 6.2), with the exception of bigger RMSE values maximum errors, which indicate worse behaviour of Method2 at certain moments.

Our expectation is that Method2 is able to compensate for the unknown orientation change of the attachment point of the smartphone. The adaptive nature of the method is visible in Table 6.5 from the fact that there is no significant difference in the observed average errors and RMSE between the two attachment positions.

6.2.3 Discussion

The final goal of the experimental measurements of the position orientation tracking subsystem is to properly model its output as a virtual sensor that senses 2D poses and has known characteristics in terms of accuracy.

The position estimation errors in Table 6.1 have two principal sources. The first source is the tracking algorithm based on the noisy depth sensor data. The second source is the random nature in which participants arrived at marked points, since during the experiment they were allowed to stop anywhere along the 0.5 m marker line inside a target square. With the current experimental design it is impossible to separate the contribution of each source to the obtained position errors, so we will impose a strict rule and assign the whole error to the tracking algorithm.

The RMSE is equal or less to 0.16 m for all the measurement points in Table 6.1, except for points 7 and 8. A greater error in these points can be explained by the combination of body position, camera placement, and depth sensor characteristics. When a person is sensed by a

depth camera, depth measurements are taken only on the side of the body directly exposed to the camera. Close to the camera in the overhead position (points 1 and 2), a depth sensor will collect more 3D points from the head and upper shoulder, which are the parts closer to the vertical body center. At the middle distances (2 m to 4 m) from the camera (points 7 and 8), the depth sensor will collect the majority of points from the exposed side of the body. In point 7 this part of the body is at the back, and in point 8 at the right side of the body. This anomaly happens only when people are exposed to the camera under orientation angles close to 0°, 90°, 180°, 270° and 360°. When a person is oriented diagonally towards camera more depth points are taken from the body center. At the bigger distances (after 4 m), depth sensor noise and smaller occupancy values influence the tracking algorithm to give more significance to height values, estimating a position more towards the true center of the person.

The comparison of the average orientation errors for the same points across Tables 6.2 and 6.4 implies that there was no significant magnetically-caused bias at any marker position. The accuracy comparison based on the maximum RMSE and maximum errors in the same tables reveals that the first method with AOE algorithm performed better when the sensor was in Position1. The RMSE values, and especially the maximum error values presented in Table 6.4, indicate that Method2 in its current implementation under-performs in terms of accuracy. The cause for this is incorrect static orientation (45° left or right from true value) registered as the external heading reference at certain moments. This could be improved by decreasing the allowed angle error between static and dynamic headings. However, this decrease in error angle can prolong the time necessary to fulfil conditions for registering the external heading reference after entering the camera scene. With the current setup, the detection time for the external reference of a person's heading can sometimes be delayed for one second, depending on how close the trajectory of the movement is aligned with the eight principal orientations of the classifier. This delay is also the reason for the algorithm failure in 11 of the recorded cases. On the positive side, our adaptive vision-inertial sensor information fusion method performed as predicted in conditions of unknown sensor placement, evidently outperforming the non-adaptive method, as seen in the results for Position2 in Tables 6.3 and 6.5.

The achieved result of RMSE = 0.16 m is sufficiently close to the required minimal distance value of 10 cm (Section 4.3.1). As the indicator of the orientation accuracy for each method we took the worst RMSE value in its related table (Table 6.2 for Method1; Table 6.4 for Method2). For Method1 we obtained RMSE = 17° which is satisfying in relation to the acceptable error of 15-20 degrees. Method2 gave RMSE = 24° which falls just outside of the desired error range. Results in Table 6.5 show similar RMSE for different attachment positions of the sensor (28° vs. 22°) which proves that Method2 is able to adapt to an unknown sensor attachment situation.

In conclusion, for the orientation data collection from patients in controlled conditions the recommendation is to use the smartphone and AOE algorithm, because it is the simplest solution with acceptable accuracy. For uncontrolled conditions, like a home environment, we propose to apply the method based on the fusion of vision and inertial sensor information. A successful real-world application of this method depends on the improvement of the algorithm

to achieve faster detection of the person's true orientation after entering the camera scene.

6.3 Summary

The experimental study demonstrated that we have obtained a localization system with a sufficient position tracking accuracy for use in the intended FOG-monitoring application. The study of orientation algorithms gave us the necessary insight into the properties of smartphones for indoor orientation tracking in the relation of FOG. The proposed method of data fusion of visual and inertial data for absolute orientation tracking demonstrated how to use data from multiple sources to improve robustness of measurements with respect to the uncertainty of the wearable sensor fixing position.

7

Person Identification in a Home Camera Network

In this chapter, we propose a solution for the problem of identification of the patient in a home monitoring system. For this purpose we developed a new method that is able to perform identification of multiple persons living in one home. The chapter starts with a necessary background of the problem that we are facing. A short state-of-the-art, specific terms and frameworks are presented. In subsequent sections, we follow up with the presentation of a new appearance learning approach that uses recently introduced feature descriptors and the combination of Support Vector Machine (SVM) (Cortes and Vapnik, 1995; Burges, 1998) and Naive Bayes (NB) (Domingos and Pazzani, 1997; Rish, 2001) classifiers. Finally, we are ending the chapter with the evaluation of the new re-identification method on the prototype of our monitoring system.

7.1 Background

7.1.1 Identification with RGB-D Camera

Up to date, there have been several systems and algorithms developed specifically for the identification of persons using RGB-D cameras. Basso et al. (2013) presented a tracking approach for

Parts of this chapter appear in (Takač et al., 2014)

multiple people in which an online classifier based on Adaboost (Freund and Schapire, 1996) was used for learning identities of people. Randomized parallelepipeds inside a 3D color (RGB) histogram space were used as weak classifiers for boosting. Since the algorithm is geared towards the use for tracking in mobile robots, the approach was not tested in the conditions matching those of domestic camera networks in which multiple static displaced cameras are used.

Similar online boosting technique, using three types of RGB-D features and the confidence maximization search in 3D space, was used to build people models for tracking by Luber et al. (2011). The system evaluation was done on the data from a populated indoor environment using the setup of three Kinect sensors with a joint field of view. The evaluation of this algorithm for a true distributed non-overlapping camera configuration is not available.

Barbosa et al. (2012) presented the set of 3D soft-biometric cues, such as skeleton-based and surface-based distances calculated on the 3D point cloud, which they used to build identity signatures. The 3D soft-biometric cues could be the best approach for home network, since the biometric signature of household members needs to be taken only once. However, at the current moment the approach does not seem to be suitable for a range of different camera viewpoints, low resolution and unconstrained poses that can be expected in a multitude of a different home network configurations, since many of the proposed anthropometric measures are hard to extract under non-perfect real-world conditions.

Satta et al. (2013) developed a fully functional prototype of a real-time multiple RGB-D camera re-identification system which uses the fast re-identification method based on their own dissimilarity representation descriptors. We consider this method to be appropriate for the application in a domestic camera network. However, the method is primarily oriented towards the typical surveillance re-identification scenario, with a large number of cameras and persons and presumed presence of the human operator for the system. To see how the method could be adapted for our application, we should discuss the advantages and the limitations of a small home camera network compared to a public surveillance network.

7.1.2 (Re)identification in a Home Camera Network

The usual approach to the re-identification for public surveillance is to build a database of descriptors generated for each person and apply some distance measure between them. The database of descriptors is known as the gallery, while the distance measure is known as the matching score. In the re-identification for public surveillance, there are two possible frameworks (Chen et al., 2007). The first framework works with the complete gallery, which means that the appearance descriptors of all the persons that need to be identified are available when the re-identification needs to be done. In this case the re-identification is solved as the ranking problem, by searching for the minimal matching score against all other descriptors in the gallery. The second framework is the one in which the complete gallery is not available, so that new persons are added to the gallery by the human operator and the re-identification problem is solved by setting the similarity threshold value on the matching score.

The re-identification in a home camera system, which has to be automatically executed during long-term daily monitoring, involves components of both frameworks. It needs to improve on the functionality of the second framework in terms of providing automatic detection of new person appearances, and it has to be able to assign available appearance when the system is operating with the complete gallery of all household members.

Once there exists a short-term re-identification approach for multiple cameras that is able to support both previously described functionalities, the advance towards absolute long-term person identification can be made by using the contextual knowledge of the living habits of inhabitants or by adding additional sensors in the environment/on the persons. For example, the primary initialization of the gallery could be supervised by contextual information, such as the time of a day and the room where the specific appearance first showed up. Or, if the person wears an inertial sensor, the association between the learned appearance model and the person identity could be reinforced using the matching of movement data between inertial and video sensor modalities.

At the moment, our final goal of automatic association between the appearance and person identity in a vision-based healthcare monitoring system is predicated by the need to firstly obtain a robust re-identification framework capable of automatically deciding whether track appearances are a part of the existing household gallery or not. Instead of solving the re-identification problem using ranking and threshold on the similarity matching score of a single probe image, we propose the use of appearance learning (similar as was done in (Barbosa et al., 2012) or Luber et al. (2011)) to find a more dynamic, multi-view appearance model of each individual in the home gallery. During re-identification, the binary classification is first applied to discriminate between all the learned appearance models of gallery members and unknown people outside the gallery. Then, if a new appearance (probe) is classified as a part of the gallery, the subsequent identification between known individuals in the gallery is performed by using a simple prediction on a multi-class classifier trained with the known gallery members.

7.1.3 Dissimilarity Representation and Appearance Descriptor

The Multiple Component Matching (MCM) framework proposed by Satta et al. (2011b) provides an unifying view of appearance-based person re-identification methods, by embedding the common concepts of multiple instance representation (e.g. patches, strips, interest points) and body part subdivision. An extension of MCM framework called the Multiple Component Dissimilarity (MCD) framework Satta et al. (2011a) adopts very compact representation of individuals, while still trying to keep the discriminative capability and robustness of the original identification method. Under the MCD framework, the descriptor I_p^D for the image of a person subdivided into $\mathcal{M}$ body parts, is obtained as the concatenated vector of $\mathcal{M}$ dissimilarity vectors, where each dissimilarity vector represents dissimilarity between each body part and a set of bag of components for that part, which are called prototypes. In this thesis, we use a specific implementation of MCD descriptor, called *MCDimpl* Satta et al. (2012). The *MCDimpl* descriptor subdivides the body into torso and legs and uses components patches randomly ex-

tracted from each body part, represented by their HSV colour histogram. A great density of information, along with a fixed small size, make the *MCDimpl* descriptor a good candidate for use in statistical classifiers.

7.2 Appearance Learning with One-class Support Vector Machine

7.2.1 One-class SVM and Naive Bayes Classifier Cascade

We have emphasized that the ability to automatically confirm or reject a track as a part of the existing set of household members is an important property. The problem of detecting an unknown identity can not be solved by any supervised binary classification method, since the training data for unknown identities can not be collected. Problem can be posed as a novelty detection case, for which a suitable solution can be found in the application of the one-class SVM (OCSVM) introduced by Schölkopf et al. (2000).

In the OCSVM, the kernel function is used to map feature vectors in a higher dimensional space, and find a hyperplane that separates the trained class from the origin. Given the training vectors $\mathrm{x}_i = \mathrm{I}_p^D$, $\mathrm{x}_i \in R^n$, $i \in [l]$, the model is estimated as follows:

$$\begin{aligned} &\min_{w,b,\text{"}\xi\text{,"}\rho} \ \frac{1}{2}\mathrm{w}^T\mathrm{w} - \varrho + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i \\ &\text{subject to} \ \ \mathrm{w}^T\varphi(\mathrm{x}_i) \geq \varrho - \xi_i, \xi_i \geq 0, \end{aligned} \tag{7.1}$$

where $\varrho/\|\mathrm{w}\|$ specifies the distance from the decision hyperplane to the origin, and ξ_i are introduced slack variables. The trade-off parameter $\nu \in (0, 1)$ corresponds to an expected fraction of outliers within the feature vectors. As is the usual case in other SVMs, a kernel $K(\mathrm{x}_i, \mathrm{x}_j)$ is needed to form a decision function. One of the most common kernel functions used in experiments is the Gaussian Radial Basis Function (RBF) $K(\mathrm{x}_i, \mathrm{x}_j) = e^{-(\ \|\mathrm{x}_i - \mathrm{x}_i\|^2)}$, with the parameter γ which sets the kernel width. For the multi-class classification of the persons inside the household gallery, the Naive Bayes classifier is used. We use it because it is simple, non-parametric and there are implementations which enable on-line training.

The identification algorithm based on the combination of the two classifiers is shown in Figure 7.1. Each descriptor of the unidentified track is first classified by the OCSVM. In the case of a positive result, the NB classification on the same descriptor is invoked. The outputs of both classifications are stored in their respective buffers. When the sufficient number *num_dsc* of SVM classifications for the track has been reached, the contents of both buffers are used for the final identification class decision according to SVM and NB decision functions (written on Figure 7.1). If the SVM decision function confirms that the current track appearance is a part of the gallery, the class given by NB decision function is forwarded to the system output. Otherwise, the appearance is declared to be unknown.

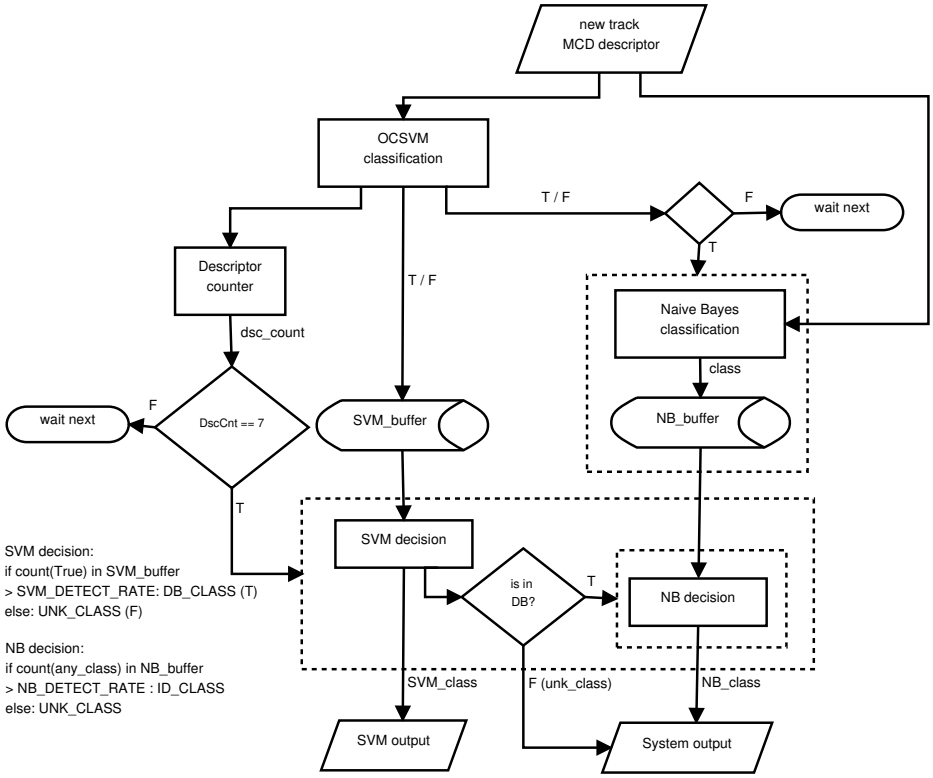
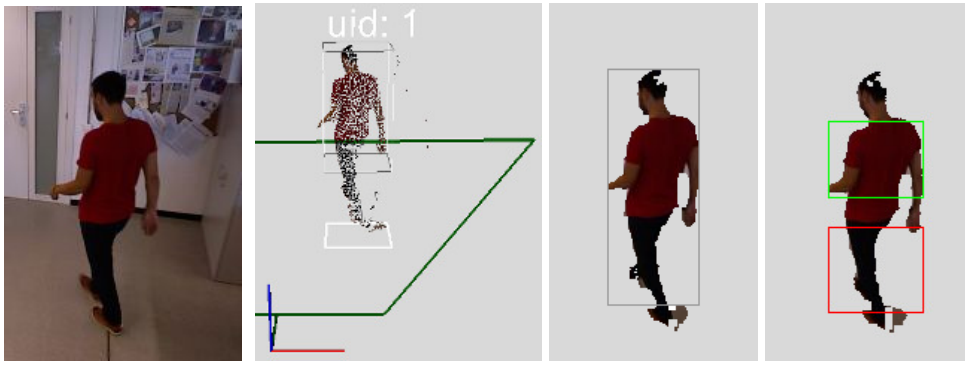


Figure 7.1: Identification using two classifier cascade.

7.2.2 Appearance Feature Vector Extraction

For building a dissimilarity representation of a person, we use a two part model of a human (torso and legs) similarly to how it was done by Satta et al. (2012). We base the body part extraction on knowing fixed ratios of human proportions towards the standing height. This approach has contextual dependency, since the model is taken only for the people which are having the standing posture. In a home environment, where frequent posture changes between standing, sitting and lying are to be expected, this approach ensures that appearances will be taken under consistent and similar conditions. The necessary contextual information about the standing posture is obtained by tracking person's height and applying pre-set height thresholds.

Starting with the 2D bounding rectangle ($a = 0.6m$) of the person track, the appearance extraction algorithm recovers the bounding cube for the 3D point cloud of all the points in the column above the tracked area (Figure 7.2b). Using the known transformation matrix between the 3D world coordinate space and 2D pixel coordinates of the Kinect camera image plane, it is possible to re-project back to the image plane the points only inside the bounding cube of the



(a) Color image.

(b) Tracked 3D point cloud with bounding cube.

(c) Re-projection of points inside the bounding cube on image plane forming person mask and bounding rectangle.

(d) Upper and lower body part mask based on metric anthropomorphic ratios.

Figure 7.2: Extraction of body part images.

person. Since the pixels of the re-projected image all belong to the tracked person, by applying a threshold operator it is straightforward to obtain the person mask and the bounding rectangle (Figure 7.2c). The masks for torso and legs are then obtained by applying body region limits on the existing mask of a person. Body region limits are given as follows, taking as the basis the anthropomorphic ratio towards height in meters:

$$hs = \frac{H}{\alpha},\ \alpha = 7.5 \tag{7.2}$$

$$ts = H * \beta * \frac{2}{3},\ \beta = 0.46 \tag{7.3}$$

$$ls = H * \delta,\ \delta = 0.5 \tag{7.4}$$

where H is standing height of a person, hs is head size, ts is torso size, ls is lower legs size, and the coefficients α, β and δ are based on values given in (ISO, 2013). By applying $2/3$ coefficient only the upper two-thirds of the torso are taken into account, leaving unmasked space between the upper torso and the legs (Figure 7.2d). This ensures that only color consistent parts will be taken by part masks, nullifying a potential influence of the change of perspective. In the end, body part images extracted in this manner are given to the original implementation of *MCDimpl* to produce person's appearance feature vector.

7.3 Re-identification Evaluation Experiment

The purpose of the experiment was to evaluate the feasibility and statistical properties of the proposed person identification approach based on the OCSVM and NB classifier cascade for learning and detection of the people living inside a home. We used the prototype of localization system consisting out of two Kinect sensors and a PC server (Intel Core i7 @ 2.3 Ghz; 8 GB RAM; Linux) on which all the processing nodes were run.

7.3.1 Dataset

We constructed a new dataset of 16 persons. For each person in a dataset there are two recorded videos. One used for the appearance learning, and the other used for the testing. A learning data video for each participant consists of the participant walking for approximatively 20 seconds in a random pattern in front of each Kinect camera. A testing data video for each participant consists of the participant walking the predefined route which alternates between the scenes in front of the two cameras. The walk along the testing route takes about a minute and a half, and it was designed to maximize the number of possible ways a person can enter camera FOV. For the given camera configuration set in a laboratory environment, there were seven possible views of a person when entering into camera FOVs. These views are shown in Figure 7.3. In route planning, we tried to introduce possible obstacles. One of the camera FOV entries (Figure 7.3-4) is performed by opening the door which were previously shut, forcing the system to extract the appearance while dealing with an occlusion. The other similar situation is entering the FOV behind the obstacle (Figure 7.3-5), when only a part of the body is visible.

Collected videos were used to build a training database and a testing database of person appearances. To obtain the training database, learning video for each person was replayed with the system put into the learning mode in which tracking nodes continuously (every 0.5 seconds) extracted one appearance (an image and two masks) and sent them to `ID_Manager` node, which recorded those appearances in the SQL database associated with that person's class label. During the testing database construction, testing videos were replayed with the system set into the normal operation mode in which appearances were extracted only when a new track was detected. For each new track detection $num\_dsc = 7$ appearances per track were taken, with the extraction frames being at least 0.1 s spaced in time. On-line re-identification in `ID_Manager` was turned off and feature vectors were recorded in the SQL database.

7.3.2 Classifier Training

For this experiment, the size of the target household gallery was set to $C = 3$. The NB and the OCSVM classifiers needed to be trained for an each separate instance of the household gallery. To get the statistical data about the classifier cascade performance, pairs of both classifiers were trained 16 times, each time with a different triplet of persons inside the gallery. The sets of three people were chosen at random, with a constraint that the same person could not be represented

Figure 7.3: Examples of FOV for each camera and eight entry events into those FOVs. There are 7 different person views (1 frontal, 2 right profile, 3 left back overhead, 4 frontal with door, 5 semi-frontal with obstacle, 6 left profile, 8 right back overhead), with one view (7 right profile) repeated twice in order to enable a continuous walking path during experiment.

more than three times. Obtained galleries are presented in the first column of Table 7.1, where numbers in the *Person IDs* column are related to the appearances in Figure 7.4.

The prototype gallery for obtaining dissimilarity descriptors using *MCDimpl* was pre-built using the 1264 pedestrian images of the VIPeR dataset (Gray et al., 2007), the same as was used by Satta et al. (2013). Given the triplet of persons forming a household gallery, their stored appearances were retrieved from the training database and turned into the dissimilarity representation forming the training data set $\mathcal{X} = \{X_{r_1}, X_{r_2}, X_{r_3}\} = \{x_i, ..., x_l\}$, where $r_1, r_2, r_3 \in \{1, .., 16\}$, $r_1 \neq r_2 \neq r_3$ are class indexes for the persons in the dataset, and $l = N_{r1} + N_{r2} + N_{r3}$ is the total number of the feature vectors used. The number of collected training descriptors for each person varied between 25 and 50, due to the random walking patterns in learning videos.

OpenCV Machine Learning library was used for training of both classifiers. The training of the Naive Bayes classifier is non-parametric, requiring only feature vectors and labels. On the other hand, the performance of the OCSVM classifier with the RBF kernel is dependent upon the used value of the hyperparameters ν and γ during the training. Since using the usual cross-validation method in order to optimize the hyper-parameters of OCSVM is not possible (Lukashevich et al., 2009), we had to find another approach.

We decided to use 642 different appearances taken from the VIPeR dataset as the validation data on which performance metrics is calculated to be used as the guidance in the grid search optimization process. Since the VIPeR dataset is diverse and does not contain the exact ap-

Figure 7.4: Appearance of each of 16 persons in the dataset. Top row, left to right, appearances 1-8. Bottom row, left to right, appearances 9-16.

pearances from the household gallery on which the OCSVM was trained, the intuition is that the fraction $\varkappa$ of the VIPeR images classified by OCSVM as belonging to the household gallery should tend towards zero as the OCSVM is improving in the rejection of the unknown appearances. Still, in order to avoid over-training, we can consider that there is a possibility of having a few appearances in the VIPeR set that are similar to the ones in the household gallery, which means that there should exist some small fraction $\varkappa$ of VIPeR images classified as positive by OCSVM. To investigate this behaviour, the OCSVM classifier was trained for each of 16 galleries in Table 7.1 with the target $\varkappa_{thresh}$ values of 0.5%, 1%, 2% and 4%.

7.3.3 Results

Statistic results for the accuracy of Naive Bayes classification are given in Table 7.1. Each row contains the test result for one NB classifier trained on the data of a randomly chosen household gallery triplet. The column *Tested* contains the sum of the entries into camera FOV taken by the persons in the triplet during testing trials for which the sufficient number $num\_dsc = 7$ of the track appearance images was collected by the underlying tracker, while the column *Correct* shows how many of those entries were correctly classified. In the final NB decision function the *nb_detect_rate* threshold set to 75% was used. According to the aggregated testing results for all 16 trained NB classifiers, the achieved average accuracy of the classification is $90.0 \pm 9.2\%$.

Table 7.2 demonstrates the influence of the OCSVM classification on the identification system output. The dependent variable is the newly introduced performance metric $\varkappa$ used for OCSVM hyperspace parameter optimization. The OCSVM classifiers for each person triplet

Table 7.1: Accuracy of Naive Bayes classifier.

Person IDs	Correct	Tested	Accuracy [%]
8,12,13	24	24	100.0
4,6,15	22	23	95.7
4,8,12	23	23	100.0
4,9,14	20	23	87.0
9,10,11	19	22	86.4
6,10,16	14	22	63.6
5,8,11	23	23	100.0
3,5,10	22	23	95.7
7,9,13	19	23	82.6
1,7,13	18	20	90.0
2,5,14	22	24	91.7
3,6,7	19	23	82.6
2,15,16	20	23	87.0
1,14,16	19	20	95.0
1,13,15	20	21	95.2
2,12,16	20	23	87.0
Total	324	360	90.0 ± 9.2

from Table 7.1 were re-trained four times, each time with a different value of ν_{thresh}. Each row of Table 7.2 shows average results obtained by testing the classifier cascade with 16 different OCSVM models, all of them trained with the same value of ν_{thresh}. To test the OCSVM and the overall system, the same set of valid FOV entries from the test database was used as when testing the NB classifier. In OCSVM decision function the *svm_detect_rate* acceptance threshold was set to 70%.

The average specificity and the sensitivity (along with the standard deviation) of the OCSVM classifier are given in the 2^{nd} and the 3^{rd} column. The best balance of positive and negative SVM classifications, with almost 80% sensitivity and specificity, can be be expected for ν_{thresh} value around 1%. The last two columns of Table 7.2 express the average sensitivity and accuracy of the classifier combination. The true positives for a given person inside a gallery are influenced by the misdetections of the NB classifier, resulting in the lower sensitivity output for the classifier cascade. Since the NB classifier only acts when there is a positive SVM classification, it has no influence on the true negatives that influence system specificity. The consequence is that the system specificity is the same as the specificity of the OCSVM classifier alone, so there is no need to show it separately in Table 7.2.

Table 7.2: Classification results for one-class SVM and combined classification system output.

	OCSVM		System		
κ_{thresh}	Sensitivity[%]	Specificity[%]	Sensitivity[%]	Specificity[%]	Accuracy[%]
0.5%	71 ± 24	87 ± 10	64 ± 21	87 ± 10	83 ± 7
1%	78 ± 17	82 ± 11	71 ± 19	82 ± 11	80 ± 8
2%	81 ± 16	75 ± 12	74 ± 19	75 ± 12	75 ± 8
4%	85 ± 10	64 ± 15	76 ± 13	64 ± 15	66 ± 11

7.4 Summary

In this chapter, we presented the method for identification of people by appearance learning in a small camera network. The method uses the combination of two classifiers. The first classifier (OCSVM) is used to discern if a new track in front of any camera has the appearance that belongs to the target group of people whose appearances are already known from before. If the inclusion in the group is confirmed, the second classifier (NB) predicts the exact person identity for the new track. Although we can target any group of people for re-identification, the original purpose of our method is to use it for the identification of the members of a PD patient's household.

In appearance learning, we employed a novel dissimilarity representation descriptor by Satta et al. (2011a), and we showed how to control the acquisition of those descriptors when a new track appears in front of a camera. Furthermore, we showed how to train the OCSVM classifier to discriminate between known and unknown groups of people by using totally independent dataset for the training parameter optimisation. Training for such discrimination presented a problem prior to the introduction of our method, because it is not possible to make a cross-validation for parameter optimization when training one-class SVM classifiers.

Our re-identification method still does not have a completely automatized way to assign appearances to people's identities at the initial stage of system operation while the appearance gallery is empty. In the presented work, this inputs were provided by human supervisor. We presume that this initial identity assignment task could in the future be automatized by using the contextual information of location or people's posture. However, at this moment solving the problem of the initial conditions of the system was not in our focus. The main intention was to find the technique for appearance learning that can be used in our patient data collection experiments. In the chapter about future work, we will offer more ideas how to solve bootstrapping problem.

Our method was tested on the 16 people dataset collected with the prototype of the real-time localization system in a laboratory environment. The experimental results have shown that the best balance of the specificity and the sensitivity of the classifier combination is achieved with around 75% for both (Table 7.2; 3^{rd} row). This results should be additionally improved if the method is to be applied in a long-term deployment in a completely uncontrolled environment.

We used the presented method to identify the patients and extract their movement data during clinical and home experiments. Details of these experiments will be presented later in Chapters 8 and 9. In clinical experiments there was medical personnel on the same scene as the patient, while in the home there was usually the patient's caretaker. Since we were only interested in following the patient, the size of target group for re-identification was set to 1 person. Such setup leaves it to OCSVM to confirm the identity of the patient, while the NB classifier step is skipped. Using only one person in the gallery minimizes the necessary training time, and minimizes system error to the level of the error of the OCSVM.

8

Posture and Activity Recognition

The relation between the patient's activity context and its importance for an improved FOG detection was established earlier in Section 3.3.1. In this chapter we present the newly developed solution for monitoring the activity context in PD patients. To get the activity context we need to asses both posture and activity.

At the beginning of the chapter we are exploring posture identification, setting the focus on the problem of detection of postures and postural transitions in the PD patient population. We present the possible sensor modalities for the task. To handle both static and dynamic postures, our posture identification algorithm relies on the height data obtained by tracking the person using the `Vision` node. We finish this part with the description of the algorithmic implementation.

The second part of this chapter belongs to the presentation of the newly developed approach for the fast activity classification. We give an explanation for the choice of the classifier, followed by the implementation details. The new activity classifier was developed using movement data from the participants without PD. After the evaluation on healthy participants, this classifier was also evaluated on the clinical data of PD patients. The training methods and the results of both evaluations are reported in the last sections of the chapter.

8.1 Posture Identification

There are two basic strategies that can be applied in posture identification with body-worn inertial sensors. The first strategy involves the usage of the accelerometer as an inclinometer.

The accelerometer measures the gravity force along each axis in order to infer the orientation of the body part on which it is attached. The problem of this approach is that static postures with similar inclination angles such as standing and sitting are easily confused (Gjoreski et al., 2011). The second strategy is based on the analysis of postural transitions. Postural transitions are recognized as independent states (in the temporal process) and taken as an indicator for determining the subsequent static body posture. Mathematical tools such as discrete wavelet transform (DWT) and FFT are usually used for the analysis of dynamic signal and posture transition identification (Najafi et al., 2002; Bao and Intille, 2004b; Bidargaddi et al., 2007).

Recent efforts by Rodriguez-Martin et al. (2013) resulted in a posture recognition algorithm that targets the population of the PD patients. The algorithm uses a hierarchical structure of 5 classifiers to identify eleven postures divided into 2 groups; static postures (stand, sit, bent, lying) and dynamic postures (walking, sit-to-stand, stand-to-sit, bending down, bending up, lying from sit, sit from lying) by means of a single tri-axial accelerometer located at the waist. The 5 classifiers were first trained and tested on 31 healthy volunteers. The posture classification algorithm achieved per-activity sensitivities of at least 97% and per-activity specificities of at least 84%. The algorithm was afterwards additionally tested with unchanged parameters on the accelerometer data from 8 patients with PD. The patient dataset had a shortened list of 5 activities (sit, sit-to-stand, sit, stand-to-sit, walking). On the PD patients dataset the algorithm achieved per-activity sensitivities of at least 87% and per-activity specificities of at least 78%, except for the sensitivity of walking which was 25%. The analysis showed that lower specificity results for walking are due to disturbances in gait balance or bradykinesia in PD patients. Except walking, other lower sensitivity and specificity results of PD patients in comparison to healthy people are the consequence of confusion in the recognition between sit-to-stand and stand-to-sit transitions. The authors assigned these errors to movements that introduce unexpected dynamic signal. Some movements, such as dyskinesias, are specific to PD, while the others are consequences of unforeseen real-life behaviour such as sitting in a chair and then making several up and down movements to correct the sitting position.

The algorithm of Rodriguez-Martin et al. (2013) already offers a solution for posture tracking using inertial sensors. The sensor configuration of our monitoring system, allows us to make a complimentary solution for static posture detection based on video data. Adding a complimentary solution has two advantages. The first advantage is that a more informative modality has a higher chance to improve on shortcomings of accelerometer-based posture identification. The second advantage is that if we have two posture identification algorithms in two separate modalities, we create the basis for a method that could improve the patient re-identification process. A periodic confirmation of the identity of the patient, that was previously assigned by the appearance based re-identification (Chapter 7), would be possible by comparing postural transitions between the two different modalities. If the same type of postural transition is detected at the same time in one of the video tracks and in the inertial sensor algorithm, there is a high chance that the tracked person is actually a patient wearing the sensor. This particular re-identification method is just a concept, that was not implemented in this thesis. Nevertheless,

we did implement vision-based posture identification.

There are several examples in the literature where video data was applied for the postural transitions detection. Often the silhouettes of people are used for privacy reasons (Demiris et al., 2009). Different 2D image features can be extracted from the silhouettes and sent to a classifier. For instance, to identify postures Allin and Mihailidis (2008) use Hu moments (Hu, 1962) and clustering, Banerjee et al. (2010) apply Zernike moments (Teague, 1980) with a decision tree, and Cucchiara et al. (2005) use Bayesian inference on silhouette projection histograms. Besides using silhouettes, another common option is to use posture identification based on simplified 2D (Goffredo et al., 2009) and 3D (Pellegrini and Iocchi, 2008) human body models.

Our goal is to implement a posture identification algorithm that uses the existing structure of the `Vision` node, and that will not require a lot of additional processing resources for image feature extraction. The reason for this is the expectation that in the future the `Vision` node might be deployed on embedded systems with limited resources. Berrada et al. (2007) used simple statistical features such as mean value and standard deviation of the blob of active pixels. Since they observe the person from the side, the vertical mean value of the blob is directly related with the height of the posture. Banerjee (2010) introduced several approaches based on voxel data captured by a system of three cameras. Banerjee's simplest approach uses three height thresholds and presumes that the person is in the upright posture when her height is above the first threshold and in the sitting posture when the height is between the other two, lower thresholds. The advantage of using height as the determining feature is its invariance to the angle and the distance that the person has from the camera. The disadvantage of this approach lies in the errors in the height measurement induced by the tracking. For example, the occlusion of the upper body part can decrease the measured height and directly influence the outcome of the posture classification.

In Section 5.2.3, we decided to base the posture classification on the height information. We observed the behaviour of the PD patients and we analysed the reported experiences from Rodriguez-Martin et al. (2013). We concluded that some of the movements can confuse a simple threshold-based posture classifier. Up and down bending or squatting to pick up an object from the floor are such types of movements. A simple height threshold algorithm would detect SIT posture every time when the height value is inside the sitting threshold zone. When there is dyskinesia or when the person is searching for a better seating position by producing up and down movements of the upper body while sitting, the height value can exceed the upper sitting threshold and be interpreted as a sit-to-stand action. Our final conclusion was that we can not only rely on the height information and two thresholds. We need to use historical information extracted from the height profile to confirm the actual posture.

8.1.1 Finite State Machine

A finite state machine (FSM) is an abstract construct in which there is a finite number of states, out of which in any moment only one state can be active. The changes between the states

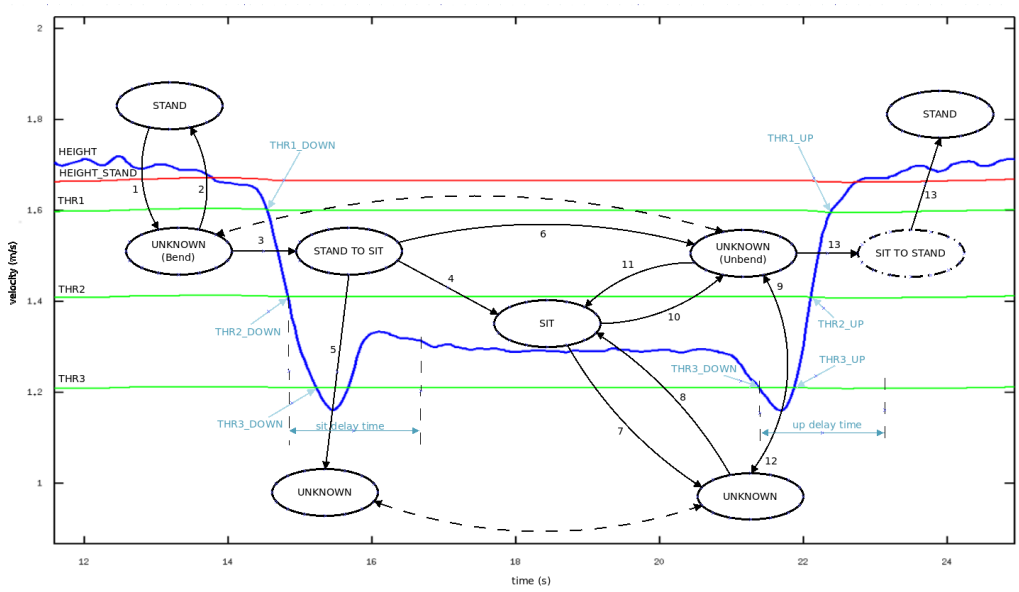


Figure 8.1: Finite state machine for posture identification.

are known as the state transitions and they happen when some triggering event or condition occurs. Figure 8.1 shows an example of a posture change from standing to sitting, sitting for around 4 seconds and going back to standing. Two signals are used as the inputs in the FSM: the person's current filtered height h and the person's estimated standing height h_{st}. There are three thresholds (green lines) that were mentioned earlier. We define them as follows:

Threshold 1 (THR1)
: The height at which a person is considered not to be in the fully standing posture any more. A person could be bent a bit or starting the posture transition. Mathematically we defined the threshold value as $h_{thr1} = h_{std} \times (1 - q)$, where q is the percentage of the standing height that is tolerated for the height signal variation.

Threshold 2 (THR2)
: The height under which a person is expected to be when sitting. The value of the threshold is defined as $h_{thr2} = h_{sit} + p \times h_{std}$, which is the estimated sitting height of the person enlarged by the percentage p of the standing height h_{std}. The sitting height is defined on the basis of the formula $h_{sit} = h_{chair} + 0.51 \times h_{std}$ (Fredriks et al., 2005). Value h_{chair} is the height of the object on which the person is sitting (usually between 0.45 m and 0.52 m), while $0.51 \times h_{std}$ is the height of the person's torso when sitting.

Threshold 3 (THR3)
: The height above which a person is expected to be when sitting. It was defined in a

similar way as THR2 by the formula $h_{thr2} = h_{sit} - p \times h_{std}$, to produce a symmetrical band of values centred at the expected sitting height.

The FSM has a finite set of 5 possible states: stand (STD), stand-to-sit (STD_SIT), sit (SIT), sit-to-stand (SIT_STD), and the unknown state (UNK). There are 6 threshold-based events and 3 time-based events. Threshold-based events are triggered each time when the height signal crosses one of the three thresholds in any of the two directions. These events are: THR1_DOWN, THR1_UP, THR2_DOWN, THR2_UP, THR3_DOWN, THR3_UP. Time-based events are events that happen a certain amount of time t_{dly} after the threshold has been crossed. Three time based events are defined in relation to their triggering threshold crossings: THR2_DOWN _TIMED, THR3_DOWN_TIMED, and THR3_UP_TIMED.

The transitions between the states are defined by the current state, the previous state, event type and additional conditions when the event occurs. A successful transition assigns a new value to the current state and the previous state. Table 8.1 shows the transitions that are marked with numbered arrows in Figure 8.1.

Table 8.1: State transition table for posture changes.

State transiiton	Event	$State_t$	$Prev.\ state_t$	Condition	$State_{t+1}$	$Prev.\ state_{t+1}$
1	THR1_DOWN	STD	any		UNK	STD
2	THR1_UP	UNK	any		STD	UNK
3	THR2_DOWN	UNK	STD		STD_SIT	UNK
4	THR2_DOWN_TIMED	STD_SIT	UNK	THR2 > h > THR3	SIT	STD_SIT
5	THR2_DOWN_TIMED	STD_SIT	UNK	h > THR2	UNK	STD_SIT
6	THR2_DOWN_TIMED	STD_SIT	UNK	h < THR3	UNK	STD_SIT
7	THR3_DOWN_TIMED	SIT	STD_SIT or UNK	h < THR3	UNK	SIT
8	THR3_UP_TIMED	UNK	SIT	THR2 > h > THR3	SIT	UNK
9	THR3_UP_TIMED	UNK	SIT	h > THR2	UNK	UNK
10	THR2_UP	SIT	UNK or STD-SIT		UNK	SIT
11	THR2_DOWN_TIMED	UNK	SIT	THR2 > h > THR3	SIT	UNK
12	THR2_DOWN_TIMED	UNK	SIT	h < THR3	UNK	UNK
13	THR1_UP	UNK	SIT		STD	SIT_STD

The main reason for using FSM to track transitions between the postures is to confirm SIT posture and assign a dynamic posture change to the part of the analysed signal. The left side of the Figure 8.1 (between $time = 13$ s and $time = 17$ s) shows a standard height profile during the stand-to-sit movement. State change (StTr2) ensures that the FSM algorithm can always return into the default state, which is to declare that the person is standing (STD) when the height is sufficiently large.

StTr1 is the start of the bending movement. Before the height crosses THR2, we are not sure if the existing forward bending is significant enough to warrant a change of posture state. The downward crossing of THR2 by height value, indicates that the current movement could actually be deep bending that is characteristic for the stand-to-sit posture change and the STD_SIT state gets declared (StTr3). The SIT state is defined by the threshold band formed by the comparison THR2 > h > THR3. Sometimes the forward lean during the normal sit down movement is so deep that the person's height can temporarily get even lower than THR3 (visible

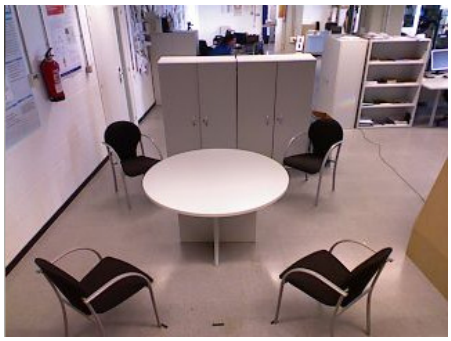
around $time = 15$ s) before it returns inside the expected threshold band for the SIT state. To avoid declaring some other posture state when the height decreases bellow THR3, the decision is delayed until the expiry of the timed event triggered by the height value crossing THR2. On the expiry of THR2_DOWN_TIMED, a person's height can be the one expected for sitting (StTr4), lower (StTr5) or higher (StTr6). Sometimes when a person is sitting, there can be a short movement in the seat upwards to fix the manner of sitting, or to lean more forward. In both cases, the height value can exceed of the thresholds for sitting (StTr7, StTr10).

If SIT was the previous state before the height value crossed THR2 or THR3, FSM algorithm allows for the possibility that the person might return into the previous SIT state (StTr8, StTr11). The postural transition of sit-to-stand is the inverse of the stand-to-sit. If there is a forward leaning with the height going lower than THR3, that is afterwards followed by a fast, upward crossing of THR2, it is possible to consider this sequence as a SIT_STD transition. However, forward leaning during getting up is usually not deep enough. Thus, the only available evidence of person getting up is when THR2 and THR1 are crossed in the upward direction (StTr13).

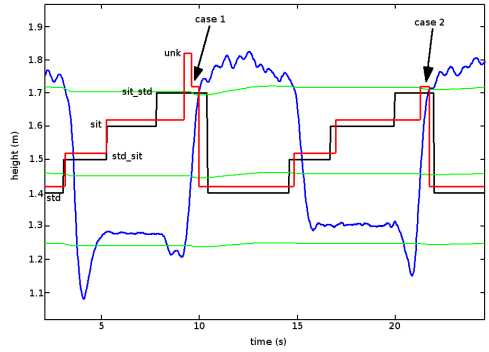
8.1.2 Parameter Optimization

The performance of the posture algorithm depends on the values of the three height thresholds (h_{thr1}, h_{thr2}, h_{thr3}) and the delay times (t_{thr2_D}, t_{thr3_D}, t_{thr3_U}) for the time based events. We used a grid parameter search method to optimize the thresholds and delay times. The search space for 6 variables is too big, so we reduced its dimension to 3 parameters. As seen before, three height thresholds are defined using two percentage parameters q and p, while the delay times for time based events are defined via a single delay parameter t_{dly} as $t_{thr2_D} = t_{thr3_D} = t_{thr3_U} = t_{dly}$. The objective function for the optimization is defined as the maximum accuracy of the posture classification on the given dataset, under the constraint $h_{thr1} > h_{thr2} > h_{thr3}$. If several sets of parameters achieve the same maximum accuracy at the end of the parameter search, the preference is given to the parameters (p, q) that enable the biggest distance $(h_{thr1} - h_{thr2})$, and the smallest parameter t_{dly}.

For the threshold optimization we collected posture transition data from eight healthy people with the average age $[28.0 \pm 7.3]$ years and average height $[168.6 \pm 7.6]$ cm. We wanted to test the invariance of the posture identification method in relation to different angles and distances of the person from the camera. Hence, we used four chairs of the same height ($h_{chair} =$ 0.46 m) that were set around a round table with angles of 45°, 135°, 225°, 315° (see Figure 8.2a). Two of the chairs were set at the distance of 2 m and another two at 4 m from the camera. Each participant was asked to sit on each chair, maintain seated for a few seconds, get-up, walk and sit on the next chair. After making a circle around the table and sitting into each chair he/she would leave the camera scene. Each participant did the sequence twice, first going to the left side and then going to the right side around the table. This produced eight instances of each static and dynamic posture per participant.



(a) Image of the scene. Chairs are 2 m and 4 m far from the camera.



(b) Result graphs for two stand-sit-stand sequences.

Figure 8.2: Data for posture FSM parameter optimization.

The calculation of the output accuracy for parameter optimization is explained on the example of the two consecutive stand-sit-stand sequences presented in Figure 8.2b. The height profile and three height thresholds are marked with the same blue and green colors as in Figure 8.1. The black line represents the ground truth for postures, which was set by the video observation, while the red line is the output of the FSM for some set of parameters (p, q, t_{dly}). For every sample of height data, we compared the ground truth class and the output posture class. We accumulated the comparisons of predicted and actual classes. For all the samples by all the participants we collected a 5×5 confusion table (CFT). The accuracy is then calculated straight from the CFT as:

$$acc = \frac{\sum diag(CFT)}{N} \tag{8.1}$$

where N is the total number of samples. The maximum achieved accuracy for the dataset was around 80%. The main reason is the inability of the state machine to infer the sit-to-stand posture transition before the moment when the height value crosses threshold THR2_UP. This causes the difference between the moment in which the FSM outputs STD_SIT posture class, compared to the same class onset in the ground truth. If we analyse a way in which sit-to-stand transition can be executed, there are actually three possible cases. Case 1 is the simplest one in which the frontal bend is not large enough to cause a drop in the height value lower than THR3. Case 2, showed in Figure 8.2b, happens when the time counter for THR3_DOWN_TIMED is activated, but it is reset by the height value crossing over THR2. This case, similarly to the previous, also gives a clean transition from SIT to SIT_STD state. In Case 3, frontal bending takes too long and THR3_DOWN_TIMED event expires before THR2_UP threshold gets crossed. Such behaviour causes a short period of UNK state, before SIT_STD state is detected.

The assessment of the FSM depends on the application. For internal parameter optimization the output was compared with the ground truth for each sample. The described evaluation method yielded a maximum accuracy of around 80 %. If the exact timing of the start of a postu-

ral state is not very important, but only the the fact that the postural state event got recognized, the assessment can be done on the principles explained in Section 3.2.2. Similar event-based evaluation was used by Rodriguez-Martin et al. (2013). The test of FSM on PD patient data is presented in the next section, within the evaluation of the complete activity classifier.

8.2 Hierarchical Activity Classifier

A decision tree classifier is a multi-stage classifier which classifies an unknown sample into an output class using one or several decision functions in a successive manner (Swain and Hauska, 1977). Its name stems from the fact that such classification strategy can easily be represented by a graph diagram in the form of a tree. The main topological elements of a decision tree are the root node, a number of interior nodes, and a number of terminal nodes. Consequently, the design of a decision tree involves the search for an appropriate tree structure, selection of features, and a definition of decision functions to be used for each internal node (Safavian and Landgrebe, 1991).

Depending on the chosen features, the decision function can be a simple comparison over a single number value, or more complex, such as a statistical classifier taking a feature vector as input. If learned statistical classifiers are used, it is necessary to give more importance to the classifiers that behave better and have less chance for error. This is achieved by allowing them to decide first, which leads to *hierarchical classification*, an often employed approach for human activity recognition (Zhang et al., 2010; Khan et al., 2010; Banos et al., 2013; Su et al., 2014).

The reason for using hierarchical classifier is that its configuration minimizes the decision error. Hierarchical classifiers for activity recognition that analyse both static postures and actions, usually first make the decision whether the activity is static or dynamic, before proceeding with the further classification. The main disadvantage of the hierarchical classification approach is that an error committed at the first levels of classification can propagate to higher levels and likely result in an erroneous decision (Banos et al., 2013). Nevertheless, due to the hierarchy of human movements and the decomposition of complex activities into more simple sub-activities, using this type of classifier for human activity recognition often yields superior results over simple multi-class classifier (Ribeiro and Santos-victor, 2005; Subramanya et al., 2006).

8.2.1 Activities

The main goal of our activity classifier is to identify 7 activities divided into 2 groups; static postures (*Stand*, *Sit*) and actions (*Forward walk*, *Non-forward move*, *Wide turn*, *Spot turn*, *Bending*). This set of activities was chosen from the set of false positive and false negative FOG-related activities described in Section 3.3. The choice was made on the basis of two criteria: 1) the importance for detecting or discriminating FOG; and 2) a realistic chance that the activity can be recognized from the available sensor data.

The inclusion of two static postures in the set of activities for classification was automatic. The *Sit* class directly eliminates the possibility of FOG, and the *Stand* class signalling no movement on the spot is an indicator that FOG might be occurring. By detecting the *Bend* class, either while the person is standing, or as a part of the sit-to-stand postural change, it is possible to directly eliminate FOG false positives. Unlike previous three activities which have clearly defined relation to FOG, the turning action is ambiguous, with both FP and FN FOG detections being observed during turns. (On-the-)*Spot turns* with small radius require significant change in the motor program, which often results in the shuffling step FOG while the turn is still ongoing. In *Wide turns* FOG patients have enough space to execute normal step lengths and sustain a higher movement speed, with FOG most often happening towards the end of the turning action. Therefore, recognizing the situation in which there is a wide turn followed by a sudden stop in locomotion, could be one of the FOG indicators. Since there is a difference in the expected sequence of actions leading to FOG between these two types of turning, they were added separately into the activity set.

Table 3.1 in Section 3.3.1 displays four activities that have the word *walking* in their name (*start walking*, *normal walking*, *stop walking* and *conditioned walking*) and two activities with the word *steps* (*small steps*, *backward/lateral steps*). A general difference between the two groups is in the speed and the length of the performed locomotor actions. The *normal walking* action is a forward movement with clearly defined steps (at least 3 of them) executed in a continuous manner. The *Forward walk* can be inferred from location data based on movement velocity. Already during the first step of normal walk people achieve linear speed of at least 0.3-0.4 m/s. During a FOG episode, the linear movement speed of patients is expected to be lower than this threshold. If we know when the person is walking and when not, we can eliminate FP FOG detections related to three of the *walking*-related cases from Table 3.1, i.e *start walking*, *normal walking*, *stop walking*. Recognizing the fourth *walking*-related case, *conditioned walking*, requires higher levels of perception and contextual cognition (e.g. where does a person hold hands) than planned for the current system, so we do not aim to recognize it.

The activities with the word *steps* contain non-regular intermittent steps that can go in any possible direction. In some cases it is very difficult to spot the difference between *small steps* and FOG, even for to a trained observer. It is natural to expect that our position tracking based system will have the same difficulties. When a person moves forward with *small steps*, his/her movement speed is very low. Hence, the accelerometer signal and the speed of the observed person are similar to the ones of the person that is advancing forward while experiencing shuffling FOG. Because it is not possible to distinguish well between FOG and the action which negates it (small intentional steps), we realized that tracking the forward *small steps* action would just introduce additional confusion in activity recognition. Unlike forward directed *small steps*, *backwards/sideways steps* are exceptional occurrences, Thus, they can be easily distinguished, but only if the person's egocentric movement direction is known. We unified *backwards/sideways steps* into the *Non-forward move* class, which is the final targeted action class added into the activity set.

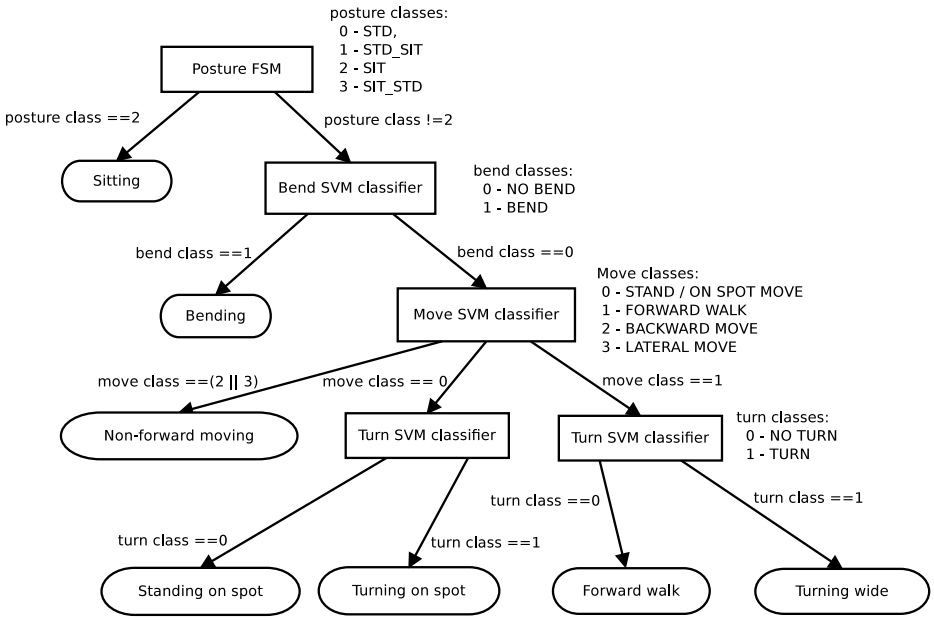


Figure 8.3: Decision tree with 4 classifier types.

8.2.2 Decision Tree Structure

The hierarchy of the classifiers in the decision tree arises from the relation between the activities that we want to recognize and the types of contextual information provided by the sensors. The decision tree that is used for seven-activity classification is presented in Figure 8.3. On the top of the decision tree is *Posture_FSM*. The output of *Posture_FSM* is taken as into a binary decision between *SIT* vs. all other classes (STD, SIT_STD, STD_SIT). On the next level, the STD class is separated from the postural transition classes (SIT_STD, STD_SIT) based on the classification by the *Bend_SVM* classifier. Positive detection of *Bending* directs towards probable postural transition, while the negative detection means that the person is in the upright posture, and either moving or standing on the spot. On the third level of the decision tree structure, when there is a confirmed STD (upright) posture class, the decision is made about the direction of the movement of the person. Possible directional classes are *Forward*, *Non-forward* and *Stand* (in place with some minimal movement). Due to unique egocentric velocities during the non-forward movement (negative forward velocity, high lateral velocity), the *Move_SVM* classifier should not have any problems to detect this particular class. In combination with the *Turn_SVM* classifier, a classification output from the *Move_SVM* leads to detection of either *Forward walk* or *Wide turning* activity, depending whether *Turn_SVM* had a positive turn detection or no. Similarly, in the case when the *Move_SVM* classifier detects *Stand* class, the distinction between *Stand* and *Spot turn* classes depends on the output of the *Turn_SVM* classifier.

8.2.3 Feature Vector

The multi-modal contextual data for recognition of FOG related activities consists out of raw accelerometer and gyroscopic measurements from IMU, estimated 2D orientation on the floor plane, position coordinates, estimated linear velocity and height. This data has to be transformed into the selected set of features for activity classification presented in Table 8.2.

The primary requirement for the extraction of features from multi-modal data in our system is the concurrent access to *IMU* and *TrackData* messages. The activity recognition algorithm needs the estimation of the patient's 2D orientation during the process of formation of feature vectors. Since the algorithm for the estimation of 2D orientation also needs synchronized data from *IMU* and *TrackData* (as seen in Chapter 6), feature extraction and the hierarchical activity classification algorithm are conveniently placed in the same callback function as the orientation estimation algorithm (in `Context` node). The synchronization of *IMU* and *TrackData* message streams in the callback function sets the sampling rate of the input data for activity recognition to 30 Hz.

Table 8.2: Features for activity recognition

Feature	Limits	Unit
Avg. forward velocity	[-0.75, 1.5]	m
Avg. lateral velocity	[-0.75, 1.5]	m
Avg. acceleration magnitude	[0, 5.0]	m/s^2
Avg. rotation velocity around vertical axis	[0, 2.5]	rad/s
Avg. rotation velocity around transversal axis	[0, 2.5]	rad/s
Displacement (position difference)	[0, 1.0]	m
Height difference	[0, 0.6]	m
Std. dev. forward velocity	[0, 0.3]	m
Std. dev. lateral velocity	[0, 0.3]	m
Std. dev. acceleration magnitude	[0, 2.0]	m/s^2
Std. dev. rotational velocity around vertical axis	[0, 0.6]	rad/s
Std. dev. rotation velocity around transversal axis	[0, 0.6]	rad/s

Avg. - Average (Mean)
Std. dev. - Standard deviation

A fixed-width sliding window, with 50% overlap, is used for feature extraction. Three types of feature calculations are utilized: mean value, standard deviation and sample difference of the signal. Features based on mean value and standard deviation are calculated using the window length of 16 samples. Under the 30 Hz sampling rate this window length corresponds to a time period $\Delta t = 0.53$ s. We choose such short window time because we need to analyse the direction of a person's movement in each step. For example, when a person is standing, he/she

can go backward for just a step and then stop. Furthermore, before a person starts to turn while walking, usually he/she will make straight steps until the first step at the start of the turn. This first step contains significant body rotation and it is possible to detect it right away. Our goal is to use the shortest possible window length in which all selected FOG-related activities can be recognized. Based on the collected activity data, window duration close to half a second was enough to capture even the fastest of the walking steps.

Some of the features in Table 8.2 request additional processing of data provided by the perception system:

Forward and lateral velocity
: These velocities reflect the movement relatively to the person's own coordinate system. To obtain the velocities, the original velocity vector of the person calculated in the camera base coordinate system ($\vec{v^{cb}}$) has to be transformed into the person's velocity expressed in the egocentric 2D coordinate system ($\vec{v^{ego}}$). The velocity transformation is performed using 2D orientation angle φ between the person and the *camera_base* frame with the following equation: $\vec{v^{ego}} = {}^{cb}_{ego}T(\varphi) \times \vec{v^{cb}}$.

Acceleration magnitude
: The accelerometer measures a body motion acceleration component and the gravitational acceleration component. The accelerometer signal has to be conditioned before it can be used in feature calculations. Only the dynamic component of the body motion acceleration is used and it is obtained by employing the 3^{rd} order high-pass Butterworth filter with a cut-off frequency of 0.3 Hz (Anguita et al., 2013). The filter design was performed similarly to the process already explained in Section 5.2.3. The magnitude of acceleration $\mathcal{M}$ is calculated from the filtered signal using the formula $\mathcal{M} = \sqrt{A_x + A_y + A_z}$, where A_x, A_y and A_z are acceleration components.

Displacement
: Displacement is the position change that happened since the last activity classification. The displacement value D is calculated every Δt as $D = \sqrt{x_{diff}^2 + y_{diff}^2}$, where x_{diff}, y_{diff} are position differences in meters along the principal axes of the camera base frame.

The rest of the features is calculated directly from the acquired signals. The absolute height difference is calculated as $H = abs(h_t - h_{t-\Delta t})$. Features for rotational velocities are extracted from the gyroscopic signal. Two gyroscopic channels of the smartphone are used: G_x for vertical rotations and G_z for transversal rotations, under the presumption that the IMU sensor is fixed accordingly to the reference position shown in Figure 6.1.a in Chapter 6. The final phase of the feature preparation includes normalization. Usually, normalization of features is performed to balance their contribution to the objective function of the classifier. The simplest

method for normalization is to rescale the feature range by applying

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{8.2}$$

where $min(x)$ and $max(x)$ are upper and lower limits for the specific feature value. The limit values used for normalization are given in Table 8.2. The limits were set on the basis of feature ranges observed in the training dataset, that we describe in the next section.

8.3 Training of Component Classifiers

8.3.1 Dataset

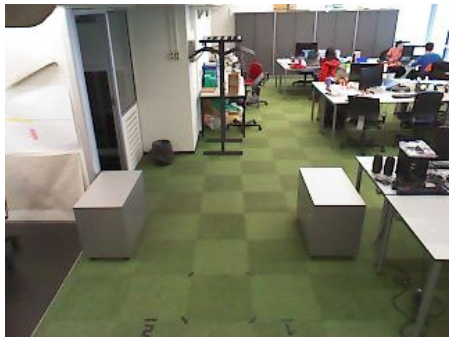
The dataset for training and testing the decision tree component classifiers (*Turn_SVM*, *Bend_SVM*, *Move_SVM*) was collected using a smartphone and one Kinect camera. The smartphone was placed at the left hip according to the convention that was previously described in Chapter 6 (Figure 6.1.b). The smartphone was fixed on the participant's body using the elastic belt that is worn under the clothes. The elastic belt has a sewn-in pocket in which the smartphone can be placed. This fixation method ensures that the device is strongly attached against the body and minimizes the possibility of undesired sensor movement. This method is a standard way of fixing sensors in user experiments performed in the CETpD laboratory.

The experiment involved 8 participants without gait problems, that we will refer to as "healthy" participants in the rest of the chapter. The age of the participants ranged from 27 to 32 years ($\mu = 29.2$, $\sigma = 2.0$) and their height from 159 cm to 190 cm ($\mu = 167.8$, $\sigma = 12.0$). A session for each participant lasted around 15 minutes and consisted of three trials. Each trial targeted a specific set of activities:

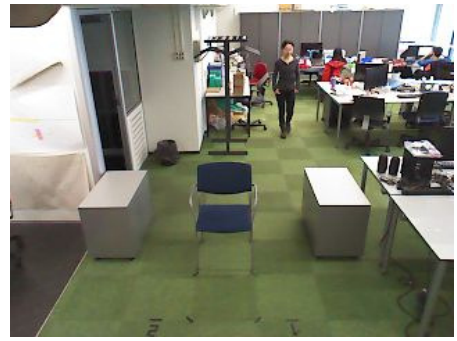
Turning

The primary target of this trial was to collect data while people are making different types of turns. A walking trajectory was designed to involve an interchange of straight walking and turning in a limited space in front of the camera. The *Turning* trial was done in two series of four short walks. In each short walk the participant would enter the camera FOV, walk from one side of the camera scene to another while following a trajectory with 3 different types of turns, and exit the camera FOV on the opposite side than he/she entered. After that a new walk was made from the current side of the camera scene, by entering again and following a slightly different trajectory.

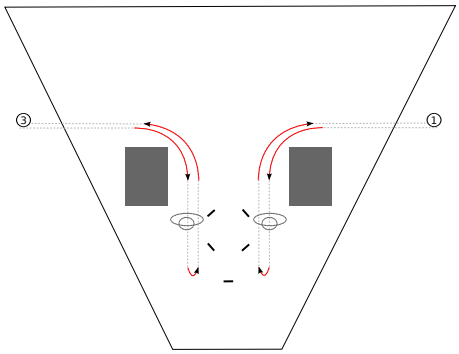
To facilitate normal walking behaviour of the participants, an obstacle was added on each side of the scene. Also, a rectangle on the floor of size 0.5 x 0.5 m at the distance 2 m centrally from the camera was marked as the basic reference for the place where turning should be executed. The image of the scene with two obstacles and the reference rectangle is visible in Figure 8.4a, while the walking trajectories that were assigned for the



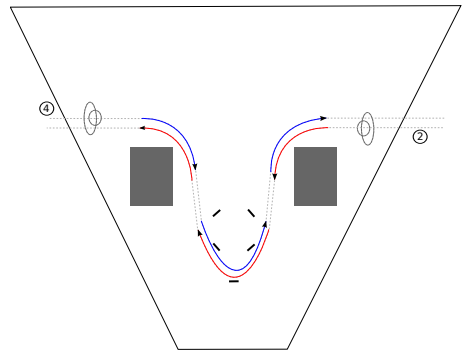
(a) Image of the scene.



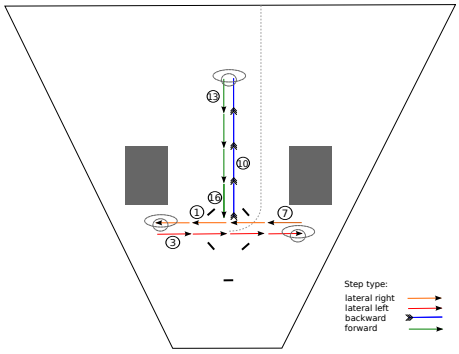
(b) Image of the scene.



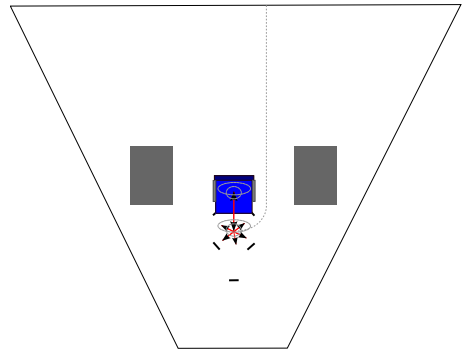
(c) Walking trajectories for spot turns.



(d) Walking trajectories for U-turns.



(e) Walking trajectories for step moves.



(f) Walking trajectories for bending.

Figure 8.4: Trajectories for the collection of data for activity recognition.

turning trial are sketched in Figures 8.4c and 8.4d. Figure 8.4c shows the first and the third walk in the series, which involve the combination of a 90° turn, a 180° spot turn and a 90° turn in the opposite direction than the first 90° turn. Figure 8.4d shows the second and the fourth walk in the series, composed of two 90° turns, with one 180° wide turn in between. The second walk starts from the right side of the camera, while the fourth walk starts from the left side.

Besides collecting data for training the *Turn_SVM* classifier, the secondary objective of the *Turning* trial was to collect data to train the recognition of the forward walking activity, regardless of walking speed. This is the main reason why the trial was split into two series of 4 walks. In the first series the participants walked with their normal walking speed, while in the second series they reduced speed to half of their normal walking speed.

Movement direction

In the second trial we targeted the collection of data for recognizing the movement direction. We took into account even the movements that last as short as it takes to make one step. To collect the specific training data, the participants were asked to perform discrete steps in four basic directions on the floor plane.

Upon approaching the referent spot, participants were asked to face the camera and make a series of 16 discrete steps. The orientation of the participant always remained with the face towards the camera. The series of steps is illustrated in Figure 8.4e. Starting from the reference rectangle, the participant had to make 2 lateral steps on his right, followed by 4 lateral steps to his left side and two additional lateral steps towards the right side, in order to return to the starting rectangle. From there the participant did 4 separate steps backwards, followed by 4 steps forward. The 16 step combination was repeated 2 times by each participant.

Bending

For the last trial, a chair was placed at the distance of 2.5 m from the camera (just behind the referent marked rectangle). Each participant started with normal walking outside of the camera FOV and walked until he/she reached the place in front of the chair (see Figure 8.4b). The participants were instructed to repeat 4 identical series of stand-sit-stand actions, once they reached the reference rectangle.

After entering the reference rectangle, the participant first had to wait 4 s before sitting down on the chair. After he/she sat on the chair, he/she was to stay seated for 4 s, before getting up from the chair. Upon getting up, a participant was instructed to simulate "restless standing" for 4 s, and then repeat everything 3 more times starting with a new sit action.

The collection of "restless standing" data was part of the secondary objective of the trial, that was not directly related with training *Bend_SVM* classifier. The data collected dur-

ing "restless standing" was intended for training the *Stand* class in the *Move_SVM* classifier, so that the classifier can achieve robustness to possible FOG and bradykinetic movements of PD patients. The instructions for achieving "restless standing" required from a participant to shift hi/hers weight from one leg to another, and/or to produce small position changes by moving the upper part of the body while being inside the reference rectangle.

Raw Kinect video and depth data, along with the smartphone inertial data for each trial, was recorded in the *.rosbag* format. Afterwards, the captured raw data was replayed as the input into `Vision` node in order to extract *TrackData* messages from person tracks. Between each entry and exit of a person in/from the active tracking area in front of the camera, the tracking algorithm produced a trajectory segment belonging to the person's track that is called *tracklet*. The first trial (*Turning*) produced eight *tracklets* per person, while the second trial (*Movement directions*) and the third trial (*Bending*) produced two *tracklets* per person each. For each 2D point in a *tracklet* its *TrackData* message was read with its synchronized IMU message, transformed into the feature vector presented in Table 8.2, and saved into a SQL database.

Once the feature vectors were available in the database to be used for training and testing, the final step was to assign the ground truth activity labels to video data. The start and the duration of activities in the labelled data were observed with the time precision expressed in hundreds of milliseconds.

8.3.2 Training Method

The first step towards the completion of the seven-class hierarchical activity classifier was to separately train SVM classifiers. Each SVM classifier was trained with data from the trials in which the participants performed the activity that the particular classifier is targeting, and the activity that can easily confuse the same classifier. For example, in the case of the *Turn_SVM* classifier, the data for training was taken from the *Turning* trial and from the *Bending* trial. The reason for using data from two trials is that the position of fixing the smartphone and its exact 3D orientation may indirectly influence a correct detection of turning, by classifying it as bending. If it happens that the smartphone is not perfectly positioned, having its X-axis parallel to the vertical axis of the person's trunk and Z-axis parallel to the person's transversal axis, the Z-axis will measure some rotational velocity when the person is turning. Proportionally to the angle of deviation of these axes from the ideal position, the confusion between turning and bending will become more probable. Data that contains examples of bending movements with false labels are used during the *Turn_SVM* classifier training, to make the classifier able to differentiate better between turning and bending.

Similarly, for training of the *Move_SVM* classifier we used feature vectors from all three trials. The trajectories from the *Turning* trial contain data with samples of different walking speeds. In the trajectories from the *Bending* trial we can find samples that are able to confuse the classification for direction of movements in the *Move_SVM* classifier. The critical fea-

ture for detecting movement direction is the change in a person's 2D position. If the detection of movement direction is based primarily on the 2D position, the sitting down action can be mistaken for backward movement and getting up action for the forward movement. Hence, *Bending* trial data is used in *Move_SVM* training as the negative example for those classes. The *Bend_SVM* classifier was trained only with the data from *Bending* trial.

The goal of the experiment with the healthy participants was to obtain the average classification results in terms of sensitivity, specificity and accuracy for each component classifier. To calculate the average results, the classifiers had to be trained and tested several times. We had to define a method to divide the available dataset into training and testing parts in order to achieve a sufficient generalization of the statistical results. Usually "leave-one-out" or a similar method can be used to make the dataset division. Another good option is to split the dataset so that 70% of the available data is used for training and 30% for testing.

To get average results, the training of the SVM component classifiers was repeated 8 times. For each training repetition, we split the dataset from 8 participants randomly into training data (taken from 5 participants), and test data (taken from 3 participants). Table 8.3 shows which participants provided data for which training repetition, and what was the distribution of the ground truth labels for each class in the training data. Similarly, Table 8.4 shows which participants provided data for testing of which trained model and the distribution of the ground truth labels for each class in the test data.

Table 8.3: Distribution of ground truth labels for feature vectors that were used in each of 8 training repetitions.

Training ID	Person IDs	Turn [0, 1]	Bend [0, 1]	Move [0, 1, 2, 3]
1	1,2,4,5,6	[3917, 796]	[2347, 577]	[2956, 2289, 161, 336]
2	2,3,5,6,8	[3650, 749]	[2318, 581]	[3136, 1968, 170, 334]
3	1,3,4,5,7	[4059, 879]	[2384, 600]	[2951, 2377, 185, 284]
4	3,4,6,7,8	[3805, 879]	[2146, 632]	[3050, 2343, 181, 346]
5	1,2,5,7,8	[4071, 797]	[2546, 551]	[3011, 2179, 158, 288]
6	1,2,3,4,8	[4196, 807]	[2512, 624]	[3046, 2323, 188, 291]
7	1,4,6,7,8	[3876, 874]	[2181, 573]	[2913, 2437, 166, 353]
8	2,3,4,6,8	[3781, 789]	[2356, 607]	[3287, 2029, 171, 328]
μ		[3919, 821]	[2349, 593]	[3044, 2243, 173, 320]
σ		[180, 49]	[140, 27]	[121, 169, 11, 28]

Turns: 0-Straight walk, 1-Turning

Bend: 0-Upright, 1-Bending

Move: 0-Stand/Move slow, 1-Walk forward, 2-Walk backward, 3-Move lateral

μ - Average (Mean),

σ - Standard deviation

Table 8.4: Distribution of ground truth labels for feature vectors that were used in testing each of 8 trained model sets.

Training ID	Person IDs	Turn [0, 1]	Bend [0, 1]	Move [0, 1, 2, 3]
1	3,7,8	[3923, 558]	[4065, 375]	[1910, 1303, 115, 179]
2	1,4,7	[3989, 609]	[4190, 363]	[1722, 1652, 103, 182]
3	2,6,8	[3809, 486]	[3943, 352]	[1918, 1215, 92, 232]
4	1,2,5	[3731, 475]	[3888, 314]	[1800, 1247, 92, 173]
5	3,4,6	[3968, 573]	[4087, 400]	[1861, 1416, 118, 220]
6	5,6,7	[3670, 573]	[3878, 324]	[1820, 1271, 89, 219]
7	2,3,5	[3864, 477]	[3964, 377]	[1966, 1153, 108, 162]
8	1,4,8	[3736, 566]	[3959, 339]	[1575, 1562, 102,190]
μ		[3919, 821]	[2349, 593]	[3044, 2243, 173, 320]
σ		[180, 49]	[140, 27]	[121, 169, 11, 28]

The procedure used for training each of the three component classifiers was the same. We used OpenCV *trainSVM()* function with 10-fold cross-validation and the following parameters: C-SVM type classifier, RBF kernel, maximum iteration number 5000 and $\varepsilon = 0.00001$.

8.3.3 Results

For each training repetition (a row in Table 8.3), all three SVM component classifiers were evaluated using the same test data (the same row in Table 8.4). The test data contained activities from the three trials (*Turning*, *Movement directions* and *Bending*). The evaluation for each SVM classifier was done based on the comparison of its output class and the ground truth label. The output class for the SVM classifier is obtained by invoking the OpenCV *predictSVM()* function with a new feature vector input every 0.25 s, since average values that form feature vectors are calculated with that specific rate (every 8 samples at 30 Hz). For each prediction of the classifier, its output class and accompanying label were added to their respective vectors (*output_vec* and *label_vec*). The addition to these two specific vectors was done for every classification of the feature vectors produced for all the persons in the evaluation data. From these two vectors we produced a confusion table with actual and predicted values. A confusion table was subsequently used for direct calculation of statistical results. The tables with complete statistical results obtained in this way can be found in Appendix C (Table C.1 - Table C.10). In the discussion of the experiment results, we will present only average values extracted from those tables that depict how the classification method behaves in general.

The detection of turns achieved a moderate sensitivity of $[76.0 \pm 8.3]$%, and a very good specificity of $[97.0 \pm 1.0]$% (see Table 8.5). The visual inspection of turning behaviour shows that turn as an event is always detected, but the difference exits in the exact period of matching between the time when a turn is detected and its corresponding ground truth label. A charac-

Table 8.5: Averaged results for *Turn_SVM* classifier on 8 healthy people dataset. Extracted from Table C.1.

Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
76.0 ± 8.3	97.0 ± 1.0	78.7 ± 3.5	96.6 ± 1.3	94.4 ± 1.0	76.9 ± 3.6

PPV: Positive predictive value - TP/(TP+FP)
NPV: Negative predictive value - TN/(TN+FN)
F1 score - 2TP/(2TP+FP+FN)

teristic situation is depicted in Figure 8.5, which shows the case of a participant (ID=3) making a walk at low speed and with a wide 180° turn. The trajectory starts from the right side of the scene. Usually, the turn classifier recognizes the start or the end of the turn for one (0.25 s) or two (0.5 s) classification periods sooner than it is marked in the label. The way in which the data is labelled directly influences the result. Each turn was manually labelled, based on video observation, as the period from the first step when the person evidently started rotating their hips, until the first step when the person started to walk straight again. We observed the average rotational velocity around the vertical axis as the main feature for turning. We noticed that the trained classifier is actually more sensitive to a change in the person's orientation than a human annotator. There seems to be a learned threshold of rotational velocity inside the *Turn_SVM* classifier with the value of around 0.6 - 0.7 rad/s.

A lower sensitivity of turning detection causes a drop in rotational velocity during a wide 180° turn. This error is the outcome of the way in which the turn is labelled, and the fact that turning during data collection was not completely constrained. A wide turn is defined by two

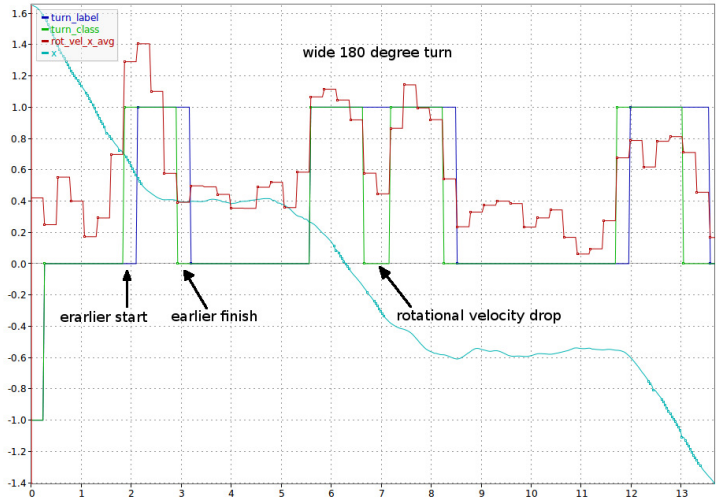


Figure 8.5: Classification of turning behaviour for one tracklet.

floor markers at the distance of 0.5 m. When going around those floor markers, a person can make a wide turn by slightly rotating during every step (similarly to walking on the perimeter of the circle), or he/she can make a turn with an almost 90° angle, followed by a straight step and another 90° angle turn (similarly to circumventing a rectangle). In the later case, a wide turn is practically broken into two smaller turns, which get separately detected by the *Turn_SVM* classifier. Since we labelled a wide turn as one ensemble, there is a period between two detected smaller turns when turning is not registered. An example of this can be seen at the 7 second mark in Figure 8.5. These false negatives influence the expressed sensitivity.

Statistical results for the binary SVM classifier targeting the recognition of bending behaviour are given in Table 8.6.

Table 8.6: Averaged results for *Bend_SVM* classifier on 8 healthy people dataset. Extracted from Table C.14.

Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
88.2 ± 4.5	98.6 ± 0.7	85.3 ± 6.5	98.9 ± 0.4	97.8 ± 0.8	86.6 ± 4.4

In Table 8.6 we can observe a very high average specificity of $[98.9 \pm 0.4]$ %, along with a bit lower average sensitivity and PPV of around 85%. Both sensitivity and PPV are dependent on the recognition of true positives, which means that the reason for the non-optimal performance lies in failed TP detections. Detailed comparison of the classifier output and its ground truth labels, shows that the lower TP detection performance has the same set of causes as the turn classifier. The situations pertaining to these causes are depicted in Figure 8.6, which shows a part of the *tracklet* captured during a bending trial.

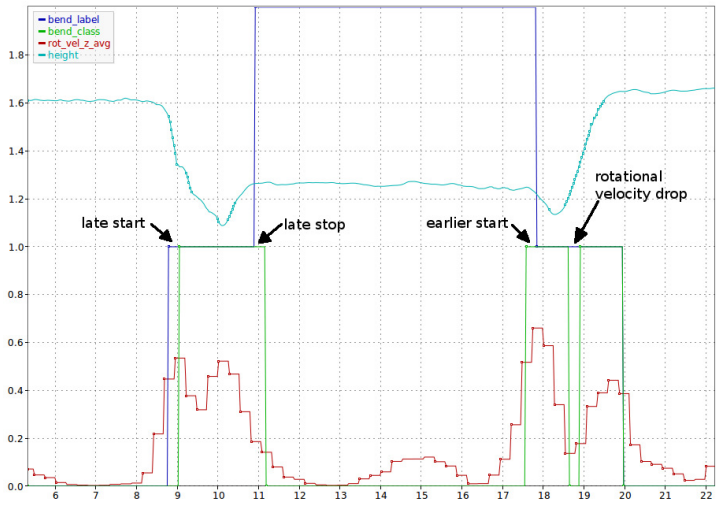


Figure 8.6: Classification of bending behaviour for one tracklet.

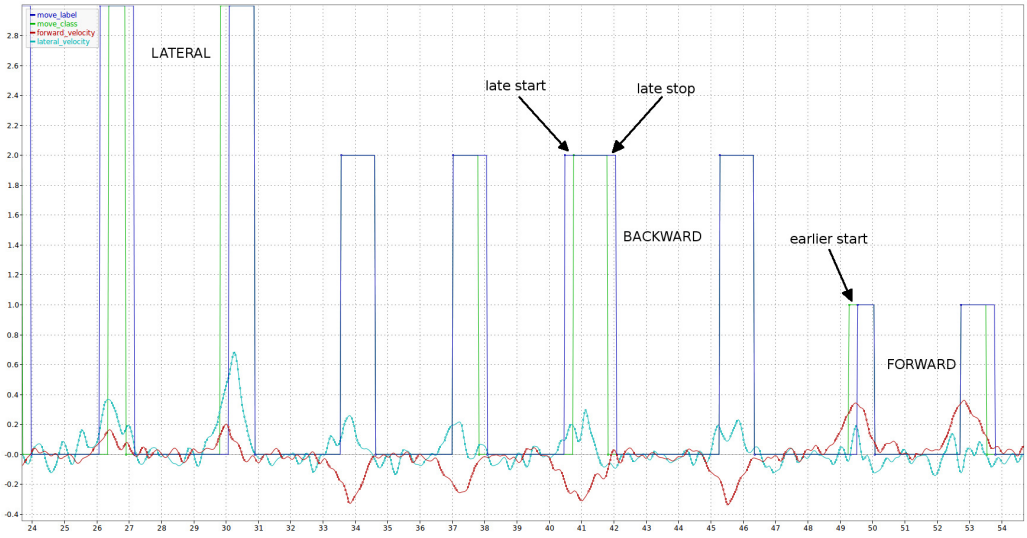


Figure 8.7: Classification of movement directions.

In Figure 8.6 we notice the sequence in which a person sits down, sits for a several seconds and stands up. The displayed data is the measured height, the measured transversal velocity, the bend label and the classifier output.

Forward bending and backward leaning during a postural transition are considered as a joint, singular target class that needs to be detected. The main dependent feature is the average rotational velocity around the transversal body axis. Similarly to turning, bending always gets detected as an event. However, there is a slight mismatch between the start and the finish of the bending event detected by the classifier and the manually set bend label. Figure 8.6 shows examples of a late start, late stop and earlier start which are all ± 0.25 s displaced in time towards their labels. All these events can be related with the average rotational velocity and its implicit threshold that is set about 0.5 - 0.6 rad/s. This implicit threshold value is also responsible for a missed detection in duration of one classification period that is visible at around the 19 second mark. The misdetection happens due to the transversal rotational velocity drop present between forward and backward bending during the sit-to-stand postural change.

The final component classifier is a 4 class SVM classifier for recognition of planar movement directions. Its statistical results are given in Table 8.7. The *Move_SVM* classifier achieves very good average sensitivity (> 96 %) in the recognition of *Stand* and *Forward* class, while lower sensitivity is achieved for *Lateral* and *Backward* movement classes. The original confusion table (Appendix C, Table C.11) reveals that *Stand* class is often detected instead of *Lateral* and *Backward* movements. The graph with overlaying output classes and labels (see Figure 8.7) shows that similarly to the *Turn_SVM* and *Bend_SVM* classifier, *Move_SVM* always detects the events of going backward or moving lateral, but the start and the end of these events are usually detected one classification iteration (0.25 s) too late or too early. The reason why non-

forward movements mostly gets confused with the *Stand* class is due to the nature of the test data. The test data contains non-forward movements collected only as discrete directional steps during the *Movement directions* trial. These steps always started and finished with a still standing. Due to small directional velocities at the beginning and the end of step movements, steps can not be detected exactly when they start. This produces a delay in the detection.

Table 8.7: Averaged results for *Move_SVM* classifier on 8 healthy people dataset. Extracted from Table C.3.

Class	Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
(0) Stand	96.7 ± 0.9	93.5 ± 1.2	94.2 ± 1.4	96.3 ± 0.7	95.2 ± 0.7	95.4 ± 0.9
(1) Forward	96.6 ± 1.5	98.1 ± 0.4	97.0 ± 0.7	97.7 ± 1.3	97.5 ± 0.7	96.8 ± 0.7
(2) Backward	82.4 ± 6.2	99.7 ± 0.1	98.4 ± 4.4	99.5 ± 0.2	99.3 ± 0.1	85.5 ± 2.5
(3) Lateral	70.2 ± 6.0	99.2 ± 0.3	84.6 ± 5.4	98.2 ± 0.4	97.6 ± 0.4	76.5 ± 3.8

Each time after all the three SVM component classifiers were trained with one of the training datasets, the hierarchical activity classifier was also evaluated. The statistical results for the seven-class hierarchical activity classifier are given in Table 8.8.

Table 8.8: Averaged results for seven-class hierarchical activity classifier on 8 healthy people dataset. Extracted from Table C.4 and C.5.

Class	Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
Stand/Slow walk (0)	93.8 ± 0.9	94.4 ± 0.9	92.1 ± 1.1	95.7 ± 0.6	94.2 ± 0.4	92.9 ± 0.7
Forward walk (1)	85.6 ± 3.9	96.0 ± 1.7	84.4 ± 3.1	96.6 ± 0.7	94.1 ± 1.1	84.9 ± 1.1
Non-forward (2)	66.0 ± 6.5	98.7 ± 0.3	80.1 ± 3.3	97.4 ± 0.5	96.4 ± 0.6	72.2 ± 4.0
Wide turn (3)	71.8 ± 11.5	97.5 ± 0.8	76.3 ± 3.2	96.8 ± 1.4	94.8 ± 1.0	73.4 ± 5.6
Spot turn (4)	73.5 ± 1.8	99.2 ± 0.2	69.1 ± 3.4	99.4 ± 0.1	98.6 ± 0.2	71.2 ± 1.8
Bend (5)	63.9 ± 3.7	98.9 ± 0.4	84.4 ± 4.9	96.7 ± 0.4	95.9 ± 0.5	72.7 ± 3.3
Sit (6)	98.5 ± 1.1	96.9 ± 0.3	81.1 ± 2.4	99.8 ± 0.2	97.1 ± 0.3	88.9 ± 1.5

The hierarchical classifier has lower average sensitivities in relation to the component classifiers, while specificities and accuracies are on the similar levels. Very high level of sensitivity (98.5 %) was achieved for the detection of the *Sit* class, which justifies putting *Posture_FSM* as the top level node. The interaction of the *Posture_FSM* and the *Bend_SVM* classifier at the second level of hierarchy, results with diminished sensitivity in the detection of the *Bend* class (63.9 % inside the hierarchical classifier vs. 88.2 % when it was tested independently). The table of confusion (Appendix C, Table C.12) shows that the decrease in sensitivity comes from the mistake of the *Bend* for the *Sit* class. Figure 8.8 illustrates this mistake. The inability of *Posture_FSM* to recognize the forward leaning phase during the sit-to-stand is the main cause of the error. *Posture_FSM* can not recognize forward leaning because during this motion the value of the person's height stays in between the upper and the lower threshold for sitting. Hence, SIT class is given as the output of *Posture_FSM*. Figure 8.8 shows that the *Bend_SVM* classifier correctly

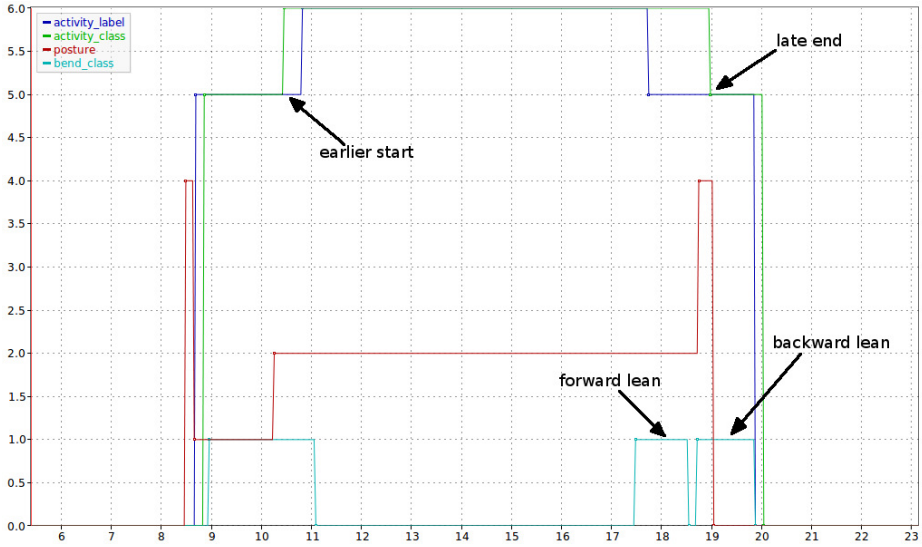


Figure 8.8: Classification of activity for stand-sit-stand sequence.

recognizes the *Bend* class during postural transitions. However, *Posture_FSM* overshadows the correct detections of bending, because of the configuration of the hierarchical classifier.

The detection of *Standing/on-the-spot move* remains at the similar level of sensitivity as it was for the *Move_SVM* classifier (93.8 % vs. 96.7%). The *Non-forward move* class generates the same problem of confusion with the *Standing* class, which was explained for backward and lateral movements in the *Move_SVM* classifier. The results for *Wide turning* and *Spot turning* classes are based on the logical conjunction of outputs of *Turn_SVM* and *Move_SVM* classifiers. Since both turning and forward walking (or standing) need to be recognized at the same moment, these classes are detected with 3-5 % less sensitivity (71.8%, 73.5%) compared to the *Turning* only in *Turn_SVM* (76.0%).

The evaluation of the hierarchical activity classifier on healthy people gives us information about the sensitivity and specificity levels for each activity that can be expected from the configuration of this particular decision tree. In the following section, we explore how the hierarchical activity classifier performs on the movement data of PD patients.

8.4 Evaluation on Parkinson's Disease Patients

8.4.1 Timed Up and Go Test

The *Timed Up and Go* (TUG) (Podsiadlo and Richardson, 1991) is a widely used clinical test for assessment of mobility and balance in elderly persons. The test consist of rising from a chair, walking the distance of 3 m with the preferred speed, turning for 180°, walking back to

the chair and sitting. Total duration of the time that a person needed to execute the test is the single measurement used as the indication of the person's locomotor performance. Because of its simplicity, the TUG test is often performed in the clinic setting to detect differences in performance between people with PD and elderly people without PD (Morris et al., 2001b). The test is an accurate assessment tool for quantification of gait and dynamic balance abilities. In the clinical setting, the medical personnel can obtain more information about the advancement of the PD of the patient by performing periodical TUG tests. Also, TUG is a good tool for the estimation of the risk from falls among the PD patients (Nocera et al., 2013).

For PD patients suffering from FOG, the TUG test contains several critical situations that can provoke a FOG episode (e.g. start of walking, turning). In fact, the TUG test contains almost the same activities as the ones that are recognized in our hierarchical activity classifier. The exceptions are backward and lateral movements. This makes clinical TUG tests a good opportunity to collect the data for evaluation of the activity classifier on its primary target group, the PD patients.

8.4.2 Clinical Dataset

In cooperation with the *Unit for Parkinson's and Movement Disorders* (esp. Unidad del Parkinson y Trastornos del Movimiento) of *Teknon Hospital* in Barcelona, we performed data collection during TUG tests with PD patients with diagnosed FOG. The participants pool consisted from 4 PD patients (3 male, 1 female) with the average H&Y score [$\mu = 2.35, \sigma = 0.38$]. The average age of participants was [$\mu = 67.8, \sigma = 6.9$] years, and their height [$\mu = 162, \sigma = 5.4$] cm.

Red tape markers were put on the floor of the experiment venue (see Figure 8.9) to label the start, the middle and the turning point for the 3 m walking path. Two Kinects were used for the experiment, along with a smartphone. The use of multiple Kinects in the clinical environment was a good opportunity to test the behaviour of our distributed system as a portable multimodal data collection platform. The Kinects were positioned in such a way that one of them was set close to the chair to have a good overview of the postural transitions (Figure 8.9a), while the other was set to have the visibility over the whole walking path (Figure 8.9b).

The position, orientation and fixation method of the smartphone on the patient's body was the same as it was during the collection of the activity dataset with healthy people (Section 8.3.1). A recording session for each participant lasted around 10 minutes and consisted out of 3 normal *Up and Go* walks in length of 3 m, and 3 *Up and Go* walks with the stop at the middle marker when going in each direction.

Raw video and depth data from the Kinects, along with the smartphone inertial data for each session, were recorded in the *.rosbag* format. The processing of recorded data in order to extract *tracklets* and obtain feature vectors, was done in the identical way as for the activity classifier for healthy persons.

(a) Image of the scene for TUG test from the viewpoint of the first camera.

(b) Image of the scene for TUG test from the viewpoint of the first camera.

Figure 8.9: Viewpoints of two cameras set in the space for clinical rehabilitation.

8.4.3 Training Method

The collected number of samples of activities in the clinical dataset was not big enough too support both training and testing of the classifier with the same data. Thus, the training of the SVM classifiers was performed using the dataset of 8 healthy people that was obtained earlier in the laboratory (Section 8.3.1), while evaluation of the classifiers was done with the PD patient data. The complete distribution of the labels in the training dataset that includes all 8 persons is given in Table 8.9.

Table 8.9: Distribution of labels in the whole dataset of 8 healthy persons.

Training ID	Person IDs	Turn [0, 1]	Bend [0, 1]	Move [0, 1, 2, 3]
1	1,2,3,4,5,6,7,8	[6271, 1314]	[3758, 949]	[4870, 3589, 276, 512]

The overall methods and SVM parameters for training each SVM component classifier were kept identical to the ones that were described in Section 8.3.2.

8.4.4 Results

After we trained all the three component SVM classifiers, we evaluated each of them separately on the TUG data of each of the 4 PD patients. The distribution of the labels in the patient data is shown in Table 8.10. The last two columns of the table, filled with zeros, demonstrate that none of the patients executed backward or lateral movements during TUG tests. It was not possible to calculate sensitivity for these two types of activities, since we did not have true positive classifications.

Table 8.11 demonstrates a decrease in the average sensitivity of detection of turning.This decrease is the consequence of FOG being experienced by one of PD patients (see Appendix C,

Table 8.10: Distribution of labels in PD patient data.

Patient ID	Turn [0, 1]	Bend [0, 1]	Move [0, 1, 2, 3]
1	[532, 97]	[590, 39]	[265, 204, 0, 0]
2	[746, 95]	[764, 77]	[323, 205, 0, 0]
3	[694, 83]	[704, 73]	[150, 221, 0, 0]
4	[728, 133]	[720, 60]	[195, 336, 0, 0]
μ	[675, 102]	[695, 62]	[233, 244, 0, 0]
σ	[98, 22]	[74, 17]	[76, 63, 0, 0]

Table C.6, 2^{nd} row). The patient suffered FOG episodes when performing 180° turns. Two types of behaviour that caused errors were observed. His turns were sometimes broken into discrete short turning segments. In between the turning segment the patient was standing still trying to unfreeze. Short turning segments lasted too short to be detected by the *Turn_SVM* classifier. Another type of FOG involved turning very slowly while simultaneously performing shuffling steps. During these turns the measured rotational velocity was too low for the correct turn detection in the classifier. Individual results of other patients who did not have a turn-specific FOG are at a similar level (approx. 70-75%) as the results for healthy people in Table 8.5.

Table 8.11: Averaged results for *Turn_SVM* classifier on 4 patient clinical dataset. Extracted from Table C.6

Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
65.3 ± 23.7	98.40 ± 1.7	84.8 ± 13.7	95.3 ± 2.4	94.3 ± 2.6	71.8 ± 20.1

The evaluation of the *Bend_SVM* classifier on PD patient data (Table 8.12) gives a 17% lesser average sensitivity in comparison to previous results with healthy people (Table 8.6) Detailed classification results per patient (Appendix C, Table C.7) revealed that the lower average results are not caused by any patient in particular. We compared the graphs between *Bend_SVM* outputs and ground truth labels. We noticed the reasons for the poorer performance. The occurrence of the bending posture change was always recognized. The sensitivity is lost because of the mismatch in timing of the start of the detected bending event and the label for the start of the labelled postural transitions. Each postural transition that contained both the forward and the backward leaning, was labelled as one *Bend* class event. The speed of postural transitions in PD patients is lower than the speed of postural transitions in healthy people used for *Bend_SVM* training.

In Table 8.13 the *Forward* class is detected with a very high sensitivity (> 96 %), but with a somewhat lesser specificity (> 84 %). The confusion table for the *Move_SVM* classifier (Appendix C, Table C.13) reveals that in around 14 % of the cases the *Stand* class gets confused for

Table 8.12: Averaged results for *Bend_SVM* classifier on 4 patient clinical dataset. Extracted from Table C.7.

Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
70.8 ± 6.9	97.1 ± 1.7	70.8 ± 15.6	97.2 ± 1.0	94.8 ± 1.1	69.7 ± 7.4

the *Forward* class. Besides the usual error of having the start and the stop of the forward movement detected one classification period later, the forward movement was also detected when FOG manifested as *shuffling*. The patient movement during *shuffling* is a border line case between the *Stand* class and the *Forward* class, for which it is very difficult to set the correct label.

Table 8.13: Averaged results for *Move_SVM* classifier on 4 patient clinical dataset. Extracted from Table C.8.

Class	Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
(0) Stand	83.3 ± 7.3	98.5 ± 1.2	98.4 ± 0.8	86.6 ± 4.1	91.6 ± 2.3	90.1 ± 4.3
(1) Forward	96.2 ± 2.0	84.7 ± 6.8	87.3 ± 3.8	95.1 ± 3.3	91.0 ± 1.6	91.5 ± 1.3
(2) Backward	-	99.9 ± 0.1	0.0 ± 0.0	99.5 ± 0.0	100.0 ± 0.1	-
(3) Lateral	-	98.1 ± 1.4	0.0 ± 0.0	100.0 ± 0.0	98.1 ± 1.4	-

Table 8.14 depicts the behaviour of the complete activity classifier. The *Sit* class retains a very high sensitivity (> 94%). The relation between the *Sit* class and the *Bend* class during postural transition stays unchanged, for the same reason that was already explained when we interpreted Table 8.8. Average sensitivities for the *Stand* and *Forward walk* classes are 5-7% percent bellow sensitivities achieved for the same classes in the *Move_SVM* classifier (Table C.13). The confusion table (Appendix C, Table C.14) shows that close to 7% of the actual *Stand* class events were mistaken for the *Spot turning* class. We conclude that the reduction in the sensitivity is caused by the configuration of the decision tree. The *Turn_SVM* classifier changes the final output class from *Stand* to *Spot turning* when it makes an incorrect prediction. A similar explanation is valid for lower sensitivity and higher specificity of the *Forward walk* class. The detection of the *Forward walk* class is directly influenced by the *Bend_SVM* classifier (2.4% error), which is positioned closer to the top of the decision tree.

Table 8.14: Averaged results for seven-class hierarchical activity classifier on 4 PD patient dataset. Extracted from Table C.9 and C.10.

Class	Sensitivity [%]	Specificity [%]	PPV [%]	NPV [%]	Accuracy [%]	F1 score
Stand/Slow walk (0)	74.0 ± 10.5	95.1 ± 2.3	78.4 ± 5.7	94.3 ± 1.7	91.8 ± 2.0	76.0 ± 7.9
Forward walk (1)	91.4 ± 3.2	92.9 ± 1.4	82.7 ± 4.7	96.5 ± 1.7	92.5 ± 1.0	86.7 ± 2.5
Non-forward (2)	-	99.2 ± 0.6	0.0 ± 0.0	100.0 ± 0.0	99.2 ± 0.6	-
Wide turn (3)	37.1 ± 13.0	97.6 ± 1.4	37.7 ± 7.0	97.5 ± 1.6	95.3 ± 2.7	36.8 ± 8.7
Spot turn (4)	41.3 ± 23.1	98.2 ± 0.5	64.1 ± 24.3	94.1 ± 2.4	92.8 ± 2.0	49.0 ± 24.4
Bend (5)	64.4 ± 7.8	96.4 ± 1.4	63.1 ± 15.2	96.7 ± 0.4	93.7 ± 1.4	63.6 ± 11.4
Sit (6)	94.3 ± 4.3	96.5 ± 0.2	91.2 ± 3.3	97.6 ± 1.7	96.0 ± 0.9	92.7 ± 3.4

For both turning classes, the sensitivity is bellow 50%. This is a significant decline in performance in relation to the detection rates that were achieved with healthy people. The 3^{rd} row of the activity classifier confusion table (Appendix C, Table C.14), shows that in more than 58% of cases, the *Wide turning* class was confused for the *Forward walk* class. This happens because the *Turn_SVM* classifier did not execute correct positive detection at the exact times when it was expected. The deterioration in performance of *Turn_SVM* on the clinical dataset affects activity the classification even more negatively, if we add the constraint of logical conjunction with a correct classification from the *Move_SVM* classifier. In the 4^{th} row of the confusion table (Table C.14), we can see that the detection of the *Spot turn* class is partially influenced by the errors originating from all SVM classifiers that are closer to the top the decision tree; *Turn_SVM* (21.1% error), *Move_SVM* (18.1% error) and *Bend_SVM* (10.5% error). The reasons for these errors are specifics of turning under FOG, but also the specifics of the TUG test dataset. In the collected TUG data, turning for 180° always came before bending and was sometimes even executed simultaneously.

8.5 Summary

In this chapter we described the complete implementation and evaluation of algorithms/methods for fast and timely recognition of the chosen set of activities. The most distinctive features of our activity recognition approach are:

- The inclusion of specific activities related to FOG (e.g. backward and lateral movement, "restless standing", different walking speeds);
- The use of multi-modal data that combines 2D positions from video tracker with wearable inertial sensor data;
- Very short buffer for activity recognition of length 0.5 s, with a very fast processing loop of 0.25 s.

During the systematization of the activity types (presented in Section 4.1.3), we made the distinction between static postures, dynamic postural transitions, actions and activities. Following this systematization, first we implemented the classifier for the detection of static postures and postural transitions based exclusively on video data. The developed video-based posture classifier complements the operation of the posture classifier based on inertial data that was developed by Rodriguez-Martin et al. (2013). Our expectations are that the two classifiers could be used together for improved postural identification, as well as for the improved person re-identification between cameras. We implemented the posture classifier as a finite state machine. The FSM uses as the input person's height data and has three height thresholds as the parameters.

The FSM for posture identification was the first building block necessary for the development of the hierarchical activity classifier. The decision tree for the hierarchical classifier was

built using two binary SVMs and one multi-class SVM as decision functions on its nodes. The method for training each of the SVM component classifiers was presented and evaluated on a laboratory dataset of 8 healthy participants. The results of the evaluation vary between classes, from very good (e.g. *Sit*, 98.5% sensitivity, 96.9 % specificity) to moderate (e.g. *Non-forward move*, 66.0% sensitivity, 98.7% specificity). These results are satisfying, having in mind the very short buffer time used in the algorithm.

The second evaluation of the new multi-modal activity recognition method was performed using the data from the clinical TUG tests by 4 PD patients. Reduced performance was noticed in comparison with the results of the activity recognition with the data from the healthy participants. The main factor for the decline in performance was that the component SVM classifiers were trained with the data of healthy people. Healthy people generally have faster and more fluid movements than PD patients, both in postural transitions and walking. This fact is especially reflected on the results of one PD patient (ID = 2) that experienced several severe episodes of FOG during the clinical data collection.

We conclude that the better classification results for a PD patient might, naturally, be achieved if the data of the same patient was used for training. To test this, it is necessary to gather a lot of very diverse data from the same patient by a long-term observation. Since the TUG trials in a clinic had a very short duration, there was no opportunity for such data collection.

An encouraging realization is that the events for each activity still get recognized in the majority of cases, even on the clinical patient data. The reported errors are caused by the off-timing in detection of the start and the stop for the particular activity events. However, the order of the events recognized by the classifier and the resulting activity profile are very similar to the activity profile formed by the ground truth labels. In the next chapter we will examine to which extent the errors in the exact timing of an event detection influence the contextually-based FOG detection.

9

Using Context for Improved Freezing of Gait Detection

In this chapter we present the final FOG contextualization algorithm and its evaluation. The contextualization algorithm combines FOG detections from the Moore-Bächlin algorithm with the activity detections from the hierarchical activity classifier. Hence, the first section of the chapter is dedicated to the description of the exact implementation of Moore-Bächlin algorithm in our system, while the FOG contextualization algorithm is introduced afterwards. The benefits of the FOG contextualization were evaluated on the data collected in the homes of the PD patients. Unlike the trials done in a laboratory or a clinic, which normally use the same kind of walking trajectories and the same number of repetitions for all the participants, during the home visits it was not possible to replicate the identical trial conditions. Data collection trials had to be tailored taking into account the uniqueness of PD, FOG and home environment of each patient. In the section about the experiment description, we provide some of our insights and experiences with such process. The collected data, along with the clearly defined evaluation method allows us to finally evaluate the new FOG contextualization algorithm. We use a patient-specific approach where each patient is treated as a specific case. The results presented at the end of this chapter are representing the final results of the thesis.

9.1 Moore-Bächlin Algorithm Implementation

In the implementation of the Moore-Bächlin algorithm there has to be a balance between the sampling frequency, the analytic window length, and the resolution of the FFT. To set the parameters for the Moore-Bächlin algorithm, we use the results of the most recent study by Moore et al. (2013). This study explored the effects of the *freeze threshold* and the window size, as well as the use of a multi-segmental sensor placement, on sensitivity and specificity of the FOG detection.

The most relevant results presented in Moore et al. (2013) in relation to our own FOG detection setup are the ones for a single acceleration sensor placed on the lumbar back region of the body. The authors used four different analytic window lengths of 2.5, 5, 7.5 and 10 s. For each window length they varied the freezing threshold (*FThr*) in the range between 0.5 and 7.0, with the step of 0.5. As the experiment outcome, the sensitivity of the sensors to FOG events was related inversely to the size of the analytic window. The single lumbar sensor exhibited robust sensitivity and specificity (70-80%) inside the *FThr* range 3.5-5, when the larger analytic window sizes (7.5 and 10 s) were used. With the smallest window size (2.5 s,) the sensitivity was very high (90-100%) for all FI values between 0.5 and 7. With the same window length, the specificity was very low, rising linearly from 0% to around 40% for the same *FThr* range between 0.5 and 7. The smallest window length value in the experiment (of 2.5 s) offers the best guarantee that the algorithm will be able to achieve high sensitivity. The additional benefit is also the reduced latency of detection that is the result of using a shorter analytic window. Therefore, with the assumption that the problem of low specificity can be solved by eliminating FPs with activity context, we targeted to use of the 2.5 s analytic window length in our implementation.

We implemented the Moore-Bächlin algorithm in the `FOG_Algorithm` node. The node takes as input the IMU data provided by `ROS_Android` node with 100 Hz. By leaving out every second sample, we artificially reduce the sampling frequency from 100 Hz to 50 Hz. The 50 Hz frequency or less was used in other works (Moore et al., 2013; Rodríguez-Martín et al., 2014), and it is sufficient to properly capture the FOG-relevant characteristics of movement. Furthermore, a smaller frequency reduces the number of samples necessary to fill the buffer of the certain time length, consequently reducing the FFT calculation processing requirements.

The accelerometer data are extracted from the IMU messages and are used in the calculation of the FFT. The subsequent calculation of the freeze threshold (*FThr*) and the power threshold (*PThr*) is done according to Equation 3.1 in Section 3.2.1. The output rate of the node is defined with the time length of the analytic window and the window overlap (50%). The analytic window time length was set to the exact value of $T_{win} = 2.56$ s. This length is the closest that we can approach to the target length of 2.5 s, when using the sampling frequency $f_{sample} = 50$ Hz and the number of samples for the FFT algorithm $N_{FFT} = 128$. The described parameter setup, results in the `FOG_Algorithm` node output frequency $f_{out} = 0.8$ Hz. The output from the node is formatted into messages of type *FOGData* (see Appendix E).

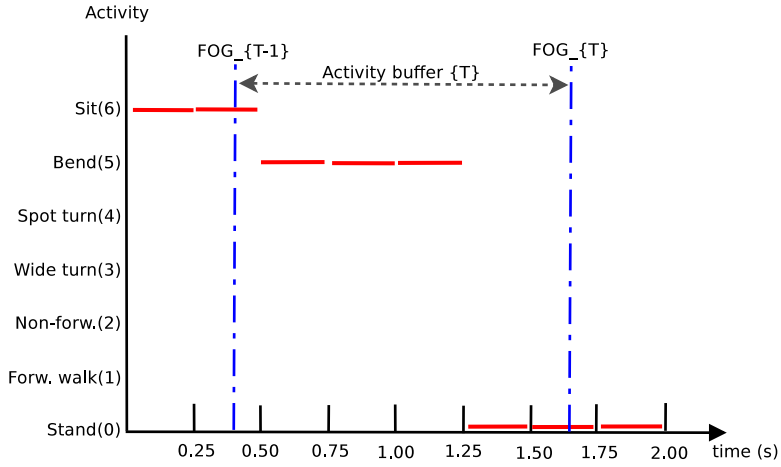


Figure 9.1: Temporal relation between activity and FOG messages.

9.2 Algorithm for Contextualization of FOG Detection

Two inputs for the FOG detection algorithm are the activity class calculated every $\Delta T_{Act} = 0.25$ s by the algorithm described in Chapter 8 and the *FOGData* messages calculated every $\Delta T_{MB} = 1.25$ s as described in Section 9.1. To describe the temporal relation between the two outputs, we use Figure 9.1. The example on the figure shows a typical sit-to-stand postural transition. A person starts in the *Sit* posture (2 detections), gets up which causes the recognition of the *Bend* class (3 detections) and finishes in the *Stand* posture (2 detections).

Between the two updates of the Moore-Bächlin algorithm, there is on average 6 outputs of the hierarchical activity classifier. These outputs are recorded in the code sequence inside the *activity_buffer* field. With the newest calculation of FI (designated FOG_T inFigure 9.1) the sequence of stored activities in the *activity_buffer*), can be queried to check if the activities recorded since the last output update of the Moore-Bächlin algorithm support or reject the newest predicted FOG class.

The sequences that support FOG are called "whitelisted", while the sequences that reject FOG are called "blacklisted". Some of the examples of possible "blacklisted" and "whitelisted" sequences are presented in Table 9.1.

Table 9.1: Example of some of the possible "whitelisted" and "blacklisted" activity sequences.

Blacklisted		Whitelisted	
Sequence code	Description	Sequence code	Description
1-1-1-1-1-1	Forw. walk	0-0-0-0-0-0	Stand
0-0-1-1-3-3	Stand - Forw. walk - Turn wide	6-6-5-5-0-0	Sit - Bend - Stand
6-6-5-5-5-1	Sit - Bend - Walk	1-1-0-0-4-4	Forw. walk - Stand - Turn spot

At first, our intention was to implement all the possible "whitelisted" and "blacklisted" sequences. However, the activity buffer with $r = 6$ members, where each member can have $n = 7$ different activity codes, gives the total of $n^r = 117649$ possible permutations. It was necessary to simplify the approach, since it is impossible to completely fill the database with all the possible sequences of activities.

We aim to improve the specificity of the Moore-Bächlin algorithm by minimizing the number of the FP detections. One way to confirm that the newest FOG detection is indeed TP, and not FP, is to validate that the current activity of the patient allows for FOG to be happening at the moment of detection. We take that a positive FOG detection is true only when the patient was upright and did not make any significant movement in the last 0.5 s. Hence, when forming the mask for the "whitelisted" activity sequence only the last two activity codes in the sequence have to be taken into consideration. Two possible activities that signal potential FOG are *Stand(0)* and *Spot turn(1)*. The examples of four possible masks produced with combination of these two activities are presented in Table 9.2.

Table 9.2: Masks for "whitelisted" activity sequence.

Mask sequences			
x-x-x-x-0-0	x-x-x-x-0-4	x-x-x-x-4-0	x-x-x-x-4-4

With the known "whitelisted" activity sequence masks, the algorithm for contextualized FOG detection executes as presented in Algorithm 1.

9.3 Evaluation of FOG Contextualization

The main objective of the experiment was to test the operation of the complete distributed system in various home environments and to accurately determine sensitivity and specificity for the new algorithm using home collected patient data.

9.3.1 Participants

The experimental study included 3 male PD patients experiencing FOG. The patients voluntary accepted to join the study, and to allow the researchers to setup the system and to record the data in their homes.

The characteristics of the patients were as follows: mean age [μ=68.4, σ=7.4], disease duration [μ=12.3, σ 8.7] years, H&Y stage [μ=2.8, σ=0.7]. None of the patients described any increase in the freezing behaviour following the administration of their usual dopaminergic therapy.

```
Data: bool moore_fog_class, double[ ] feature_vector, int[ ] sequence_mask
Result: bool final_fog_class
while patient_track_exists do
    hierarchical_activity_classifier.predict_activity(feature_vector[ ]) →
    activity_buffer[last];
    if moore_fog_class == new message then
        if (activity_buffer[ ] ∧ sequence_mask[ ]) then
            context_fog_class = True;
        else
            context_fog_class = False;
        end
        if moore_fog_class ∧ context_fog_class then
            final_fog_class = True;
        else
            final_fog_class = False;
        end
    else
        continue;
    end
end
```

Algorithm 1: FOG contextualization algorithm in the `Context` node.

9.3.2 Home Visit Procedure

A home visit for data collection was scheduled to last up to 4 hours for each participant. This time period granted enough time to researchers to install and remove the sensor equipment, record sensor data during walking trials, and interview the patient twice (once before and once after the trials). After the arrival to the patient's home, the patient was familiarized with the experiment via the description in the participant information sheet. Prior to any data collection, the patient and his caretaker signed a consent form. Before the walking trial the researchers had to find out more about the patient, his PD history, the characteristics of his FOG, and his life in the home. To facilitate this step, the patient was asked to answer the custom questionnaire (see Appendix D). Besides the usual questions about the PD and the FOG manifestations, the questionnaire included the sections about the indoor FOG triggers, the methods that the patient uses to exit from a FOG state, and technology acceptance.

9.3.3 Home Environment and Locomotion

The information from the questionnaire and the initial interview helped us to detect the potential places in the home where a FOG might occur and to choose optimal positions for the

placement of cameras in the patient's living space. Figure 9.2 shows two scenes from homes of each of the three participants.

The participants reported that at home they spend the majority of the time in the living room. Their preferred micro-locations are usually a sofa or the chairs around the dining table. The most often used trajectories involve going to the kitchen and to the toilet. We identified the potential FOG triggering places in the home of each patient. In Figure 9.2a we can see Patient1 standing between the living room and the kitchen doorway. This is a high risk situation for FOG occurrence, since both doorways are FOG triggers. An additional FOG trigger is the 90° turn when going from one door to another. The living room in Figure 9.2b provides several narrow passages and tight spots. One narrow passage is between the dinning table and the red sofa chair. The patient mentioned that FOG episodes occur at that spot more often when there is another person sitting in the sofa chair. The small table in the middle of the room creates two narrow passages: between the small table and the sofa, and between the corner of the small table and the chair next to the dining table. If the chair is pulled outward from the table because someone is sitting in it, the passage gets too narrow for normal walking.

Similarly, in the living room of Patient2 (see Figure 9.2c) a narrow passage is formed by the small table and the corner sofa. When the patient approaches the sofa from the kitchen, he can go straight in the longer narrow passage, or he can go around the table and turn into the shorter narrow passage. Both passages are potential FOG triggers. Besides the narrow passages, Patient2 reported open space hesitation at the location near the open part of the low wall (prior to approaching the sofa). The wall opening is the beginning of the stairs leading to the ground floor. Several walking tests during the data collection showed that the missing part of the wall has influence only when the patient walks close to it (< 1 m distance). On the other part of the same living room (see Figure 9.2d), the space between the two doorways and the beginning of the stairway to the upper floor is another micro-location with a high risk for FOG. The kitchen and the living room are connected with a straight line path, while all the other trajectories, such as going from the kitchen to the bathroom or from the living room to the stairway, require additional 90° turn.

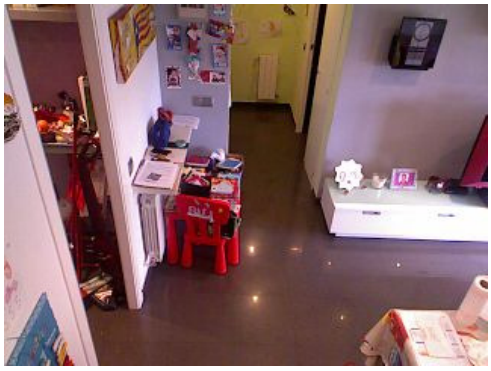
In the living room of Patient3 (see Figure 9.2e) the configuration of the furniture in the space is set to facilitate movement and to minimize the number of FOG zones. When the patient wants to go from his usual sitting spot on the sofa to the kitchen, he needs to make a 90° turn around the corner of the dinning table. The space for turning is not limited on the side away from the table and the turn does not have to be sharp, which considerably helps to avoid FOG. The critical spot in this environment is the sliding kitchen door (Figure 9.2f). With the presence of the sliding door, the doorway passage becomes even narrower. Just after the doorway and inside the kitchen, there is a fridge on the left side, requiring an additional 90° turn on the spot when the patient wants to use it. The FOG episodes occur with the equal tendency on either side of this kitchen doorway.

(a) Patient1. Camera1. Hall with doors towards living room (straight) and kitchen (right).

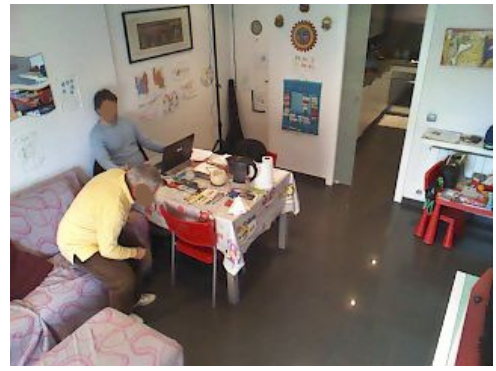
(b) Patient1. Camera2. Living room.

(c) Patient2. Camera1. Living room.

(d) Patient2. Camera2. Living room connection to stairs (left), kitchen (straight), and bathroom (rigth).



(e) Patient3. Camera2. Living room connection to kitchen (door left) and hall towards bedrooms.



(f) Patient3. Camera1. Living room.

Figure 9.2: Home environment coverage with Kinects.

9.3.4 Data Collection

The prototype of the monitoring system consisting of two Kinects and a Galaxy Nexus smartphone was used to record raw sensor data.The wearable sensor positioning and the data recording parameters were the same as in the clinical experiment described in Section 8.4.2.

During the data collection the patients were in the clinically-defined *OFF* state that follows the withdrawal of dopaminergic therapy. They executed a version of *Up-and-Go/FOG Provocation Test* that was adapted to their living environment. Instead of walking along the straight 5 m path and passing through an artificially created FOG zone at 2.5 mark, the patients were instructed to follow their usual walking trajectories inside the home. The trajectories that pass through natural FOG trigger zones were repeated more often. A sofa in the living room was usually used as a starting point, with sitting as a starting posture. The patient was instructed to go to the kitchen, to the bathroom, or any other location of interest in his usual manner of locomotion. A variation of this approach involved setting an additional seat in the kitchen. This seat was sometimes used as a secondary starting point towards the living room, the bathroom or the hall.

Even with several different trajectories, after some time the walking trial can become too repetitive for the patient. Multiple postural transitions and a lot of movement in a short time span are very cumbersome, especially in the *OFF* state. To counter these effects, the patients were encouraged to take rest as much as they wanted. Although the data collection sessions were usually 2 hours long, we collected on average around 15 minutes of movement data per patient, while for the rest of the time the patient was sitting.

9.3.5 Ground Truth Labels

A clinician experienced in FOG used his best clinical judgement to identify FOG episodes based on the video recordings. The editing tools from the ROS *rxbag* package enabled slowing down, pausing and rewinding of videos in order to conduct a detailed observation. The onset of a freeze was tagged by value '1' and the end of the episode was tagged by value '0'. The episode duration was observed with the time resolution in hundreds of milliseconds (e.g. 1.27 s, 3.29 s). Besides FOG labels provided by the clinician, it was necessary to obtain the ground truth labels for patients' postures and activities. The labelling of the postures and activities did not require special medical personnel.

9.3.6 Evaluation Method

The evaluation was done by comparing on the same dataset the sensitivity and specificity of the FOG contextualization algorithm with the sensitivity and specificity of the Moore-Bächlin algorithm. To ensure optimal results, the *FThr* and *PThr* parameters were optimized for each patient separately.

The raw sensor data from the home experiments was replayed with the monitoring system simulating a real-time operation. For each tracklet we merged the patient's contextual data (position, orientation, detected activity), raw IMU data, the output of the *FOG_Algorithm* and the ground truth labels into a *.rosbag* file with a new message type - *ContextData* (See Appendix E). A dedicated application was produced to automatically read all *ContextData* messages for one patient in a batch mode, to find optimal threshold parameters, to compare the output of the optimized FOG algorithm against the ground truth label, and to calculate the statistical results for the classification.

We used a *window-based* approach to compare the detection algorithm outputs and the ground truth labels. For each activity window with a duration of 1.25 s, there was a new binary FOG value. For the same timestamp, we compared the new binary FOG value and the corresponding ground truth label. Table 9.3 shows the total number of processed analytic windows for each of the three patients and the distribution of the analytic windows with the positive and negative FOG ground truth labels.

Table 9.3: Distribution of FOG labels and average episode duration for each patient.

Experiment	Duration [s]/[min]	# Windows	# True FOG	# False FOG	# FOG episodes	Avg. ep.duration [s]
Home 1	890/14.8	712	10	702	6	2.11 ± 0.52
Home 2	860/14.3	688	143	545	34	6.51 ± 5.41
Home 3	716/11.9	573	13	560	13	1.33 ± 0.89
Total	2466/41.1	1973	166	1807	53	-

To find the optimal *FThr* and *PThr* parameters, we used the grid search optimization. During the grid search, a pair of sensitivity and specificity values (*Sens*, *Spec*) is obtained for each pair of the threshold parameters (*FThr*, *PThr*). The final result of this process is a pair of tables (such as Tables 9.4 and 9.5), that contain the calculated sensitivity and specificity as the function of the threshold parameters.

Table 9.4: Values of sensitivity for Moore-Bächlin algorithm for Patient2 as the function of threshold parameters.

		FThr								
		0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
PThr	0.0	100.0	100.0	97.2	90.2	80.4	65.0	55.2	47.6	38.5
	0.5	88.8	88.8	87.4	82.5	74.1	61.5	51.7	44.1	35.7
	1.0	70.6	70.6	69.9	67.8	60.8	50.3	43.4	37.1	29.4
	1.5	53.8	53.8	53.1	52.4	49.0	40.6	35.0	30.1	25.2
	2	30.8	30.8	30.1	30.1	29.4	24.5	21.7	17.5	14.7
	2.5	22.4	22.4	21.7	21.7	21.7	16.8	16.1	12.6	10.5
	3.0	15.4	15.4	14.7	14.7	14.7	11.2	11.2	9.1	7.0

Table 9.5: Values of specificity for Moore-Bächlin algorithm for Patient2 as the function of threshold parameters.

		FThr								
		0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
PThr	0.0	0.0	3.5	10.5	20.7	35.4	50.1	61.1	68.4	77.1
	0.5	52.5	55.2	59.1	64.8	70.8	77.1	81.3	85.3	90.1
	1.0	69.0	70.6	72.7	75.6	79.3	82.9	86.4	8 9.4	92.7
	1.5	80.6	81.8	82.9	84.8	86.6	89.2	91.4	93.9	96.1
	2	89.4	90.3	90.5	91.6	92.1	93.2	94.7	96.7	98.0
	2.5	93.8	93.9	93.9	94.5	94.9	95.4	96.7	97.8	98.9
	3.0	96.1	96.1	96.1	96.3	96.7	97.1	98.0	98.9	99.4

Rodríguez-Martín et al. (2014) proposed that the optimal solution in terms of balance of sensitivity and specificity, can be found using the information from a pair of the obtained grid search result tables and the rule of geometrical mean according to the following equation:

$$(PThr, FThr)_{optimal} = max\,(\sqrt{Sensitivity_{(PThr,FThr)} \times Specificity_{(PThr,FThr)}}) \\ Subject\ to:\ Sensitivity > 0.7, Specificity > 0.7 \tag{9.1}$$

In Tables 9.4 and 9.5 we highlighted the optimal sensitivity and specificity values of the Moore-Bächlin algorithm Patient2, that are obtained after the equation 9.1 has been applied. We can note that $PThr = 0.5$ and $FThr = 2.0$ are the optimal threshold parameters in this particular case.

9.4 Results

For each patient we found the optimal threshold parameters for the original Moore-Bächlin algorithm and the optimal threshold parameters for the FOG contextualization algorithm. Each of the two algorithms were separately evaluated using the *window-based* approach to produce statistical results in a form of the sensitivity, specificity and general accuracy. Table 9.6 shows the results for the first patient.

For the first patient we captured only 6 separate FOG episodes spread over 10 analytic windows. Out of 6 instances of freezing, 3 of them were a hesitation after getting up from a sofa, while the other 3 manifested as a shuffling gait during turns. The Moore-Bächlin algorithm successfully detected FOG in 9 out of 10 analytic windows. The contextualized FOG algorithm missed one additional TP detection, which lowered its sensitivity for an additional 10%. After we have reviewed the video and compared the FOG contextualization output signals with the ground truth labels, we noticed that this additional error happened due to a falsely detected

Table 9.6: Patient1: Comparison of the best results of Moore-Bächlin algorithm and our new method with added activity context.

Algorithm	*FThr*	*PThr*	TP	FP	FN	TN	Sens. [%]	Spec. [%]	Acc. [%]	F1 score
Moore-Bächlin	1.0	1.0	9	180	1	522	90.0	74.4	74.6	0.09
Added context	0.5	1.0	8	23	2	670	80.0	96.7	96.4	0.39
Difference	-	-	-	-157	+1	+157	-10.0	+22.3	+21.9	+0.30

Forward walk (1) activity at the moment when the patient was still experiencing a start hesitation. A 10 % decrease in the sensitivity of the new algorithm was compensated with a 22.3 % improvement of the specificity. The specificity raised because the number of FP detections was reduced from 180 to 23 instances. The positive influence of the contextualization on a such high portion of the FP detections reflected also positively on the improvement of the overall algorithm accuracy (+21.9%) and the F1-score (+0.30).

During the data collection with Patient2, we recorded many various examples of a domestic FOG. The FOG episodes were longer and more pronounced. The type and the intensity of the episodes also reflected on the classification results for Patient2 that are presented in Table 9.7.

Table 9.7: Patient2: Comparison of the best results of Moore-Bächlin algorithm and our new method with added activity context.

Algorithm	*FThr*	*PThr*	TP	FP	FN	TN	Sens [%]	Spec [%]	Acc [%]	F1 score
Moore-Bächlin	2.0	0.5	106	159	37	386	74.1	70.8	71.5	0.52
Added context	0.5	0.5	108	113	35	432	75.5	79.3	78.5	0.59
Difference	-	-	-10	-79	+10	+79	+1.4	+8.4	+7.0	+0.07

The freezing episodes in this dataset take a fifth of the time that the patient spent walking (143/688 detections). Various types of FOG manifestation were captured: leg trembling in narrow passages and doors, shuffling and suddenly stopping in front of passages, and suddenly stopping in the middle or in the end of a 90° turn. Such data provide much better statistical support for sensitivity calculation, compared to the previous case. The results show that the contextualization algorithm achieved slightly better sensitivity (+1.4 %), while at the same time it also improved the specificity for additional 8%. In Table 9.7 besides the statistical results, we can also observe the difference between the chosen optimal threshold parameters for the two algorithms. The Moore-Bächlin algorithm achieves the balance of specificity and sensitivity by using the parameter pair with values $FThr = 2.0$ and $PThr = 0.5$. The algorithm with added context is able to use lowered parameter values ($FThr = 0.5$, $PThr = 0.5$) and to still provide better results. In all the three home experiments the contextualization algorithm used comparably lower values for one or both thresholds parameters,than the ones used for by the Moore-Bächlin algorithm.

The third patient experienced very short FOG episodes that lasted 1.33 s on average (see Table

9.3). These episodes usually manifested as a very short starting hesitation with trembling after postural change, or as a shuffling gait while turning towards the fridge (and away from it) after entering through the kitchen door. A relatively small number of episodes (13) was observed. Each episode was labelled by one analytic window.

Table 9.8: Patient3: Comparison of the best results of the Moore-Bächlin algorithm and our new method with added activity context.

Algorithm	*FThr*	*PThr*	TP	FP	FN	TN	Sens [%]	Spec [%]	Acc [%]	F1 score
Moore-Bächlin	3.5	2.0	11	52	2	508	84.6	90.7	90.6	0.29
Added context	1.0	1.5	9	21	4	539	69.2	96.3	95.6	0.42
Difference	-	-	-2	-31	+2	+31	-15.4	+5.5	+5.1	+0.13

Table 9.8 shows the results for Patient3. The percentage difference for sensitivity is -15.4% for the contextualization algorithm. The contextualization introduced two additional false negative detections: a short hesitation was recognized as a forward movement, and a FOG that manifested as a forward shuffling was falsely detected as *Forward walk*. The Moore-Bächlin algorithm for Patient3 uses higher values of the optimal threshold parameters, in comparison with the threshold parameter values used for the other two patients. Higher thresholds values result with a high specificity (> 90%). By adding context, even so high specificity got improved for an additional 5%.

9.5 Summary

In this chapter we presented experiments conducted in the homes of three PD patients with FOG. We setup the system according to the spatial conditions in each home. The collected data were analysed with the newly introduced FOG contextualization algorithm and the results were compared with the Moore-Bächlin algorithm. The FOG contextualization algorithm was designed specifically to target the elimination of the false positive detections and to increase the specificity. The secondary goal of the contextualization was to increase the sensitivity, by enabling the use of lower values for the freezing and the power thresholds in the Moore-Bächlin algorithm.

The results from the analysed data indeed indicate the clear improvement in the specificity. The potential for the specificity improvement was measured as an increase between 5% and 22%, based on the optimal threshold parameters calculated specifically for each patient. When the collected data contained a low number of episodes with very short FOG durations, there was around 10%-15% decrease in the sensitivity. For the data from the patient who experienced a high number of longer FOG episodes, the sensitivity was slightly improved (+1%) over the the Moore-Bächlin algorithm.

10

Conclusion

Technical innovations of the information age will soon allow us to continually collect vast amounts of data about our bodies. Such trends are expected to have a favourable impact on the future treatment of chronic diseases; long-lasting health conditions that can be controlled, but not cured. The availability of miniaturized sensors, processing units and automatic medication dispensers enables a new way for the management of the chronic conditions, and facilitates the realization of the concept of advanced health monitoring system. Parkinson's disease (PD) is one of the chronic diseases for which such monitoring systems can be extremely beneficial, by enabling a continuous control of the desired medication dosage and the elimination of some of its most unpleasant symptoms.

The main goal of this thesis is to improve the detection of Freezing of Gait (FOG), a disabling symptom that commonly occurs in the later stages of Parkinson's disease. As a novelty, the approach taken in the thesis relies on using the information about the situation and the location of the patient, the two of the aspects of the patient's overall context that have a special relation with FOG. To observe the context and to monitor the motor impairment, we needed a system capable of a synchronous capture, replay, storage and fusion of multi-modal sensor data collected from the patient and his environment. To answer these requirements, we developed a completely new context-aware system that targets to monitor the PD patients at their home.

10.1 Research Objectives and Contributions

In this section, we reflect on the committed work and the realized contributions during the fulfilment of each of the three objectives described in the thesis introduction in Chapter 1.

Objective ①: Design of the system for home monitoring of FOG

To design a system that solves a specific medical problem and becomes the permanent part of a person's life, we need to understand both the essence of the medical problem and the future user. In Chapter 1 we gave a general description of all the major symptoms of PD and we focused our attention towards the investigation of the properties of the FOG symptom. FOG is characterized by brief episodes of inability to step, or by extremely short steps that typically occur during gait initiation or when turning while walking. Out of all PD symptoms, FOG is the symptom with most peculiarities. Its unpredictability directly affects the quality of life of the individual. There are three different manifestations of FOG: trembling, shuffling, and complete freeze; and a set of distinctive situations that perspire it, known as triggers. The triggers have many influences, both internal (e.g. stress, anxiety) and external (e.g. narrow passages, turning). A distinctive trait of FOG, that makes its management quite difficult, is its resistance to the parkinsonian medications. Alternatively, FOG can be alleviated by external stimuli, such as lines on the floor or rhythmic sounds, which can influence the attention of the person experiencing an episode, and help him/her to initiate gait. Such approach is known as the *sensory cueing*. The optimal effectiveness of the cueing can be achieved through a timely activation of a cueing device, triggered by the accurate detection of a FOG episode. The timely activation depends on a correct and timely detection, which is still an open problem.

The chronological analysis of the state-of-the-art in Chapter 2 recognized the algorithm for FOG detection invented by Moore, and later perfected by Bächlin, as the state-of-the art algorithm. We recognized the important parameters and requirements of the wearable FOG detection systems: the capability for online processing, minimal latency, minimal number of sensors, and the optimal position for fixing the sensor on the body. We observed the nature of the datasets used in the evaluation of the already existing FOG detection algorithms and singled out the fact that there have not been any datasets that used natural home environments to trigger the FOG episodes. Our conclusions on the state-of-the-art were about the necessity to collect the patient data in their homes, and the realization of the untapped potential for the FOG detection improvement that exists in capturing the unused aspects of the FOG context.

The database of the FP7 REMPARK project offers an excellent opportunity to get a high quality input for the design of the future home monitoring system. In Chapter 3 we made a conclusive analysis of the video and inertial data from 17 PD patients in the database. The data was captured from the PD patients wearing one accelerometer sensor in their homes. The analysis focused on the effects that the home environment and the patient activity have on the onset of FOG. We categorized 11 activities as the instigators for false positive detections by the Moore-Bächlin algorithm. It became clear that one of the ways to improve the specificity of

the existing algorithm would be to recognize these activities, and use them to eliminate incorrect detections. Besides false positives, we also observed false negative detections by the same detection algorithm. We related false negatives with the necessity to track the person's location in an indoor environment in order to correct them. To find the exact relations between FOG and indoor locations, we analysed videos in the database and categorized micro-locations and macro-locations in relation with the types of FOG manifestation. This analysis provided sufficient information to set the requirements and the concept of the system.

In Chapter 4, we presented the requirements for the system and we introduced the software and the hardware platform on which the new system will be developed. We set the patient's location, orientation, and activity as the main context information types that have to be tracked. We designed the future system to be distributed, real-time, modular, portable, and scalable. Physically, the system was envisioned as a network of Microsof Kinect cameras placed in the patient's home, that interacts with a wearable inertial sensor on the patient (smartphone). Since we wanted to use the commercial of the-shelf hardware, we foresaw the production of the software modules (for position tracking, orientation tracking, activity recognition) as the main objective during the system development. We aimed to speed up the development process with the use of the middle-ware and open source libraries.

Objective ②: Development of algorithms for extraction of context

The main functionality of the monitoring system depends on the ability to track the patient's *location*. The location data is provided by multiple person tracking based on the Kinect data. The implementation of the tracking was described in Chapter 5. Video tracking was implemented with the requirements to extend the nominal working range of the Kinect depth sensor to 5.5 m and to keep the same level of tracking accuracy over the whole scene in front of the camera. Besides the algorithm for the multiple person position tracking, we implemented several additional algorithms for the extraction of relevant information from image and depth data. The extracted information includes the person's height, the person's height image features, and the color image features necessary to learn the person's appearance.

The second major functionality of the system, presented in Chapter 6, is the tracking of the *orientation* based on the information from the smartphone. We developed a new method which combines image features (person's height patterns) with the gyroscopic data and provides the absolute 2D orientation in reference to the camera. The accuracy of the orientation estimation was evaluated in the experiment, along with the accuracy of the position tracking. We confirmed that the new method is able to provide the general orientation of the tracked person with the sufficient accuracy. Using such orientation estimation, it can be easily inferred if the person is facing some obstacle in the environment.

Chapter 7 presented the *identity recognition* method that is capable to learn the appearances of a small set of people and to classify between the people inside the set. The classification method is based on the classifier cascade that employs a one-class Support Vector Machine clas-

sifier followed by a Naive Bayes classifier. The main challenge of this method is to optimally train the OCSVM without performing the cross-validation step. We proposed the use of the second independent gallery of appearances in the training stage. We train the OCSVM on the target set of appearances with the condition of achieving the targeted minimal detection accuracy on one another, independent appearance gallery. With this method it is also possible to train OCSVM using the appearance data of only one person, the patient. This last feature of the proposed identification method enabled us to easily recognize, and automatically track the patients in the video data collected during clinical and home experiments.

In Chapter 8 we presented the algorithm for the recognition of the chosen set of FOG-related activities. First, we implemented a vision-based posture classifier in the form of a finite state machine. The posture classifier uses height information as input, and has 5 postural states that are achieved through 13 internal state transitions. A hierarchical decision tree was used to combine the posture finite state machine with three additional movement SVM classifiers. The SVM classifiers were trained to detect turning and bending behaviour, and to recognize the four main movement directions on the floor plane. All three SVM classifiers use as the input both the data produced by video tracking (position, height and orientation data) and the data captured by the smartphone (accelerometer and gyroscope signals). The particular property of the presented activity classification is a very short duration of the activity data buffers (only 0.5 s). By employing such very short buffers, we ensured that the detection of activities will be performed with a small latency. The training of the movement SVM classifiers was executed with the dataset of 8 healthy people captured in laboratory conditions. The primary evaluation on laboratory data was expanded with the results of activity recognition for 4 PD patients doing the *Timed Up and Go* trials in a clinic. The results showed that for some activities, such as standing or sitting, the activity recognition method achieves excellent classification, while for some other activities, like turns, there is still space for a significant improvement.

Objective ③: Contextualization of FOG

In Chapter 9 we described the *FOG contextualization* method, which uses the Moore-Bächlin algorithm to confirm or disprove the detected FOG, by including the recognized activity of the patient into the decision process. A dataset with home data of 3 PD patients was produced by two Kinect cameras and a smartphone in synchronous recording. The improvement potential of our context-based detection method was strictly proven by the comparison with the unchanged Moore-Bächlin algorithm on the same dataset. The context-based algorithm very positively influenced the reduction of false positives, which was expressed through a higher specificity. In some cases, the context-based algorithm also eliminated true positives, reducing sensitivity to a lesser extent. The final comparison between the two algorithms on the basis of their means of the sensitivity, specificity, accuracy and the F1-score showed improvement in the overall FOG detection achieved with the new system.

10.2 Limitations and Future Work

The work described in this thesis required knowledge from several scientific and technical fields, such as medicine, industrial design and computer science, with the special focus on computer vision, sensor fusion and machine learning. A great amount of the project time was spent on the software engineering, developing and implementing the major system software modules. A good overview and awareness of the state-of-the art in video tracking, data fusion, people re-identification and activity recognition was essential. The diverse set of subjects and the limited development time left certain aspects of the system open for future improvements.

Since every major functionality of the system was tested with the participant data collected in laboratory experiments, there are some limitations in the methods of evaluation, algorithm parametrization and the size of the datasets, that should be mentioned. For example, the accuracy of the position and the orientation tracking algorithms was evaluated by having the participants stand on the markers on the floor. If we had on our disposal another more sophisticated and accurate tracking system, we would be able to conduct a more rigorous evaluation based on the dynamic trajectories. The evaluation of the re-identification was conducted only with the gallery size of 3 persons, which we took as the average number of people in a PD patient's household. Although the chosen number of persons represents a more complicated case, it would be relevant to verify the results for galleries containing 2 or only 1 person. In the clinical and the home experiments, where we were collecting the data for evaluation of activity recognition and FOG detection, we wanted to collect a lot of positive FOG examples. Unfortunately, the nature of the research on the FOG phenomenon is such that it is extremely difficult to obtain high quantities of positive examples by conducting only short term experiments.

There are several ways in which the implemented algorithms could be improved in the future. For the activity classification, we opted to use a hierarchical classification based on one particular decision tree configuration. The possible improvements in the activity recognition algorithm could come from the use of a different decision tree configuration, other types of internal node classifiers, or even a different type of classifier instead of the decision tree (e.g. multi-class SVM, Bayesian network). Furthermore, in the activity classification, the performance improvements should be sought by varying the length of the activity buffer, and by adding new elements to the feature vector that is used at the input of the SVM component classifiers. The FOG contextualization algorithm could benefit from exploring the influence of the temporal sequences of activities on FOG. Modelling the temporal relation between the activities and FOG in a Bayesian Network type of approach might be an elegant solution.

The biggest space for a future development, we find in the use of the patient location data for the detection of false negatives, especially in relation to the akinetic type of freezing. We implemented scene mapping tool and context-zone editor with the clear intention to handle such problem. However, in our home data sets we did not collect enough data for the development of the algorithm dedicated to this problem. The future work in this direction demands that the monitoring system is permanently installed in the homes of PD patients, in order to collect data during long periods of time. That is a challenging feat, that can be performed only by the

big multi-institutional projects that have a high number of researchers and adequate logistics support.

Finally, there are non-functional aspects of the system that also require additional research. Usability is key in the acceptance of a clinical monitoring technology by its users, and it is necessary to confirm it by the usability testing (Daniels et al., 2007). During the home experiments we used the Home Experiment Questionnaire (Appendix D) to pose to our participants a few basic questions about the technology acceptance. Their replies on how they perceive our system were mainly positive. However, for a more conclusive usability testing, it is necessary to have a longer period of use of the system, and to use a recognized usability questionnaire such as the System Usability Scale (Brooke, 1996).

With this thesis we brought the system to a functional prototype stage and demonstrated the feasibility of the proposed approach based on *context*. We believe that our system is a highly valuable tool, not only for use in the freezing of gait management, but also for the application in the management of other types of chronic diseases, where location and activity play an important role.

A

REMPARK Database Characterization Data

Table A.1: Experiment information and basic PD patient data for D2FOG dataset.

Patient	Age	Sex	H& Y	State	Duration exp. [min]	PThr	Fthr	TP	FP	FN	Duration TP [sec]	Duration FP [sec]	Duration FN [sec]
1	74	F	3	OFF	20	2.5	0.5	15	13	7	157	81	93
				ON	10	1	1.5	0.80	0	2	8	0	7
2	64	F	2.5	OFF	21	0.5	1	0	24	8	0	223	23
				ON	15	2	1	3	2	2	9	8	3
3	79	M	3	OFF	25	1.5	1.5	3	5	6	12	16	34
4	75	M	2	OFF	19	2	0.5	5	5	4	23	13	17
				ON	24	3	0.5	5	5	2	17	45	7
5	75	M	3	OFF	19	4	0.5	5	11	1	27	35	5
				ON	23	3	2.5	2	10	0	6	69	0
6	63	F	3	OFF	20	0.5	1	23	1	9	206	5	44
				ON	20	0.5	3	4	14	3	19	64	7
7	75	M	2.5	OFF	13	0	0.5	1	0	3	2	0	23
				ON	15	0	0.5	1	4	2	3	13	27
8	72	M	3	OFF	10	3.5	0.5	21	5	2	183	23	9
				ON	18	1.5	2.5	9	10	0	59	30	0
9	69	F	3	OFF	17	1.	1	25	6	15	149	18	53
10	74	M	4	OFF	18	0.5	1.5	4	4	15	20	19	114
11	60	M	3	OFF	12	2	1	16	5	2	255	16	5
				ON	12	2.5	1.5	2	5	0	7	20	0
12	73	M	2.5	OFF	11	1.5	1	5	3	5	25	10	15
13	75	M	3	OFF	14	1	0.5	0	0	2	0	0	7
14	73	M	3	OFF	11	3	1.5	15	9	3	91	23	7
				ON	42	2.5	2.5	12	6	4	80	18	19
15	75	F	3	ON	15	0.5	0.5	0	4	11	0	16	86
16	77	M	3	OFF	25	3	0.5	20	6	10	114	15	61
17	75	M	3	OFF	12	2	1	16	1	5	367	2	33
				ON	22	2	1	7	0	4	120	40	120
Total	-	-	-	-	483	-	-	221	158	127	1969	782	739

Table A.2: Categorization of false positive (FP) detections for Moore-Bächlin algorithm on D2FOG dataset. Details for each patient.

Patient	State	Turns	Small steps	Init. walk	Stopping	Cond. walk	Back / Lateral	Normal walk	Stand / lower move	Stand / upper move	Posture change	Sit
1	OFF		9	1	1	1					1	
	ON											
2	OFF	9	7	1			2	5				
	ON	2										
3	OFF	2	1		1		1					
	ON											
4	OFF	1		1			3					
	ON	2		1					1	1		
5	OFF	7	1				2				1	
	ON	3	2							5		
6	OFF									1		
	ON	4		2			1	2		4		1
7	OFF											
	ON		1							1		2
8	OFF			1		2		1				1
	ON	5			1			3				1
9	OFF			1				3	2			
	ON											
10	OFF	1			1							2
	ON											
11	OFF	4			1							
	ON				1			1				3
12	OFF	2	1									
	ON											
13	OFF											
	ON											
14	OFF	3	4						2			
	ON		2	1	1		2					
15	OFF											
	ON						2	1				1
16	OFF	3						2	1			
	ON											
17	OFF								1			
	ON											
Total	-	48	28	9	7	3	13	18	7	12	2	11

Table A.3: Categorization of false negative (FN) detections for Moore-Bächlin algorithm on D2FOG dataset. Details for each patient.

Patient	State	Manifestation			Freezing Type					Locations / Objects influencing FOG
		Shuffle	Trem-ble	Aki-netic	Start	Turn	Tight	Desti-na-tion	Open	
1	OFF		5	2	1	1			5	None
	ON		1	1		2				None
2	OFF	8			3	2	2		1	Tight spot chair-table-furniture, passage bed – wall, passage table - kitchen element, door
	ON	1	1		2					Passage table -furniture, doorway
3	OFF	5		1		4			2	None
4	OFF	4			1	1	2			Passage table - furniture piece (0.8 m wide), doorway
	ON	2			2					Front of bed
5	OFF		1			1				Door
6	OFF	2	4	3	2	4	2		1	Door
	ON			3		3				Front of kitchen sink
7	OFF			3		1	2			Tight spot chair-table-furniture, door, passage between chairs
	ON			2			2			Door
8	OFF	1	1		1	1				After door, front of table
9	OFF	6	2	7	4	1	5		5	Front of chair, next to table, passage wall-chair, passage table-chair, doorway
10	OFF		14	1	3	7	4		1	None
11	OFF	1		1			1		1	Passage between the edges of two sofas (1 m wide)
12	OFF	3		2	1	1	3			Front of chair, next to table, doorway
13	OFF		1	1		1		1		Front of chair
14	OFF	3			1		1	1		Front of chair, passage between corners of two couches (0.5 m wide)
	ON	3		1		1	2		1	Stairway bottom, front of doorway, passage wall-furniture piece
15	ON	1	4	6	6	2		1	2	Next to table, next to sofa, stairway bottom, front of chair, doorway
16	OFF	4	5	1	5	3	1		1	None
17	OFF	1	4		1	1			3	Passage between two chairs (0.8 m wide)
	ON		1	3	1	1			2	Front of chair
Total	-	45	44	38	34	38	27	3	25	-

Table A.4: True positive (TP) detections for Moore-Bächlin algorithm on D2FOG dataset in dependence of micro location. Detailed data for each patient.

Patient	State	Open space	Narrow space	Door-way	Chair	Bed	Sofa / Couch	Table	Lift door	Other	*Total*
1	OFF	12			3						15
	ON	2									2
2	OFF										
	ON	1		2							3
3	OFF	1			2						3
	ON										
4	OFF	1		4							5
	ON	1		3	1						5
5	OFF			5							5
	ON			1	1						2
6	OFF	4	1	5	6		1		6		23
	ON	1				1		2			4
7	OFF		1								1
	ON		1								1
8	OFF	6	1	13				1			21
	ON			1	8						9
9	OFF	10	4	4	4			1	2		25
	ON										
10	OFF			4							4
	ON										
11	OFF	3	2	10	1						16
	ON	1	1								2
12	OFF	1		2	1				1		5
	ON										
13	OFF										
	ON										
14	OFF	7	3		2		2		1		15
	ON	5	1		3					3	12
15	OFF										
	ON										
16	OFF	12	1	2	4				1		20
	ON										
17	OFF	7	3		4		2				16
	ON	6			1						7
Total	-	81	19	56	41	1	5	4	11	3	221

Table A.5: False negative (FN) detections for Moore-Bächlin algorithm on D2FOG dataset in dependence of micro location. Detailed data for each patient.

Patient	State	Living room	Kitchen	Hall	Bed-room	Bath-room	Other indoor	Terrace	Building	Outside	*Total*
1	OFF	7									7
	ON	2									2
2	OFF	2	3	2				1			8
	ON		1	1							2
3	OFF	4		1	1						6
	ON										
4	OFF	1	2	1							4
	ON	1				1					2
5	OFF			1							1
	ON										
6	OFF	2	1	3	2			1			9
	ON									3	3
7	OFF		2	1							3
	ON			2							2
8	OFF			1				1			2
	ON										
9	OFF	4	3	3	2			3			15
	ON										
10	OFF	4	1	5	5						15
	ON										
11	OFF	1	1								2
	ON										
12	OFF			3	1			1			5
	ON										
13	OFF			1	1						2
	ON										
14	OFF		1		2						3
	ON	1	1	1						1	4
15	OFF										
	ON	6		1	1		1	1		1	11
16	OFF	4	1	5							10
	ON										
17	OFF	4	1								5
	ON	3			1						4
Total	-	46	18	32	16	1	1	8	0	5	127

Table A.6: True positive (TP) detection for Moore-Bächlin algorithm on D2FOG dataset in dependence of macro location. Detailed data for each patient.

Patient	State	Living room	Kitchen	Hall	Bed-room	Bath-room	Other indoor	Terrace	Building	Outside	*Total*
1	OFF	6		2				7			15
	ON			1				1			2
2	OFF										
	ON	1	1		1						3
3	OFF	2		1							3
	ON										
4	OFF	3		2							5
	ON	1	1	1	1		1				5
5	OFF	3	1	1							5
	ON		1	1							2
6	OFF	2		7	1		1		9	3	23
	ON	2			1					1	4
7	OFF	1									1
	ON		1								1
8	OFF	6			3			5	7		21
	ON	9									9
9	OFF	13	3	4				1	4		25
	ON										
10	OFF			4							4
	ON										
11	OFF	10	6								16
	ON	2									2
12	OFF	1		3					1		5
	ON										
13	OFF										
	ON										
14	OFF	9							2	4	15
	ON	3	1	1			3		1	3	12
15	OFF										
	ON										
16	OFF	7	1	11					1		20
	ON										
17	OFF	16									16
	ON	7									7
Total	-	104	16	39	7	0	5	14	25	11	221

Table A.7: False negative (FN) detections for Moore-Bächlin algorithm on D2FOG dataset in dependence of macro location. Detailed data for each patient.

Patient	State	Living room	Kitchen	Hall	Bed-room	Bath-room	Other indoor	Terrace	Building	Outside	*Total*
1	OFF	4						3			7
	ON							2			2
2	OFF	2	2	2	1					1	8
	ON	1							1		2
3	OFF	2	1	1						2	6
	ON										
4	OFF	3		1							4
	ON				1	1					2
5	OFF								1		1
	ON										
6	OFF	1	1	2	1				3	1	9
	ON		3								3
7	OFF	1				2					3
	ON	1				1					2
8	OFF	1							1		2
	ON										
9	OFF	10		4					1		15
	ON										
10	OFF	4	5	6							15
	ON										
11	OFF	2									2
	ON										
12	OFF	2	1						2		5
	ON										
13	OFF	1			1					2	
	ON										
14	OFF	3									3
	ON	1		3							4
15	OFF										
	ON	9							1	1	11
16	OFF	2		7					1		10
	ON										
17	OFF	5									5
	ON	4									4
Total	-	59	13	26	3	5	0	5	11	5	127

B

Position and Orientation Evaluation Trajectories

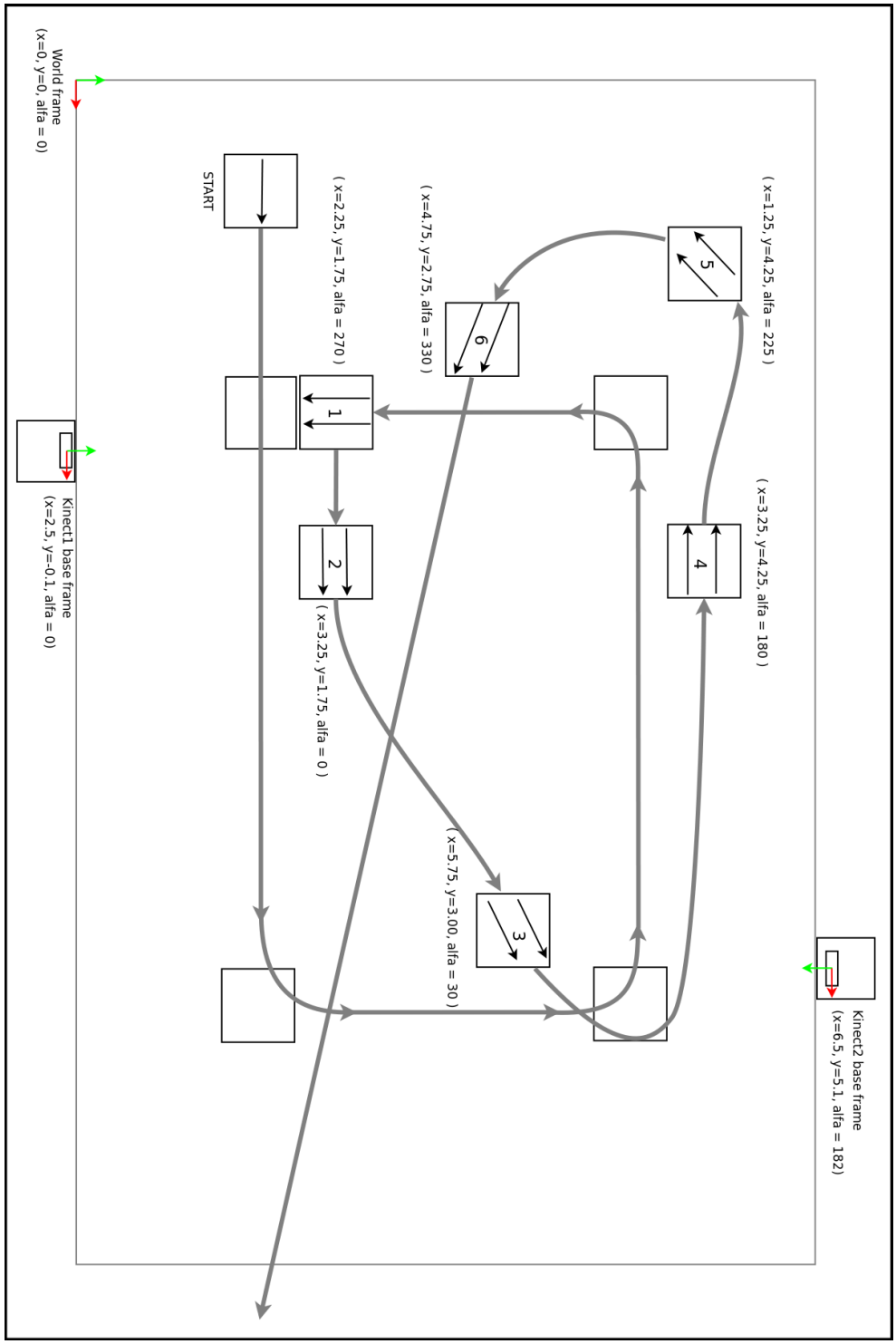


Figure B.1: Schematic of marker positions and numbering for walks starting from the left side.

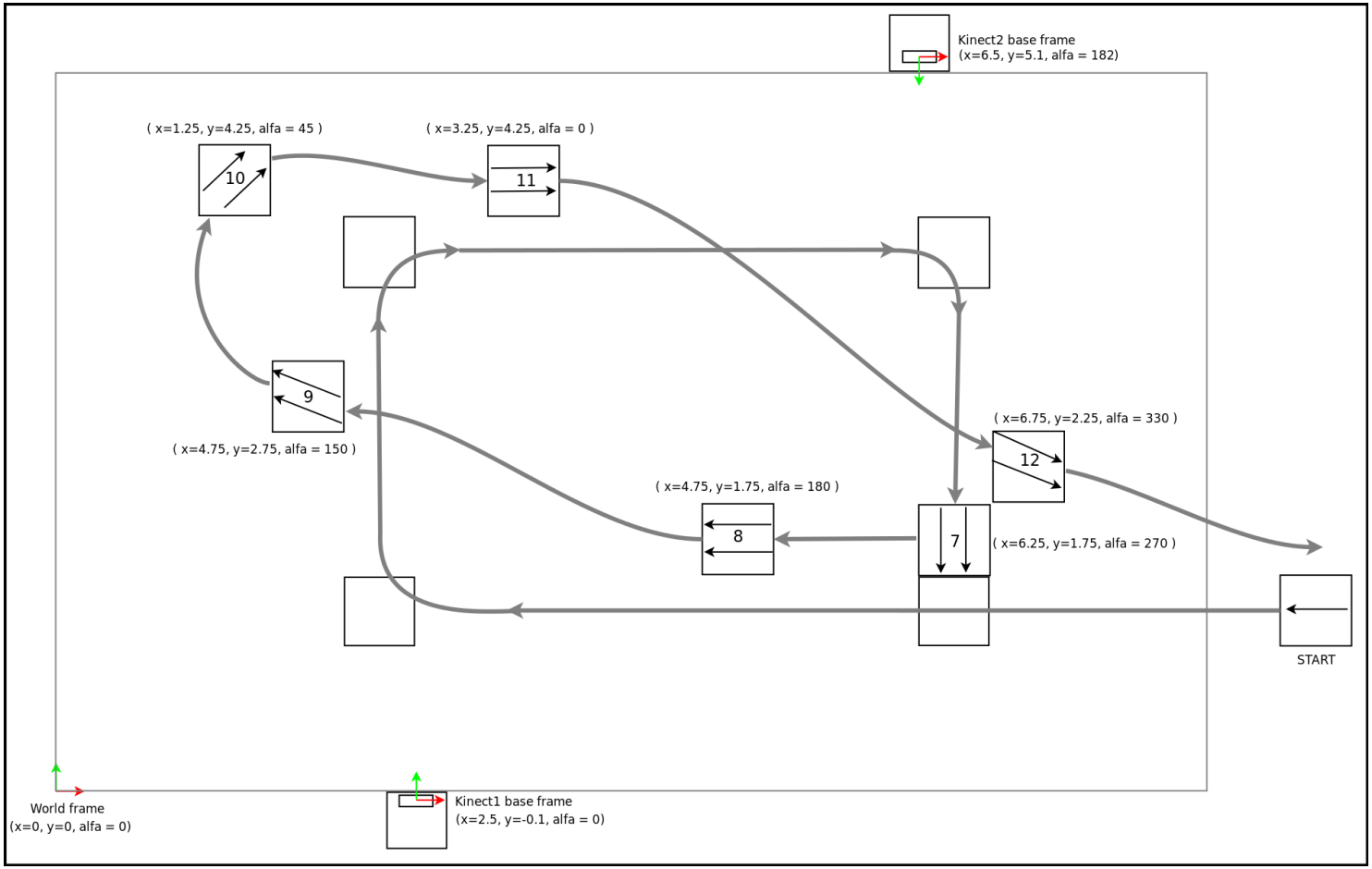


Figure B.2: Schematic of marker positions and numbering for walks starting from the right side.

C

Posture and Activity Recognition Data

Table C.1: Statistical results for *Turn_SVM* classifier on 8 healthy people dataset. Detailed data for each training repetition.

Training ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
1	460	177	98	3746	0.82	0.95	0.72	0.97	0.94	0.77
2	408	98	201	3891	0.67	0.98	0.81	0.95	0.93	0.73
3	382	85	104	3724	0.79	0.98	0.82	0.97	0.96	0.80
4	369	90	106	3641	0.78	0.98	0.80	0.97	0.95	0.79
5	403	97	170	3871	0.70	0.98	0.81	0.96	0.94	0.75
6	489	167	84	3503	0.85	0.95	0.75	0.98	0.94	0.80
7	398	110	79	3754	0.83	0.97	0.78	0.98	0.96	0.81
8	356	85	210	3651	0.63	0.98	0.81	0.95	0.93	0.71
μ[%]	-	-	-	-	76.0	97.0	78.7	96.6	94.4	76.9
σ[%]	-	-	-	-	8.3	1.0	3.5	1.3	1.0	3.6

Sens. - Sensitivity
Spec. - Specificity
Acc. - Accuracy

Table C.2: Statistical results for *Bend_SVM* classifier on 8 healthy people dataset. Detailed data for each training repetition.

Training ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
1	343	40	32	4025	0.91	0.99	0.90	0.99	0.98	0.91
2	298	44	65	4146	0.82	0.99	0.87	0.98	0.98	0.85
3	326	41	26	3902	0.93	0.99	0.89	0.99	0.98	0.91
4	289	32	25	3856	0.92	0.99	0.90	0.99	0.99	0.91
5	326	40	74	4047	0.82	0.99	0.89	0.98	0.97	0.85
6	296	71	28	3807	0.91	0.98	0.81	0.99	0.98	0.86
7	332	53	45	3911	0.88	0.99	0.86	0.99	0.98	0.87
8	292	119	47	3840	0.86	0.97	0.71	0.99	0.96	0.78
μ[%]	-	-	-	-	88.2	98.6	85.3	98.9	97.8	86.6
σ[%]	-	-	-	-	4.5	0.7	6.5	0.4	0.8	4.4

Table C.3: Statistical results for $Move\_SVM$ classifier on 8 healthy people dataset. Detailed data for each training repetition.

Class	Training ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(0) Stand	1	1857	106	53	1491	0.97	0.93	0.95	0.97	0.95	0.96
	2	1647	137	75	1773	0.96	0.93	0.92	0.96	0.94	0.94
	3	1863	87	55	1452	0.97	0.94	0.96	0.96	0.96	0.96
	4	757	80	43	1432	0.98	0.95	0.96	0.97	0.96	0.97
	5	1811	149	50	1605	0.97	0.92	0.92	0.97	0.94	0.95
	6	1737	83	83	1496	0.95	0.95	0.95	0.95	0.95	0.95
	7	1915	112	51	1311	0.97	0.92	0.94	0.96	0.95	0.96
	8	1511	108	64	1746	0.96	0.94	0.93	0.96	0.95	0.95
	μ[%]	-	-	-	-	96.7	93.5	94.2	96.3	95.2	95.4
	σ[%]	-	-	-	-	0.9	1.2	1.4	0.7	0.7	0.9
(1) Forward	1	1256	46	47	2158	0.96	0.98	0.96	0.98	0.97	0.96
	2	1517	39	108	1968	0.93	0.98	0.97	0.95	0.96	0.95
	3	1195	50	20	2192	0.98	0.98	0.96	0.99	0.98	0.97
	4	1219	24	28	2041	0.98	0.99	0.98	0.99	0.98	0.98
	5	1357	34	59	2165	0.96	0.98	0.98	0.97	0.97	0.97
	6	1231	50	40	2078	0.97	0.98	0.96	0.98	0.97	0.96
	7	1122	36	31	2200	0.97	0.98	0.97	0.99	0.98	0.97
	8	1509	44	53	1823	0.97	0.98	0.97	0.97	0.97	0.97
	μ[%]	-	-	-	-	96.6	98.1	97.0	97.7	97.5	96.8
	σ[%]	-	-	-	-	1.5	0.4	0.7	1.3	0.7	0.7
(2) Backward	1	95	7	20	3385	0.83	1.00	0.93	0.99	0.99	0.88
	2	91	14	12	3515	0.88	1.00	0.87	1.00	0.99	0.88
	3	73	11	19	3354	0.79	1.00	0.87	0.99	0.99	0.83
	4	75	7	17	3213	0.82	1.00	0.91	0.99	0.99	0.86
	5	98	8	20	3489	0.83	1.00	0.92	0.99	0.99	0.88
	6	78	14	11	3296	0.88	1.00	0.85	1.00	0.99	0.86
	7	75	3	33	3278	0.69	1.00	0.96	0.99	0.99	0.81
	8	89	17	13	3310	0.87	0.99	0.84	1.00	0.99	0.86
	μ[%]	-	-	-	-	82.4	99.7	89.4	99.5	99.2	85.5
	σ[%]	-	-	-	-	6.2	0.1	4.4	0.2	0.1	2.5
(3) Lateral	1	113	27	66	3301	0.63	0.99	0.81	0.98	0.97	0.71
	2	139	48	43	3402	0.76	0.99	0.74	0.99	0.97	0.75
	3	160	18	72	3207	0.69	0.99	0.90	0.98	0.97	0.78
	4	131	19	42	3120	0.76	0.99	0.87	0.99	0.98	0.81
	5	137	21	83	3374	0.62	0.99	0.87	0.98	0.97	0.72
	6	169	37	50	3143	0.77	0.99	0.82	0.98	0.97	0.80
	7	107	19	55	3208	0.66	0.99	0.85	0.98	0.98	0.74
	8	137	14	53	3225	0.72	1.00	0.91	0.98	0.98	0.80
	μ[%]	-	-	-	-	70.2	99.2	84.6	98.2	97.6	76.5
	σ[%]	-	-	-	-	6.0	0.3	5.4	0.4	0.4	3.8

Table C.4: Statistical results for hierarchical activity classifier on 8 healthy people dataset.

Class	Training ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(0) Stand	1	1685	147	98	2350	0.95	0.94	0.92	0.96	0.94	0.93
	2	1500	146	99	2741	0.94	0.95	0.91	0.97	0.95	0.92
	3	1699	127	120	2210	0.93	0.95	0.93	0.95	0.94	0.93
	4	1639	111	103	2209	0.94	0.95	0.94	0.96	0.95	0.94
	5	1648	181	101	2399	0.94	0.93	0.90	0.96	0.93	0.92
	6	1556	122	126	2281	0.93	0.95	0.93	0.95	0.94	0.93
	7	1796	152	93	2099	0.95	0.93	0.92	0.96	0.94	0.94
	8	1367	119	111	2562	0.92	0.96	0.92	0.96	0.94	0.92
	μ[%]	-	-	-	-	93.8	94.4	92.1	95.7	94.2	92.9
	σ[%]	-	-	-	-	0.9	0.9	1.1	0.6	0.4	0.7
(1) Forward straight walk	1	629	95	150	3406	0.81	0.97	0.87	0.96	0.94	0.84
	2	932	191	119	3244	0.89	0.94	0.83	0.96	0.93	0.86
	3	615	116	81	3344	0.88	0.97	0.84	0.98	0.95	0.86
	4	647	116	89	3210	0.88	0.97	0.85	0.97	0.95	0.86
	5	763	177	99	3290	0.89	0.95	0.81	0.97	0.94	0.85
	6	584	73	152	3276	0.79	0.98	0.89	0.96	0.94	0.84
	7	549	85	113	3393	0.83	0.98	0.87	0.97	0.95	0.85
	8	1 843	217	110	2989	0.88	0.93	0.80	0.96	0.92	0.84
	μ[%]	-	-	-	-	85.6	96.0	84.4	96.6	94.1	84.9
	σ[%]	-	-	-	-	3.9	1.7	3.1	0.7	1.1	1.1
(2) Backward/lateral move	1	181	47	112	3940	0.62	0.99	0.79	0.97	0.96	0.69
	2	222	64	70	4130	0.76	0.98	0.78	0.98	0.97	0.77
	3	203	60	124	3769	0.62	0.98	0.77	0.97	0.96	0.69
	4	193	35	77	3757	0.71	0.99	0.85	0.98	0.97	0.78
	5	211	44	125	3949	0.63	0.99	0.83	0.97	0.96	0.71
	6	213	70	98	3704	0.68	0.98	0.75	0.97	0.96	0.72
	7	151	32	119	3838	0.56	0.99	0.83	0.97	0.96	0.67
	8	207	46	90	3816	0.70	0.99	0.82	0.98	0.97	0.75
	μ[%]	-	-	-	-	66.0	98.7	80.1	97.4	96.4	72.2
	σ[%]	-	-	-	-	6.5	0.3	3.3	0.5	0.6	4.0
(3) Wide turn	1	332	137	84	3727	0.80	0.96	0.71	0.98	0.95	0.75
	2	276	83	206	3921	0.57	0.98	0.77	0.95	0.94	0.66
	3	291	77	84	3704	0.78	0.98	0.79	0.98	0.96	0.78
	4	288	83	94	3597	0.75	0.98	0.78	0.97	0.96	0.76
	5	282	74	176	3797	0.62	0.98	0.79	0.96	0.94	0.69
	6	377	146	70	3492	0.84	0.96	0.72	0.98	0.95	0.78
	7	308	97	68	3667	0.82	0.97	0.76	0.98	0.96	0.79
	8	259	69	199	3632	0.57	0.98	0.79	0.95	0.94	0.66
	μ[%]	-	-	-	-	71.8	97.5	76.3	96.8	94.8	73.4
	σ[%]	-	-	-	-	11.5	0.8	3.2	1.4	1.0	5.6

Table C.5: Statistical results for hierarchical activity classifier on 8 healthy people dataset – continuation from the previous page.

Class	Training ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(4) Spot turn	1	90	45	29	4116	0.76	0.99	0.67	0.99	0.98	0.71
	2	84	35	32	4335	0.72	0.99	0.71	0.99	0.99	0.71
	3	68	29	22	4037	0.76	0.99	0.70	0.99	0.99	0.73
	4	56	21	23	3962	0.71	0.99	0.73	0.99	0.99	0.72
	5	76	41	30	4182	0.72	0.99	0.65	0.99	0.98	0.68
	6	80	40	27	3938	0.75	0.99	0.67	0.99	0.98	0.70
	7	63	32	23	4022	0.73	0.99	0.66	0.99	0.99	0.70
	8	70	24	25	4040	0.74	0.99	0.74	0.99	0.99	0.74
	μ[%]	-	-	-	-	73.5	99.2	69.1	99.4	98.6	71.2
	σ[%]	-	-	-	-	1.8	0.2	3.4	0.1	0.2	1.8
(5) Bend	1	246	27	124	3883	0.66	0.99	0.90	0.97	0.96	0.77
	2	211	32	162	4081	0.57	0.99	0.87	0.96	0.96	0.69
	3	239	43	120	3754	0.67	0.99	0.85	0.97	0.96	0.75
	4	210	41	110	3701	0.66	0.99	0.84	0.97	0.96	0.74
	5	245	37	150	3897	0.62	0.99	0.87	0.96	0.96	0.72
	6	220	38	112	3715	0.66	0.99	0.85	0.97	0.96	0.75
	7	247	46	124	3723	0.67	0.99	0.84	0.97	0.96	0.74
	8	213	77	135	3734	0.61	0.98	0.73	0.97	0.95	0.67
	μ[%]	-	-	-	-	63.9	98.9	84.4	96.7	95.9	72.7
	σ[%]	-	-	-	-	3.7	0.4	4.9	0.4	0.5	3.3
(6) Sit	1	515	104	5	3656	0.99	0.97	0.83	1.00	0.97	0.90
	2	571	139	2	3774	1.00	0.96	0.80	1.00	0.97	0.89
	3	476	113	14	3553	0.97	0.97	0.81	1.00	0.97	0.88
	4	521	101	12	3428	0.98	0.97	0.84	1.00	0.97	0.90
	5	418	132	5	3774	0.99	0.97	0.76	1.00	0.97	0.86
	6	468	98	2	3517	1.00	0.97	0.83	1.00	0.98	0.90
	7	471	111	15	3543	0.97	0.97	0.81	1.00	0.97	0.88
	8	525	123	5	3506	0.99	0.97	0.81	1.00	0.97	0.89
	μ[%]	-	-	-	-	98.5	96.9	81.1	99.8	97.1	88.9
	σ[%]	-	-	-	-	1.1	0.3	2.4	0.2	0.3	1.5

Table C.6: Statistical results for *Turn_SVM* classifier on 4 patient clinical dataset. Detailed data for each patient.

Patient ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
1	83	4	14	528	0.86	0.99	0.95	0.97	0.97	0.90
2	30	6	65	740	0.32	0.99	0.83	0.92	0.92	0.46
3	56	29	27	665	0.67	0.96	0.66	0.96	0.93	0.67
4	102	6	31	722	0.77	0.99	0.94	0.96	0.96	0.85
μ[%]	-	-	-	-	65.3	98.4	84.8	95.3	94.3	71.8
σ[%]	-	-	-	-	23.7	1.7	13.7	2.4	2.6	20.1

Table C.7: Statistical results for *Bend_SVM* classifier on 4 patient clinical dataset. Detailed data for each patient.

Patient ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
1	31	30	8	560	0.79	0.95	0.51	0.99	0.94	0.62
2	51	25	26	739	0.66	0.97	0.67	0.97	0.94	0.67
3	47	13	26	691	0.64	0.98	0.78	0.96	0.95	0.71
4	60	9	22	720	0.73	0.99	0.87	0.97	0.96	0.79
μ[%]	-	-	-	-	70.8	97.1	70.8	97.2	94.8	69.7
σ[%]	-	-	-	-	6.9	1.7	15.6	1.0	1.1	7.4

Table C.8: Statistical results for $Move\_SVM$ classifier on 4 patient clinical dataset. Detailed data for each patient.

Class	Patient ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(0) Stand	1	240	6	25	198	0.91	0.97	0.98	0.89	0.93	0.94
	2	279	4	44	201	0.86	0.98	0.99	0.82	0.91	0.92
	3	110	2	40	219	0.73	0.99	0.98	0.85	0.89	0.84
	4	162	1	33	336	0.83	1.00	0.99	0.91	0.94	0.91
	μ[%]	-	-	-	-	83.3	98.5	98.4	86.6	91.6	90.1
	σ[%]	-	-	-	-	7.3	1.2	0.8	4.1	2.3	4.3
(1) Forward	1	192	23	12	242	0.94	0.91	0.89	0.95	0.93	0.92
	2	201	41	4	282	0.98	0.87	0.83	0.99	0.91	0.90
	3	216	37	5	113	0.98	0.75	0.85	0.96	0.89	0.91
	4	319	30	17	166	0.95	0.85	0.91	0.91	0.91	0.93
	μ[%]	-	-	-	-	96.2	84.7	87.3	95.1	91.0	91.5
	σ[%]	-	-	-	-	2.0	6.8	3.8	3.3	1.6	1.3
(2) Backward	1	0	1	0	468	-	1.00	0.00	1.00	1.00	-
	2	0	0	0	528	-	1.00	-	1.00	1.00	-
	3	0	1	0	468	-	1.00	0.00	1.00	1.00	-
	4	0	0	0	532	-	1.00	-	1.00	1.00	-
	μ[%]	-	-	-	-	-	99.9	0.0	100.0	99.9	-
	σ[%]	-	-	-	-	-	0.1	-	0.0	0.1	-
(3) Lateral	1	0	7	0	462	-	0.99	0.00	1.00	0.99	-
	2	0	3	0	525	-	0.99	0.00	1.00	0.99	-
	3	0	6	0	365	-	0.98	0.00	1.00	0.98	-
	4	0	20	1	511	-	0.96	0.00	1.00	0.96	-
	μ[%]	-	-	-	-	-	98.1	0.0	100.0	98.1	-
	σ[%]	-	-	-	-	-	1.4	0.0	0.1	1.4	-

Table C.9: Statistical results for hierarchical activity classifier on 4 patient dataset.

Class	Patient ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(0) Stand	1	153	25	33	410	0.82	0.94	0.86	0.93	0.91	0.84
	2	182	47	36	562	0.83	0.92	0.79	0.94	0.90	0.81
	3	64	22	39	639	0.62	0.97	0.74	0.94	0.92	0.68
	4	56	20	26	732	0.68	0.97	0.74	0.97	0.94	0.71
	μ[%]	-	-	-	-	74.0	95.1	78.4	94.3	91.8	76.0
	σ[%]	-	-	-	-	10.5	2.3	5.7	1.7	2.0	7.9
(1) Forw. straight	1	157	24	23	417	0.87	0.95	0.87	0.95	0.92	0.87
	2	189	40	10	588	0.95	0.94	0.83	0.98	0.94	0.88
	3	153	48	14	549	0.92	0.92	0.76	0.98	0.92	0.83
	4	269	46	24	495	0.92	0.91	0.85	0.95	0.92	0.88
	μ[%]	-	-	-	-	91.4	92.9	82.7	96.5	92.5	86.7
	σ[%]	-	-	-	-	3.2	1.4	4.7	1.7	1.0	2.5
(2) Back/lateral	1	0	4	0	617	-	0.99	0.00	1.00	0.99	-
	2	0	2	0	825	-	1.00	0.00	1.00	1.00	-
	3	0	4	0	760	-	0.99	0.00	1.00	0.99	-
	4	0	14	0	820	-	0.98	0.00	1.00	0.98	-
	μ[%]	-	-	-	-	-	99.2	0.0	100.0	99.2	-
	σ[%]	-	-	-	-	-	0.6	0.0	0.0	0.6	-
(3) Wide turn	1	8	13	7	593	0.53	0.98	0.38	0.99	0.97	0.44
	2	3	6	10	808	0.23	0.99	0.33	0.99	0.98	0.27
	3	14	30	31	689	0.31	0.96	0.32	0.96	0.92	0.31
	4	18	20	26	770	0.41	0.97	0.47	0.97	0.94	0.44
	μ[%]	-	-	-	-	37.1	97.6	37.7	97.5	95.3	36.8
	σ[%]	-	-	-	-	13.0	1.4	7.0	1.6	2.7	8.7
(4) Spot turn	1	55	9	24	533	0.70	0.98	0.86	0.96	0.95	0.77
	2	12	13	67	735	0.15	0.98	0.48	0.92	0.90	0.23
	3	12	19	25	708	0.32	0.97	0.39	0.97	0.94	0.35
	4	52	10	57	715	0.48	0.99	0.84	0.93	0.92	0.61
	μ[%]	-	-	-	-	41.2	98.2	64.1	94.1	92.8	49.0
	σ[%]	-	-	-	-	23.1	0.5	24.3	2.4	2.0	24.4

Table C.10: Statistical results for hierarchical activity classifier on 4 patient dataset – continuation from the previous page.

Class	Patient ID	TP	FP	FN	TN	Sens.	Spec.	PPV	NPV	Acc.	F1 score
(5) Bend	1	21	28	18	554	0.54	0.95	0.43	0.97	0.93	0.48
	2	55	19	21	732	0.72	0.97	0.74	0.97	0.95	0.73
	3	49	16	26	673	0.65	0.98	0.75	0.96	0.95	0.70
	4	54	36	27	717	0.67	0.95	0.60	0.96	0.92	0.63
	μ[%]	-	-	-	-	64.6	96.4	63.1	96.7	93.7	63.6
	σ[%]	-	-	-	-	7.8	1.4	15.2	0.4	1.4	11.4
(6) Sit	1	108	16	14	483	0.89	0.97	0.87	0.97	0.95	0.88
	2	238	21	4	564	0.98	0.96	0.92	0.99	0.97	0.95
	3	317	16	20	411	0.94	0.96	0.95	0.95	0.95	0.95
	4	217	22	8	587	0.96	0.96	0.91	0.99	0.96	0.94
	μ[%]	-	-	-	-	94.3	96.5	91.2	97.6	96.0	92.7
	σ[%]	-	-	-	-	4.3	0.2	3.3	1.7	0.9	3.4

Table C.11: Confusion table for *Move_SVM* classifier for 8 healthy people dataset. Table contains averaged numbers of classified events calculated from evaluation of 8 trained models. Parentheses hold average value expressed as the percentage of actual class.

		Predicted			
		(0) Stand	(1) Forward	(2) Backward	(3) Lateral
Actual	(0) Stand	1762 (96.7)	32 (1.7)	10 (0.5)	18 (1.5)
	(1) Forward	42 (3.1)	1301 (96.4)	0 (0.0)	7 (0.5)
	(2) Backward	18 (17.1)	0 (0.0)	84 (82.3)	1 (0.6)
	(3) Lateral	49 (25.0)	9 (4.6)	0 (0.2)	137 (70.2)

Table C.12: Confusion table for hierarchical activity classifier for 8 healthy people dataset. Table contains averaged numbers of classified events calculated from evaluation of 8 trained models. Parentheses hold average value expressed as the percentage of actual class.

		Predicted						
		(0) Stand	(1) Forw. str.	(2) Back./Lat.	(3) Wide turn	(4) Spot turn	(5) Bend	(6) Sit
Actual	(0) Stand	1611 (93.8)	26 (1.5)	44 (2.5)	1 (0.1)	10 (0.6)	27 (1.5)	0 (0.0)
	(1) Forw. str.	26 (3.2)	695 (85.9.4)	3 (0.4)	81 (10.0)	0 (0.0)	5 (0.6)	0 (0.0)
	(2) Back./Lat.	83 (27.7)	8 (2.5)	198 (66.0)	2 (0.5)	5 (1.5)	5 (1.7)	0 (0.0)
	(3) Wide turn	2 (0.6)	100 (23.5)	2 (0.4)	302 (71.1)	19 (4.4)	0 (0.0)	0 (0.0)
	(4) Spot turn	10 (9.5)	1 (1.0)	1 (1.4)	12 (12.3)	73 (73.6)	2 (2.3)	0 (0.0)
	(5) Bend	14 (4.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	229 (63.8)	115 (32.1)
	(6) Sit	3 (0.6)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	4 (0.8)	496 (98.5)

Table C.13: Confusion table for *Move_SVM* classifier for 4 patient clinical dataset. Table contains averaged numbers of classified events calculated from evaluation on 4 patients with one trained model. Parentheses hold average value expressed as the percentage of actual class.

		Predicted			
		(0) Stand	(1) Forward	(2) Backward	(3) Lateral
Actual	(0) Stand	198 (84.9)	33 (14.0)	0 (0.0)	3 (1.3)
	(1) Forward	3 (1.2)	232 (96.1)	0 (0.0)	7 (2.9)
	(2) Backward	0 (0.0)	0 (0.0)	84 (0.0)	0 (0.0)
	(3) Lateral	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)

Table C.14: Confusion table for hierarchical activity classifier for 4 patient clinical dataset. Table contains averaged numbers of classified events calculated from evaluation on 4 patients with one trained model. Parentheses hold average value expressed as the percentage of actual class.

		Predicted						
		(0) Stand	(1) Forw. str.	(2) Back./Lat.	(3) Wide turn	(4) Spot turn	(5) Bend	(6) Sit
Actual	(0) Stand	114 (77.6)	17 (11.6)	2 (1.4)	1 (0.7)	11 (7.5)	3 (2.0)	0 (0.0)
	(1) Forw. str.	7 (3.3)	192 (91.4)	4 (1.9)	2 (1.0)	1 (0.5)	5 (2.4)	0 (0.0)
	(2) Back./Lat.	0 (0.0)	0 (0.0)	10 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)
	(3) Wide turn	1 (3.4)	17 (58.6)	0 (0.0)	11 (37.9)	0 (0.0)	1 (3.4)	0 (0.0)
	(4) Spot turn	16 (21.1)	5 (6.6)	0 (0.0)	14 (18.4)	33 (43.4)	8 (10.5)	1 (0.0)
	(5) Bend	3 (4.4)	1 (1.5)	0 (0.0)	0 (0.0)	1 (1.5)	45 (66.2)	18 (26.5)
	(6) Sit	2 (0.9)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	9 (3.9)	220 (94.8)

D

Home Experiment Questionnaire

DEFROST: Home System for Monitoring Freezing of Gait in Parkinson's Disease
[Home Experiment Questionnaire]

QUESTIONAIRE NUMBER: DATE:

NAME: IDENTIFYING LABEL:
SURNAME:
PHONE NUMBER:
ADDRESS:

INFORMATION:

The object of the study is to develop algorithms able to detect FoG episodes using contextual information of the patients. To do so, we will construct the database of ambulatory movement of Parkinson's disease patients. Data collection consists of recording the signals from two Kinect cameras and the inertial sensor (smartphone on waist) in order to obtain examples of Freezing of Gait (FoG) symptom episodes in the patient's home environment. Recorded video data will also be used in the labelling process to achieve the golden standard.

INTERVIEWER

- □ I have informed the patient of the study procedures
- □ I have left a copy of the patient information sheet

INCLUSION CRITERIA:

I.1. Did he/she sign the consent form?
- □ No
- □ Yes

A. SOCIODEMOGRAPHIC AND HOME INFORMATION

A.1. Sex: □ Male □ Female

A.2. Age …..... years old

A.3. Height ….....meters

A.4. Marital Status: 1) Single 2) Married 3) Living with partner 4) Separated 5) Widowed

A.5. How many rooms is there in the house?

….....

A.6 How long do you live here? …..... years

B. ON-OFF STATES AND TREMOR

B.1. Are “off” periods predictable?

- □ Yes
- □ No

B.2. Are “off” periods unpredictable?

- □ Yes
- □ No

B.3. Do “off” periods come suddenly?

- □ Yes
- □ No

B.4. What proportion of the walking day is the patient “off” on average?

- □ None
- □ 1-25% of a day
- □ 26-50% of a day
- □ 51-75% of a day
- □ 76-100% of a day

B.4. Do you experience tremor during the day? If, so, where?

- □ None
- □ Yes. Lower parts. Legs
- □ Yes. Upper parts. Arms.

C. FREEZING OF GAIT - General

All answers except in response to item 3, should be based on the experience over the last week.

This questionnaire should be completed by the researcher after asking and demonstrating freezing phenomenon if necessary.

C.1. During your worst state - do you walk:

- □ normally
- □ almost normally..somewhat slow
- □ slow, but fully independent
- □ need assistance or walking aid
- □ unable to walk

C.2. Are you gait difficulties affecting your daily activities and independence?

- □ not at all
- □ mildly
- □ moderately
- □ severely
- □ unable to walk

C.3. Do you feel that your feet get glued to the floor while walking, making a turn or when trying to initiate walking (freezing)?

- □ never
- □ very rarely: about once a month
- □ rarely: about once a week
- □ often: about once a day
- □ always: whenever walking

C.4. How long is your longest freezing episode?

- □ never happened
- □ 1-2 seconds
- □ 3-10 seconds
- □ 11-30 seconds
- □ unable to walk for more than 30 seconds

C.5. How long is your typical start hesitation episode (freezing when initiating the first step)?

- □ none
- □ takes longer than 1 second to start walking
- □ takes longer than 3 seconds to start walking
- □ takes longer than 10 seconds to start walking
- □ takes longer than 30 seconds to start walking

C.6. How long is your typical turning hesitation: (freezing when initiating the first step)?

- □ none
- □ resume turning in 1 to 2 seconds
- □ resume turning in 3 to 10 seconds
- □ resume turning in 11 to 30 seconds
- □ unable to resume turning for more than 30 seconds

D. FREEZING OF GAIT – Indoor triggers

D.1. Do you experience freezing more often indoors or outdoors?

- □ indoor
- □ outdoor

D.2. Is there a specific spot in the house where you always have freezing? If yes what is it?

- □ No
- □ Yes Place:.....

D.3. In what room in the house would you say that you experience freezing of gait most often?

- □ living room
- □ kitchen
- □ bathroom
- □ bedroom(s)
- □ hall

D.4. Do you have more problem with high or with low obstacles? I.e. is it easier to pass through the door or between the chair and the table?

- □ no difference
- □ high obstacles are harder
- □ low obstacles are harder

E. FREEZING OF GAIT – Exit methods

E.1. How do you usually get out of your freezing episode?

- □ just by waiting
- □ using some of the self discovered methods
- □ using some of the methods others trained me in
- □ by direct help of someone else

E.2. Have you been trained in sensory cueing (visual, audio)? If so, do you find it helpful?

- □ No
- □ Yes Found useful: No/Yes

F. TECHNOLOGY ACCEPTANCE

F.1. Would you use a belt with the sensor like you did today every day, if it could help you get out of the freezing episodes faster?

- □ No
- □ Yes

F.2. Would you agree of having cameras in your living room and hall, if it could help you get out of the freezing episodes faster?

- □ No
- □ Yes

F.3. Would you agree of both wearing a belt with the sensor and having cameras in your living room and hall, if it could help you get out of the freezing episodes faster?

- □ No
- □ Yes

F.4. Out of the two technologies, which one do you find more acceptable for helping you every day?

- □ Cameras
- □ Belt with the sensor

G. USUAL ACTIVITIES DURING THE DAY

The participant will describe what he/she usually does in a day.

Morning

1.-.....

2.-.....

3.-.....

4.-.....

5.-.....

Afternoon

1.-.....

2.-.....

3.-.....

4.-.....

5.-.....

E

Custom ROS Messages

`TrackData.msg`

```
Header header                          % Header with timestamp information
float64 original_ts                    % Original timestamp
float32 x                              % Position in x coordinate of camera_base frame
float32 y                              % Position in y coordinate of camera_base frame
float32 z                              % distance from center of camera_base frame
float32 height                         % Unfiltered height
int32 orientation_class                % Output orientation from NN classifier
float32 orientation_probability        % Probability for NN class
int32 scene_id                         % Unique numeric identification for camera
int32 person_id                        % Unique numeric identification for track
string camera_name                     % User given camera name
float32 movement_speed                 % Speed as scalar value
float32 velocity_x                     % Velocity in x coordinate of camera_base frame
float32 velocity_y                     % Velocity in y coordinate of camera_base frame
string posture                         % Classification output of posture FSM
float64[] height_filtered              % Filtered height
float64[] height_velocity              % Velocity from filtered height
float32 height_standing                % Estimated standing height
```

`PersonModel.msg`

```
int32 action_type                      % Filtered height
uint32 pid                             % Unique numeric identification for person
string camera_name                     % Camera which took person model
int32 identity_state                   % Information whether identity is confirmed
float32[] marker_color                 % Color assigned to person
sensor_msgs/Image MCD_rgb              % Scaled image of person
sensor_msgs/Image                      % Mask for the up part of body
MCD_mask_upper
sensor_msgs/Image                      % Mask for the down part of body
MCD_mask_lower
uint32 update_count                    % Counts appearance updates
```

`FOGData.msg`

Header header	% Header with timestamp information
float32 freeze_power	% FFT power in freezing band
float32 loco_power	% FFT power in locomotor band
float32 total_power	% FFT power in all bands
float32 freezing_index	% Freezing index
int32 fog_label	% Ground truth label for FOG
sensor_msgs/Imu imu	% The raw sensor data from IMU

`ContextData.msg`

Header header	% Header with timestamp information
float32 x	% Position in x coordinate of *camera_base* frame
float32 y	% Position in y coordinate of *camera_base* frame
float32 displacement	% Difference of positions
float32 forward_velocity	% Forward velocity of a person
float32 lateral_velocity	% Lateral velocity of a person
float32 velocity_x_avg	% Average forward velocity
float32 velocity_y_avg	% Average lateral velocity
float32 velocity_x_std	% Std. deviation of forward velocity
float32 velocity_y_std	% Std. deviation of lateral velocity
float32 movement_speed	% Movement speed
float32 height	% Person height (latest)
float32 height_stand	% Person height (standing)
float32 height_diff	% Height difference
int32 posture	% Posture class code
float32 freeze_power	% FFT power in freezing band
float32 loco_power	% FFT power in locomotor band
float32 total_power	% FFT power in all bands
float32 rotation_angle	% Angle of rotation in *camera_base* frame
float32 rotation_velocity	% Velocity of rotation in *camera_base* frame
float32 freezing_index	% Freezing index
float32 calculated_value1	% Space for additional calculation results
float32 calculated_value2	% Space for additional calculation results
float32 imu_acc_x	% Raw acceleration x axis
float32 imu_acc_y	% Raw acceleration y axis
float32 imu_acc_z	% Raw acceleration z axis
float32 acc_magn_avg	% Average magnitude of acceleration
float32 acc_magn_std	% Std. deviation magnitude of acceleration
float32 imu_ang_vel_x	% Raw angular velocity around x axis
float32 imu_ang_vel_y	% Raw angular velocity around y axis

```
float32 imu_ang_vel_z          % Raw angular velocity around z axis
float32 rot_vel_x_avg          % Average velocity around vertical axis
float32 rot_vel_z_avg          % Std. deviation velocity around vertical axis
float32 rot_vel_x_std          % Average velocity around transversal axis
float32 rot_vel_z_std          % Std. deviation velocity around transversal axis
int32 bend_class               % Output class of SVM_Bend classifier
int32 turn_class               % Output class of SVM_Turn classifier
int32 move_class               % Output class of SVM_Move classifier
int32 activity_class           % Output class of activity classifier
int32 moore_fog_class          % Output class of Moore-Bächlin algorithm
int32 context_fog_class        % Possibility for FOG based on context only
int32 fog_class                % Output class of contextualized M-B algorithm
int32 bend_label               % Ground truth label for bending
int32 turn_label               % Ground truth label for turns
int32 move_label               % Ground truth label for movement direction
int32 activity_label           % Ground truth label for activity
int32 fog_label                % Ground truth label for FOG
```

Bibliography

Abadie, F., Codagnone, C., and van Lieshout, M. (2011). Strategic intelligence monitor on personal health systems (SIMPHS): Report on typology/segementation of the PHS market. Technical report.

Abowd, D., Dey, A. K., Orr, R., and Brotherton, J. (1998). Context-awareness in wearable and ubiquitous computing. *Virtual Reality*, 3(3):200–211.

Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., and Steggles, P. (1999). Towards a better understanding of context and context-awareness. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, HUC '99, pages 304–307, London, UK, UK. Springer-Verlag.

Aggarwal, J. and Ryoo, M. (2011). Human activity analysis: A review. *ACM Comput. Surv.*, 43(3):16:1–16:43.

Aggarwal, J. K. and Cai, Q. (1999). Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428–440.

Allin, S. and Mihailidis, A. (2008). Sit to stand detection and analysis. Arlington, VA.

Almeida, Q. J. and Lebold, C. A. (2010). Freezing of gait in parkinson's disease: a perceptual cause for a motor impairment? *Journal of neurology, neurosurgery, and psychiatry*, 81(5):513–518.

Ambani, L. M. and Van Woert, M. H. (1973). Start hesitation–a side effect of long-term levodopa therapy. *The New England journal of medicine*, 288(21):1113–1115.

Andersen, M., Jensen, T., Lisouski, P., Mortensen, A., Hansen, M., Gregersen, T., and Ahrendt, P. (2012). Kinect depth sensor evaluation for computer vision applications. Technical ECE-TR-6, Aarhus University, Aarhus, denmark.

Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2012). Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Proceedings of the 4th International Conference on Ambient Assisted Living and Home Care*, IWAAL'12, pages 216–223, Berlin, Heidelberg. Springer-Verlag.

Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2013). Energy efficient smartphone-based activity recognition using fixed-point arithmetic. *The Journal of Universal Computer Science*, 19(9):1295–1314.

Azulay, J. P., Mesure, S., Amblard, B., Blin, O., Sangla, I., and Pouget, J. (1999). Visual control of locomotion in parkinson's disease. *Brain: a journal of neurology*, 122 (Pt 1):111–120.

Bagley, S., Kelly, B., Tunnicliffe, N., Turnbull, G. I., and Walker, J. M. (1991). The effect of visual cues on the gait of independently mobile parkinson's disease patients. *Physiotherapy*, 77(6):415–420.

Banerjee, T. (2010). *Activity Segmentation With Special Emphasis on Sit-to-Stand Analysis*. Master thesis, University of Missouri-Columbia, Missouri, USA.

Banerjee, T., Keller, J., Skubic, M., and Abbott, C. (2010). Sit-to-stand detection using fuzzy clustering techniques. In *2010 IEEE International Conference on Fuzzy Systems (FUZZ)*, pages 1–8.

Banos, O., Damas, M., Pomares, H., Rojas, F., Delgado-Marquez, B., and Valenzuela, O. (2013). Human activity recognition based on a sensor weighting hierarchical classifier. *Soft Computing*, 17(2):333–343.

Bao, L. and Intille, S. S. (2004a). Activity recognition from user-annotated acceleration data. pages 1–17. Springer.

Bao, L. and Intille, S. S. (2004b). Activity recognition from user-annotated acceleration data. In Ferscha, A. and Mattern, F., editors, *Pervasive Computing*, number 3001 in Lecture Notes in Computer Science, pages 1–17. Springer Berlin Heidelberg.

Barbeau, A. (1969). L-dopa therapy in parkinson's disease. *Canadian Medical Association Journal*, 101(13):59–68.

Barbosa, I. B., Cristani, M., Bue, A. D., Bazzani, L., and Murino, V. (2012). Re-identification with RGB-d sensors. In Fusiello, A., Murino, V., and Cucchiara, R., editors, *Computer Vision – ECCV 2012. Workshops and Demonstrations*, number 7583 in Lecture Notes in Computer Science, pages 433–442. Springer Berlin Heidelberg.

Basso, F., Munaro, M., Michieletto, S., Pagello, E., and Menegatti, E. (2013). Fast and robust multi-people tracking from RGB-d data for a mobile robot. In Lee, S., Cho, H., Yoon, K.-J., and Lee, J., editors, *Intelligent Autonomous Systems 12*, number 193 in Advances in Intelligent Systems and Computing, pages 265–276. Springer Berlin Heidelberg.

Batlle, J., Mouaddib, E., and Salvi, J. (1998). *Recent progress in coded structured light as a technique to solve the correspondence problem: a survey*.

Berardelli, A., Rothwell, J. C., Thompson, P. D., and Hallett, M. (2001). Pathophysiology of bradykinesia in parkinson's disease. *Brain*, 124(11):2131–2146.

Beringer, R., Sixsmith, A., Campo, M., Brown, J., and McCloskey, R. (2011). The "acceptance" of ambient assisted living: Developing an alternate methodology to this limited research lens. In *Proceedings of the 9th International Conference on Toward Useful Services for Elderly and People with Disabilities: Smart Homes and Health Telematics*, ICOST'11, pages 161–167, Berlin, Heidelberg. Springer-Verlag.

Berrada, D., Romero, M., Abowd, G., Blount, M., and Davis, J. (2007). Automatic administration of the get up and go test. In *Proceedings of the 1st ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments*, HealthNet '07, pages 73–75, New York, NY, USA. ACM.

Beymer, D. (2000). Person counting using stereo. In *Proceedings of the Workshop on Human Motion (HUMO'00)*, HUMO '00, pages 127–, Washington, DC, USA. IEEE Computer Society.

Bidargaddi, N., Sarela, A., Boyle, J., Cheung, V., Karunanithi, M., Klingbei, L., Yelland, C., and Gray, L. (2007). Wavelet based approach for posture transition estimation using a waist worn accelerometer. In *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2007. EMBS 2007*, pages 1884–1887.

Bloem, B. R., Hausdorff, J. M., Visser, J. E., and Giladi, N. (2004). Falls and freezing of gait in parkinson's disease: a review of two interconnected, episodic phenomena. *Movement disorders: official journal of the Movement Disorder Society*, 19(8):871–884.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Brichetto, G., Pelosin, E., Marchese, R., and Abbruzzese, G. (2006). Evaluation of physical therapy in parkinsonian patients with freezing of gait: a pilot study. *Clinical rehabilitation*, 20(1):31–35.

Brooke, J. (1996). SUS: A quick and dirty usability scale. In Jordan, P., Weerdmeester, B., Thomas, A., and Mclelland, I., editors, *Usability evaluation in industry*. Taylor and Francis.

Brown, P., Bovey, J., and Chen, X. (1997). Context-aware applications: from the laboratory to the marketplace. *IEEE Personal Communications*, 4(5):58–64.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Bächlin, M., Hausdorff, J. M., Roggen, D., Giladi, N., Plotnik, M., and Tröster, G. (2009a). Online detection of freezing of gait in parkinson's disease patients: A performance characterization. In *Proceedings of the Fourth International Conference on Body Area Networks*,

BodyNets '09, pages 11:1–11:8, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Bächlin, M., Plotnik, M., Roggen, D., Giladi, N., Hausdorff, J. M., and Tröster, G. (2010a). A wearable system to assist walking of parkinson s disease patients. *Methods of information in medicine*, 49(1):88–95.

Bächlin, M., Plotnik, M., Roggen, D., Maidan, I., Hausdorff, J. M., Giladi, N., and Tröster, G. (2010b). Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society*, 14(2):436–446.

Bächlin, M., Roggen, D., Plotnik, M., Hausdorff, J. M., and Tröster, G. (2012). Experiences in developing a wearable gait assistant for parkinson's disease patients. In *Healthcare Sensor Networks*, pages 303 – 338. CRC Press Taylor & Francis Group, Florida, USA.

Bächlin, M., Roggen, D., Troster, G., Plotnik, M., Inbar, N., Meidan, I., Herman, T., Brozgol, M., Shaviv, E., Giladi, N., and Hausdorff, J. (2009b). Potentials of enhanced context awareness in wearable assistants for parkinson's disease patients with the freezing of gait syndrome. In *International Symposium on Wearable Computers, 2009. ISWC '09*, pages 123–130.

Cabestany, J., Perez Lopez, C., Sama, A., Moreno, J., Bayes, A., and Rodriguez-Molinero, A. (2013). REMPARK: When AI and technology meet parkinson disease assessment. In *Mixed Design of Integrated Circuits and Systems (MIXDES), 2013 Proceedings of the 20th International Conference*, pages 562–567.

Carlbom, I. and Paciorek, J. (1978). Planar geometric projections and viewing transformations. *ACM Comput. Surv.*, 10(4):465–502.

Chaudhuri, K. R. and Schapira, A. H. V. (2009). Non-motor symptoms of parkinson's disease: dopaminergic pathophysiology and treatment. *Lancet neurology*, 8(5):464–474.

Chaudhuri, K. R., Yates, L., and Martinez-Martin, P. (2005). The non-motor symptom complex of parkinson's disease: a comprehensive assessment is essential. *Current neurology and neuroscience reports*, 5(4):275–283.

Chee, R., Murphy, A., Danoudis, M., Georgiou-Karistianis, N., and Iansek, R. (2009). Gait freezing in parkinson's disease and the stride length sequence effect interaction. *Brain: a journal of neurology*, 132(Pt 8):2151–2160.

Chen, D., Pittsburgh, S. C. S., and Wactlar, H. (2007). People identification across ambient camera networks.

Chen, J., Kam, A. H., Zhang, J., Liu, N., and Shue, L. (2005). Bathroom activity monitoring based on sound. In *Proceedings of the Third International Conference on Pervasive Computing*, PERVASIVE'05, pages 47–61, Berlin, Heidelberg. Springer-Verlag.

Chen, L., Hoey, J., Nugent, C., Cook, D., and Yu, Z. (2012). Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 42(6):790–808.

Cho, Y., Nam, Y., Choi, Y.-J., and Cho, W.-D. (2008). SmartBuckle: Human activity recognition using a 3-axis accelerometer and a wearable camera. In *Proceedings of the 2Nd International Workshop on Systems and Networking Support for Health Care and Assisted Living Environments*, HealthNet '08, pages 7:1–7:3, New York, NY, USA. ACM.

Cole, B., Roy, S., and Nawab, S. (2011). Detecting freezing-of-gait during unscripted and unconstrained activity. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society,EMBC*, pages 5649–5652.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577.

Cootes, T. F., Edwards, G. J., and Taylor, C. J. (1998). Active appearance models. In Burkhardt, H. and Neumann, B., editors, *Computer Vision — ECCV'98*, number 1407 in Lecture Notes in Computer Science, pages 484–498. Springer Berlin Heidelberg.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, 20(3):273–297.

Coughlin, J., D'Ambrosio, L. A., Reimer, B., and Pratt, M. R. (2007). Older adult perceptions of smart home technologies: implications for research, policy & market innovations in healthcare. *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, 2007:1810–1815.

Coughlin, L. and Templeton, J. (1980). Hip fractures in patients with parkinson's disease. *Clinical orthopaedics and related research*, (148):192–195.

Cowie, D., Limousin, P., Peters, A., Hariz, M., and Day, B. L. (2012). Doorway-provoked freezing of gait in parkinson's disease. *Movement disorders: official journal of the Movement Disorder Society*, 27(4):492–499.

Cubo, E., Leurgans, S., and Goetz, C. G. (2004). Short-term and practice effects of metronome pacing in parkinson's disease patients with gait freezing while in the 'on' state: randomized single blind evaluation. *Parkinsonism & related disorders*, 10(8):507–510.

Cucchiara, R., Grana, C., Prati, A., and Vezzani, R. (2005). Probabilistic posture classification for human-behavior analysis. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 35(1):42–54.

CuPID (2011). Closed-loop system for personalized and at-home rehabilitation of people with parkinson's disease. EU Project. FP7-ICT-2011-7-288516.

Cédras, C. and Shah, M. (1995). Motion-based recognition a survey. *Image and Vision Computing*, 13(2):129–155.

Daniels, J., Fels, S., Kushniruk, A., Lim, J., and Ansermino, J. M. (2007). A framework for evaluating usability of clinical monitoring technology. *Journal of Clinical Monitoring and Computing*, 21(5):323–330.

DAPHNET (2006). Dynamic analysis of PHysiological NETworks. EU Project. EU FP6 018474-2.

Darrell, T., Demirdjian, D., Checka, N., and Felzenszwalb, P. (2001). Plan-view trajectory estimation with dense stereo background models. In *in Proceedings of the International Conference on Computer Vision*, pages 628–635.

Demiris, G., Oliver, D. P., Giger, J., Skubic, M., and Rantz, M. (2009). Older adults' privacy considerations for vision based recognition methods of eldercare applications. *Technology and Health Care: Official Journal of the European Society for Engineering and Medicine*, 17(1):41–48.

Demiris, G., Rantz, M., Aud, M., Marek, K., Tyrer, H., Skubic, M., and Hussam, A. (2004). Older adults' attitudes towards and perceptions of "smart home" technologies: a pilot study. *Medical Informatics and the Internet in Medicine*, 29(2):87–94.

Dey, A. K. (1998). Context-aware computing: The CyberDesk project.

Dietz, M. A., Goetz, C. G., and Stebbins, G. T. (1990). Evaluation of a modified inverted walking stick as a treatment for parkinsonian freezing episodes. *Movement disorders: official journal of the Movement Disorder Society*, 5(3):243–247.

Dishongh, T. J. and McGrath, M. (2009). *Wireless Sensor Networks for Healthcare Applications*. Artech House, Boston, 1 edition edition.

Djurić-Jovičić, M., Jovicic, N., Milovanović, I., Radovanović, S., Kresojević, N., and Popovic, M. (2010). Classification of walking patterns in parkinson's disease patients based on inertial sensor data. In *2010 10th Symposium on Neural Network Applications in Electrical Engineering (NEUREL)*, pages 3–6.

Domingos, P. and Pazzani, M. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Mach. Learn.*, 29(2-3):103–130.

Dutta, T. (2012). Evaluation of the kinect™ sensor for 3-d kinematic measurement in the workplace. *Applied Ergonomics*, 43(4):645–649.

Edgcomb, A. and Vahid, F. (2012). Privacy perception and fall detection accuracy for in-home video assistive monitoring with privacy enhancements. *SIGHIT Rec.*, 2(2):6–15.

Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57.

ElHelw, M., Pansiot, J., McIlwraith, D., Ali, R., Lo, B., and Atallah, L. (2009). An integrated multi-sensing framework for pervasive healthcare monitoring. In *3rd International Conference on Pervasive Computing Technologies for Healthcare, 2009. PervasiveHealth 2009*, pages 1–7.

ElSayed, M., Alsebai, A., Salaheldin, A., El Gayar, N., and ElHelw, M. (2010). Body and visual sensor fusion for motion analysis in ubiquitous healthcare systems. In *2010 International Conference on Body Sensor Networks (BSN)*, pages 250–254.

Evennou, F. and Marx, F. (2006). Advanced integration of WIFI and inertial navigation systems for indoor mobile positioning. *EURASIP J. Appl. Signal Process.*, 2006:164–164.

Fahn, S. (1995). The freezing phenomenon in parkinsonism. *Advances in neurology*, 67:53–63.

Fajen, B. R., Warren, W. H., Temizer, S., and Kaelbling, L. P. (2003). A dynamical model of visually-guided steering, obstacle avoidance, and route selection. *Int. J. Comput. Vision*, 54(1-3):13–34.

Fernandez-Sanchez, E. J., Diaz, J., and Ros, E. (2013). Background subtraction based on color and depth using active. *Sensors*, 13(7):8895–8915.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

Fishkin, K., Philipose, M., and Rea, A. (2005). Hands-on RFID: wireless wearables for detecting use of objects. In *Ninth IEEE International Symposium on Wearable Computers, 2005. Proceedings*, pages 38–41.

Foresight, P. H. S. (2013). Personal health systems: State of the art. Technical report.

Frank, K., Vera Nadales, M. J., Robertson, P., and Angermann, M. (2010). Reliable real-time recognition of motion related human activities using MEMS inertial sensors. Portland, Oregon, USA.

Fredriks, A., van Buuren, S., van Heel, W. J. M., Dijkman-Neerincx, R., Verloove-Vanhoric..., S., and Wit, J. (2005). Nationwide age references for sitting height, leg length, and sitting height/height ratio, and their diagnostic value for disproportionate growth disorders. *Archives of Disease in Childhood*, 90(8):807–812.

Freedland, R. L., Festa, C., Sealy, M., McBean, A., Elghazaly, P., Capan, A., Brozycki, L., Nelson, A. J., and Rothman, J. (2002). The effects of pulsed auditory stimulation on various gait measurements in persons with parkinson's disease. *NeuroRehabilitation*, 17(1):81–87.

Freund, Y. and Schapire, R. E. (1996). *Experiments with a New Boosting Algorithm*.

Friedewald, M., Vildjiounaite, E., Punie, Y., and Wright, D. (2007). Privacy, identity and security in ambient intelligence: A scenario analysis. *Telematics and Informatics*, 24(1):15–29.

Gavrila, D. M. (1999). The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98.

Gentile, C., Braga, A. J., and Kik, A. (2008). A comprehensive evaluation of joint range and angle estimation in indoor ultrawideband location systems. *EURASIP J. Wirel. Commun. Netw.*, 2008:36:1–36:11.

Giladi, Shabtai, Simon, Biran, Tal, and Korczyn (2000). Construction of freezing of gait questionnaire for patients with parkinsonism. *Parkinsonism & related disorders*, 6(3):165–170.

Giladi, N. and Hausdorff, J. M. (2006). The role of mental function in the pathogenesis of freezing of gait in parkinson's disease. *Journal of the neurological sciences*, 248(1-2):173–176.

Giladi, N., McDermott, M. P., Fahn, S., Przedborski, S., Jankovic, J., Stern, M., Tanner, C., and Parkinson Study Group (2001a). Freezing of gait in PD: prospective assessment in the DATATOP cohort. *Neurology*, 56(12):1712–1721.

Giladi, N., McMahon, D., Przedborski, S., Flaster, E., Guillory, S., Kostic, V., and Fahn, S. (1992). Motor blocks in parkinson's disease. *Neurology*, 42(2):333–339.

Giladi, N. and Nieuwboer, A. (2008). Understanding and treating freezing of gait in parkinsonism, proposed working definition, and setting the stage. *Movement disorders: official journal of the Movement Disorder Society*, 23 Suppl 2:S423–425.

Giladi, N., Tal, J., Azulay, T., Rascol, O., Brooks, D. J., Melamed, E., Oertel, W., Poewe, W. H., Stocchi, F., and Tolosa, E. (2009). Validation of the freezing of gait questionnaire in patients with parkinson's disease. *Movement Disorders*, 24(5):655–661.

Giladi, N., Treves, T. A., Simon, E. S., Shabtai, H., Orlov, Y., Kandinov, B., Paleacu, D., and Korczyn, A. D. (2001b). Freezing of gait in patients with advanced parkinson's disease. *Journal of neural transmission (Vienna, Austria: 1996)*, 108(1):53–61.

Gjoreski, H., Lustrek, M., and Gams, M. (2011). Accelerometer placement for posture recognition and fall detection. In *2011 7th International Conference on Intelligent Environments (IE)*, pages 47–54.

Goffredo, M., Schmid, M., Conforto, S., Carli, M., Neri, A., and D'Alessio, T. (2009). Markerless human motion analysis in gauss #x2013;laguerre transform domain: An application to sit-to-stand in young and elderly people. *IEEE Transactions on Information Technology in Biomedicine*, 13(2):207–216.

Gray, D., Brennan, S., and Tao, H. (2007). Evaluating appearance models for recognition, reacquisition, and tracking. In *In IEEE International Workshop on Performance Evaluation for Tracking and Surveillance, Rio de Janeiro*.

Hadim, S. and Mohamed, N. (2006). Middleware: middleware challenges and approaches for wireless sensor networks. *IEEE Distributed Systems Online*, 7(3):1–1.

Han, B. and Davis, L. (2005). On-line density-based appearance modeling for object tracking. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005*, volume 2, pages 1492–1499 Vol. 2.

Han, J. H., Lee, W. J., Ahn, T. B., Jeon, B. S., and Park, K.-S. (2003). Gait analysis for freezing detection in patients with movement disorder using three dimensional acceleration system. In *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2003*, volume 2, pages 1863–1865 Vol.2.

Handojoseno, A. M. A., Shine, J. M., Nguyen, T. N., Tran, Y., Lewis, S. J. G., and Nguyen, H. T. (2012). The detection of freezing of gait in parkinson's disease patients using EEG signals based on wavelet decomposition. *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2012:69–72.

Harville, M. (2004). Stereo person tracking with adaptive plan-view templates of height and occupancy statistics. *Image and Vision Computing*, 22(2):127–142.

Harville, M. (2005). Stereo person tracking with short and long term plan-view appearance models of shape and color. In *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005. AVSS 2005*, pages 522–527.

Harville, M. and Li, D. (2004). Fast, integrated person tracking and activity recognition with plan-view templates from a single stereo camera. In *Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition*, CVPR'04, pages 398–405, Washington, DC, USA. IEEE Computer Society.

Hausdorff, J. M., Balash, Y., and Giladi, N. (2003a). Time series analysis of leg movements during freezing of gait in parkinson's disease: akinesia, rhyme or reason? *Physica A: Statistical Mechanics and its Applications*, 321(3–4):565–570.

Hausdorff, J. M., Schaafsma, J. D., Balash, Y., Bartels, A. L., Gurevich, T., and Giladi, N. (2003b). Impaired regulation of stride variability in parkinson's disease subjects with freezing of gait. *Experimental brain research*, 149(2):187–194.

Hayashi, K., Hashimoto, M., Sumi, K., and Sasakawa, K. (2004). Multiple-person tracker with a fixed slanting stereo camera. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings*, pages 681–686.

Higuchi, T., Cinelli, M. E., Greig, M. A., and Patla, A. E. (2006). Locomotion through apertures when wider space for locomotion is necessary: adaptation to artificially altered bodily states. *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale*, 175(1):50–59.

Hoehn, M. M. and Yahr, M. D. (1967). Parkinsonism: onset, progression and mortality. *Neurology*, 17(5):427–442.

HOME (2008). Home-based empowered living for parkinson's disease patients. EU Project. AAL-2008-1-022.

Hong, X. and Nugent, C. D. (2013). Segmenting sensor data for activity monitoring in smart environments. *Personal Ubiquitous Comput.*, 17(3):545–559.

Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., and Burgard, W. (2013). OctoMap an efficient probabilistic 3d mapping framework based on octrees. *Auton. Robots*, 34(3):189–206.

Hu, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187.

Hughes, A. J., Daniel, S. E., Blankson, S., and Lees, A. J. (1993). A clinicopathologic study of 100 cases of parkinson's disease. *Archives of neurology*, 50(2):140–148.

Hull, R., Neaves, P., and Bedford-Roberts, J. (1997). Towards situated computing. In *, First International Symposium on Wearable Computers, 1997. Digest of Papers*, pages 146–153.

IEEE, S. (1990). IEEE standard glossary of software engineering terminology. *IEEE Std 610.12-1990*, pages 1–84.

ISO (2013). Basic human body measurements for technlogical design part 3: Worldwide and regional design ranges for use in ISO product standards. Technical Report ISO TC 159/SC 3 N 416.

Jacobs, J. V., Nutt, J. G., Carlson-Kuhta, P., Stephens, M., and Horak, F. B. (2009). Knee trembling during freezing of gait represents multiple anticipatory postural adjustments. *Experimental neurology*, 215(2):334–341.

Jain, V., Kimia, B., and Mundy, J. (2007). Background modeling based on subpixel edges. In *IEEE International Conference on Image Processing, 2007. ICIP 2007*, volume 6, pages VI – 321–VI – 324.

Jankovic, J. (1985). Long-term study of pergolide in parkinson's disease. *Neurology*, 35(3):296–299.

Jankovic, J. (2008). Parkinson's disease: clinical features and diagnosis. *Journal of Neurology, Neurosurgery & Psychiatry*, 79(4):368–376.

Jansen, S. E. M., Toet, A., and Werkhoven, P. J. (2011). Human locomotion through a multiple obstacle environment: strategy changes as a result of visual field limitation. *Experimental Brain Research. Experimentelle Hirnforschung. Experimentation Cerebrale*, 212(3):449–456.

Jaqaman, K., Loerke, D., Mettlen, M., Kuwata, H., Grinstein, S., Schmid, S. L., and Danuser, G. (2008). Robust single particle tracking in live cell time-lapse sequences. *Nature methods*, 5(8):695–702.

Jian, X., Xiao-qing, D., Sheng-jin, W., and You-shou, W. (2008). Background subtraction based on a combination of texture, color and intensity. In *9th International Conference on Signal Processing, 2008. ICSP 2008*, pages 1400–1405.

Jiang, Y. and Norman, K. E. (2006). Effects of visual and auditory cues on gait initiation in people with parkinson's disease. *Clinical rehabilitation*, 20(1):36–45.

Jovanov, E., Wang, E., Verhagen, L., Fredrickson, M., and Fratangelo, R. (2009). deFOG–a real time system for detection and unfreezing of gait of parkinson's patients. *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2009:5151–5154.

Jurie, F. and Dhome, M. (2002). Real time robust template matching. In *in British Machine Vision Conference 2002*, pages 123–131.

KaewTraKulPong, P. and Bowden, R. (2002). An improved adaptive background mixture model for real-time tracking with shadow detection. In Remagnino, P., Jones, G. A., Paragios, N., and Regazzoni, C. S., editors, *Video-Based Surveillance Systems*, pages 135–144. Springer US.

Kerr, G. K., Worringham, C. J., Cole, M. H., Lacherez, P. F., Wood, J. M., and Silburn, P. A. (2010). Predictors of future falls in parkinson disease. *Neurology*, 75(2):116–124.

Khan, A., Lee, Y. K., and Lee, S. (2010). Accelerometer's position free human activity recognition using a hierarchical recognition model. In *2010 12th IEEE International Conference on e-Health Networking Applications and Services (Healthcom)*, pages 296–301.

Khan, Z., Balch, T., and Dellaert, F. (2005). MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1805–1819.

Khoshelham, K. (2011). Accuracy analysis of kinect depth data. In *ISPRS workshop laser scanning*, volume 38, pages 133–138, Calgary Canada.

Khoshelham, K. and Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454.

Kim, K., Chalidabhongse, T. H., Harwood, D., and Davis, L. (2005). Real-time foreground-background segmentation using codebook model. *Real-Time Imaging*, 11(3):172–185.

Kleine-Cosack, C., Hennecke, M. H., Vajda, S., and Fink, G. A. (2010). Exploiting acoustic source localization for context classification in smart environments. In *Proceedings of the First International Joint Conference on Ambient Intelligence*, AmI'10, pages 157–166, Berlin, Heidelberg. Springer-Verlag.

Kompoliti, K., Goetz, C. G., Leurgans, S., Morrissey, M., and Siegel, I. M. (2000). "on" freezing in parkinson's disease: resistance to visual cue walking devices. *Movement disorders: official journal of the Movement Disorder Society*, 15(2):309–312.

Lalkhen, A. G. and McCluskey, A. (2008). Clinical tests: sensitivity and specificity. *Continuing Education in Anaesthesia, Critical Care & Pain*, 8(6):221–223.

Lane, N., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., and Campbell, A. (2010). A survey of mobile phone sensing. *IEEE Communications Magazine*, 48(9):140–150.

Ledger, S., Galvin, R., Lynch, D., and Stokes, E. K. (2008). A randomised controlled trial evaluating the effect of an individual auditory cueing device on freezing and gait speed in people with parkinson's disease. *BMC Neurology*, 8:46.

Lees, A. J. (1989). The on-off phenomenon. *Journal of Neurology, Neurosurgery, and Psychiatry*, 52(Suppl):29–37.

Leone, A., Diraco, G., and Siciliano, P. (2013). Context-aware AAL services through a 3d sensor-based platform. *Journal of Sensors*, 2013:e792978.

Lewis, G. N., Byblow, W. D., and Walt, S. E. (2000). Stride length regulation in parkinson's disease: the use of extrinsic, visual cues. *Brain: a journal of neurology*, 123 (Pt 10):2077–2090.

Luber, M., Spinello, L., and Arras, K. O. (2011). People tracking in rgb-d data with on-line boosted target models. In *In IEEE/RSJ Int. Conf. on*.

Lukashevich, H., Nowak, S., and Dunker, P. (2009). Using one-class SVM outliers detection for verification of collaboratively tagged image training sets. In *IEEE International Conference on Multimedia and Expo, 2009. ICME 2009*, pages 682–685.

Macht, M., Kaussner, Y., Möller, J. C., Stiasny-Kolster, K., Eggert, K. M., Krüger, H.-P., and Ellgring, H. (2007). Predictors of freezing in parkinson's disease: a survey of 6,620 patients. *Movement disorders: official journal of the Movement Disorder Society*, 22(7):953–956.

Madgwick, S. O. H., Harrison, A. J. L., and Vaidyanathan, A. (2011). Estimation of IMU and MARG orientation using a gradient descent algorithm. *IEEE ... International Conference on Rehabilitation Robotics: [proceedings]*, 2011:5975346.

Maggio, D. E. and Cavallaro, D. A. (2011). *Video Tracking: Theory and Practice*. Wiley Publishing, 1st edition.

Maidan, I., Plotnik, M., Mirelman, A., Weiss, A., Giladi, N., and Hausdorff, J. M. (2010). Heart rate changes during freezing of gait in patients with parkinson's disease. *Movement disorders: official journal of the Movement Disorder Society*, 25(14):2346–2354.

Marconi, R., Lefebvre-Caparros, D., Bonnet, A. M., Vidailhet, M., Dubois, B., and Agid, Y. (1994). Levodopa-induced dyskinesias in parkinson's disease phenomenology and pathophysiology. *Movement disorders: official journal of the Movement Disorder Society*, 9(1):2–12.

Martinez, M. and Stiefelhagen, R. (2013). Kinect unleashed: Getting control over high resolution depth maps. In *MVA2013 IAPR International Conference on Machine Vision Applications*, Kyoto, Japan.

Mathias, S., Nayak, U. S., and Isaacs, B. (1986). Balance in elderly patients: the "get-up and go" test. *Archives of physical medicine and rehabilitation*, 67(6):387–389.

Maurer, U., Smailagic, A., Siewiorek, D., and Deisher, M. (2006). Activity recognition and monitoring using multiple sensors on different body positions. In *International Workshop on Wearable and Implantable Body Sensor Networks, 2006. BSN 2006*, pages 4 pp.–116.

Mazilu, S., Blanke, U., Roggen, D., Tröster, G., Gazit, E., and Hausdorff, J. M. (2013a). Engineers meet clinicians: Augmenting parkinson's disease patients to gather information for gait rehabilitation. In *Proceedings of the 4th Augmented Human International Conference*, AH '13, pages 124–127, New York, NY, USA. ACM.

Mazilu, S., Calatroni, A., Gazit, E., Roggen, D., Hausdorff, J. M., and Tröster, G. (2013b). Feature learning for detection and prediction of freezing of gait in parkinson's disease. In

Perner, P., editor, *Machine Learning and Data Mining in Pattern Recognition*, number 7988 in Lecture Notes in Computer Science, pages 144–158. Springer Berlin Heidelberg.

Mazilu, S., Hardegger, M., Zhu, Z., Roggen, D., Troster, G., Plotnik, M., and Hausdorff, J. (2012). Online detection of freezing of gait with smartphones and machine learning techniques. In *2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*, pages 123–130.

McKenna, J. and Charif, N. (2004). Summarising contextual activity and detecting unusual inactivity in a supportive home environment. *Pattern Anal. Appl.*, 7(4):386–401.

McRae, C., Diem, G., Vo, A., O'Brien, C., and Seeberger, L. (2002). Reliability of measurements of patient health status: a comparison of physician, patient, and caregiver ratings. *Parkinsonism & Related Disorders*, 8(3):187–192.

Mitchell, S. L., Harper, D. W., Lau, A., and Bhalla, R. (2000). Patterns of outcome measurement in parkinson's disease clinical trials. *Neuroepidemiology*, 19(2):100–108.

Miyake, A., Friedman, N. P., Emerson, M. J., Witzki, A. H., Howerter, A., and Wager, T. D. (2000). The unity and diversity of executive functions and their contributions to complex "frontal lobe" tasks: a latent variable analysis. *Cognitive psychology*, 41(1):49–100.

Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2–3):90–126.

Moore, O., Peretz, C., and Giladi, N. (2007). Freezing of gait affects quality of life of peoples with parkinson's disease beyond its relationships with mobility and gait. *Movement disorders: official journal of the Movement Disorder Society*, 22(15):2192–2195.

Moore, S. T., MacDougall, H. G., and Ondo, W. G. (2008). Ambulatory monitoring of freezing of gait in parkinson's disease. *Journal of neuroscience methods*, 167(2):340–348.

Moore, S. T., Yungher, D. A., Morris, T. R., Dilda, V., MacDougall, H. G., Shine, J. M., Naismith, S. L., and Lewis, S. J. (2013). Autonomous identification of freezing of gait in parkinson's disease from lower-body segmental accelerometry. *Journal of NeuroEngineering and Rehabilitation*, 10(1):19.

Moravec, H. (1988). Sensor fusion in certainty grids for mobile robots. *AI Mag.*, 9(2):61–74.

Moreau, C., Defebvre, L., Bleuse, S., Blatt, J. L., Duhamel, A., Bloem, B. R., Destée, A., and Krystkowiak, P. (2008). Externally provoked freezing of gait in open runways in advanced parkinson's disease results from motor and mental collapse. *Journal of neural transmission (Vienna, Austria: 1996)*, 115(10):1431–1436.

Morris, M. E., Huxham, F., McGinley, J., Dodd, K., and Iansek, R. (2001a). The biomechanics and motor control of gait in parkinson disease. *Clinical biomechanics (Bristol, Avon)*, 16(6):459–470.

Morris, M. E., Iansek, R., Matyas, T. A., and Summers, J. J. (1994). Ability to modulate walking cadence remains intact in parkinson's disease. *Journal of Neurology, Neurosurgery, and Psychiatry*, 57(12):1532–1534.

Morris, S., Morris, M. E., and Iansek, R. (2001b). Reliability of measurements obtained with the timed "up & go" test in people with parkinson disease. *Physical Therapy*, 81(2):810–818.

Muñoz-Salinas, R. (2008). A bayesian plan-view map based approach for multiple-person detection and tracking. *Pattern Recognition*, 41(12):3665–3676.

Naismith, S. L., Shine, J. M., and Lewis, S. J. G. (2010). The specific contributions of set-shifting to freezing of gait in parkinson's disease. *Movement disorders: official journal of the Movement Disorder Society*, 25(8):1000–1004.

Najafi, B., Aminian, K., Loew, F., Blanc, Y., and Robert, P. (2002). Measurement of stand-sit and sit-stand transitions using a miniature gyroscope and its application in fall risk evaluation in the elderly. *IEEE Transactions on Biomedical Engineering*, 49(8):843–851.

Ni, L., Liu, Y., Lau, Y. C., and Patil, A. (2003). LANDMARC: indoor location sensing using active RFID. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)*, pages 407–415.

Niazmand, K., Tonn, K., Zhao, Y., Fietzek, U. M., Schroeteler, F., Ziegler, K., Ceballos-Baumann, A. O., and Lueth, T. (2011). Freezing of gait detection in parkinson's disease using accelerometer based smart clothes. In *2011 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 201–204.

Nieuwboer, A. (2008). Cueing for freezing of gait in patients with parkinson's disease: a rehabilitation perspective. *Movement disorders: official journal of the Movement Disorder Society*, 23 Suppl 2:S475–481.

Nieuwboer, A., De Weerdt, W., Dom, R., and Lesaffre, E. (1998). A frequency and correlation analysis of motor deficits in parkinson patients. *Disability and rehabilitation*, 20(4):142–150.

Nieuwboer, A. and Giladi, N. (2013). Characterizing freezing of gait in parkinson's disease: Models of an episodic phenomenon. *Movement Disorders*, 28(11):1509–1519.

Nieuwboer, A., Kwakkel, G., Rochester, L., Jones, D., van Wegen, E., Willems, A. M., Chavret, F., Hetherington, V., Baker, K., and Lim, I. (2007). Cueing training in the home improves gait-related mobility in parkinson's disease: the RESCUE trial. *Journal of Neurology, Neurosurgery, and Psychiatry*, 78(2):134–140.

Nocera, J. R., Stegemöller, E. L., Malaty, I. A., Okun, M. S., Marsiske, M., Hass, C. J., and National Parkinson Foundation Quality Improvement Initiative Investigators (2013). Using the timed up & go test in a clinical setting to predict falling in parkinson's disease. *Archives of Physical Medicine and Rehabilitation*, 94(7):1300–1305.

Nummiaro, K., Koller-Meier, E., and Van Gool, L. (2002). An adaptive color-based particle filter. *Image and Vision Computing*, 21(1):99–110.

Nutt, J. G., Bloem, B. R., Giladi, N., Hallett, M., Horak, F. B., and Nieuwboer, A. (2011). Freezing of gait: moving forward on a mysterious clinical phenomenon. *The Lancet Neurology*, 10(8):734–744.

Nutt, J. G., Woodward, W. R., Hammerstad, J. P., Carter, J. H., and Anderson, J. L. (1984). The "on-off" phenomenon in parkinson's disease. relation to levodopa absorption and transport. *The New England journal of medicine*, 310(8):483–488.

Oh, Y., Schmidt, A., and Woo, W. (2007). Designing, developing, and evaluating context-aware systems. In *International Conference on Multimedia and Ubiquitous Engineering, 2007. MUE '07*, pages 1158–1163.

Orr, R. J. and Abowd, G. D. (2000). The smart floor: A mechanism for natural user identification and tracking. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '00, pages 275–276, New York, NY, USA. ACM.

Pansiot, J., Stoyanov, D., McIlwraith, D., Lo, B. P. L., and Yang, G. Z. (2007). Ambient and wearable sensor fusion for activity recognition in healthcare monitoring systems. In Leonhardt, P. D.-I. D. m. S., Falck, D.-I. T., and Mähönen, P. D. P., editors, *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007)*, number 13 in IFMBE Proceedings, pages 208–212. Springer Berlin Heidelberg.

Parette, P. and Scherer, M. (2004). Assistive technology use and stigma. *Education and Training in Developmental Disabilities*, 3(39):217–226.

Parkinson, J. (2002). An essay on the shaking palsy. *The Journal of Neuropsychiatry and Clinical Neurosciences*, 14(2):223–236.

Patterson, D. J., Fox, D., Kautz, H., and Philipose, M. (2005). Fine-grained activity recognition by aggregating abstract object usage. In *Proceedings of the Ninth IEEE International Symposium on Wearable Computers*, ISWC '05, pages 44–51, Washington, DC, USA. IEEE Computer Society.

Pellegrini, S. and Iocchi, L. (2008). Human posture tracking and classification through stereo vision and 3d model matching. *J. Image Video Process.*, 2008:7:1–7:12.

PERFORM (2008). A soPhisticatEd multi-paRametric system FOR the continuous effective assessment and monitoring of motor status in parkinson's disease and other neurodegenerative diseases. EU Project. FP7-ICT-2007-1-215952.

Petrushin, V. A., Wei, G., and Gershman, A. V. (2006). Multiple-camera people localization in an indoor environment. *Knowl. Inf. Syst.*, 10(2):229–241.

Piccardi, M. (2004). Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 3099–3104 vol.4.

Plotnik, M., Giladi, N., Balash, Y., Peretz, C., and Hausdorff, J. M. (2005). Is freezing of gait in parkinson's disease related to asymmetric motor function? *Annals of neurology*, 57(5):656–663.

Plotnik, M., Giladi, N., and Hausdorff, J. M. (2008). Bilateral coordination of walking and freezing of gait in parkinson's disease. *The European journal of neuroscience*, 27(8):1999–2006.

Plotnik, M., Giladi, N., and Hausdorff, J. M. (2012). Is freezing of gait in parkinson's disease a result of multiple gait impairments? implications for treatment. *Parkinson's Disease*, 2012:1–8.

Podsiadlo, D. and Richardson, S. (1991). The timed "up & go": a test of basic functional mobility for frail elderly persons. *Journal of the American Geriatrics Society*, 39(2):142–148.

Poppe, R. (2007). Vision-based human motion analysis: An overview. *Comput. Vis. Image Underst.*, 108(1-2):4–18.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990.

Pérez, P., Hue, C., Vermaak, J., and Gangnet, M. (2002). Color-based probabilistic tracking. In Heyden, A., Sparr, G., Nielsen, M., and Johansen, P., editors, *Computer Vision — ECCV 2002*, number 2350 in Lecture Notes in Computer Science, pages 661–675. Springer Berlin Heidelberg.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. (2009). {ROS}: an open-source robot operating system.

Rahimpour, M., Lovell, N. H., Celler, B. G., and McCormick, J. (2008). Patients' perceptions of a home telecare system. *International journal of medical informatics*, 77(7):486–498.

REMPARK (2011). Personal health device for the remote and autonomous management of parkinson's disease. EU Project. FP7-ICT-2011-7-287677.

Retscher, G. (2007). Test and integration of location sensors for a multi-sensor personal navigator. *The Journal of Navigation*, 60(01):107–117.

Ribeiro, P. C. and Santos-victor, J. (2005). Human activity recognition from video: modeling, feature selection and classification architecture. In *International Workshop on Human Activity Recognition and Modeling(HAREM)*.

Riess, T. J. (1998). Gait and parkinson's disease: a conceptual model for an augmented-reality based therapeutic device. *Studies in health technology and informatics*, 58:200–208.

Rish, I. (2001). An empirical study of the naive bayes classifier.

Roalter, L., Kranz, M., and Möller, A. (2010). A middleware for intelligent environments and the internet of things. In *Proceedings of the 7th International Conference on Ubiquitous Intelligence and Computing*, UIC'10, pages 267–281, Berlin, Heidelberg. Springer-Verlag.

Rochester, L., Nieuwboer, A., Baker, K., Hetherington, V., Willems, A.-M., Chavret, F., Kwakkel, G., Van Wegen, E., Lim, I., and Jones, D. (2007). The attentional cost of external rhythmical cues and their impact on gait in parkinson's disease: effect of cue modality and task complexity. *Journal of neural transmission (Vienna, Austria: 1996)*, 114(10):1243–1248.

Rodriguez-Martin, D., Samà, A., Perez-Lopez, C., Català, A., Cabestany, J., and Rodriguez-Molinero, A. (2013). SVM-based posture identification with a single waist-located triaxial accelerometer. *Expert Systems with Applications*, 40(18):7203–7211.

Rodríguez-Martín, D., Pérez-López, C., Cabestany, J., Català, A., and Rodriguez-Molinero, A. (2014). Enhancing FoG detection by means of postural context using a waist accelerometer. Dead Sea, Israel.

Rodríguez-Martín, D., Pérez-López, C., Samà, A., Cabestany, J., and Català, A. (2013). A wearable inertial measurement unit for long-term monitoring in the dependency care area. *Sensors (Basel, Switzerland)*, 13(10):14079–14104.

ROS.org (2014). http ://wiki.ros.org/message_filters. Online source.

Safavian, S. and Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man and Cybernetics*, 21(3):660–674.

Saldana, J. (2009). *The Coding Manual for Qualitative Researchers*. SAGE.

Samà, A., Perez Lopez, C., Rodríguez Martín, D., Cabestany, J., Moreno, J., and Rodriguez-Molinero, A. (2013). A heterogeneous database for movement knowledge extraction in parkinson's disease. In *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Puerto de la Cruz, Spain.

Satta, R., Fumera, G., and Roli, F. (2011a). Exploiting dissimilarity representations for person re-identification. In *Proceedings of the First international conference on Similarity-based pattern recognition*, SIMBAD'11, pages 275–289, Berlin, Heidelberg. Springer-Verlag.

Satta, R., Fumera, G., and Roli, F. (2012). Fast person re-identification based on dissimilarity representations. *Pattern Recogn. Lett.*, 33(14):1838–1848.

Satta, R., Fumera, G., Roli, F., Cristani, M., and Murino, V. (2011b). A multiple component matching framework for person re-identification. In *Proceedings of the 16th international conference on Image analysis and processing - Volume Part II*, ICIAP'11, pages 140–149, Berlin, Heidelberg. Springer-Verlag.

Satta, R., Pala, F., Fumera, G., and Roli, F. (2013). Real-time appearance-based person re-identification over multiple KinectTM cameras. Barcelona, Spain.

Schaafsma, J. D., Balash, Y., Gurevich, T., Bartels, A. L., Hausdorff, J. M., and Giladi, N. (2003). Characterization of freezing of gait subtypes and the response of each to levodopa in parkinson's disease. *European journal of neurology: the official journal of the European Federation of Neurological Societies*, 10(4):391–398.

Schilit, B. and Theimer, M. (1994). Disseminating active map information to mobile hosts. *IEEE Network*, 8(5):22–32.

Schölkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J., and Platt, J. (2000). *Support Vector Method for Novelty Detection*.

Shafique, K. and Shah, M. (2003). A non-iterative greedy algorithm for multi-frame point correspondence. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 51–65.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*.

Simon, D. and Boring, J. R. (1990). Sensitivity, specificity, and predictive value. In Walker, H. K., Hall, W. D., and Hurst, J. W., editors, *Clinical Methods: The History, Physical, and Laboratory Examinations*. Butterworths, Boston, 3rd edition.

Sinha, N., Gupta, M., and Rao, D. H. (2000). Dynamic neural networks: an overview. In *Proceedings of IEEE International Conference on Industrial Technology 2000*, volume 1, pages 491–496 vol.2.

Smith, K. (2007). *Bayesian Methods for Visual Multi-object Tracking with Applications to Human Activity Recognition*.

Snijders, A. H., Nijkrake, M. J., Bakker, M., Munneke, M., Wind, C., and Bloem, B. R. (2008). Clinimetrics of freezing of gait. *Movement disorders: official journal of the Movement Disorder Society*, 23 Suppl 2:S468–474.

Sobral, A. and Vacavant, A. (2014). A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21.

Spildooren, J., Vercruysse, S., Desloovere, K., Vandenberghe, W., Kerckhofs, E., and Nieuwboer, A. (2010). Freezing of gait in parkinson's disease: the impact of dual-tasking and turning. *Movement disorders: official journal of the Movement Disorder Society*, 25(15):2563–2570.

Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages –252 Vol. 2.

Steele, R., Lo, A., Secombe, C., and Wong, Y. K. (2009). Elderly persons' perception and acceptance of using wireless sensor networks to assist healthcare. *International Journal of Medical Informatics*, 78(12):788–801.

Steinhauer, H. J., Marsland, S., and Guesgen, H. W. (2012). Context awareness for a smart environment utilizing context maps and dempster-shafer theory. In Donnelly, M., Paggetti, C., Nugent, C., and Mokhtari, M., editors, *Impact Analysis of Solutions for Chronic Disease Prevention and Management*, number 7251 in Lecture Notes in Computer Science, pages 270–273. Springer Berlin Heidelberg.

Stoyanov, T., Mojtahedzadeh, R., Andreasson, H., and Lilienthal, A. J. (2013). Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications. *Robot. Auton. Syst.*, 61(10):1094–1105.

Su, X., Tong, H., and Ji, P. (2014). Activity recognition with smartphone sensors. *Tsinghua Science and Technology*, 19(3):235–249.

Subramanya, A., Raj, A., Bilmes, J., and Fox, D. (2006). Hierarchical models for activity recognition. In *2006 IEEE 8th Workshop on Multimedia Signal Processing*, pages 233–237.

Sundaresan, A. and Chellappa, R. (2009). Multicamera tracking of articulated human motion using shape and motion cues. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 18(9):2114–2126.

Suteerawattananon, M., Morris, G. S., Etnyre, B. R., Jankovic, J., and Protas, E. J. (2004). Effects of visual and auditory cues on gait in individuals with parkinson's disease. *Journal of the neurological sciences*, 219(1-2):63–69.

Swain, P. H. and Hauska, H. (1977). The decision tree classifier: Design and potential. *IEEE Transactions on Geoscience Electronics*, 15(3):142–147.

Takač, B., Català, A., Cabestany, J., Chen, W., and Rauterberg, M. (2012a). A system for inference of spatial context of parkinson's disease patients. *Studies in health technology and informatics*, 177:126–131.

Takač, B., Català, A., Rauterberg, M., and Chen, W. (2014). People identification for domestic non-overlapping RGB-d camera networks. In *Multi-Conference on Systems, Signals Devices (SSD), 2014 11th International*, pages 1–6.

Takač, B., Català, A., Rodríguez Martín, D., Chen, W., and Rauterberg, M. (2012b). Ambient sensor system for freezing of gait detection by spatial context analysis. In *Proceedings of the 4th international conference on Ambient Assisted Living and Home Care*, IWAAL'12, pages 232–239, Berlin, Heidelberg. Springer-Verlag.

Takač, B., Català, A., Rodríguez Martín, D., van der Aa, N., Chen, W., and Rauterberg, M. (2013). Position and orientation tracking in a ubiquitous monitoring system for parkinson disease patients with freezing of gait symptom. 1.

Tapia, E. M., Intille, S. S., and Larson, K. (2004). Activity recognition in the home using simple and ubiquitous sensors. In Ferscha, A. and Mattern, F., editors, *Pervasive Computing*, number 3001 in Lecture Notes in Computer Science, pages 158–175. Springer Berlin Heidelberg.

Tastan, B. and Sukthankar, G. (2011). Leveraging human behavior models to predict paths in indoor environments. *Pervasive and Mobile Computing*, 7(3):319–330.

Teague, M. (1980). Image analysis via the general theory of moments. *Journal of the Optical Society of America (1917-1983)*, 70:920–930.

Teixeira, T., Dublon, G., and Savvides, A. (2010). A survey of human-sensing: Methods for detecting presence, count, location, track, and identity. Technical report, ENALAB, Yale University.

Tesoriero, R., Gallud, J., Lozano, M., and Penichet, V. (2008). Using active and passive RFID technology to support indoor location-aware systems. *IEEE Transactions on Consumer Electronics*, 54(2):578–583.

Thome, N., Merad, D., and Miguet, S. (2008). Learning articulated appearance models for tracking humans: A spectral graph matching approach. *Signal Processing: Image Communication*, 23(10):769–787.

Thompson, P. D. and Marsden, C. D. (2000). Walking disorders. In *Neurology in Clinical Practice*, pages 341–354. Butterworth Heinemann, 3rd edition.

Thrun, S. and Bü, A. (1996). Integrating grid-based and topological maps for mobile robot navigation. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, pages 944–950, Portland, Oregon. AAAI Press.

Torres-Solis, J., H., T., and Chau, T. (2010). A review of indoor localization technologies: towards navigational assistance for topographical disorientation. In Villanueva Molina, F. J., editor, *Ambient Intelligence*. InTech.

Tripoliti, E. E., Tzallas, A. T., Tsipouras, M. G., Rigas, G., Bougia, P., Leontiou, M., Konitsiotis, S., Chondrogiorgi, M., Tsouli, S., and Fotiadis, D. I. (2013). Automatic detection of freezing of gait events in patients with parkinson's disease. *Computer methods and programs in biomedicine*, 110(1):12–26.

Tuzel, O., Porikli, F., and Meer, P. (2005). A bayesian approach to background modeling. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, 2005. CVPR Workshops*, pages 58–58.

van Wegen, E., Lim, I., de Goede, C., Nieuwboer, A., Willems, A., Jones, D., Rochester, L., Hetherington, V., Berendse, H., Zijlmans, J., Wolters, E., and Kwakkel, G. (2006). The effects of visual rhythms and optic flow on stride patterns of patients with parkinson's disease. *Parkinsonism & related disorders*, 12(1):21–27.

Veenman, C., Reinders, M. J. T., and Backer, E. (2001). Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(1):54–72.

Wallhagen, M. I. and Brod, M. (1997). Perceived control and well-being in parkinson's disease. *Western journal of nursing research*, 19(1):11–25; discussion 25–31.

Weidmann, U. (1993). *Transporttechnik der Fussgänger: transporttechnische Eigenschaften des Fussgängerverkehrs ; Literaturauswertung*. IVT.

Welford, B. P. (1962). Note on a method for calculating corrected sums of squares and products. *Technometrics*, 4(3):419–420.

Willems, A. M., Nieuwboer, A., Chavret, F., Desloovere, K., Dom, R., Rochester, L., Jones, D., Kwakkel, G., and Van Wegen, E. (2006). The use of rhythmic auditory cues to influence gait in patients with parkinson's disease, the differential effect for freezers and non-freezers, an explorative study. *Disability and rehabilitation*, 28(11):721–728.

Willems, A.-M., Nieuwboer, A., Chavret, F., Desloovere, K., Dom, R., Rochester, L., Kwakkel, G., van Wegen, E., and Jones, D. (2007). Turning in parkinson's disease patients and controls: the effect of auditory cues. *Movement disorders: official journal of the Movement Disorder Society*, 22(13):1871–1878.

Williams, D. R., Watt, H. C., and Lees, A. J. (2006). Predictors of falls and fractures in bradykinetic rigid syndromes: a retrospective study. *Journal of Neurology, Neurosurgery & Psychiatry*, 77(4):468–473.

Wilson, D. H. and Atkeson, C. (2005). Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors. In *Proceedings of the Third International Conference on Pervasive Computing*, PERVASIVE'05, pages 62–79, Berlin, Heidelberg. Springer-Verlag.

Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. (1996). Pfinder: real-time tracking of the human body. In *, Proceedings of the Second International Conference on Automatic Face and Gesture Recognition, 1996*, pages 51–56.

Wren, C. R. and Tapia, E. M. (2006). Toward scalable activity recognition for sensor networks. In *Proceedings of the Second International Conference on Location- and Context-Awareness*, LoCA'06, pages 168–185, Berlin, Heidelberg. Springer-Verlag.

Xia, L., Chen, C.-C., and Aggarwal, J. (2011). Human detection using depth information by kinect. In *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 15–22.

Xiang, Z. and al, e. (2004). *A wireless LAN-based indoor positioning technology*.

Yang, J.-Y., Chen, Y.-P., Lee, G.-Y., Liou, S.-N., and Wang, J.-S. (2007). Activity recognition using one triaxial accelerometer: A neuro-fuzzy classifier with feature reduction. In Ma, L., Rauterberg, M., and Nakatsu, R., editors, *Entertainment Computing – ICEC 2007*, number 4740 in Lecture Notes in Computer Science, pages 395–400. Springer Berlin Heidelberg.

Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking: A survey. *ACM Comput. Surv.*, 38(4).

Yilmaz, A., Li, X., and Shah, M. (2004). Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1531–1536.

Yim, J., Park, C., Joo, J., and Jeong, S. (2008). Extended kalman filter for wireless LAN based indoor positioning. *Decision Support Systems*, 45(4):960–971.

Yogev, G., Giladi, N., Peretz, C., Springer, S., Simon, E. S., and Hausdorff, J. M. (2005). Dual tasking, gait rhythmicity, and parkinson's disease: which aspects of gait are attention demanding? *The European journal of neuroscience*, 22(5):1248–1256.

Zabaleta, H., Keller, T., and Fimbel, E. (2008). Gait analysis in frequency domain for freezing detection in patients with parkinson's disease. *Gerontechnology*, 7(2).

Zhang, C., Tian, Y., and Capezuti, E. (2012). Privacy preserving automatic fall detection for elderly using RGBD cameras. In *Proceedings of the 13th International Conference on Computers Helping People with Special Needs - Volume Part I*, ICCHP'12, pages 625–633, Berlin, Heidelberg. Springer-Verlag.

Zhang, H. and Xu, D. (2006). Fusing color and texture features for background model. In Wang, L., Jiao, L., Shi, G., Li, X., and Liu, J., editors, *Fuzzy Systems and Knowledge Discovery*, number 4223 in Lecture Notes in Computer Science, pages 887–893. Springer Berlin Heidelberg.

Zhang, S., McCullagh, P., Nugent, C., and Zheng, H. (2010). Activity monitoring using a smart phone's accelerometer with hierarchical classification. In *2010 Sixth International Conference on Intelligent Environments (IE)*, pages 158–163.

Zhao, Y., Tonn, K., Niazmand, K., Fietzek, U. M., D'Angelo, L., Ceballos-Baumann, A., and Lueth, T. (2012). Online FOG identification in parkinson's disease with a time-frequency combined algorithm. In *2012 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 192–195.

Zhu, C. and Sheng, W. (2011). Motion- and location-based online human daily activity recognition. *Pervasive and Mobile Computing*, 7(2):256–269.

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004*, volume 2, pages 28–31 Vol.2.

Publications

Journals

- Takač, B., Català, A., Rodríguez Martín, D., van der Aa, N., Chen, W., and Rauterberg, M. (2013). Position and Orientation Tracking in a Ubiquitous Monitoring System for Parkinson's Disease Patients With Freezing of Gait Symptom. JMIR Mhealth Uhealth 2013;1(2):e14

Conferences

- Takač, B., Català, A., Cabestany, J., Chen, W., and Rauterberg, M. (2012). A System for Infer- ence of Spatial Context of Parkinson's Disease Patients. Proceedings of the 9th International Conference on Wearable Micro and Nano Technologies for Personalized Health Technologies for Personalized Health (pHealth 2012), 26-28 June 2012, Porto, Portugal, (Studies in Health Technology and Informatics, 177, pp. 126-131). Amsterdam: IOS Press.

- Takač, B., Català, A., Rodríguez Martín, D., Chen, W., and Rauterberg, M. (2012). Ambient Sensor System for Freezing of Gait Detection by Spatial Context Analysis. In Proceedings of the 4th interna- tional conference on Ambient Assisted Living and Home Care, IWAAL'12, pages 232–239, Berlin, Heidelberg. Springer-Verlag.

- Takač, B., Català, A., Chen, W. and Rauterberg, M. (2014). People Identification for Domestic Non- overlapping RGB-D Camera Networks. In Multi-Conference on Systems, Signals Devices (SSD), 2014 11th International, pages 1–6.

Other

- Takač, B., Chen, W, and Rauterberg, M.,. (2013). Toward a Domestic System to Assist People with Parkinson's. SPIE Newsroomg. Online, DOI: 10.1117/2.1201305.004884

Acknowledgments

This dissertation was produced in two partner universities, under the guidance of three supervisors. I thank my primary supervisor, prof. Andreu Català from Technical University of Catalonia, for welcoming me into his reseach group and providing me with all the logistics and experience for practical research with the patients. Thank you Andreu for giving me the freedom to explore, to find new technical challenges that interested me, and for always having patience and trust that I will be able to overcome those same challenges. In equal measure, I would like to give thanks to my supervisors from Eindhoven University of Technology, my promotor prof. Matthias Rauterberg and co-promotor dr. Wei Chen. Thank you Matthias, for being my focusing and propelling force. You helped me to find direction in the moments when it was needed, reminded me always to think like a scientist and gave incentives and ideas for publication writing when it was important. Thank you also for your guidance through the thesis writing process in the final year. Thank you Wei for being available for dicussion every time I would drop by your office (which was almost always unannounced). I appreciate your advices and help that you provided in planning future research and publication steps.

My great gratitude goes towards the memebers of my reading committee, prof. Loe Feijs, prof. Juan Manuel Moreno Arróstegui, dr. A. Rodríguez-Molinero and dr. Nico van der Aa. Thank you all for your effort to read and give detailed feedback on my manuscript. Your comments and remarks were very valuable inputs that helped me to improve the quality of the thesis.

The final results of the dissertation depended on collecting the data from the Parkinson's disease patients. I was privileged to have the access to the therapy sessions in Unidad del Parkinson y Trastornos del Movimiento of Teknon Hospital in Barcelona. For this privilege and help during data collection experiments, I would like to thank to dr. Angels Bayés and Sheila Alcaine García. Also, I would like to thank to all the people in Catalunya who have agreed to participate in my data collection experiments. Thank you for sharing the privacy of your home and finding strength for one more walk.

A joint doctoral programme which requires reallocation between two universities is very logistically challenging. In four years, I have moved the same number of times; probably most than any other student in the programme. I was lucky to avoid any major visa and accomodation problems thanks to the help from Mercè Cabané and Neus Salleras at UPC, and Ellen Konijnenberg and Gaby Jansen at TU/e. They were also very helpful when I had any other

organizational issue during my stays at each of the two institutions.

The best part of the joint doctoral programme is that by being a member of two different research groups, I met twice as many colleagues as one normally would have. Many thanks to my fellows at both universities - Jorge Luis Reyes, Anh-Tuan Nguyen, Leonid Ivonin, Huang-Ming Chang, John NA Brown, Wilbert Aguilar, Diego Montero, Daniel Rodríguez-Martín, Albert Samà, Sibrecht Bouwstra, Misha Croes, Bram van der Vlist, Iyinoluwa Ayoola and Ehsan Baha - for sharing the PhD experience with me, through our daily lunch discussions and other fun events that we had outside of the lab.

I dedicate this thesis to my parents and my sister, for all their support, patience and love that they have given me while I was living far away from home, pursuing my academic goals. My final and biggest thanks goes to my life partner, best friend and colleague Marija Nakevska. Thank you for being there for me every step of the way.

Curriculum Vitae

Boris Takač was born on 22nd of April 1981 in Sisak, Croatia. He received a diploma of a Graduate Engineer in Electrical Engineering (5 years) and a postgraduate degree of Magister in Electrical Engineering from the Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia, in 2004 and 2009 respectively.

Between 2004 and 2008, he worked as a research and development engineer for measurement and monitoring systems in Končar Electrical Engineering Institute in Zagreb, Croatia. From 2008 to 2010, he was a student of Erasmus Mundus Master in Advanced Robotics, after which he obtained a MSc in Computer Science and Engineering from the Faculty of Engineering, University of Genova, Italy and a MSc in Robotics and Automation from École Centrale de Nantes, France.

Since January 2011, he has been a doctoral candidate in Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments. He spent his first and third year of the doctoral studies at Universitat Politècnica de Catalunya in the Technical Research Centre for Dependency Care and Autonomous Living in Vilanova i la Geltrú, Spain. During the second and the fourth year of the doctorate he was a member of the Designed Intelligence research group, at the Department of Industrial Design of Eindhoven University of Technology, The Netherlands. Through his PhD project he specialized in design and development of systems in the area of Assistive Technologies for Health and Dependency Care.