University College
of Southeast Norway

www.usn.no

FMH606 Master's Thesis 2017

Industrial IT and Automation

AI Techniques in Assisting Elderly people at Home with Unobtrusive Supervision of Events Related to Health and Safety

Karina Kaspersen

Faculty of Technology, Natural sciences and Maritime Sciences
Campus Porsgrunn

University College of Southeast Norway

www.usn.no

Course: FMH606 Master's Thesis, 2017

Title: AI Techniques in assisting elderly people at home with unobtrusive supervision of events related to health and safety

Number of pages: 105

Keywords: AI, Classification, Assistance Technology, Wireless Sensor Networks

Student: Karina Kaspersen

Supervisor: Saba Mylvaganam and Alexander Jonsaas

Availability: Open

Approved for archiving: ____

(supervisor signature)

Summary:

AI techniques are used widely in merging technologies, such as smartphones, PC, Internet of Things (IoT) with cheap and computational power on chips. Device sizes, range of wireless communication and governmental initiatives are part of the focus on implementing assistive technology (AT) in the elderly care sector, enabling elderly to live longer independently at home, while still feel safe.

As fast response and comfort are key issues in AT monitoring systems for elderly, the objective of this present work is to identify sensors and placement on the body and in the ambient environment for timely and correct detection of critical events for unobtrusive monitoring. An experimental procedure was performed involving recordings from various numbers of sensors with different placement to distinguish proper placement and types of sensors to be applied in such systems through training and evaluation of AI models to detect critical events.

A background study on techniques and sensors used in the field, a survey on this type of technology in other parts of the world, as well as in Norway together with ethical aspects of using such technology for elderly, formed the basis of the experimental procedures arranged. Different experiments involving recording of both ciritical as well as trivial events were performed to test and evaluate the generated AI models.

The AI models gave high accuracy (>92 %) of properly identifying fall from non-fall events. An AI model using only wrist-based sensors gave 100 % accuracy after feature reduction, which is promising in such a system for the user to only wear the sensor unit on the wrist, preferably in form of a smartwatch. An geofence threshold-based AI model for detecting if a user had not come home after a pre-set time was also simulated, which gave a satisfactory result. A proposed system architecture for implementing these detection models in an application software for smartwatch and smartphone are also elaborated. For further work, a suggestion of collecting more experimental data from several subjects has been made to study the model performance from a larger data set, as well as analysing power consumption of such a system running on wireless communication

Preface

As part of the master's program in Industrial IT and Automation at University College of Southeast Norway, a master thesis on applying AI techniques for a proposed unobtrusive monitoring system of elderly was carried out during the spring semester of 2017.

Sensor networking and data fusion techniques has been used at USN since the beginning of the 90s. Today, it is of interest to both elderly, the care sector, relatives, and the Norwegian government to implement monitoring systems for elderly, enabling them to live longer at home, while still being provided safety and care, as well as being given necessary help if a critical event was to occur. This formed the background for the project task, were different critical events for such an unobtrusive monitoring system is to be identified, types and placement and number of sensors to detect such events using AI models is to be generated and evaluated. The AI models will be evaluated to investigate the possibility of such a system to be developed using these sensors and placement on the body and ambient environment.

I would like to thank my supervisors Professor Saba Mylvaganam and Alexander Jonsaas, in addition to Janne Dugstad, Director of Vitensenteret in Drammen, for valuable advice, guidance, and encouragement throughout the project process. I would also like to thank The University College of Southeast Norway, USN, for supplying the necessary equipment.

The report requires some pre-existing knowledge in the field of AI techniques.

The following software's was utilized throughout the project:

- Arduino
- MATLAB R2016a
- NI LabVIEW 2016
- Unscrambler® X
- MS. Office

Porsgrunn, 15.05.17

Karina Kaspersen

Note on Terminology

Description of expressions that are used extensively throughout this report that can be unclear, or have other definitions in other literature of the discipline regarding the topic of the present work, assistive technology (AT)

Expression	***Description***
Assistive Technology	Assistive technology (AT) is a term that includes assistive, adaptive, and rehabilitative devices for people with disabilities. It also includes the process used in selecting, locating, and using the AT [1]. In the present work, used in the context of technology aimed for elderly
Care Personnel	Includes nursing care staff and informal caregivers, (carrying, daily care and lifting). Elderly care personnel/care staff
Caretakers	Often close relatives or someone responsible for the elderly user
Critical Event	A critical event that are being detected by a monitoring system that must be handled by alarm sentral, care personell or caretakers. This could be that the user has fallen, gone outside etc.
Data Collection Units	A unit; wearable, ambient or a smartphone that collects experimental data to be used in generation of AI models to detect critical events of elderly living independently at home (ELIAH)
Elderly Home Care Patients	Often the users, but these people are being provided care by the community that includes Care Personell, Pysician or other Health Workers. Often refered to as persons over 65 years old.
Feature	Used in machine learning. An individual measurable property of a phenomenon being observed [2]
The Present Work	The work performed throughout this specific project
Unobtrusive Monitoring	Unobtrusive is "*not blatant, arresting, or aggressive*" [3]. Monitoring that does not attempt to invade the users personal space, by excessive sensors, events to be monitored, data storing and so on. Only the necessary events will be monitored based on individual use while keeping the aim of the monitoring activities
Users	Users of the AT, the elderly citizens these systems are being provided for. Not only patients, but users of Smart Homes and Fall events, etc. These users could also be provided the monitoring systems by close family members or other caregivers. Often refered to as persons over 65 years old.

Nomenclature

List of Symbols

Symbol	Description	Unit	Symbol	Description	Unit
a	Acceleration	$[g]$	l	No. of E / Inputs	
v	Velocity	$[m/s]$	j	Feature Ranks	
C	Capacitance	$[fF]$	p	Percentile	
ω	Angular velocity	$[°/s]$	A	Accuracy	[%]
θ	Angle	[°]	i	th percentile	
t	Time	[s/min/h]	f	Function	
Φ	Magnetic Flux	$[Wb]$	k	No. of nearest neighbours	
δ	Door Activity	[0,1]	Y	Output/Target/Classification Class	
φ	Latitude	[°]	β	Normal vector	
λ	Longitude	[°]	d	Dimension	
ø	Radius	[°]	b	Hyperplane Parameter	
E	Experiment/Input		e	Error	[%]
M	Magnitude		W	Weight	
S	Sum		σ	Standard Deviation	
F	Feature		X	Signal	
c	Coefficients of Wavelet Transform		Q	Level Wavelet Decomposition	
P	Peak		L	Vector that contains the No. of c by Q	
^	Or				
||	And				
$r = 1 \ldots R$	Data points in E signals				
n	No. of Features				

Abbreviations

Abbreviations of relevant and extensive used expressions in terms of the topic of this report.

List of Abbreviations	
AI	Artificial Intelligence
ALD	Activity of Daily Living
ANN	Artificial Neural Networks
AT	Assistive Technology
BLE	Bluetooth Low Energy
CE	Cross-Entropy
DCU	Data Collection Unit
DMP	Digital Motion Processor
DOF	Degrees of Freedom
ELIAH	Elderly Living Independently at Home
EMR	Electronic Medical Record
I2C	Inter-Integrated Circuit
ICT	Information- and Communication-Technology
IoT	Internet of Things
kNN	k-Nearest Neighbour
MEMS	Manufacture Microelectromechanical Systems
NWTP	National Welfare Technology Programme
PCA	Principal Component Analysis
PWM	Pulse Width Modulation
SCI	Spinal Cord Injury
SVM	Support Vector Machine
USN	University College of Southeast Norway
Xcorr	Cross-correlation

Contents

1 Introduction

This chapter describes the background, objectives and goals of the project conducted during the present work, and description of project activities and report disposition

1.1 Background and Objectives

The elderly care sector is on the verge of being reshaped by technology as the sensors and devices has become smaller, communication technology has higher range and applicability and the powerful use of smart technology in mobile phones, watches, home appliances among other has become commonplace [4]. As more elderly are familiar with the use of mobile technology nowadays, it is easier for them to adapt to the use of unobtrusive monitoring upon their daily life to ensure their independence and safety. The implementation of welfare technology, with the focus on assistive living systems in the present work, in the Norwegian care sector is a governmental initiative to help elderly live longer independently at home as this will increase quality of life for elderly user, moving into care homes can be put off, critical events are detected and relevant actors are contacted to help the user i.a. [5].

Many of today's assistive monitoring technology solutions used by elderly living at home are characterized by the user alarming or contacting the care sector themselves using bulky and noticeably sensors, care personnel attending the technical equipment themselves and difficulty in merging the equipment on a common platform [6]. The aim of the present work is to focus on an unobtrusive monitoring system to recognize typical crucial events of elderly still living independently at home, and technology equipment that is easy to handle by the individual themselves as well as the care personnel or relatives, which involves minimal operations. Approaches involving monitoring elderly users with special and complex needs or health related problems are not involved in this study.

The main goals of AT are described in [7] as:

- *"To increase self-dependency*
- *To allow community dwelling: demographic changes have led a large number of elderly people to live alone*
- *To increase the elderly user's participation* in ICT-based assistance
- *To provide insightful data to health professionals, caregivers, familiars, psychologist, system designers, and so forth"*

These goals are used as guidelines during the present work.

1.2 Goals of the Present Work

The main goal of the present work is to address and classify different scenarios in an unobtrusive monitoring system for elderly users, enabling them to live longer independently in their own home, while still feel safe and being provided necessary care from relevant actors. Both wearable and ambient sensors will be used to evaluate AI models for a proposed unobtrusive monitoring system to proper detect these critical events.

Based on the project description in Appendix A.1, the aim of the present work will:

1. Address critical events based on a background study during the current work and advances in the field of AT to be implemented into a proposed unobtrusive monitoring system of elderly living independently at home (ELIAH)
2. Identify type of wearable and ambient sensors to be used in the proposed unobtrusive monitoring system
3. Identify proper placement of the wearable and ambient sensors
4. Identify proper number of sensors used
5. Generate and evaluate AI models together with different combinations and placement of the sensors used
6. Establish a proposed system architecture for implementing these AI models using a smartphone and/or related solutions. The focus here is that the proposed unobtrusive monitoring system can easily merge with widely used technologies in the field, as well as involve good usability aimed for all relevant actors involved

1.3 Project Activities and Report Disposition

The project activities follow the report disposition based on the activities involved in the project description Appendix A.1. Figure 1-1 shows a workflow diagram describing the project activities, outcome of each activity which will be used in the following activities and which chapters these activities are addressed in throughout this report.

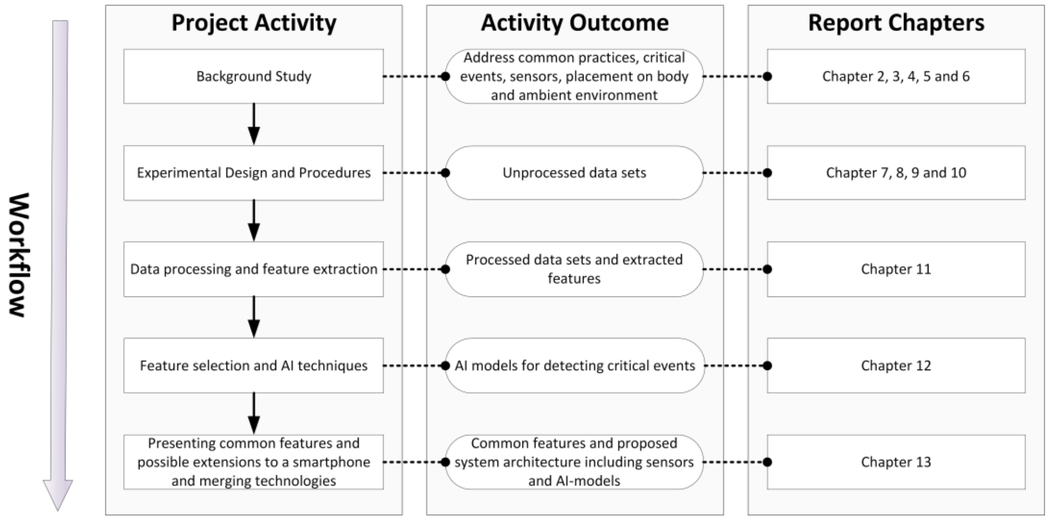


Figure 1-1. A workflow diagram showing the project activities and the outcome of each activity that are used in the following activity. The report chapters that outlines the specific activities are also pointed to in the diagram

The report chapters and project activities are further described.

Chapter 2. Literature Study on Existing Techniques in the Field of Unobtrusive Monitoring of Elderly as well as Related Fields: Outlines the findings in recent studies on the project topics to address important functionality the current technology has and/or and lacks. The findings give a foundation of what functionality the proposed system architecture to be established in this project can benefit from

Chapter 3. Sensors used in Assistance to Elderly and Supervision of Events Related to their Health and Safety: An overview of typical sensors used, their placement and function in assistive monitoring systems as well as studies involving these sensor architectures

Chapter 4. AT for Elderly in Japan and other Countries: A brief survey on the characteristics and approaches of AT in Japan and relevant Western European countries

Chapter 5. Welfare Technology in the Norwegian Care Service: Aspects of Norwegian Welfare Technologies, focus areas, how the technology solutions are managed and experiences from users and relevant actors in the Care Services

Chapter 6. Ethical Aspects of Sole Dependence on AT for Elderly Users: Addressing aspects of AT to be used by elderly to keep them living longer at home, while still feeling safe and being provided necessary help

Chapter 7. Measurands in Experimental Architecture for Unobtrusive Monitoring of Critical Events: Overview of measurands to be used in the experimental procedure of the proposed unobtrusive monitoring system for elderly established in the present work, their definitions, measurement principles and purpose in the system

Chapter 8. Data Collection Units and Data Acquisition: Description of modules used to create data collection units (DCUs) for sensor recordings of experiments in the present study. The description involves how the modules are connected and how experimental data sets will be recorded and stored by a PC during the experimental procedure

Chapter 9. Experimental Procedures for Data Collection: Description of the experiments performed and their purpose for AI model generation

Chapter 10. Characteristics of Experimental Signal Trains: Comparable plots of the experiment signal trains is to be studied for extraction of characteristics that will distinguish critical events from non-critical events

Chapter 11. Data Processing and Feature Extraction of Experimental Data Sets: How the experimental data sets were cleaned and prepared for feature extraction. Definitions, equations, and plots showing the characteristics of features extracted from the experiments that will help AI models classifying or detect critical events

Chapter 12. AI Models for Classification of Critical *Events***:** Description of AI algorithms, how the AI models or classifiers were established using software tools and how necessary features were selected to reduce number of features used while still give acceptable accuracy of the models

Chapter 13. Common Features in the AI Models used with Possible Extension to a Smartphone and a Smartwatch: Presenting distinct common features found when selecting features for the different approaches involving placement and number of sensors used in the AI models. A proposed system architecture of using these AI models or classifiers in a smartphone and/or relevant technologies are elaborated

Chapter 14. Results: Lists the accuracies of the different number and placement of the sensors used in the AI models for fall detection. Present simulation results of the geofence threshold model for detection if the user has not come home

Chapter 15. Discussion: Discussion of the findings of the results, common features and how the results reflects the proposed system architecture for extension to a smartphone and smartwatch application together with the findings from the background study chapters

Chapter 16. Conclusions and Future Work: This chapter summarizes the performed experiments and if the project aims are fulfilled. Possibilities for potential improvements and future work are discussed in the end of the chapter

2 Literature Study on Existing Techniques in the Field of Unobtrusive Monitoring of Elderly as well as Related Fields

Unobtrusive monitoring of elderly means that elderly is being looking out for without care personnel or other relevant actors being physically presence in their home or having relatives calling them only to ensure that everything is fine. Unobtrusive monitoring of elderly is expanding as AI techniques are being integrated into wireless devices such as smartphones, smartwatches, wearable and ambient sensor networks and smarthome systems. The purpose of an unobtrusive monitoring system is to only report or alert if a critical event has occurred, or for diagnostic and rehabilitation purposes, for example if the health status is changing over time. Such unobtrusive monitoring systems are beneficial for alerting relevant actors, while the elderly user is not stigmatized or feeling unnecessary observed in any way.

This chapter presents relevant advances in technology that enables unobtrusive monitoring of elderly, common AI techniques applied in such systems and related research as well as other relevant technologies.

2.1 Relevant Technologies that Enables Unobtrusive Monitoring of Elderly

Elderly people are nowadays more technical capable than preceding generations [8]. Facilitating independent living at home for elderly people by use of the expanding technological advances in welfare technology are important for relatives, elderly care sector as well as the government who has the intention of saving money, while still offering good care for those who need it. During the recent years, modern concepts such as wearable technologies like smartwatches and trending smarthome automation systems can be applied to unobtrusive monitoring systems of elderly living at home, enabling minimal user action as well as provide safety and proper user care by care personnel.

2.1.1 Architecture of Monitoring Systems for Elderly

A typical architecture of a remote health monitoring system for elderly can be shown in Figure 2-1 from the study by Patel et al. [4], including the sensor communication network consisting of wearable sensors, a home terminal that handles and stores data and generates alarms, as well as the actors involved, handling the alarms and reports. The major development in terms of wireless communications and smart technology is adaptable for integration of monitoring systems for elderly living at home. Health monitoring systems are merging towards using wearable sensor technology not only to be used by elderly, but also for rehabilitation purposes for people with various diagnosis, like spinal cord injury [9]. Such system relies on an excellent communication platforms to ensure safety for the user and to report to health personnel instantaneously if a critical event has occurred (e.g. falls or fire in living area). Wearable sensors should be comfortable to wear and have low energy consumption.

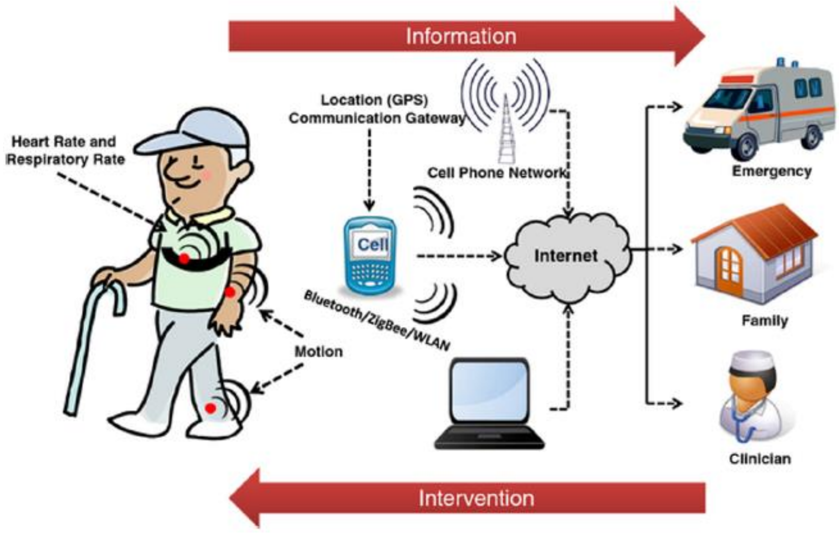


Figure 2-1. Typical architecture of remote health monitoring system showing the wearable sensors located on the user. Physiological and motion data are being constantly recorded using devices supporting wireless communication technology. The system contains a gateway computer with an application for data handling and storing and alert relevant actors if necessary [4]

2.1.2 MEMS Technology

The advances in MEMS (microelectromechanical systems) technology has enabled reduction of both size and costs of microelectronic products and modules such as sensors, data acquisition devices, and communication chips i.a. These factors has contributed to the advancement and usability of unobtrusive health and safety monitoring systems during the recent years [4], especially in terms of wearable sensors, which has to be comfortable to wear for long periods of time.

2.1.3 Wireless Communication Technology

An increasingly advancement in communication technology, especially wireless protocols such as Wi-Fi, ZigBee and bluetooth [4], enables monitoring systems to expand in terms of usability and range for remote measurements. These protocols provide the monitoring systems to work effectively, also outside the home environment, as the sensor data is transmitted and processed quickly, and relevant actors are alarmed instantaneous in case of a crucial event. Low energy consumption protocols such as Bluetooth Low Energy (BLE) and ZigBee are commonly used in various health and fitness applications nowadays, enabling the applications to be powered and record data for long periods of time, as various modes such as sleep mode etc. are implemented into their functionalities [10].

2.1.4 Smartphones and Smartwatches with immense Data Processing Power to be Applied Everywhere

Smartphones have become universal accessible for persons of all ages today, accommodating wireless mobile telecommunication such as 4G, as well as a lot of sensors and wireless technology, enabling it to be applied outside the home. A smartphone is used in the fall application developed by Lee et al. [11] were an Android application is created to detect fall of elderly living at home by the use of the sensors embedded in the device. Smartphones include integrated GPS systems or localization tracking using internet, enabling relevant actors to locate the user in case of an emergency outside the home [4]. Localization technology can be valuable for the elderly users that desires to be able to go outside but can be at risk at losing their orientation and possible forget the way back home, for instance people with dementia. If these users are equipped with a localization application that both enables them to go outside without supervisor and still feel safe, as well as the society saves both resources on extensive search operations, as the application increases the possibility of finding the person in case of emergency.

Applications that’s becoming commonplace lately is smartwatches, often intending to guide people to easily keep track of their health status [12]. These smartwatches contain an incredible number of sensors integrated in small chips realized by MEMS technology. The smartwatch can record movement and health related parameters continuously, providing expert guidance and statistic based on various AI techniques integrated in user applications to be downloaded with the product. An relatively new invention (launched in 2017) by the Norwegian company ContinYou has launched the healthwatch, ‘Contact’, for patients suffered from stroke, have dementia or other illnesses [13]. This smartwatch, is equipped with a SIM-card (among other communication technologies) and can therefore have mobile coverage in 95 % of Norway. The smartwatch can detect several critical events related to the user as well as clinical-diagnostic analysis of pulse, blood oxygen level, i.a., which also is reassuring for relatives and caretakers [13]. The watch is connected to a home central gateway for alerting relevant actors in case of an emergency.

2.2 AI Techniques used in Detection of Critical Events and Health-Related Issues

According to dictionary definition, intelligence means the ability to comprehend, reason and learn [14]. Intelligent systems are characterized by the fact that they can produce control actions in a flexible, adaptive, and robust manner, without much prior knowledge of the environment. The methodology used are computational intelligence method using mathematical algorithms which find hidden patterns in the input data to classify, predict and make decisions.

In intelligent unobtrusive monitoring system for ELIAH, AI techniques can be used for detection of critical events like falls or fire, monitor activity or movement pattern like sitting and lying down, diagnostic and rehabilitation purposes i.a. Remote wearable and ambient sensor systems for movement and activity monitoring will record a massive amount of raw input data which has to be managed and processed to derive relevant information from it [4]. Some of the techniques applied on these data sets after being recorded are signal processing, normalization, feature extraction and AI techniques to obtain a usable AI model. Features are characteristics that describes the raw inputs. A general flow for creating AI models in use for determining outputs are shown in Figure 2-2.

Figure 2-2. Typical flow of activities when creating AI models in use for determining outputs. White boxes indicate parameters and grey boxes indicate algorithms or models for processing and determining parameters

2.2.1 Threshold-based Classification

Threshold as an AI technique can be applied for various purposes in determining a sought output. For example, in a system that constantly measures pulse, a threshold-based classifier could generate alarms if the pulse is under a certain threshold, or an alarm can be stored in electronical medical records (EMR) if it the pulse has been over or under a threshold limit for a given period, which can indicate some health-related issues. A threshold classifier have been developed by Boyle et al. [15]. The algorithm was used to classify type of activity performed, like lying, standing/sitting or walking/Activities of Daily Living (ALDs), based on torso angle measured by an accelerometer located around the torso as shown in Figure 2-3.

$$\theta = \frac{180}{\pi} \arctan\left(\frac{\bar{a}_v}{\sqrt{1-\bar{a}_v^{\,2}}} \right)$$

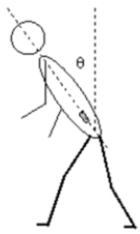


Figure 2-3. The formula used in [15] to calculate torso angle θ based on acceleration measurement a from an accelerometer placed on the torso to classify the activity type performed

2.2.2 Artificial Neural Network (ANN)

Artificial neural networks (ANNs) consists of nodes of mathematical functions, modelled in the way the human brain and neurons works. Input data, or features is fed into the network in form of inputs, and the ANN determines outputs based on the relationship between these. ANN are commonly used for activity classification [16].

Using an ANN, Zhang et al. [17] developed a portable device including a measurement system shown in the schematic overview in Figure 2-4. The system identifies type of intensity of a locomotion by placing a composition of multiple pressure sensors between insoles of shoes. The portable device manages to record and measure foot-ground contact information in every step. Studying various activities, the ANN classifier embedded in the device correctly identified the type of activity with an accuracy > 97 %, as well as predict the speed of walking and running.

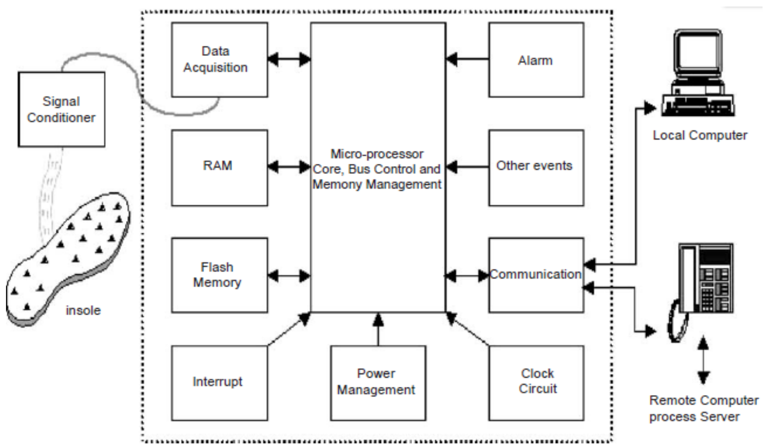


Figure 2-4. Overview of the system classifying the intensity of human locomotion as well as speed of walking and running using pressure sensor in the insoles of shoes in the study by Zhang et al. [17]

The ANN structure established for the study by Zhang et al. [17] involves two feedforward back-propagation ANN consisting of two hidden layers and one output layer shown in Figure 2-5. The number of nodes in the two hidden layers of the study by Zhang et al. shown in Figure 2-5 were decided optimally by the minimization of mean square output of a training set. Figure 2-5 also illustrates that both weights calculated and outputs from the preceding node is the input to the next node handled by an activating function.

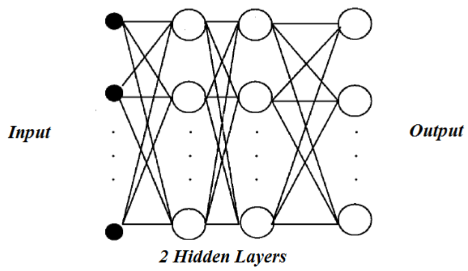


Figure 2-5. The figure is taken from the study by Zhang et al. [17], showing the input, the two hidden, and the output layer of the ANN used in classifying type of intensity by locomotion

2.2.3 Support Vector Machine (SVM)

Given a set of input data, support vector machines (SVM) can classify them into several categories based on an optimal separating decision hyperplane, found by training the classifier algorithm. SVM is moderately used in activity classification studies [16]. SVM can project the data points in the original space featured in, to another higher dimension by applying kernel methods which converts the inputs in the input space to a feature space, and from there find hidden patterns of the input data points [16].

García-Massó et al. [9] conducted a study to identify physical activity type of spinal cord injury patients equipped with four body-worn 3-axial accelerometer. SVM was one of the classifier algorithms evaluated in the study, among others. SVM produced a classification accuracy of 94 % when using two wrist-, one chest- and one waist accelerometer as shown in Figure 2-6.

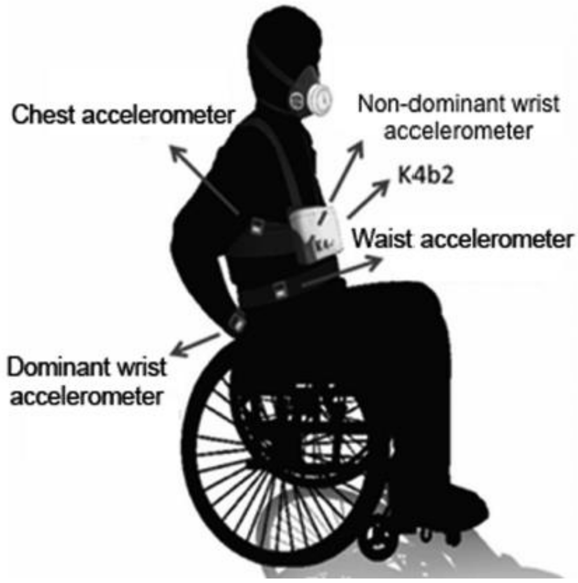


Figure 2-6. Placements of the 3-axial accelerometers on the spinal cord injured patients in the study by García-Massó et al. [9]. A SVM classifier produced the best accuracy for identifying physical activity type when applying all data sets from all the four accelerometers combined

2.2.4 k-Nearest Neighbour (kNN)

The kNN (k-Nearest Neighbour) algorithm are applied in several classification studies related to activity detection [4]. The algorithm uses a multi-dimensional feature space, where each dimension represents a different feature [18]. All the training data points that has been collected from observation or experiments are plotted in the feature space. The aim is to identify the k-nearest points (or neighbours) of the training data, which contain unknown window distances in-between the observations. The observations are classified by the majority of the kNN's. The value of k typically varies from 1 to a small percentage of the training data and is determined preferably using cross-validation procedures or trial and error [16]. Figure 2-7 shows data points plotted in a feature space, which is inputs of the kNN classifier.

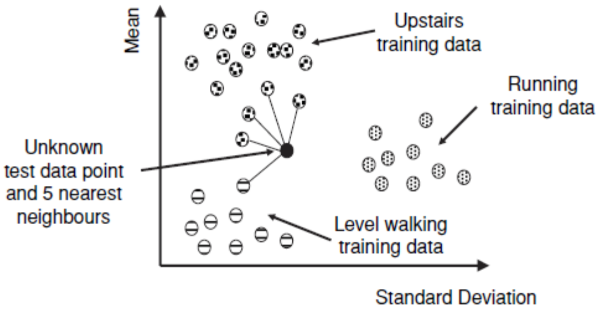


Figure 2-7. Showing a 2D feature space where different types of activities have been plotted along the feature axes Mean and Standard Deviation [18].

Finding out that the uppertrunck of the body, below the neck and above the waist, is the most suitable region on the body to place sensors for fall detection, Jian et.al developed a wearable system for detecting falls and ALDs using the kNN algorithm [19]. The activities were monitored by a 6 degrees of freedom (DOF) accelerometer and gyroscope combined sensor board located at the neck of a wearable west the subject carried.

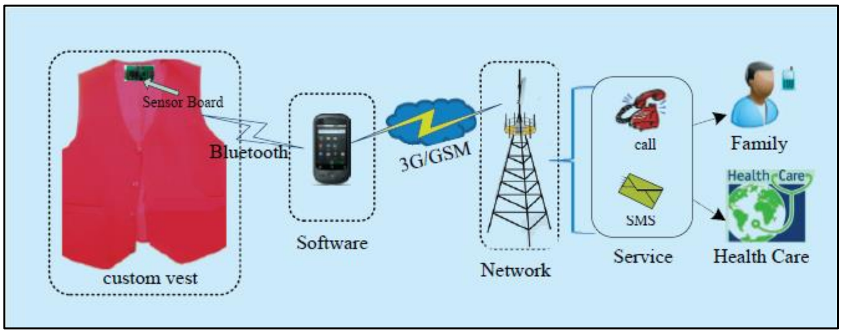


Figure 2-8. The architecture of the system developed by Jian et.al. [19]. The sensor signals are processed by the software in a smartphone and alerts relevant actors like family or health care sector, if necessary

With this approach, the kNN algorithm demonstrated an accuracy of 92,5-100 % when classifying the falls and ADLs, were the fall activities gave the lowest accuracies and walking-turning-walking activity gave the highest accuracy of 100 %.

2.3 Summary

Table 2-1 summarizes some of the technologies and techniques elaborated in this chapter that enables unobtrusive monitoring of critical events for elderly.

Table 2-1. Summary of relevant technologies and tehniques that enables unobtrusive monitoring of elderly as well as some examples and benefits of these

Technology and Techniques	Example	Details
Communication Technology	Wi-Fi, ZigBee, bluetooth, bluetooth low energy (BLE), mobile communication (2G, 3G, 4G)	Good range, wireless, low power. Enables more features to be included in AI techniques because faster data transfer and processing
Devices	Smartphones, smartwatches, tablets	Works easily together with many types of merging technologies, easily to bring along, use outside the home, small sized, fashionable, wireless, immense data power. Enables relevant actors to be alerted using application software
AI Techniques	Threshold, ANN, SVM, kNN	Widely accepted classifiers in activity recognition

3 Sensors used in Assistance to Elderly and Supervision of Events Related to their Health and Safety

Sensors are used for supervision of health and safety related events in assistance technology for ELIAH. Both wearable and ambient sensors are commonly used to detect abnormal trends of health parameters and critical events, like change of blood sugar levels over time and night wandering [20]. The type of sensors used to monitor events could vary depending on the individual situation of the elderly user. Some users may not carry wearable sensors because they tend to take them off and generate unnecessary alarms for relevant actors. Other users could need other combinations of sensors, like Smart Home automation and sensors used to detect fire and so on.

This chapter lists some of the common sensors used in assistance to elderly to enable them to live longer at home and supervision of events related to their health and safety, typical placement of the sensor as well as the purpose of monitoring of these parameters.

3.1 Wearable Sensors

Wearable sensors are sensors that are attached to the human body which are small sized, often low-cost, enabling them to be imbedded into belts, clothes, wristwatches, shoes and mobile devices [20]. Wearable sensors used in assistance technology are categorized into inertial sensors, which records measurement of body movement, and vital sign sensors (or biosensors), which records measurement of the users health condition [20]. This section summarizes commonly used inertial and vital signs sensors for monitoring events related to elderly's safety and health, respectively.

3.1.1 Wearable Sensors for Supervision of Events related to Safety

Gyroscope and accelerometers are the most common wearable sensors for recording a human motion, also referred to as inertial sensors [4]. A gyroscope measures rotation, while acceleration measures acceleration. GPS is used for localization, and are embedded into wearable devices, often safety alarms for elderly with cognitive impairment [21]. Table 3-1 lists common wearable sensors for monitoring of human motion.

Table 3-1. Table listing common wearable sensors for motion and activity detection of events related to safety. The table lists selected sensors with their functionalities. The table includes type of sensor, common placement on the body and the purpose of using this sensor in monitoring systems for elderly

Sensor Type	Common Placement	Purpose
Inertial (gyroscope and accelerometer)	Wrist [9, 13], foot [17], waist/chest [9], neck [19], hip [22]	Motion, fall and activity detection
GPS	Wrist [3], neck [21], could also be used in mobile device placed in pocket	Localization of users with cognitive impairment if necessary, or for people that are at risk of suffering critical health related events like a heart attack

3.1.2 Wearable Sensors for Supervision of Vital Signs (Biosensors)

Wearable sensors for supervision of vital signs like skin temperature, blood oxygen level, pulse are advantages to be used by elderly patients everyday as they support continuous non-invasive health monitoring outside the physician's office, and provide much more data to be analysed by health-personnel, as the health watch 'Contact' from ContinYou [13]. A disadvantage of vital signs sensors are uncomeatable feeling due to the long-time skin attachment, and reliability issues [20].
Table 3-2 lists common wearable sensors for monitoring of vital signs.

Table 3-2. Listing selected wearable sensors for detecting events related to vital signs. The table lists type of sensor, common placement on the body and the purpose of using this sensor in monitoring systems for elderly users

Sensor Type	Common Placement	Purpose

Skin temperature	Wrist [13], arm [23]	Detects activities (sleep vs. activity) abnormal skin temperature. Could indicate heart attack
ECG (Pulse)	Wrist [13], chest/waist, arm [24]	Detect abnormal pulse, pulse state during activities. Could indicate heart attack
Heart and Blood Pressure	Wrist [13], arm [24], fingertip [25]	Detect abnormal pressure levels, distinguish changes over time, detect heart rate or heart failure

3.2 Ambient Sensors

Various sensors that are placed in different areas of the living area. These sensors are deployed to reflect the user-object interactions [20]. Some disadvantages of ambient sensors are that they could easily measure other people, like care personnel or relatives, visiting the living area of the user and generate unnecessary alarms.

3.2.1 Ambient Sensors for Supervision of Critical Events related to Health and Safety

Table 3-3 lists some common ambient sensors located around the living area of the user to notify care personnel or relatives about critical events like night wandering, possible fire and type of activity based on which room motion is detected in.

Table 3-3. Table listing common ambient sensors used for detecting critical events related to health and safety of the elderly user. The table lists type of sensor and placement in the ambient environment and purpose of the sensor in monitoring systems for elderly

Sensor Type	**Common Placement**	**Purpose**
Motion Sensor	Ceiling [26], wall [27]	Detect activity type, detect present of individual in current room, detect night wandering
Door Sensor	Door [27]	Detect night wandering, for patients with cognitive impairment
Bed Sensor	Bed [27]	Detect night wandering, patient has not come back to bed after a pre-set time
Fire Alarm/Oven watch	In ceiling or near oven [27]	Detect possible fire. Detects if oven turned on
Temperature Sensor	In room (typical on wall) [27]	Detect abnormal room temperature. Indication of window open, or for controlling heating if individual is present in room
Water Sensor	Near water sources to detects water, like on the floor near water tap [27]	Detects water leaking

4 AT for Elderly in Japan and other Countries

Many developed countries experience a decreasing birth-rate and at the same time is the increasing number of elderly citizen a ticking demographic bomb [28] that forces them to innovate and automate parts of the elderly care. Various national and international research and cooperation projects that originates from governmental initiatives takes place in these countries now to change the elderly care for tackling this demographic bomb [8].

Japan is a country that merges their technological advancement within robotics and automation into the healthcare sector [28] as less ‘warm-hands’ will be available to take care of the expanding number of elderly. Many Western European countries focuses on tele-care and unobtrusive monitoring technology into their well-established elderly care sector [8].

This chapter will address the characteristics of the assistance technology in Japan and other countries where AT is merging into the elderly-care.

4.1 Japan takes Bold Steps in Automating the Elderly Care

Japan faced the challenges of the rapidly aging populating early on [29]. Roughly 25 % of Japan’s population are over the age of 65, and will increase to 40 % by year 2060 according to the statistics shown in Figure 4-1 [29]. A statistic of the population growth of people over 65-plus in Japan contra the other countries in the world are shown in Figure 4-1.

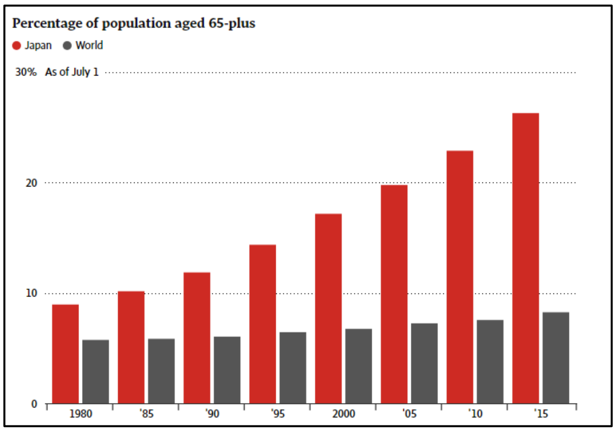


Figure 4-1. Statistics of the percentage of population aged 65-plus in Japan (red) and other countries (gray) from year 1980 to 2015 [29]

Japan has a strong technological history of implementing high-tech robots and automation into almost all parts of their industrial sectors, like the car-industry, where almost all parts of the manufacturing process are done by robots, and technicians only verify that everything works properly [28]. Robotics and automated solutions are characteristics of Japan’s highly technological elderly care sector. Professor Hiro Hirakuwa predicts that Japan will lack 400.000 care personnel in a few years, and that the elderly care sector will collapse if robots are not applied [28].

4.1.1 Robotic Animals for Therapeutic Stimuli

Sony was early on when creating the robotic dog Aibo for the first time in 1998. Aibo is a talking robotic dog that makes conversation to the elderly. Aibo can remind the user of taking their medicine, talk about the everyday things and fetch bones [28]. Paro is another Japanese robotic invention especially developed for dementia patients that contribute to stimulate interaction between patients and caregivers, reducing stress levels and improves socialisation. Paro are used in care-homes in several countries and certified as the world's most therapeutic robot by Guinness World Record [28, 30]

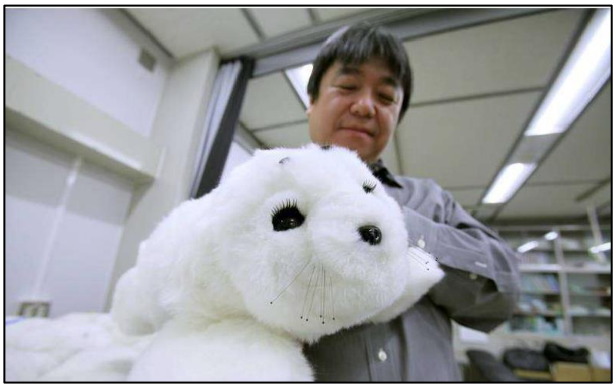


Figure 4-2. Paro, the Robotic Seal is a Japanese invention specially developed to stimulate dementia patents and are certified as the world's most therapeutic robot by Guinness World Record [29, 30]

4.1.2 Health-Care Robots

As there will become a lack in care personnel to take care of the elderly, Japan has for a long time been pioneers in developing high-tech health robots, able to perform many of the care personnel's tasks [29]. The robot Pepper, are used to increase the social atmosphere in care-homes, and hold physical classes for elderly, which have had a great impact on both the elderly's physical and mental health [28].

Japan has also developed robots that can help patients out of bed and deliver food plates i.a. features. These robots, like 'The Twendy-One' robot in Figure 4-3, are rarely used in the care sector as they are extremely costly. A philosophical divide regarding the use of robots to take care of people are experienced between Japan and the western countries. Japanese elderly is not afraid of being taken care of by a robot as there have been a cultural acceptance of this technology through the years. Common reflections regarding being taken care of by robots are presumed with fear and deep suspicion among western societies [29].

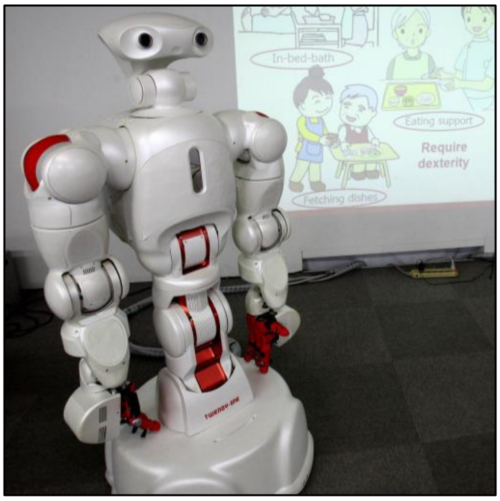


Figure 4-3. 'The Twendy-One' robot is a Japanese invention that can help patients out of bed and deliver food plates i.a. features [29]

4.2 Western European Approaches of Implementing AT for the expanding Elderly Population

Many of the western developed countries have well-established elderly care sector. Some of the countries that stand out are the UK, especially Scotland, Netherlands, and the Nordic Countries. Unlike Japan that focuses on robotic and automation technology in the elderly care, the European approach is to use AT solutions like telecare and other unobtrusive monitoring techniques to support elderly and help them to live longer independently at home [8]

4.2.1 Scotland Invests strongly in Telehealth

Scotland has since 2006 been implementing telehealth into the elderly care. Scotland has initiated several pilot projects to implement technology packets consisting of alarms and other supporting technologies to secure the daily life of elderly over 60 years old. The aim of the project was to prepare the elderly on having technology in their home before the need emerged. Scotland is committed in restructuring their elderly care using AT. Efficiency and enhanced quality, with good indication measurement and reporting are key factors for the basis of common guidelines on the use of AT in Scotland [8]

4.2.2 The Nordic Model

Denmark has made huge initiatives to encourage the implementation of AT for elderly to live longer at home [8]. Denmark established the governmental foundation called '*Fonden for Velfærdsteknologi*' (The Foundation for Welfare Technology) already in 2006 to support public organization and the country's communes financially for the implementation of welfare technology [8]. Many of the welfare technologies utilized by elderly still living at home in Denmark are ICT (information communication technology) based, like alarms, localization technology and smart home solutions [31].

The Nordic countries are quite aware of the benefits of implementing welfare technology in many of their public sectors [8]. The countries also utilize many of the same technologies in the elderly care sector, often unobtrusive monitoring solutions including mobile platforms, various sensors installed in the home and so on. The differences of using welfare technologies in the Nordic Countries often lies in which degree or level they are on in implementing the technology into their public sectors, or what level of the governmental administration that administer the deployment of this technology, municipality or county [32].

The Nordic Welfare Centre has initiated a joint project for organizations in The Nordic Countries to participate in [32]. The project aim is for The Nordic Countries to collect knowledge and experience to discuss best practices on problem existing in the countries, create toolkits and frameworks for the utilization of welfare technology for optimizing the chance of successful implementation [32].

4.3 Summary

This chapter presented an overview of distinct differences of AT with focus on Japan compared to western European countries like Scotland and the Nordic Countries. AT in Japan are characterized by robots or robotic animals replacing the functionality of care personnel. In western Europe, the AT developed in the elderly care sector focus on ECT technology for alarming personnel or relevant actors when necessary. The focus in western Europe is to enable elderly to live longer independently at home.

The cost of a robot that replaces a carer compared to ECT technology that contacts a carer if necessary are quite contrasting. Sensor technology are cheap and easy to maintain compared to a robot.

5 Welfare Technology in the Norwegian Care Service

Welfare technology are, as indicated by the name, technology developed to promote peoples welfare [33]. The aim of welfare technology is to help those who need it, freeing resources and being an area of business development as well as growth [34].

Pilot projects are initiated in the elderly care sector in several municipality throughout Norway to identify the benefits and challenges regarding the implementation and use of welfare technology for ELIAH [5]. The Norwegian Directorate of Health publishes yearly reports during this implementation period based on independent research groups chosen by the municipalities to provide an overview of the current situation and to recommend measures regarding the use of the technology. This process is initiated for the establishment of proper regulations and guidelines for these systems to be implemented eventually in all elderly care sectors in Norway [5].

This chapter outlines the findings during a survey regarding the current situation of elderly in Norway, which factors forces elderly into care homes, and relevant welfare technologies that are commonly used. Feedback from the elderly users and care staff involved in the pilot projects of implementing welfare technologies in their municipality are also included in this survey. The topics related to Norwegian welfare technology considered in this chapter focused on welfare technology in Norway with respect to the project description of the present work, such as sensor technology, unobtrusive monitoring, alarming relevant actors and the current status of how this technology is deployed today.

5.1 Situation and Evolvement of the Elderly Care Sector in Norway

The Norwegian care sector are about 50 years old, as before that, elderly was often taken care of by other family members [35]. Nowadays among a quarter of elderly over 66 years old receive some sort of assistance in their home from the elderly care service sector as shown in Figure 5-1. The use of elderly care services increases with age [35].

It is of interest that elderly continue living independently at home even when receiving care services as this could possibly promote better mental health [36]. This is one of the main reasons for the implementation of welfare technology in the elderly care services, enabling elderly to live longer at home while still feel safe and being provided proper care services before their health condition oblige them to move into care homes or institutions [37].

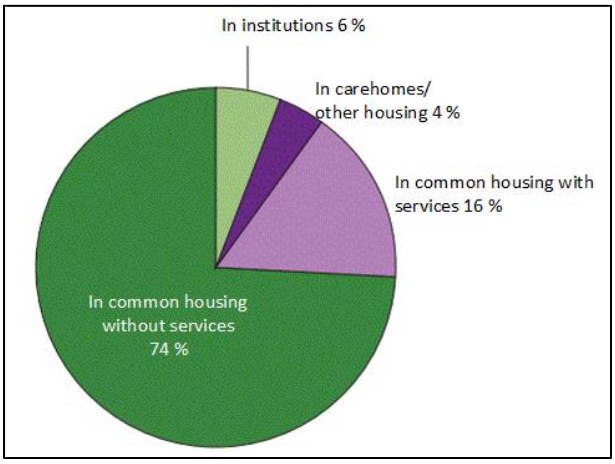


Figure 5-1. The pie chart showing the living situation of elderly in Norway in 2011. A quarter of elderly over 66 years old receives some sort of assistance from the elderly care services [35]

The extensive growth of elderly in other countries in the world deliberated in chapter 4 also includes the elderly population in Norway and will in the future years offer several challenges for the elderly care sector affecting available care home spots, care personnel, and the organization throughout. Figure 5-2 shows how the elderly population in Norway over 67 years old will increase in the following years. On the other hand, the future elderly generations will experience improved abilities in terms of having better health and physics, economy, living situation and higher education. An important aspect is that future elderly generations are also more familiar with use of technology, which will help shape the future elderly health care sector [37].

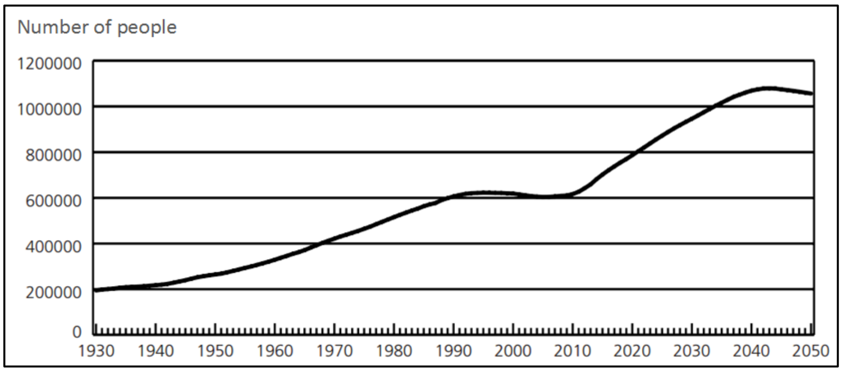


Figure 5-2. Statistics showing how the elderly population in Norway over the age of 66 years old will increase in the following years. In 2040 arises a slight decline in this growth [38]. This statistic describes the need for implementing welfare technology in the elderly care as there will be a shortage of care home spots, personnel, and economics

5.1.1 Disabilities among Elderly in Norway

Figure 5-3 shows common disabilities among elderly and how using welfare technology to ensure elderlies safety and health can reduce the risk of other disabilities occurring [33].

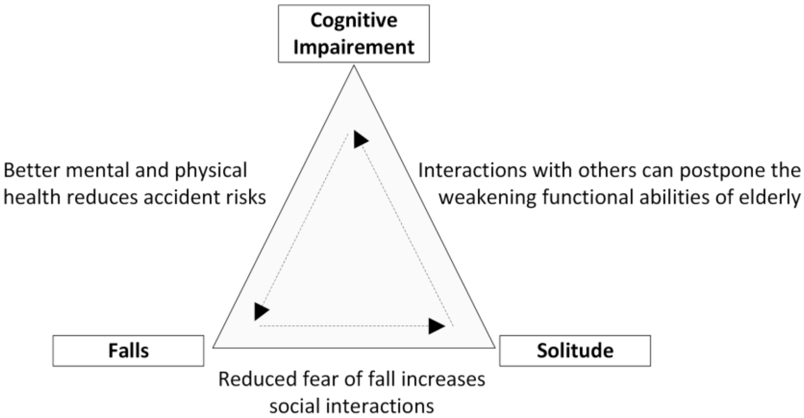


Figure 5-3. Graphical overview illustrated by a triangle showing the common disabilities elderly is affected by and how welfare technology for one of them can reduce the risk of another disability occurring. The figure is translated from [33] based on [39]

Impaired Physical Abilities

Impaired physics, hearing and visibility are common consequences of aging. About 34 % of elderly over 66 years experiences some sort of health related disabilities [38]. One third of elderly between 67 and 78 years needs healthcare services and around 14% of elderly of 80 years or older needs care in terms of hygiene care and housework daily.

Almost half of all hip-thigh injuries treated by specialist healthcare service between year 2009-2011 involved people of 80 years or older. Fall events are also the main reason elderly 80 years or older are treated by specialist healthcare services in Norway [40]. The common type of injuries among elderly 80 years or older are accidental injuries and account for 95,4% of all causes whereas 81% are fall-related. and the common scene of the injury are in their home or their residential area which account for 57,7 % of all causes [40]. For elderly people, fall related injuries can have many negative consequences and in worst cases even be a precipitating cause of death. Between 10 and 20 % of the falls occurring in care homes leads to serious injuries. In addition to fractures and head injuries, permanent disability, fear of falling again and loss of independence are other consequences related to fall-events of elderly as indicated in Figure 5-3.

Cognitive impairment

Cognitive impairment are common symptoms experienced by people of all ages but especially among elderly. Some common cognitive impairments include difficulty thinking, reflecting, remembering, planning, take judgement, take initiative, and take actions. Many ‘physical healthy’ elderly have these cognitive disabilities in a moderate form [41]. Cognitive impairments may affect people from a mild degree to a severe form of dementia diseases such as Alzheimer's. Dementia involves symptoms such as memory loss, orientation disabilities and difficult to perform everyday activities among others [42].

Statistics shows that about 84 % of the people that lives in care homes and about 40 % of people over 70 years that are being provided home care services in Norway have a dementia diagnosis. From this statistics, about 71 000 have dementia in Norway today [43]. About every fifth person will get dementia during their lifetime [44] and an increase in the number of elderly in Norway will further involve more people with dementia as the occurrence of the disease increases with ageing. The amount of people dying with dementia as a leading cause of death increased from 2,2 % in 2000 to 7,6 % in 2014 [43]. It also shows that people with dementia often needs extensive assistance from the care sector and the disease will gradually degenerate and eventually the concerned will become totally dependent on help on every aspect of their daily life [43].

5.1.2 The Norwegian National Welfare Technology Programme

The Norwegian government has initiated the Norwegian National Welfare Programme (NNWP) for development and implementation of welfare technology in the health- and care services. Improvement of the users’ ability to manage their daily life, increase the user’s safety and reduce relatives worrying about them are important targets of the project. An important measure with welfare technology is to enable better communications with users, relatives and the healthcare services through some of these technologies [37].

5.2 Technical Solutions used in Welfare Technology supporting Assistive Living for Elderly Persons and Recommendations

Sensor modules, safety alarms with localization technology and house central for communication are some of the common technical devices enabling elderly to live safe and independently at home while being supported by the care services. Management systems for taking care of alarms generated by the technical equipment are also established for data management and communication to relevant actors as family members and care services. In Norway, the type and combinations of welfare technologies for elderly are individual customized according to their disabilities and living situation [6]. There are various types of welfare technologies available which are divided into the following categories by the Norwegian Ministry of Health and Care Services [37]:

1. *"Safety- and security technology: Creates secure framework around the user. Safety alarm are the most widely used solution in this group*

2. *Compensation and wellness technology: Often used if the user has physical and cognitive impairment. Can involve smart home automation system including control of lightening and heating*

3. *Technology for social contact: Helps individual to be in contacts with other. Video communication technology is a type of solution in this group*

4. *Technology for treatment and care: User are provided technology for better handling of their health status. Technologies in this group involves automatic measurement of blood glucose levels and blood pressure as well as multi-dose medicine dispensers"*

These technical solutions contribute to safer and independent living, less resource intensive visits by care personnel. Altogether, the use of these welfare technologies in the care sector will ultimately save money and change the care sectors organizational structure, which will be beneficial as the elderly populations will grow extensively in the years to come.

5.2.1 Technical Devices Provided for Users by The Care Services in Norway

Safety Alarms

The current setup of common safety alarm systems includes a wristlet equipped with a button that communicates with a stationary house central [45]. When the user manually triggers the alarm, the house central will provide a two-way call to the healthcare services to decide if help is needed. The alarm are often received by the same person in the local healthcare service regardless of the users individual needs [45].

A new mobile safety alarm setups are developed and currently being tested out to establish user strategies for this device [45]. This device is carried around the neck and are connected to the mobile network (GSM) so the user can contact the care services outside the home and being located by GPS tracking if necessary. This system requires a comprehensive communication platform including instant communication with relevant actors that ensures the users safety and wellbeing [45].

Figure 5-4. To the left, a mobile safety alarm of type 'Safemate', carried around the neck which includes GPS tracking and mobile communication versus a more commonly stationary safety alarm, to the right, that cannot be applied outside the living area [21]

Fall Sensor with Alarm

The commonly used fall sensors for elderly in Norway today looks like wristwatches as shown in Figure 5-5, and can resemble the commonly used safety alarm shown in Figure 5-4 [6]. The fall sensor communicates with a house central when experiencing a sudden severe impact. The fall sensor with alarm is waterproof and can be used in the shower or bathtub which is a common scene for falls.

Figure 5-5. A fall sensor of type "Vital Base" that both automatic and manually triggers an alarm. It includes a sensor for movement and one for fall detection [46]

The Scandinavian research organization SINTEF has developed a fall sensor that can detect all types of fall, also "slower falls" which aren't picked up by todays commonly used fall sensors [47]. This sensor uses pressure sensors to detect these types of falls.

Door Sensor with Alarm

A door alarm as shown in Figure 5-6 is installed at the outside of a door, and informs the house central when triggered and generates an alarm when opened [6]. This alarm is commonly used for people with cognitive disabilities, that should not wander alone outside as they can lose their orientation etc. [48].

A problem that arises with the door sensor with alarm is that it does not distinguish between the user or people coming to visit. The door sensor can also be timed to only give an alarm during the night hours. Health personnel are often left to manage the sensors power when visiting, which can cause implication if not turned on again [6].

Figure 5-6. A door alarm commonly provided by the local care services which registers if a door is opened and can be turned on and off [6]

5.2.2 Technical Management of Monitoring Systems for Elderly

The technical management system can communicate with all the devices utilized by the elderly users. It processes the data received and communicates with relevant actors like care personnel and emergency central. The management system is a software solution that handles and filters alarms, patient journal etc. The architecture of the Technical Management systems can vary as mentioned in chapter 1.

Data Acquisition Platforms

Data acquisition platforms are commonly the link between the elderly users, the alarm central, relatives and other relevant actors. Telenor Objects has created a generic data acquisition platform called 'Shepherd', which manages the communication and data transmission between the users house central stationed in their home and the other actors involved as shown in Figure 5-7 [49]. The Shepherd data acquisition platform are created so that devices from all vendors can be integrated into the platform. This enables the municipality and local care services that desires to use this platform to keep applying the sensors and devices already in use by the elderly users.

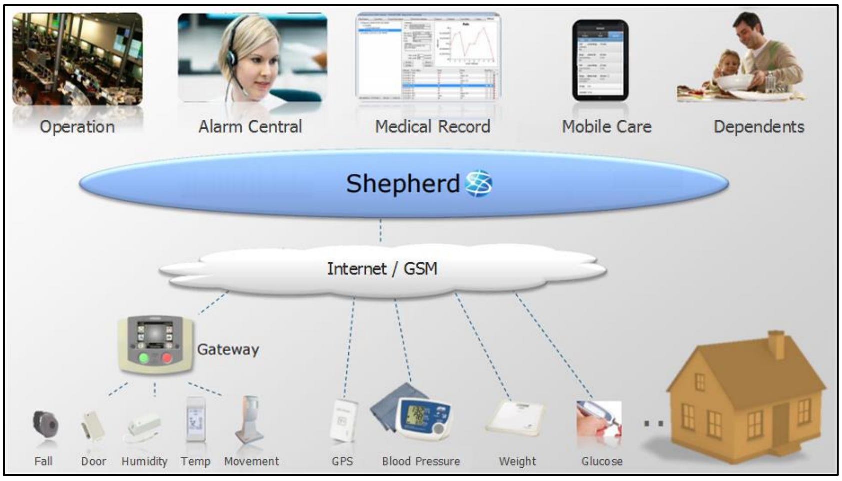


Figure 5-7. Overview of the actors (top), devices (bottom) and communication systems involved the Shepherd data acquisition platform 'Shepherd' by a Norwegian telecommunication vendor Telenor Objects. The figure is translated from [49]

House Central

A house central is installed in the home of the user, as shown in Figure 5-7. This house central is a gateway that receives the alarms generated by the various sensors and are connected to an overall communication service like internet and GSM network.

Mobile Platforms

Care personnel and relatives can access relevant information regarding the user on mobile platforms. Visma and Telenor Objects has created a software solution for care personnel to create reports, manage alarms and access of the patient medical [49]. This is a total welfare technology solution that is integrated into the municipality's existing EMR [49]. Nurses can access EMR for home visits, alarms are automatic documented in the patient journal and other information are accessible that are beneficial for general patient assessment when applying mobile platforms.

5.3 Response from Elderly and Care Staff utilizing Welfare Technologies for Homebased Care

Through the continuous research of the NNWTP, feedback from all parts involved in implementing welfare technologies in the Norwegian Elderly Care Services are collected and evaluated. The result of this evaluation is recommendation for changes and future involvement of the common regulations established in this field. The feedback includes both positive and negative experiences from the elderly users and care personnel working with these technical systems on a day-to-day basis. The vendors developing these technologies are also involved in this research as they aim to continue improving their products for them to be more applicable for use in this sector.

The aim of the research is to measure the level of achievement for the implementation of welfare technology in form of improving the elderly's life situation and releasing unnecessary workload for the care personnel. It is important that the technology contribute to fulfilling the intentions of welfare technology as listed in the introduction of this chapter. Individual situations and conditions are factors for the users to be satisfied. The most frequently mentioned qualitative benefits are increased safety and relief of inability for both service recipients (users), their relatives as well as care personnel [5].

5.3.1 Response from Elderly being provided Welfare Technology

An outcome of elderly using welfare technology in cooperation with care services is that they often can postpone or reduce the number of visits from care personnel. Some elderly can also postpone their given place on institution [5]. For some elderly has the visits from the care staff become customary, while others are relieved that the number of visits are being reduced by using welfare technology [5].

Safety alarm and localization technology have been tested out at several local elderly care services. Using this technology have contributed to increased activation of people with cognitive disabilities. Many users are less disruptive and experience achievement on other areas. Some of the safety alarms and localization devices include two-way communication. Many users have challenges with explaining the cause of the situation once they communicates with the alarm central [5].

5.3.2 Response from Care Personnel using Welfare Technology on Work

Responses from the elderly care services is that use of welfare technology releases burdens of the service, and the use of force on service recipients, like not permit them to go outside [5]. On the other hand, it is challenging to find technology which are reliable enough and has good usability. It can be time-consuming to log into the mobile solution platforms and be able to find the alarms generated by a user and take care of these alarms [6].

A lot of uncertainty concerning the alarms generated by the sensors. Unfortunate placement of the sensors generates false alarms, especially movement sensors. Other times, the care personnel were the cause of an alarm being generated if they forgot to turn them off after visiting the users. Poor and unstable coverage by the cellular network could in some locations contribute to uncertainty concerning if alarms actually *were* received. All this provided more stress during work hours [6].

5.4 Summary

This chapter has elaborated the status regarding common issues elderly in Norway today are faced with, and reasons the Norwegian government has initiated the implementation of AT solutions to still ensure elderly's welfare by the care sector now and in the future, despite the growing elderly population in Norway.

The common issues for elderly in Norway today are falls, cognitive impairment and solitude, one of these disabilities or issues can provoke another, which evidently could lead to the elderly being moved into care homes earlier on than necessary. The use of welfare technology will decrease the risk of these common factor to occur, and at the same time, enable ELIAH.

The devices commonly being provided by the care services are characterized that they are uncomeatable, unfashionable, unreliable and requires maintenance and service often by the care personnel themselves. The solutions generating alarms are unreliable, which provides unnecessary stress for the care personell.

The technology vendors in Norway cooperate with both the government and the care sector to improve their solutions to be more applicable for the care services.

6 Ethical Aspects of Sole Dependence on AT for Elderly Users

Looking at AT for elderly and its entrance into the established care sector, several moral questions arises regarding this, and who the technology specifically serves. AT combines the fundamental boundaries between two established "cultures", the technological and the humanistic perspective [34].

The challenges arise as the AT is developed for an untraditional target group for advanced technology – the elderly. ATs will break down the organizational structure of the elderly care services and are introduced on new arenas – the home.

It is challenging to decide whether or not the AT are morally good or bad [34]. It is important to consider individual measures for the elderly users of such technology. AT arises an important question: what are a good life for a human, and when technology should break this boundary in order to keep individuals safe from sickness and other harms [34].

This chapter will outline the ethical aspects of sole dependence on using AT for the elderly user group. The humanistic and the technological aspects are important to address together as AT changes the establishment of the elderly care sector as we know it, and third party actors as technological services. It is important that the main goals of welfare technology are preserved when introducing this type of technology into someone's home and comfort zone.

6.1 Changing Relations to Relevant Actors

Implementing AT into the homes of elderly users influences family roles and relations. It also changes the workload of care personnel and the insight they get to the alarms and data generated by the user.

6.1.1 Ability to Consent for People with Dementia

Dementia patients will have reduced ability to consent using suitable AT, especially alarm systems and technical surveillance, to ensure their safety and health [8]. People with dementia have challenges to comprehend the complex technical information about how the workability of the technical equipment, as well as to the need to use the information extracted to ensure their wellbeing at home [50].

A possible solution to the challenges of giving informed consent is to introduce a voluntary consent check for all over 65 years [8]. Care personnel are often afraid of utilizing technical solutions for these users. This is why it can be preferred that informed consent are already given before being diagnosed with dementia and what type of technology the person wished to use in their home if necessary [8].

6.1.2 Care Personnel and Relatives

Some issues are raised related to the paid care personnel using this technology to do their job more efficiently by taking care of alarms, health-related statistics etc. by the elderly users. AT that monitors activities of daily life in the home can impose unintended consequences for social relations between the paid care personnel, family carers and the cared-for-person [50].

6.1.3 Limitations of AT and Effectiveness from Actors Perspective

Reliability of the AT are part of the concern for all parts involved and the need for upkeep of the system and constant advances could feel as a burden for both care personnel and relatives. AT need maintenance, such as change of batteries and components, reset systems and turning devices on and off to avoid generating unnecessary alarms [50].

Too many alarms, statistics and prompts may lead to lack of attention when it really matters. Too much data to be reviewed may be distracting, leading the relevant actors to avoid checking on a regular basis [50]. Having an unobtrusive monitoring system that requires minimum user interaction in terms of handling user interface will be beneficial. It is also mentionable that many of the closest relatives that utilizes these systems or receives the alarms have health problems on their own [50].

6.2 Privacy and Surveillance

A crucial ethical aspect that arises when monitoring people and storing data from their movements is how this should be stored in an appropriate manner. Some monitoring technologies uses video and localization technologies, while other can monitor unobtrusively using only sensors. How to choose the monitoring technology that preserves privacy while still ensures safety for the user are one of the key consideration.

6.2.1 Type of Monitoring Technology

The use of localization technology is both practical and saves local authorities money and resources on search operations. Using localization technology to track users provided ATs are defined as *force* by the Patient Rights Act UK (section 4A) and requires special approval to be utilized [34]. It is a crucial measure to consider who has the most benefit of using localization technology; is it the user with dementia, or the society and care personnel? Since dementia patients often use it. it also arises the question of who decides whether this technology will be applied or if it will be applied by *force*.

Using sensors in assistive monitoring systems are less obtrusive than using a video camera. But allowing the use of video surveillance could enable emergency units to be more certain of an alarm generated if they have the possibility to quickly look at the event generating the alarm on a videoclip. Using surveillance technology raises challenges of defining what is “normal” and acceptable events to be monitored [34].

6.2.2 Personal Data Protection

Using AT for health and safety related issues will eventually establish electronic registrations of the users and their private life. Large amounts of data are transmitted and stored, more actors are involved, which will have access to these sensitive information’s. An important measure is that the proportion of the data that is registered must be considered based on what is to be achieved by the technology itself [8]. It is recommended that the local care services themselves are responsible for procurement of the ATs and establish proper regulations and routines for impact assessment policies. This can also be beneficial for the technology vendors to ensure their products are fulfilling all the requirements [8].

6.2.3 Third Party Actors

All the various types of AT that are available also introduces a range of third party individuals and organizations who are involved in the services, installations and management of these types technological devices and platforms. This could be service workers, internet providers, heating suppliers and mobile phone companies to mention a few. This raises a crucial problem for the data security, when third party actors have access to these data [50]. On the other hand, health logging application will encounter tragic consequences if internet service is lost while monitoring critical health conditions of a patient.

6.3 Introducing New Technology in the Home and Established Environments

Some of the ATs, such as smartwatches and tablets attract positive associations with modernity and youthfulness, whilst others attract negative associations with ageing and dependency of others.

6.3.1 Changing the Sense of Home

A challenge of introducing AT in the home is that it is a new arena for this type of technology. These technologies and patient information's where previously only accessible in care institutions or health related arenas like the hospital or physician's office [34]. This intervention can change the relation the user have to the home environment and contribute to reduced privacy and violation of the users integrity [34].

If the monitoring technologies and advanced health technologies merges from established arenas like care homes and hospitals and into the home environment, common dilemmas that takes place in the established arenas will now also take place in the home. Care and hospital services in the home can result in blurred or unfortunate responsibilities of the various actors involved [34].

6.3.2 Aspects of Autonomy for Elderly using AT at Home

From the elderly user's perspective will having AT in the home with homebased care instead of moving into an institution contribute to increased autonomy. Patient autonomy is defined as a patient's right to decide for themselves and their body. It is common for people to fear for their own independence in later life [50]. People of western society has been more dependable of technology through their lifetime. This could have a positive effect when elderly is being introduced to AT in later life and may contribute for them to be more acceptable for this dependence.

The effect of the establishment of future requirements and regulation of AT has not yet been shown. AT applied in a large scale by the care sector can increase influence of health insurance companies by the legislations of the government. This could ultimately influence whether or not an elderly or their caretakers have the right to refuse such technologies at all [50].

6.4 Summary

This chapter has addressed ethical aspects of sole dependence of ATs for elderly users. The purpose of AT is good, but many aspects of the technology can benefit other actors more than the elderly user it is aimed for.

With relevant actors having more insight of the users' movement will change the relations between them. As ATs often records a huge amount of data, the questions arise of how these data will be stored and who is to manage these data. Video technology would in many cases be beneficial for relevant actors to check what generated a specific alarm, but it is fundamental to distinguish which events are important enough to be recorded by video.

The effect of introducing AT are no yet disclosed. It is fundamental that ethical aspects are highly considered when creating guidelines for AT. When this technology is applied in a large scale, the health insurance companies can have the possibility to influence whether users and caretakes can have the right to refuse such technologies at all.

7 Measurands in Experimental Architecture for Unobtrusive Monitoring of Critical Events

A measurand is a physical variable obtained in any experiment and are sensed by a sensing element, in this case, several sensors, of an experimental measurement system [51] as illustrated in the graphical overview in Figure 7-1.

Based on the knowledge obtained in the background study elaborated in chapters 2-7 of this report, an experimental sensing system to record data sets for generating AI models is to be constructed. This proposed system will to measure physical measurands for unobtrusive monitoring of critical events detected by the AI models. This chapter describes the definition of the measurands chosen to be recorded by the experimental measurement system to gather the data sets for generation of AI models. The AI models will ultimately be evaluated for possibly implementation into software of a smartphone.

Considering the triangle overview in Figure 5-3 illustrating the relation between falls, cognitive impairment, and solitude for elderly, and how one disability can provoke another, which evidently could lead to the elderly being moved into care homes earlier on than necessary forms the foundation of the selected measurand to be recorded handled in the present work. AI models will detect falls and detect if a user has not come home after a pre-set time. Figure 7-1 is a graphical overview showing the measurand source from the environment, measurand type, type of sensing unit that inherits the sensing element to measure the specific measurands with symbols, and the critical events to be detected by AI models including these measurands.

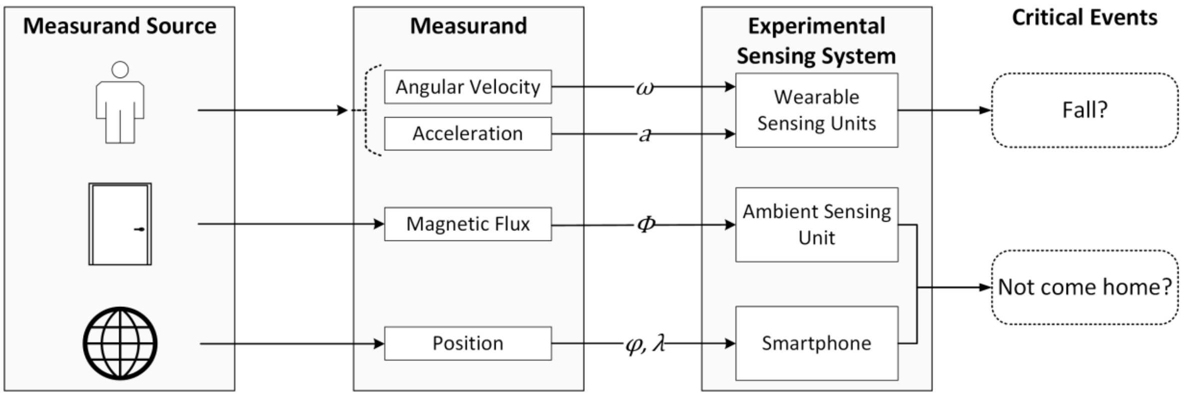


Figure 7-1. Graphical overview of the measurands to be recorded in the experimental sensing system and the critical events to ultimately be detected using sensor fusion. Showing the measurand source, measurands type, type of sensing unit that inherits the sensing element to measure the specific measurands including symbol, and the critical events to be detected by AI models to be generated

Falls could be detected by sensing ω and a of a users' movement. Using a door sensor that detects Φ when a door is closed, together with measured posion (ϕ, λ), could produce more valid alarms, which would make the relevant actors more certain that the door alarm was generated by the user and ultimately provide more freedom to users with lighter cognitive impairment, and will only be localized if necessary after a pre-set time. An AI model can generate alarms when detecting that the user has been outside a coordinate boundary or '*geofence*' of the home environment for over a pre-set time.

7.1 Measurands Recorded by the Wearable Sensing System

Falls and other movements produce movement patterns that can be described by angular velocity, ω, and acceleration, a, measurands. This section describes basic concept of the measurands and measuring principle applied by the sensors included on the wearable sensor units placed around the chest and left wrist.

7.1.1 Angular Velocity

Angular velocity (ω) measures the rotation in $[°/s]$ around the x, y and z axes of an object as shown in Figure 7-2. A more exact formulation of ω is found in [52]:

“a body subjected to a rotation $d\vec{\theta}$ about a fixed point will have an angular velocity $\vec{\omega}$ defined by the time derivative $d\vec{\theta}/dt$, in a direction collinear with $d\vec{\theta}$”, where θ is angle in degrees [°].

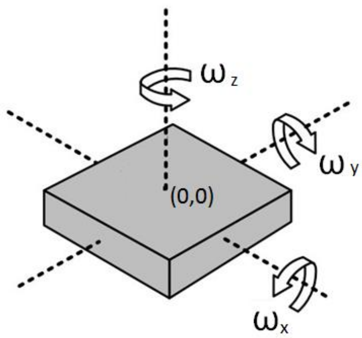


Figure 7-2. Showing the angular velocities $\omega_x, \omega_y,$ and ω_z around the x, y and z axes of a body. Origo is indicated as (0,0) and the rotation can occur in both direction, $\pm\omega$. Figure based on [53]

In the present work, a MEMS gyroscope will be used to measure the angular velocity of the chest and left wrist area of a subject performing experiments. The two gyroscopes will then measure a total of six angular velocities ($\omega_{x,y,z}^{chest,wrist}$) when employed.

A MEMS gyroscope uses the *coriolis effect*, which is the force experienced by the resonating mass on a rotating body [54] to determine ω. The resulting displacement of the mass is then read using a capacitance sensing element, which converts capacitance, $C\ [fF]$, into electrical signals that can be read by a microcontroller [53]. The principle of the resonating mass within the gyroscope is illustrated in Figure 7-3.

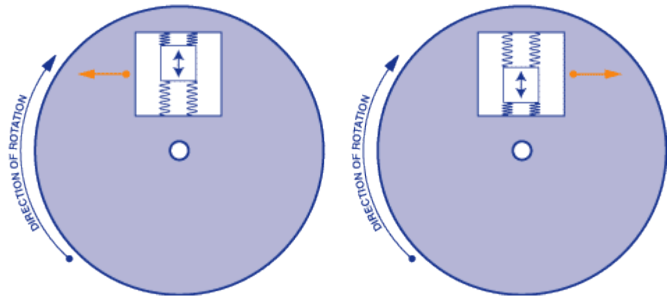


Figure 7-3. Internal perspective of a gyroscope sensor showing a mass on a spring system (white box) that creates motion when the body rotates. This motion produces a voltage output based on the sensor relative to the rotation. Figure from [53]

7.1.2 Acceleration

Acceleration (a) is defined as the rate of change of the velocity $d\vec{v}$ of an object and is commonly measured in $[m/s^2]$ or g-force $[g]$, as used in the present work [55]. Figure 7-4 illustrates a of the x, y and z axes of a body.

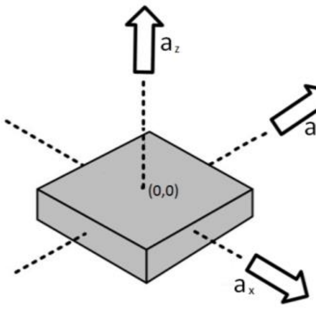


Figure 7-4. Showing the accelerations a_x, a_y, and a_z of the x, y, and z axes of a body. Origo is indicated as a can occur in both directions of an axes [53]

In the present work, a MEMS accelerometer will be placed around the chest and left wrist together with a MEMS gyroscope (7.1.1) to measure fall and non-activity patterns when performing the experiments. The two accelerometers will then measure a total of six accelerations ($a_{x,y,z}^{chest\,,wrist}$) when employed.

MEMS accelerometers measure the inertial a when a core moving beam structure moves. The beam structure is connected to two sets of *fingers*: one connected to a solid ground plate, the other attached to a known mass mounted on a spring that moves according to the applied a. When a is applied on the sensor, the fixed and moving beam fingers produces a change in C [56].

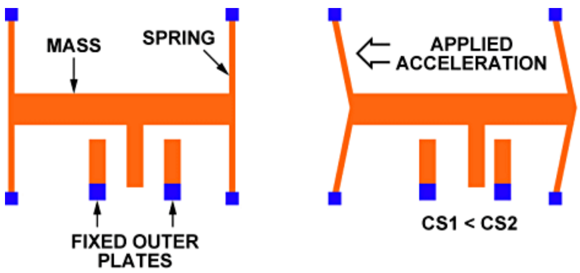


Figure 7-5. Illustration showing the measuring principle of a MEMS sensor that produced C relative to the applied a on the core moving beam of the sensor structure [56]

7.2 Measurand recorded by the Ambient Sensing System

The ambient sensing system consist of a door sensor that detects opening of a door when magnetic field is *not* presence. The door activity or state are denoted $\delta = 1$ when door is opened.

7.2.1 Magnetic Flux

The presence of magnetic flux, Φ measured in Weber $[Wb]$, forces conductance between two contacts inside a reed switch [52]. The switch generates electrical signals only when closed. The switch is then applied as a sensor which senses Φ by closing the wires as shown in Figure 7-6.

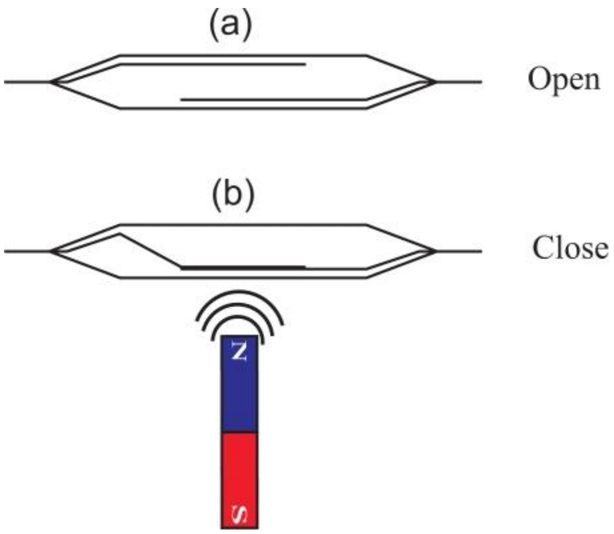


Figure 7-6. Two states of a reed switch: (a) open, no Φ is presence and the switch does not generate electric signals, or (b) closed, is Φ presence and the switch generates electric signals that could be read by a microcontroller [57]

7.3 Measurands recorded by a Smartphone

A smartphone including GPS and 4G network functionality is used for the collection of position data sets in the present work.

7.3.1 Position

Position in the geographical coordinate system as illustrated in Figure 7-7, are determined by the vertical and horizontal values of latitude (φ) and longitude (λ) and are measured in degrees [°] of the directions (North: N, +, South: S, -, East: O, +, or West, W, -) [52]. Equator is located at 0°. The North pole and South pole are located at $90°\ N$ and $90°\ S$, respectively.

All smartphones nowadays are equipped with GPS or other location services. GPS is the US satellite navigation system and the position parameters are determined using the signals from satellites spinning around the earth [52]. Other location services determine the position parameters by querying to an online database. A smartphone will often combine these methods to provide the most accurate position of the device [58].

A smartphone is a device well-suited to collect position measurement as it is a device that can apply internet almost everywhere.

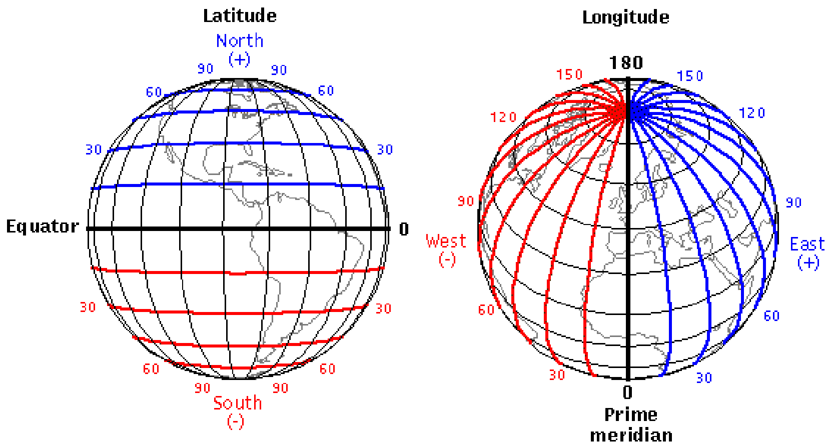


Figure 7-7. The geographical coordinate system showing the latitude (φ) and longitude (λ) coordinates and directions. The prime meridian goes through the Royal Observatory, Greenwich, in London [59]

8 Data Collection Units and Data Acquisition

This chapter describes the modules and parts involved in collecting the measurands of the unobtrusive monitoring system for ELIAH. The measurands described in chapter 7 are recorded using several DCUs including the sensing elements transmitted wirelessly to a PC by a bluetooth transmitter, as shown in Figure 8-1. A majority of the DCUs are assembled and programmed using the Arduino open platform. A smartphone is also used as a DCU of position data.

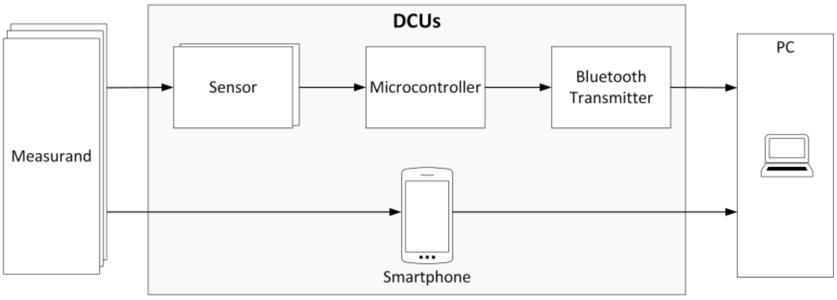


Figure 8-1. Overview of the sections involved in the data collection procedure; measurands being measured, processed, and transmitted by a DCU, and eventually saved into data sets by a PC

Two pairs of wearable, one ambient and a smartphone are the DCUs used to collect measurement data and transmit it wirelessly to a PC, where it will be stored into experimental data sets. These data sets will eventually be used to develop AI models for classification of critical events for ELIAH. An experimental data collection program is established using LabVIEW for receiving the data transmitted from the wearable and ambient DCUs and saving it into data sets. The measurands recorded by a smartphone are collected using an Android application and saved into a specified folder using the cloud service Dropbox. Figure 8-2 shows an illustrational overview of the DCUs, corresponding origin of measurand and transmitting technology to the PC where the data sets are saved into text files.

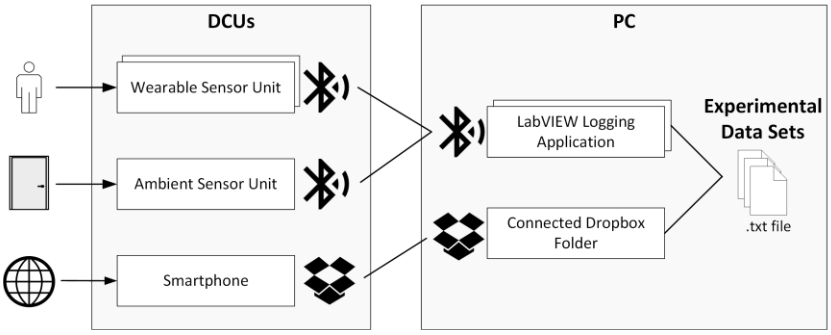


Figure 8-2. Illustrational overview of all the wireless DCUs of the unobtrusive monitoring system for ELIAH. The units use bluetooth communication and cloud service for data transmission of the recorded data to the PC. The PC includes a LabVIEW program to acquire data transmitted by bluetooth, and connected Dropbox folder for collecting the data sets. Bluetooth and Dropbox symbols acquired from [60] [61]

8.1 Wearable and Ambient DCU System

All the wearable and ambient DCUs are equipped with sensor modules, a microcontroller board for data handling, a bluetooth transceiver module to transmit experimental data to a PC and a battery for powering the wireless units.

For space reduction on the wearable DCUs, jumper wires were stripped and soldered into all the instrument modules. A block diagram of the modules involved in the wearable and ambient DCUs are shown in Figure 8-3.

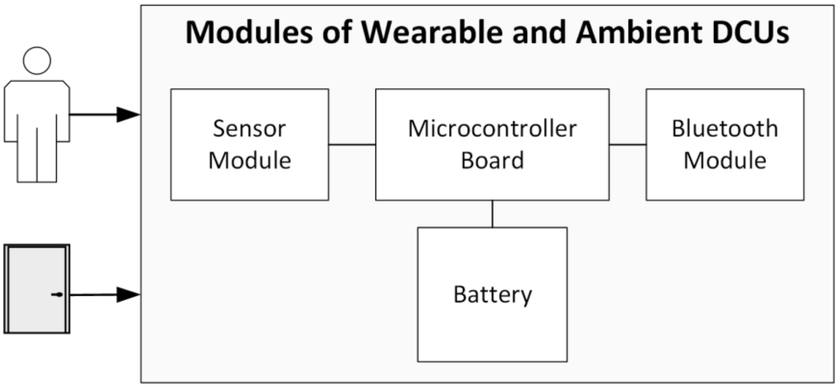


Figure 8-3. Block diagram showing the modules involved in the wearable and ambient DCUs and how they are connected

8.1.1 Microcontroller Board

The Arduino Pro Mini shown in Figure 8-4, is a small microcontroller board (33x17mm), suitable for the limited space on the wearable and ambient DCUs. The board is powered by 5-12 V and includes 14 digital input/output pins, in which 6 of them can be used as PWM based analog outputs, and 6 pins are analog input pins [62]. The board regulates 5 V power supply (VCC) on three pins. The Arduino Pro Mini is not equipped with USB connection for communication and power, but the six-pin header located on one of the short sides of the board (Figure 8-4) can be connected to a FTDI (USB to serial) adapter for this purpose. Figure 8-4 shows the Arduino Pro Mini and location of the various pins it is equipped with.

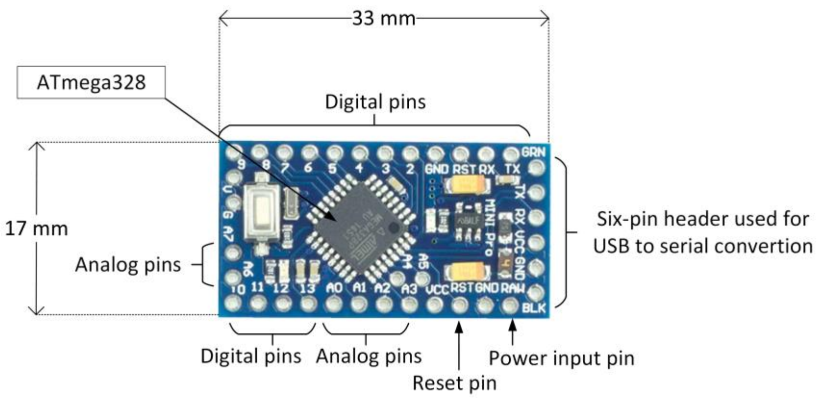


Figure 8-4. Arduino Pro Mini microcontroller board (33x14mm) showing the subtle amount of pinouts it is equipped with. and the ATmega328 microprocessor [63]

8.1.2 Bluetooth Module

The HC-06 is a bluetooth transceiver that enables serial communication with another paired bluetooth-compatible module. The HC-06 module is powered by 3,3 – 6 V. Illustrational overview of the HC-06 bluetooth transceiver and corresponding connections to the Arduino Pro Mini are shown in Figure 8-5. The RXD and TXD pins are connected to the Arduino Pro Mini's digital TX and RX pins, respectively. Other digital pins on the Arduino Pro Mini can be used as TX and RX pins if declaring so in the Arduino IDE program. A led on the bluetooth module blinks if the module is not paired, and lights static if paired with another device.

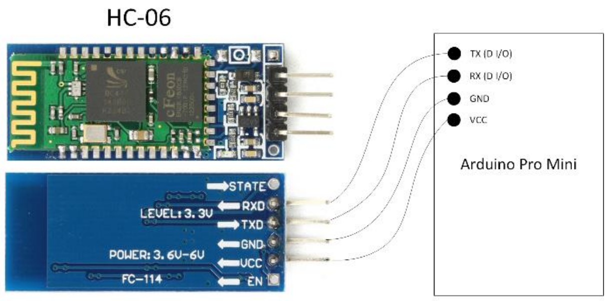


Figure 8-5. The HC-06 bluetooth module and its connections to the Arduino Pro Mini Microcontroller board [64]

8.1.3 Battery for Powering the Microcontroller

All the wearable and ambient data collection modules are wireless and requires power. A 9 V alkaline battery is connected to the Arduino Pro Mini for supplying power to these units. The Arduino Pro Mini requires to be powered using 5 – 12 V so no circuit handling are needed using these batteries. Figure 8-6 illustrates how the alkaline battery is connected to the pins on the Arduino Pro Mini.

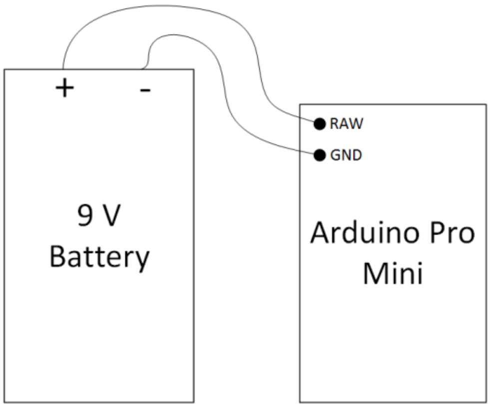


Figure 8-6. Illustrational diagram showing how the alkaline battery are connected to the pins on the Arduino Pro Mini. The battery's plus pole are connected to the 'RAW' pin, and the minus pole is connected to the 'GND' pin of the microcontroller

8.1.4 Inertial Sensor Module for Motion Measurement

The inertial sensor is a 6 DOF "breakout" sensor board connected to the wearable DCUs used for measuring motion. The sensor is a MPU-6050 which is located on the sensor board that combines a MEMS 3-axis gyroscope and a 3-axis accelerometer together with an onboard

Digital Motion Processor (DMP) chip [65]. The MPU-6050 also includes a temperature sensor which is not utilized during the present work. The gyroscope measures angular velocity, ω $[°/s]$, while the accelerometer measures linear acceleration, a $[g]$, along the x, y, and z axes [65]. The DMP chip contains a 16-bit analog to digital conversion on each channel enabling the module to record the x, y, and z channel at the same time [65].

Gyroscopes and accelerometer are inertial sensors commonly used for activity classification in body worn monitoring applications (elaborated in chapter 3). The combination of the two sensors produce six measurement-vectors during each recording, enabling the motion it records to be sufficiently described.

Figure 8-7 shows a block diagram of the pins used on the MPU-6050 sensor board and which pin-types they are connected to on the Arduino board as well as the axis orientation and placement of the wearable DCUs.

Using the datasheet for the MPU-6050 in [66], the sensors are configured to be applied during experimental procedures. The sensor board are connected to the microprocessor using I^2C or I2C (Inter-Integrated Circuit) bus. This bus requires a specific library, '*Wire*', to be included in the program code to work. The bus is typically used when attaching lower-speed peripheral ICs to processors and microcontrollers in short-distance [67]. Using the code found in [68], the user programmable full scale range of the gyroscope and accelerometer measurands are configured by changing the configuration bits in the setup of the Arduino Code (See Appendix B.1 for code of calculating sensor values from MPU-6050 and transmitting the data with bluetooth communication). The full-scale range of ω are chosen to be $\pm 250\ °/_{s}$, and the full-scale range of a are chosen to be $\pm 2\ g$. The selection of the full-scale ranges was made based on the measurements recorded on previous studies of this topic, for example in [19].

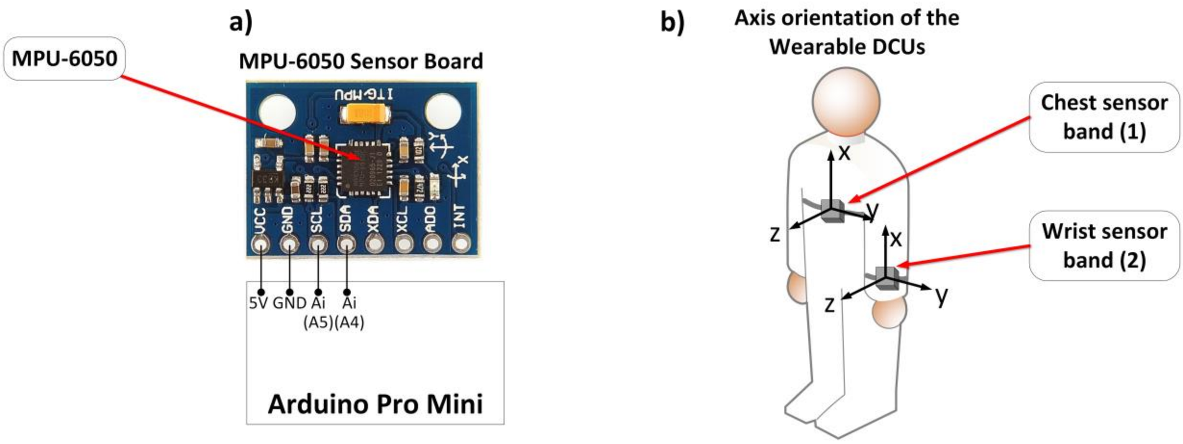


Figure 8-7. a) shows a block diagram of the pins of the MPU-6050 and how they are connected to the Arduino Pro Mini. b) shows the orientation of the x, y, and z axes of the wearable DCUs placed around the chest and left wrist are shown in the illustration to the right. The axes shown in the illustration are the absolute direction and not the positive and negative directions

8.1.5 Reed Switch Measurement of Door Activity

A door sensor consists of a reed switch, which is fixed to a door frame, and a magnet, which is fixed on the door. When the door is closed, the switch and magnet are connected and produces electric signals, while no electrical signals is detected when the door opens. This is how the sensor can detect opening of a door.

The wire ends of the reed switch are soldered to two pin headers to be able to connect the modules of the door sensor shown in Figure 8-8 to a small-sized breadboard for the experimental setup created in the present work. The reed switch circuitry shown in Figure 8-8 fixed to the door frame, consists of a voltage divider. The capacitor is used to handle signal bounce from the switch when it changes states.

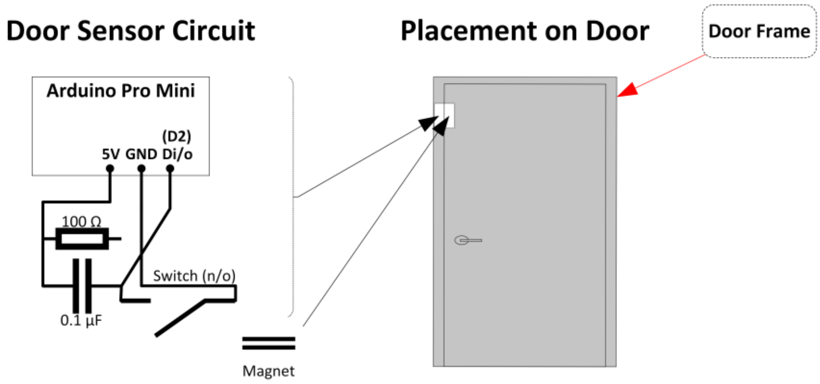


Figure 8-8. Showing the door sensor including the reed switch circuitry connected to the Arduino Pro Mini fixed to the door frame, and the magnet that triggers the switch states that is connected to the door

The code created for the door sensor transmits boolean values, $\delta = 0$ for when the door is closed, and $\delta = 1$ for when the door is detected open. Full code for the door sensor can be studied in detail in Appendix B.2.

8.1.6 Logging Application for Wearable and Ambient DCUs

Each HC-06 bluetooth module must make a connection with the selected COM-port specified to the LabVIEW program '*Logging Application for Data Collection Units'* in Figure 8-9 before the main loop that constantly records the measurands in the input buffer being received can start running. The LabVIEW application structure and running-sequence are described in the illustrational overview including the LabVIEW graphical program structure displayed in Figure 8-9.

A minor difference made in the LabVIEW program when logging data from the ambient DCU is that it does not convert comma to dot (punctum) because there is no comma involved in the data being transmitted from the ambient DCU, only one boolean value is transmitted.

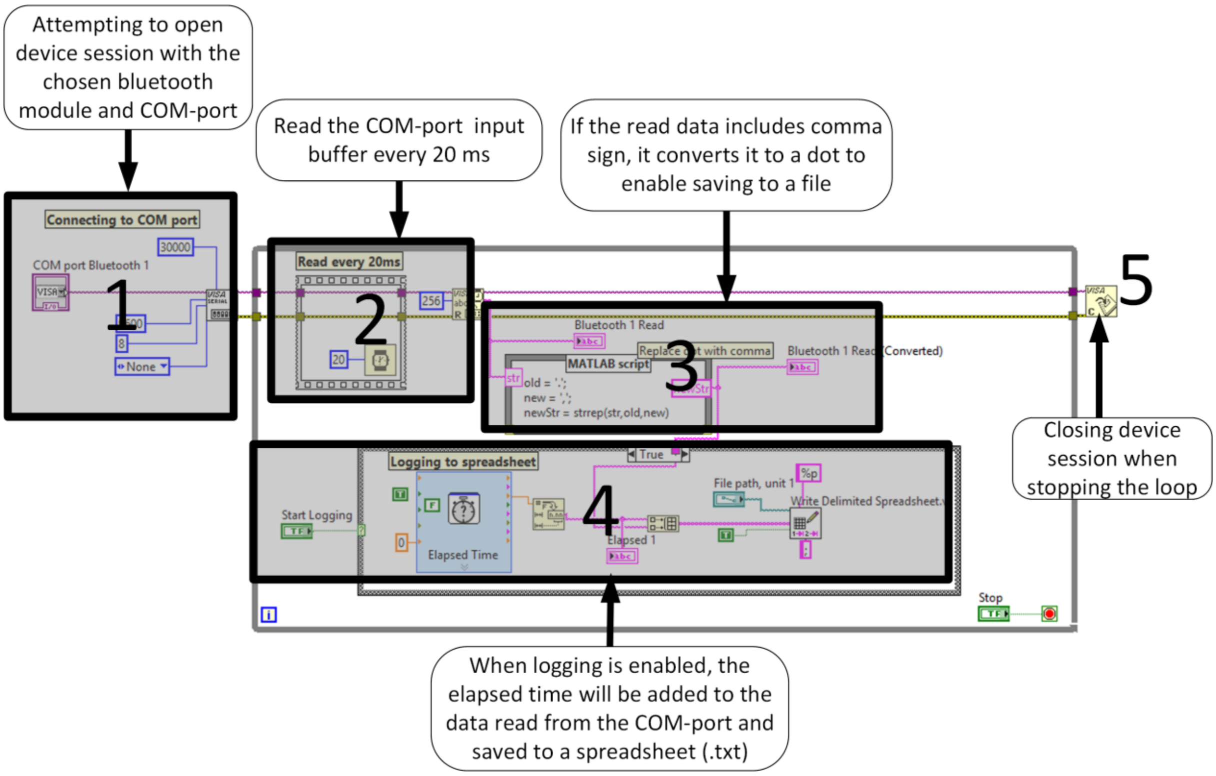


Figure 8-9. Illustrational overview showing the graphical program structure describing each part of the running sequence involved in the Logging Application for the DCUs, sequentially from 1-5. The illustration shows only the program structure of collecting data from one bluetooth transmitter. When collecting fall and non-fall experiments, two such program including loops are used. Collecting recording from the door sensor, only one bluetooth module is involve, so only the blocks specified in the illustration are used, excluding number 3, converting comma to dot

Example of the recorded experimental data obtained from the wearable DCUs are shown in Figure 8-10. The recorded measurement data are transmitted from the DCU in this manner every loop iteration in the Arduino program, without the elapsed time which is added in the LabVIEW logging application.

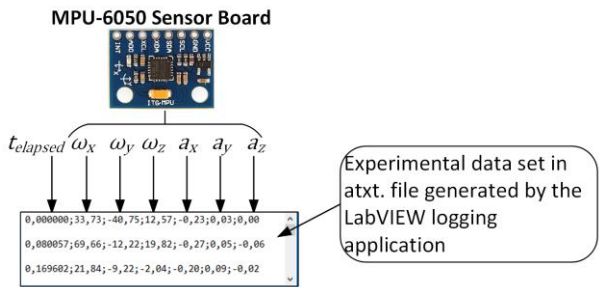


Figure 8-10. Showing a part of the experimental data set and how it is presented in the logged txt. file and pointing to each specific measurement using symbolism. The data recording in this illustration stems from an inertial sensor recording from one of the wearable DCUs

8.2 Smartphone DCU

For recording of position data, a smartphone is deployed using a downloaded Android application. This application records position measurement every minute and saves them in a text file placed in a connected Dropbox folder, which must be appended manually from the application when connected to internet. The positon measurements, φ, λ, together with the door sensor are to be used to create a geofence threshold AI model that detects whether the user has gone out of the living area and not come home after a pre-set time.

8.2.1 Android Application for Position Data Collection

An Android application called '*GPS Logger for Android'* are used for collecting positon measurement from the smartphone. This application has flexible options, is easy to use and requires minimum battery power when used. The application enables the user to choose which type of cloud-service the position data will be transmitted through, and what type of file format it will be saved as. The application saves the position each minute if the value of acceleration sensor embedded in the smartphone has changed since the last time it checked. This is convenient since the position data does not require time values when developing AI models for geofencing. In the present work, the Dropbox cloud-service was used to transmit the measured position data.

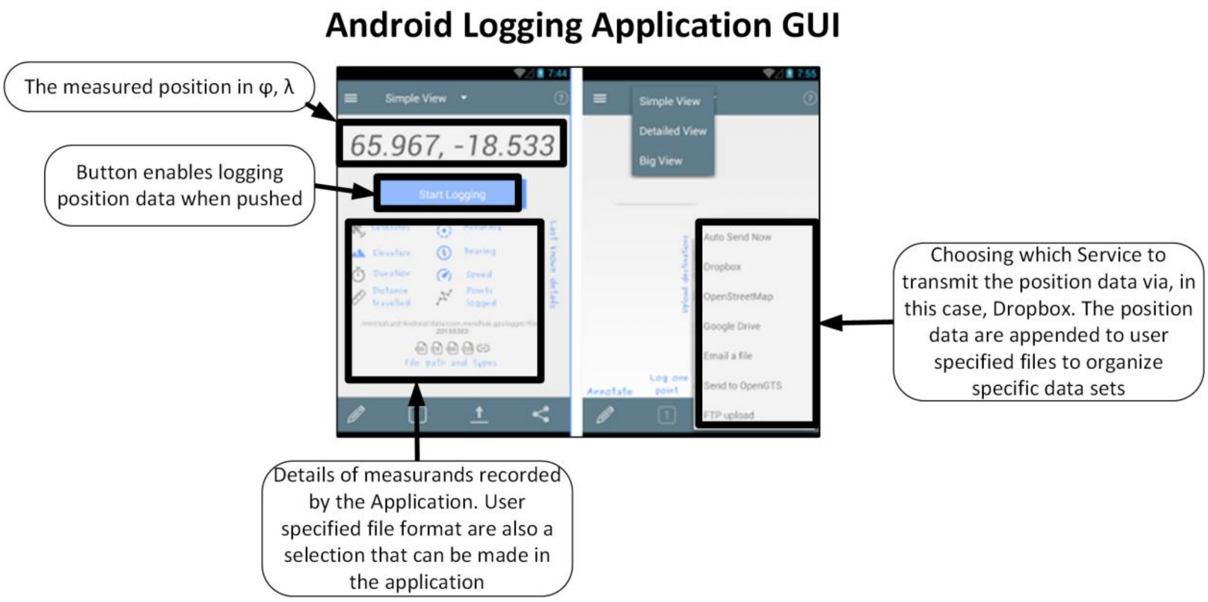


Figure 8-11. Screen shots showing some of the features of the 'GPS Logger for Android' application. The user can choose when to log, which cloud-service it will use to transmit and saving the position data. The application can also measure other parameters like number of satellites, distance travelled, and accuracy of the measurement. Figure from [69]

9 Experimental Procedures for Data Collection

The DCUs described in chapter 8 are used for experimental data collection procedures. Figure 9-1 is an illustrational overview showing the specific experiments that are to be collected by the DCUs using the LabVIEW logging application described in chapter 8.1.6.

This chapter describes the experimental plan and procedures for data collection of experiments, E to distinguish critical events from non-critical events in the proposed system for unobtrusive monitoring of ELIAH.

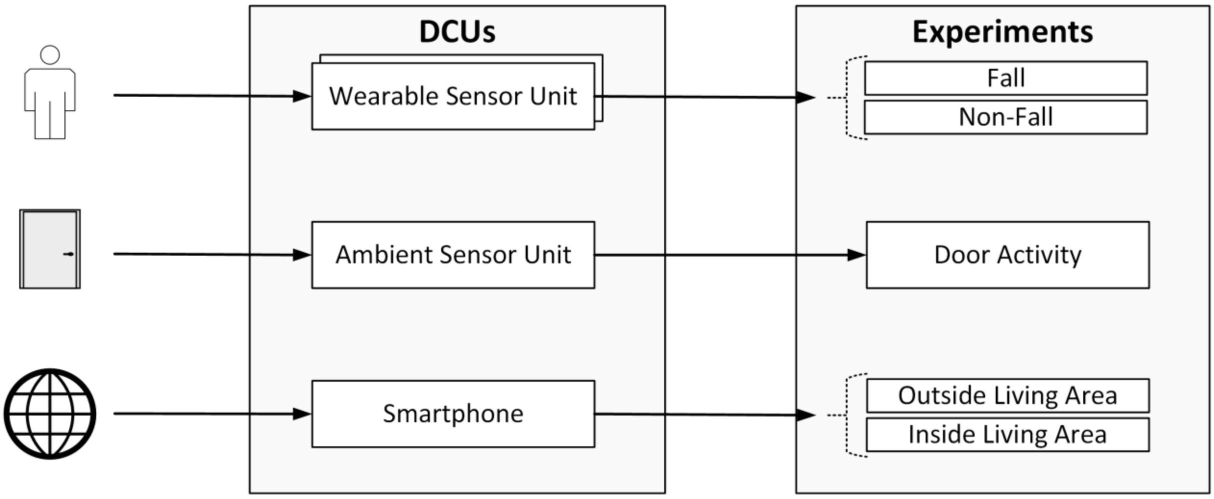


Figure 9-1. Graphical overview showing the experiments recorded by the DCUs. The datasets from these experiments will be later applied to generate AI models to distinguish critical events from non-critical events

9.1 Data Collection Procedures of the Wearable DCUs

The fall and non-fall experiments are performed using different type of landing grounds and furniture's commonly placed in all home environments. Bed, sofa, chair, and an exercise mat representing the floor, were utilized when recording falls and non-fall experiments. Non-fall experiments include transitional motion like standing to sitting, sedentary motion like sitting and lying and continuous motion like walking and running. Table 9-1 lists the experiments recorded by the wearable DCUs located around the chest and the left wrist.

Table 9-1. Description of the experiments collected by the wearable DCUs located around the chest and the left wrist

Experiment Type	Sub Experiment Categories and No. of Experiments	Description
Fall	Fall: $E_{fall} = E_{1-56}$	Various fall scenarios using different furniture's commonly stationed in the home including forward and backward falls
Non-Fall	Transition: $E_{transition} = E_{57-108}$ Sedentary: $E_{sedentary} = E_{109-120}$ Walking: $E_{walking} = E_{121-130}$ Running: $E_{running} = E_{131-140}$	Various non-fall scenarios using different furniture's commonly stationed in the home as well as a treadmill for walking and running, to distinguish fall from non-fall events

9.1.1 Description of Fall and Non-Fall Data Collection Procedure of Wearable DCUs

A total of 9 types of fall and 10 types of non-fall experiments are collected. Table 9-2 provides additional description of the fall and non-fall experimental data collection procedures as well as duration of the experiments.

Table 9-2. Description of the fall and non-fall experimental procedures using the wearable DCUs placed around the chest and the left wrist

Experiment Type	Sub Experiment	No. of Experiments	Sub Experiment Type	Description	Duration
Fall	E_{fall}	E_{1-6}= 6	Backward fall (BF)	From standing position to seating on ground	10 s
		E_{7-12}= 6		From standing position to lying on ground	10 s
		E_{13-18}= 6		From seating positon to lying on ground	10 s
		E_{19-24}= 6	Forward fall (FF)	From standing, knee flexion to seating on ground	10 s
		E_{25-30}= 6		From standing, knee flexion to lying on ground	10 s

		E_{31-36}= 6		From seating, knee flexion to seating on ground	10 s
		E_{37-42}= 6		From seating, knee flexion, lying on ground	10 s
		E_{43-46}= 4	Falling from bed	Seating (forward position) in bed, falling to ground	10 s
		E_{47-50}= 4		Lying in bed, falling to ground	10 s
		E_{51-56}=6	Walking motion and so fall forward	• Falling to seating position • Falling to lying position • Falling to seating and lying position and some movement after the fall	10 s
Non-Fall	$E_{transition}$	E_{57-70}=14	Stand to sit (StSi), sit to stand (SiSt)	Slow transition	10 s
		E_{71-84}=14		Fast transition	10 s
		E_{85-92} =8	Sit to lying (SiLy), lying to sit (LySi)	Slow transition	10 s
		E_{93-100}= 8		Fast transition	10 s
		$E_{101-104}$= 4	Sit to lying (SiLy), lying to sit (LySi)	Slow transition	10 s
		$E_{105-108}$= 4		Fast transition	10 s
	$E_{sedentary}$	$E_{109-114}$= 6	Sitting	Sedentary • Chair • Bed • Sofa	5 min
		$E_{115-120}$= 6	Lying	Sedentary • Sofa, sleeping/relaxing • Sofa, TV watching • Bed, sleeping/relaxing	5 min
	$E_{walking}$	$E_{121-130}$=10	Walking on treadmill	2, 3, 4, 5 and 6 [km/h]	60 s
	$E_{running}$	$E_{131-140}$=10	Running on treadmill	7, 8, 9, 10 and 11 [km/h]	60 s

9.2 Data Collection Procedures of the Ambient- and Smartphone DCU

Description of the data collection procedures of the ambient DCU (door sensor) to detect if the door has been opened, and the smartphone measures position inside and outside the living area for geofencing and detection of the user has gone outside or not are described in Table 9-3.

Table 9-3. Description of the experiments collected by the ambient DCU placed on the front door and smartphone placed in pocket

Experiment Type	**Sub Experiment**	**No. of Experiments**	**Description**
Door Activity	E_{door}	1	Assuring that the door sensor works properly
Position	E_{pos}	2	• Outside living area • Inside living area

10 Characteristics of Experimental Signal Trains

The experiments, E, described in chapter 9 are in this chapter plotted to study and extract characteristics that can possibly distinguish critical events from non-critical events.

This chapter will combine plots of the same sub experiments of various signal trains from the experimental data sets for analysis. Studying the signal trains will form the basis of the feature extraction procedure.

10.1 Characteristics of Signal Trains of the Wearable DCUs Experiments

The experiments E_{1-140} of the wearable DCUs placed around the chest and left wrist includes the inertial sensor that measures ω and a. Different fall and non-fall experiments are in this section compared by plotting the signal trains so it is clearer to present their distinctive characteristics.

10.1.1 Characteristics of Experiments from the Wearable DCUs

Fall are the critical event the unobtrusive monitoring system is to be detected from other non-fall events like transition, sedentary and walking. The plots in Figure 10-1 shows a typical characteristic of signal trains from all the four sensor measurements during a fall scenario. The absolute signal amplitudes of the measurements from all the four sensor measurements are quite large. All the fall sensor signals are characterized by being compact and peaky in a small period. All the x, y and z axes measurements are distinct during a fall. The signal motion after a fall are often characterized by a low frequency amplitude.

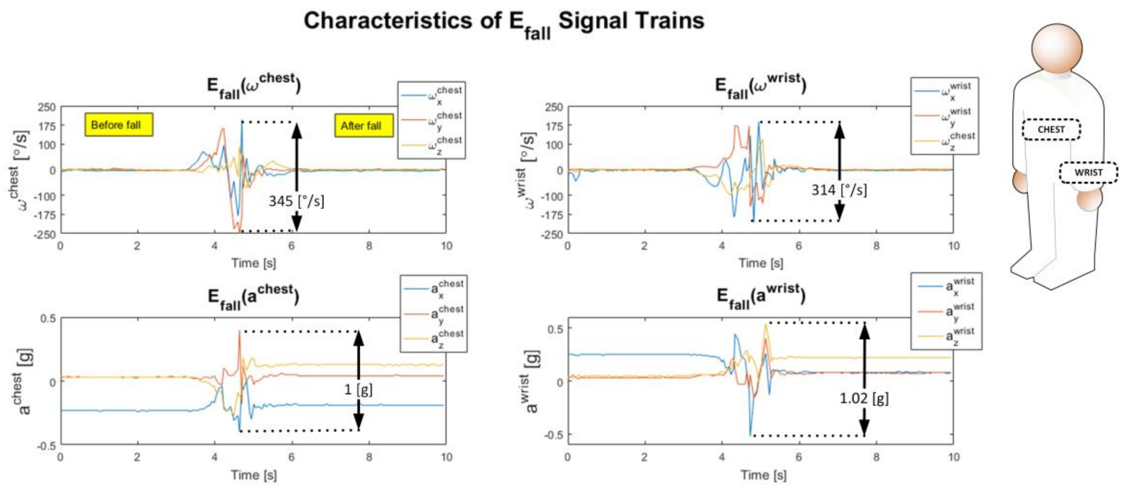


Figure 10-1 Typical ω and a signal train patterns during a fall. The plots in the figure represent the signal train is of E_1(see Table 9-2), using an exercise mat as landing ground. The distinct sensor signal train during a fall is characterized by being compact and peaky during a short period. Also, all x, y and z signals involve notable rapid peaks during a fall

10.1.2 Comparison of Sensor Signal Trains during Fall and Non-Fall Experiments

Figure 10-2 compares all ω, while Figure 10-3 compares all a of various fall and non-fall experiments. Fall signal trains are characterized by the rapid change of peaks both of ω and a signal trains. All three axes measurements during fall experiences rapid peaks, while only two axes have peaky tendencies during transition activity. The wrist signal trains ω^{wrist} and a^{wrist}, produce quite rapid peaks during fall, transition, and walking experiments. Fall signal trains are characterized that all the x, y, and z axes have more noticeable peaks compared to the not-fall signal trains, especially the y-axis. Large scale between highest and lowest peaks are also noticeable of fall experiments compared to non-fall experiments.

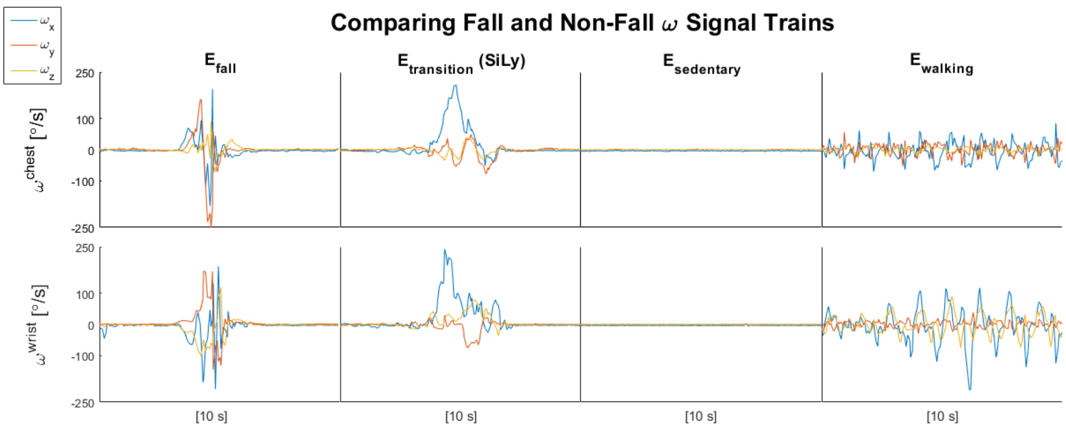


Figure 10-2. Signal trains plotting ω of fall and different non-fall E types; $E_{fall}(E_1)$, $E_{transition}$ *(StSi,* E_{11}*),* $E_{sedentary}(E_{18})$*, and* $E_{walking}$ (E_{19})*.* E_{fall} *signal trains involve quite rapid peaks in during a small period, especially compared to* $E_{transition}$*.* $E_{walking}(\omega^{wrist})$ *signals experiences high peaks, but mostly around the x and z axes*

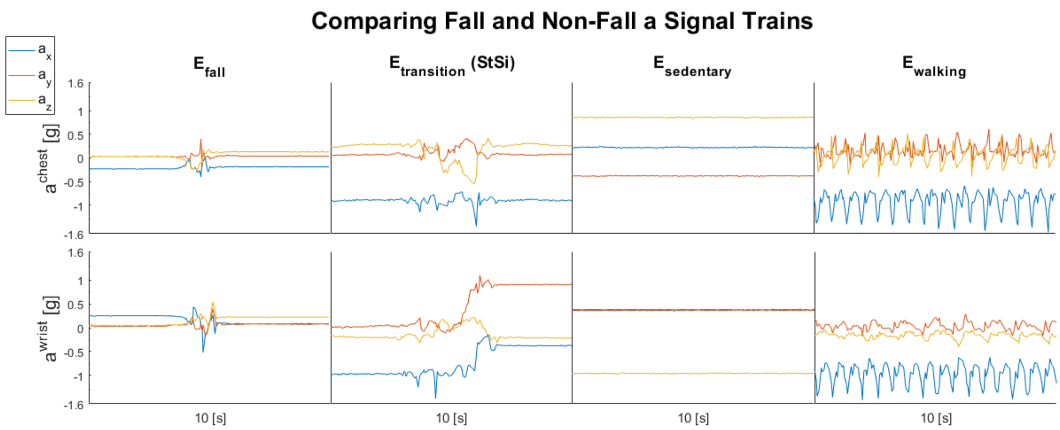


Figure 10-3. Signal trains plotting a of fall and different non-fall E types; $E_{fall}(E_1)$, $E_{transition}$ *(StSi,* E_{11}*),* $E_{sedentary}(E_{18})$*, and* $E_{walking}$ (E_{19})*.* E_{fall} *signal train of a signals also involve rapid peaks in during a small period, but experiences just as much g-force as* $E_{transition}$ *and* $E_{walking}$ *in some axes direction.* $E_{transition}$ *contains just as large peak sizes ad* E_{fall}*, but during longer time periods*

10.2 Characteristics of Signal Trains of the Ambient and Smartphone DCUs

To detect if a user has been outside the living area longer than a pre-set time period, the door sensor and position measurement sensor signals are fused together. This will make the system surer that it is the user that went out, not a visitor.

10.2.1 Characteristic of Door Sensor Signal Train

Figure 10-4 shows the states when door opens and closes. The sensor detects the state when the door is opened, denoted $\delta = 1$, and then closes again, $\delta = 0$.

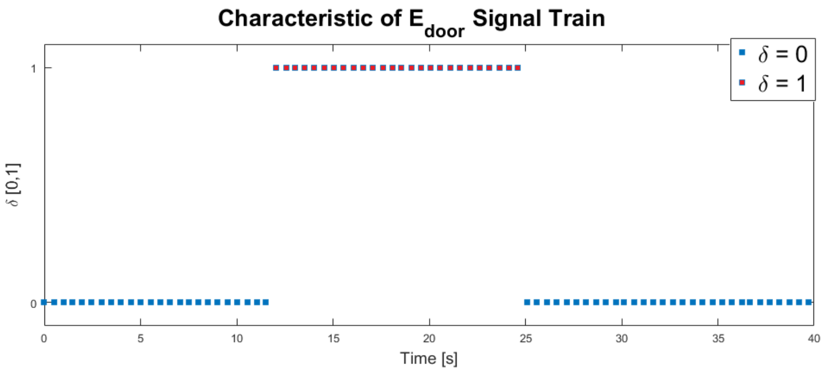


Figure 10-4. Showing E_{door} state change when door is initially locked, then detected opened and lastly closed again

10.2.2 Characteristic of Position Data Signal Train

Figure 10-5 shows the position data inside $(\varphi, \lambda)^{in}$and outside, $(\varphi, \lambda)^{out}$ the living area. Since position data are determined from both satellites and internet based services, the data will always consist of some uncertainty. For data collected in E_{pos}, the GPS based (satellites) data points consists of smaller uncertainty than the internet based ones. Position data is recorded each minute.

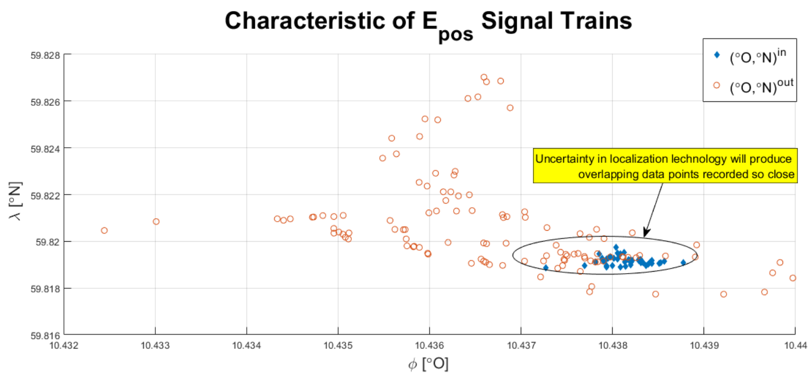


Figure 10-5. E_{pos} measurements plotted, that shows the logged φ and λ data points inside and outside the living area. Uncertainty in localization technology such as GPS by satellites and network will generally produce overlapping data points

10.3 Summary

In this chapter, the various E signal trains are plotted for analysis to extract important characteristics of measurements.

Distinct characteristic of that separated fall from non-fall experiments is that fall experiments include higher signal difference between highest and lowest peak in the signal. The fall movement pattern included higher peak amplitudes compared to non-fall experiments. Some non-fall experiments included quite high peaks, but not in all the x, y and z axes. The y axis signals had quite high amplitude during fall signal trains, and was the most distinct signal that characterized fall from non-fall experiment.

The position measurement during E_{pos} created overlapping measurement of the data collected inside and outside the living area.

11 Data Processing and Feature Extraction of Experimental Data Sets

After characteristics of the E signal trains are studied in chapter 10, the data sets are processes and made ready for feature extraction. The data sets E_{1-140} (fall and non-fall) from the wearable DCUs require most handling before launching the generation of AI models. E_{door} and E_{pos} will not be processed. The activities involved in data processing and feature extraction of E_{1-140} data sets are illustrated in Figure 11-1, where $\boldsymbol{F}$ is the feature matrix involving the experiments, E, and features F. The final feature matrix $\boldsymbol{F}(E(130)xF(80))$ does not have the same order of the Es as E_{1-140} indicate, because of variable name handling in MATLAB.

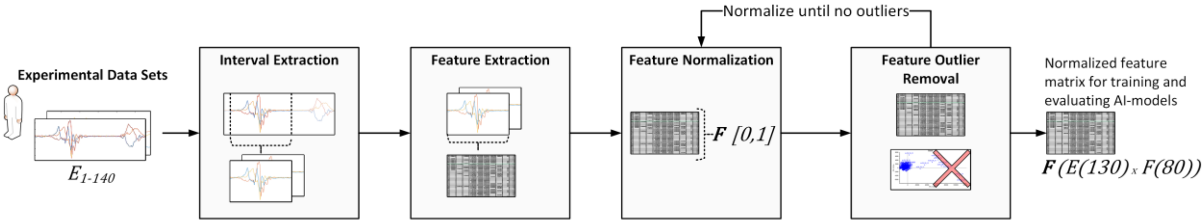


Figure 11-1. Graphical overview explaining the flow of activities involving interval and feature extraction as well as feature normalization and feature outlier removal. Interval extraction involves extracting the necessary time intervals from the signal trains, feature extraction involves determining the features of all E_{1-140} and construct the feature matrix $\boldsymbol{F}$. Normalization is applied to $\boldsymbol{F}$ so no features will be weighted higher for involving larger values than others. Some of the E are removed during outlier removal, as these Es did not represent the data good enough to create AI models for fall detection, which could possibly mislead the training process

11.1 Interval Extraction of Experimental Data Sets from the Wearable DCUs

The data processing involves extracting 10 s time intervals of E_{1-140}. Some Es includes signals in the beginning and the end during experimental procedures, that originates from logging the data from the LabVIEW logging application, which is removed. $E_{sedentary}$, $E_{walking}$, and $E_{running}$ were divided into several 10 s signal trains. This was handled so all inputs of the AI models for fall detection involves the same time interval lengths, $t_{interval}(s)$. Using Excel, the $\omega_{x,y,z}^{chest,wrist}$ and $a_{x,y,z}^{chest,wrist}$ data sets of each E_{1-20} were combined and shares the same time column, $t_{interval}$. All E_{1-140} were so imported into a workspace in MATLAB as individual variables that includes 13 columns each. Figure 11-2 shows and example of the MATLAB variable of E_{51} signals.

	$t_{interval}$	ω_x^{chest}	ω_y^{chest}	ω_z^{chest}	a_x^{chest}	a_y^{chest}	a_z^{chest}	ω_x^{wrist}	ω_x^{wrist}	ω_x^{wrist}	a_y^{wrist}	a_z^{wrist}	ω_x^{wrist}
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	-2.6000	-2.3100	-0.0800	-0.2300	-0.0100	0.0700	2.5600	0.2100	-0.7100	0.2400	0.0700	0.0400
2	0.0676	-3.1100	-3.1500	-1.6000	-0.2200	-0.0100	0.0700	3.1100	1.2100	-0.5000	0.2400	0.0800	0.0400
3	0.1512	-4.3700	-4.0700	-2.0700	-0.2300	-0.0100	0.0700	-11.1800	0.5100	-1.4700	0.2400	0.0800	0.0400
4	0.1912	-4.5500	0.0500	-0.4300	-0.2300	-0.0100	0.0700	-3.8000	-0.0500	-0.0100	0.2400	0.0700	0.0400
5	0.2663	-5	0.6600	-0.0200	-0.2200	-0.0100	0.0700	-7.4400	-0.0100	-1.4000	0.2400	0.0800	0.0300
6	0.3153	-3.1800	0.7300	-1.2700	-0.2300	-0.0100	0.0700	-3.6400	-0.1100	-1.3400	0.2400	0.0700	0.0400
7	0.3659	-3.3200	-2.2500	-2.2100	-0.2200	-0.0100	0.0700	-6.7300	0.0500	-1.9100	0.2400	0.0700	0.0300
8	0.4229	-3.1500	-0.2300	-1	-0.2200	-0.0100	0.0700	-3.8500	0.5600	-1.1900	0.2400	0.0700	0.0400
9	0.4985	-2.0800	-0.0700	-1.1400	-0.2200	-0.0100	0.0700	-0.9600	-0.3200	-0.7300	0.2400	0.0700	0.0400
10	0.5612	-3.0400	0.6000	-0.9700	-0.2200	-0.0100	0.0700	-3.2000	0.8200	-0.1800	0.2400	0.0800	0.0400
11	0.6235	-2.7000	-0.8400	-0.9800	-0.2200	-0.0100	0.0700	-6.3700	-0.7800	0.1200	0.2400	0.0700	0.0400
12	0.6866	-3.0400	-2.5500	-3.2400	-0.2200	-0.0100	0.0700	-8.1500	-0.4700	-0.4200	0.2400	0.0700	0.0400
13	0.7401	-7.0400	3.5200	1.1800	-0.2200	-0.0100	0.0800	-5.5000	0.0800	0.1100	0.2400	0.0700	0.0400
14	0.7987	-3.5400	0.5300	0.5000	-0.2200	-0.0100	0.0700	-5.0800	-0.0200	-1.0200	0.2400	0.0700	0.0400
15	0.9065	-3.4500	1.2700	0.5300	-0.2200	-0.0100	0.0700	-3.6600	-0.3800	-1.0900	0.2400	0.0700	0.0400
16	0.9412	-3.9100	3.8600	1.6000	-0.2300	-0.0100	0.0700	-1.2900	-0.0300	0.1800	0.2400	0.0700	0.0400
17	0.9718	-2.6600	1.9200	1.3900	-0.2200	-0.0100	0.0700	-4.5900	1.1800	-1.1700	0.2400	0.0700	0.0400
18	1.0346	-4.3300	3.0600	1.4400	-0.2200	-0.0100	0.0700	-3.3200	0.7400	-0.3400	0.2400	0.0700	0.0300
19	1.0972	-3.5300	3.7600	1.7300	-0.2200	-0.0100	0.0700	-2.1500	0.6400	0.3700	0.2400	0.0700	0.0400
20	1.1657	-1.2900	0.3100	0.2600	-0.2200	-0.0100	0.0700	-1	0.9200	0.0500	0.2400	0.0700	0.0400

Figure 11-2. Showing the MATLAB variables after chest and wrist signals are combined with the same time interval $t_{interval} = 10\ s$

11.2 Feature Extraction Procedure of Experiments from Wearable Data Sets

Relevant features, F, are extracted from the experimental data sets based on the domain knowledge obtained by studying the characteristics of the signal trains in chapter 10. Initial features are generated into a feature matrix, $\boldsymbol{F}(E(l = 140)xF(n = 80))$ representing all features for E_{1-20} (samples), were l is number of experimental data sets E and n is number of features F. The feature extraction procedure using the MATLAB script are illustrated in Figure 11-3 and can be studied in detail in Appendix B.4.

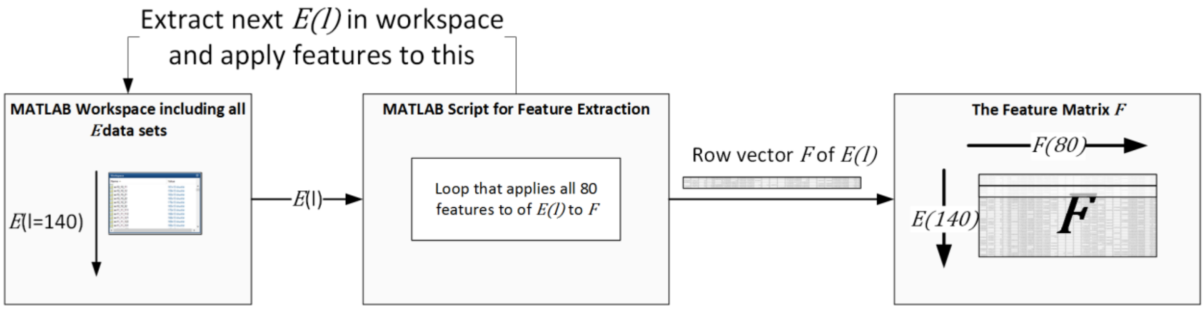


Figure 11-3. Workflow of the activities involved when extracting features from the pre-processed data sets

Magnitude $\boldsymbol{M}$, described by equation (11-1) together with sum $\boldsymbol{S}$ described by equation (11-2) are applied to *some* of the features E_{1-140}. The purpose of this is obtain a feature values that describes the feature spectrum of all x, y and z axes at the same time. The $\boldsymbol{M}$ and $\boldsymbol{S}$ combination can help distinguish fall from non-fall events where high peaks of all three axes are distinct in E_{fall}.

$$magnitude = \boldsymbol{M}(F(x, y, z)) = \sqrt{(F(x))^2 + (F(y))^2 + (F(z))^2} \tag{11-1}$$

$$sum = \boldsymbol{S}(F(x, y, z)) = F(x) + F(y) + F(z) \tag{11-2}$$

Table 11-1. Feature category and number of features F in each category of E_{1-20}

Feature Category	**No. of Features of the Feature Categories**
Peak Features	$F_{noOfPeak} = F_{1-12}$ $F_{S(noOfPeak)} = F_{13-16}$ $F_{thPeak} = F_{17-28}$ $F_{S(thPeak)} = F_{29-32}$ $F_{S(diffPeak)} = F_{77-80}$
Percentile Features	$F_p = F_{34-60}$
Wavelet Features	$F_c = F_{61-64}$
Cross-Correlation Features	$F_{xcorr} = F_{65-76}$

11.2.1 Peak Features

Features $F_{1-32,77-80}$, are features describing peaks of the signal trains. Table 11-2 describes the peak features in detail together with the equation of the four peak feature categories described in Table 11-2. The plot in Figure 11-4 shows the deviation of the combined peak features, $F_{S(noOfPeak)}$, $F_{S(thPeak)}$ $F_{S(diffPeak)}$ for E_{fall}, $E_{transition}$, $E_{sedentary}$, and $E_{walking}$. Standard deviation, σ, of all the normalized features (after outlier detection, see 11.3 and 11.4) are the values plotted in Figure 11-4. The σ is applied to the features compare the variation in the feature spectrum of the plotted experiment categories.

Table 11-2. Additional description and equations of the peak features $F_{1-32,77-80}$

Feature Category	**Feature Description**
$\boldsymbol{F_{noOfPeak}}$	Number of peaks, P, above/under a threshold in of all data points R of data sets $\omega_{x,y,z}^{chest,wrist}$ and $a_{x,y,z}^{chest,wrist}$
	$F_{noOfPeak} = \sum_{r=1}^{R} P(\vert\omega(r) > 50\vert \wedge \vert a(r) > 0.5\vert)$ where $\wedge$ means *or* and $r = 1..R$ is all the time variant data points in the E signals
$\boldsymbol{F_{S(noOfPeak)}}$	Sum of all $F_{noOfPeak}$of all ω and a experiments of both chest and wrist sensors, respectively
	$F_{S(noOfPeak)} = \boldsymbol{S}(F_{noOfPeak})$
$\boldsymbol{F_{thPeak}}$	Number of peaks, P, that are *at least* higher/lower than a *threshold* value of their neighbouring data point $r \pm 1$, above/under a threshold in all data points, R, of all $\omega_{x,y,z}^{chest,wrist}$ and $a_{x,y,z}^{chest,wrist}$
	$F_{thPeak} = \sum_{r=1}^{R} P(\vert(\omega(r-1) \vert\vert\ \omega(r+1) > 50\vert \wedge \vert(a\,(r-1)\vert\vert\ a(r+1) > 0.5\vert\,)$ where $\vert\vert$ means *and*
$\boldsymbol{F_{S(thPeak)}}$	Sum of all F_{thPeak} of all ω and a experiments of both chest and wrist sensors, respectively
	$F_{S(thPeak)} = \boldsymbol{S}(F_{thPeak})$
$\boldsymbol{F_{diffPeak}}$	Highest difference in peak of $\omega^{chest,wrist}$ and $a^{chest,wrist}$
	$F_{diffPeak} = \vert max(P)\vert + \vert min(P)\vert$ Where P can be the peak of any of the x, y or z axes

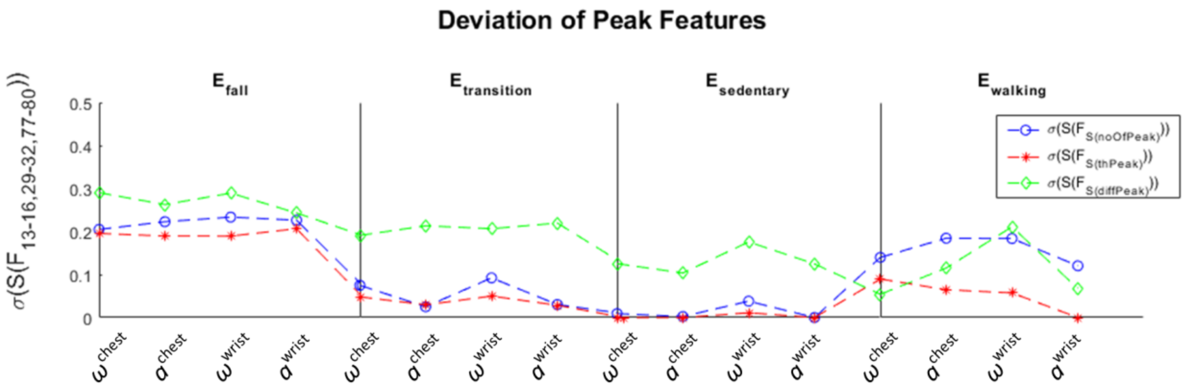


Figure 11-4. Plotting the σ of all normalized P features of E_{fall}, $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ experiment categories. The fall experiments all have high peak feature values, especially compared to walking

11.2.2 Percentile Features

Percentile, p, is a statistical measure that indicates a certain percentage of score that fall below that percentage. For example, the 10^{th} percentile is the value below which 10% of the observations may be found. 28 features are constructed consisting of several p ranges, which could indicate values that will distinguish fall from non-fall events. Table 11-3 describes the p features F_p further, as well as equations of how the feature are constructed. Figure 11-5 compares all F_p of the percentile $p^{10}, p^{20} \ldots p^{95}$ of, E_{fall}, $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ signal trains. The σ of the normalized F_p are applied before plotting.

Table 11-3. Description of F_p and equation describing how the p type features F_p are determined

	Feature Description
$\boldsymbol{F_p}$	Percentiles of each signal $\omega^{chest,wrist}$ and $a^{chest,wrist}$ (10^{th}, 20^{th}, 35^{th}, 50^{th}, 70^{th}, 80^{th} and 95^{th} percentiles) $p^{10}, p^{20} \ldots p^{95}$
	$$F_p = \boldsymbol{M}(p^i_{x,y,z}), \qquad p^i = \frac{i(R+1)}{100}$$ where i is the th percentile p or observation in the signal r. . R

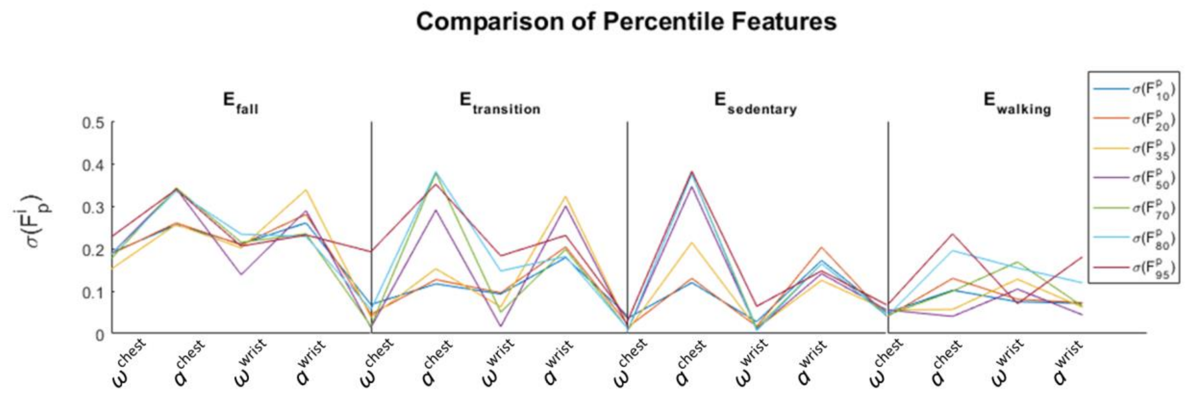


Figure 11-5. Plotting the σ of all normalized F_p of E_{fall}, $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ experiment categories. The fall experiments all have high feature values of the different signal trains, especially compared to walking and transition

11.2.3 Wavelet Features

A two-level wavelet transform is applied to all x, y and z signals of $\omega_{x,y,z}^{chest,wrist}$ and $a_{x,y,z}^{chest,wrist}$, where the mother wavelet being Daubechies. Wavelets is a good way of explaining *abrupt changes* in a signal, which is often the most interesting part of the data [70]. Features including wavelet coefficients c can therefore contribute to detect fall from non-fall events. Fall signal trains contain distinct abrupt changes when fall occurs. Using the MATLAB function 'wavedec()' to extract wavelet decomposition of a signal X at level Q:

```
[C,L] = wavedec(E,Q,'wname')
[cx_g1] = wavedec(experiment(:,2),2,'db2');
```

Where `experiment(:,2)` refers to ω_x^{chest} signal as indicated by the column indexes described in chapter. 11.1 The wavelet decomposition structure of the MATLAB function 'wavedec()' returns the wavelet decomposition vector c and a vector L, which contains the number of c by Q. Figure 11-6 displays graphically how a 3-level wavelet decomposition is obtained for the signal X[71]. The MATLAB script for creating all the features in $\boldsymbol{F}$ can be studied in Appendix B.4. Table 11-4 provides further description of the F_c and the plots of Figure 11-7 shows the deviation of the wavelet feature F_c of all E_{fall}, $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ signal trains by normalizing and applying σ to display the feature spectrum for all signals and motions.

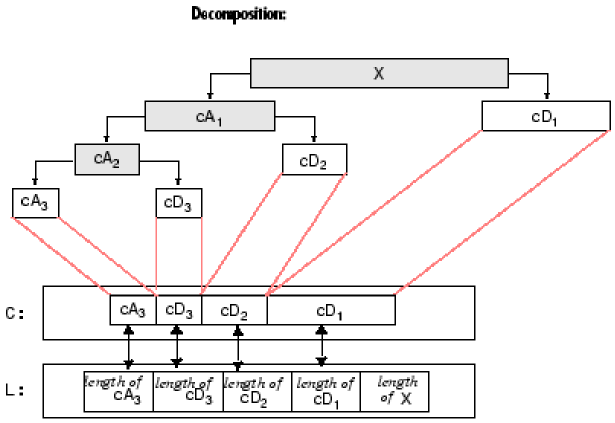


Figure 11-6. Showing the decomposition of a 3-level wavelet transform using X as the input signal to be decomposed [71]

Table 11-4. More detailed description of the wavelet feature F_c

	Feature Description
$\boldsymbol{F_c}$	$\boldsymbol{M}(c)$ extracted from a 2-level wavelet transform (the mother wavelet being Daubechies), of $\omega_{x,y,z}^{chest,wrist}$ and $a_{x,y,z}^{chest,wrist}$

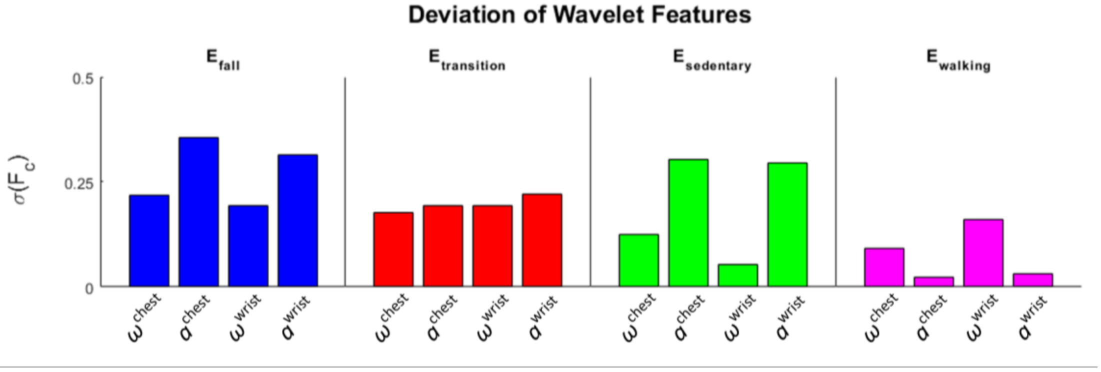


Figure 11-7. Comparison of the F_c wavelet features. The E_{fall} holds overall higher feature values than the other experiment categories; $E_{transition}$, $E_{sedentary}$ and $E_{walking}$. More abrupt changes are experienced by all signals in E_{fall}, compared to the others. Interestingly walking has relatively low wavelet feature values. A reason for this is that walking motion produce a constant frequency motion that repeats itself and results in fewer abrupt changes through of the signal range R

11.2.4 Cross-Correlation Features

The cross-correlation is a measure of how similar two time series are. Two time series that are similar will have a high correlation, $xcorr$. Combining the $xcorr$ of all x, y and z signals in an E, $\boldsymbol{M}(xcorr)$, can possibly distinguish the fall from non-fall events. Table 11-5 describes the cross-correlation feature F_{xcorr} further and Figure 11-8 shows the deviation of the $xcorr$ features by plotting the σ of the normalized features of E_{fall} $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ experiment categories (after outlier removal). The signals of the sedentary signal trains have high correlation compared to fall, transition and walking as shown in Figure 11-8. The reason for the high correlation of $\mathrm{E_{sedentary}}$ signals is that all the signals have low frequency because there is little or no movement involved.

Table 11-5. Detail description of the xcorr features F_{xcorr} and the equation for calculating xcorr applied to the ω_x^{chest} signal for demonstration purposes

	Feature Description
$\boldsymbol{F_{xcorr}}$	Cross-Correlation of x-y, x-z, and y-z signals in each measurement. $xcorr_{xy}$, $xcorr_{xz}$, $xcorr_{yz}$ of all $\omega^{chest,wrist}$ and $a^{chest,wrist}$
	$$F_{xcorr} = \boldsymbol{M}(xcorr),$$ $$xcorr_{xy}(\omega_{xy}^{chest}) = \frac{\sum_{r=1}^{R}(\omega_x^{chest}(r) - \bar{\omega}_x^{chest})(\omega_y^{chest}(r) - \bar{\omega}_y^{chest})}{\sqrt{\sum_{i=1}^{R}(\omega_x^{chest}(r) - \bar{\omega}_x^{chest})^2 \sum_{i=1}^{R}(\omega_y^{chest}(r) - \bar{\omega}_y^{chest})^2}}$$ Applied to all $\omega^{chest,wrist}$ and $a^{chest,wrist}$

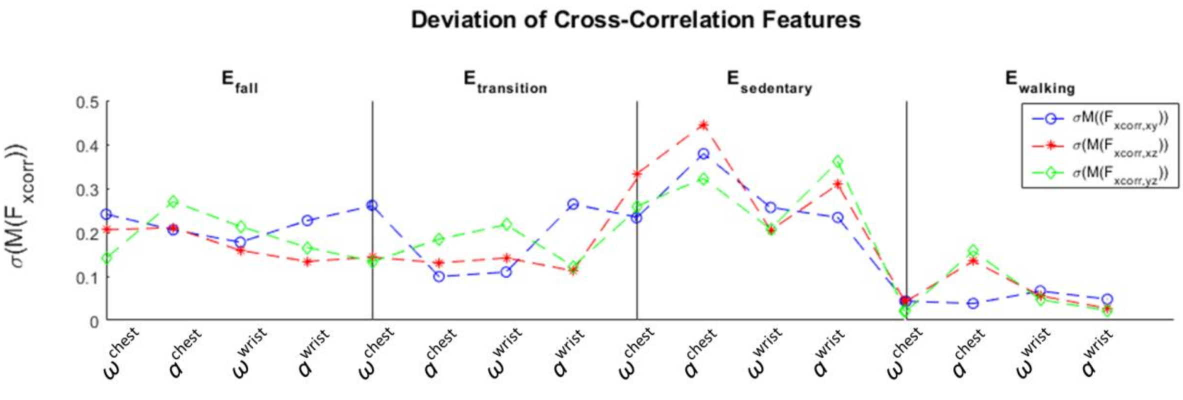


Figure 11-8. Comparing the xcorr features F_{xcorr} of all E_{fall} $E_{transition}$, $E_{sedentary}$ and $E_{walking}$ experiment categories after applying the σ to the normalized features. The reason for the high correlation of the sedentary signals is because there is little to no movement involved in all the x, y and z signals trains

11.3 Normalization of Features

After the $\boldsymbol{F}$ is generated or extracted using the *feature extraction* script in Appendix B.4, the features are normalized between [0,1]. It is important to normalize the features, so feature that includes high values are not being granted higher weights, W, or importance when the AI models are trained. A consequence of training AI models without normalized features is that the features granted higher W will be considered more important than features with lower W(or lower feature values) during feature selection further elaborated in chapter 12 [72]. All the columns in $\boldsymbol{F}$ (F_{1-80}) are normalized individually between [0,1] using *max-min* normalization by formula (11-3).

$$\sum_{n=1}^{80} \boldsymbol{F}_n^{norn} = \frac{\boldsymbol{F} - F_{n,min}}{F_{n,max} - F_{n,min}} \tag{11-3}$$

Where n is each feature column of the feature matrix $\boldsymbol{F}$. The normalized matrix is referred to as $\boldsymbol{F}$ throughout the report.

11.4 Feature Outlier Removal

The multivariate data analysis tool and software *Unscrambler*, is used to identify outliers of the normalized feature set $\boldsymbol{F}$**.** Outliers that deviates far from the other data points can mislead the training of AI models. Performing a simple PCA on $\boldsymbol{F}$ without validation in Unscrambler, a score plot (Figure 11-9) is generated, which are studied to identify outliers in $\boldsymbol{F}$. $E_{running}$, described in Table 9-2 is an experiment category that includes features that deviates far from the other experiment samples. All the $E_{runnnig}(l = 10)$ experiments are therefore removed from $\boldsymbol{F}$.

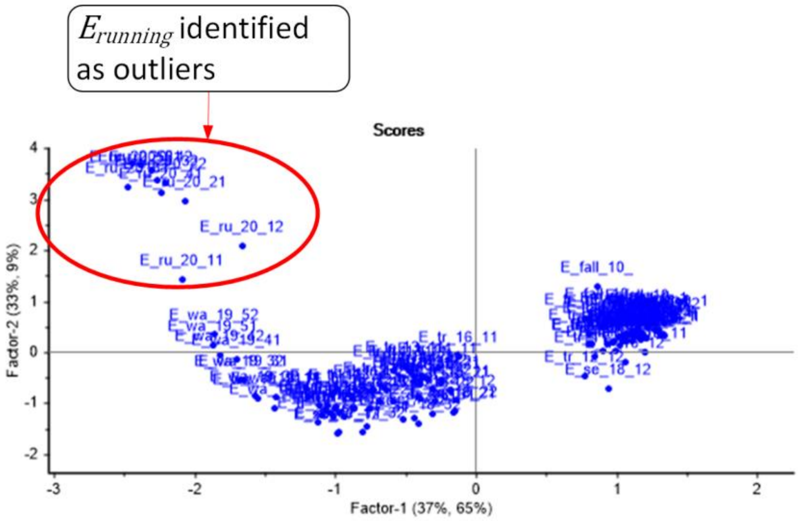


Figure 11-9. Score plot generated after performing PCA on $\boldsymbol{F}$ *in Unscrambler. The* $E_{running}$ *experiments inside the red circle, is identified as outliers, and will be removed from* $\boldsymbol{F}$ *before training AI models for fall detection*

11.5 Geofence Feature Extraction

The geofence feature F_{geo} depends on a defined radius $ø_{thresh}$ of the geofence, which is determined by the home coordinates $(\varphi, \lambda)^{home}$, as shown in Figure 11-10. Table 11-6 provides a detailed description of F_{geo}, which a state value that is equal to $F_{geo} = 1$ if true (inside geofence), and equal to $F_{geo} = 0$ if false (outside geofence).

Table 11-6. Description of the geofence feature F_{geo}, and the equation for calculating F_{geo}

	Feature Description
$\boldsymbol{F_{geo}}$	The absolute values of the coordinate distance or [°] from the home coordinates $(\varphi, \lambda)^{home}$. The user-defined radius, $ø_{thresh}$, are determined before utilizing the unobtrusive monitoring system for elderly
	$F_{geo} = (ø_{\varphi} \wedge ø_{\lambda}) > ø_{thresh}$ Where: $ø_{\varphi} = \lvert\varphi^{home} - \varphi\rvert$ $ø_{\lambda} = \lvert\lambda^{home} - \lambda\rvert$ The user is: • outside geofence if $F_{geo} > ø_{thresh}$ • inside geofence if $F_{geo} \leq ø_{thresh}$

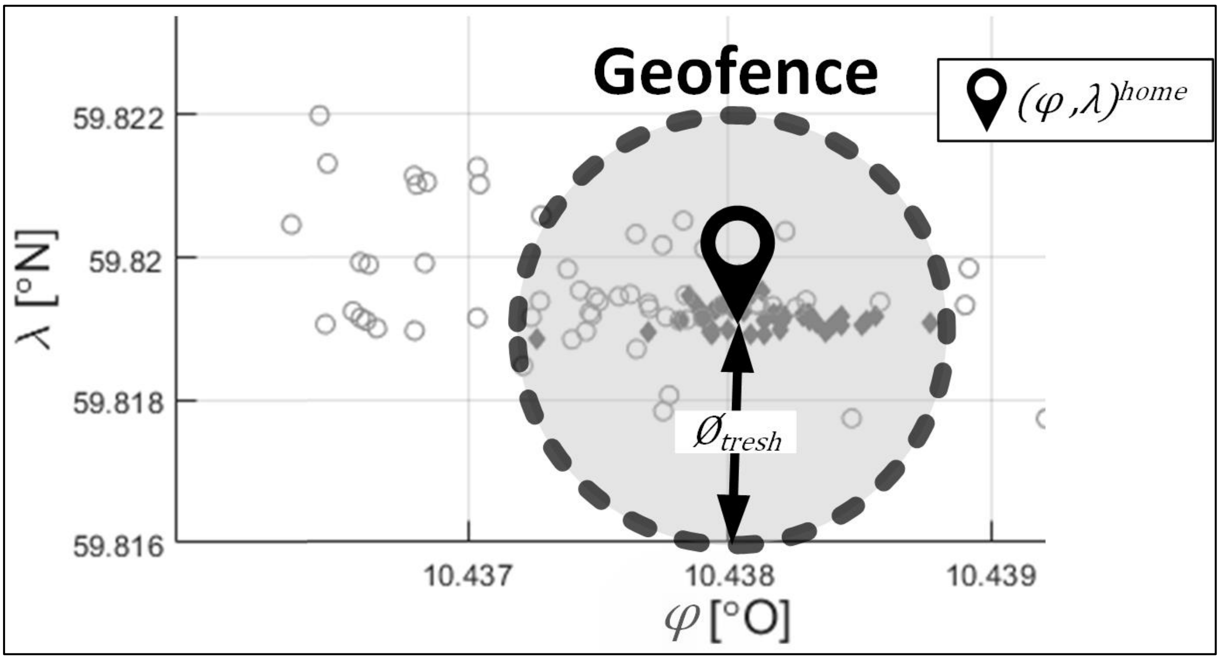


Figure 11-10. Graphical display of the parameters included in the geofence feature F_{geo}, how the geofence is defined, the radius $ø_{thresh}$, and coordinated of the home environment $(\varphi, \lambda)^{home}$

12 AI Models for Classification of Critical Events

Two AI models are created; one for detection of fall, the other for detecting if the user has not come home after a pre-set time after going outside the living area. Generating and evaluating the fall detection AI model involves most workload, and is the model most this chapter concerns. The AI model for detecting if a user has not come home after a pre-set time involves generating a simulation procedure based on the measurements involved in E_{door} and E_{pos}.

A general overview of the development of an AI model are shown in Figure 12-1, where known data are fed into an AI model for generation or training of the model and minimize the error of the known responses. The AI model should then detect the corresponding responses of new, unseen data sets being fed into it, which is a simulation of how the AI model would work when being implemented into a functioning system. Critical events are to be classified or ultimately detected by AI models, or *classifiers*, in a real-time system.

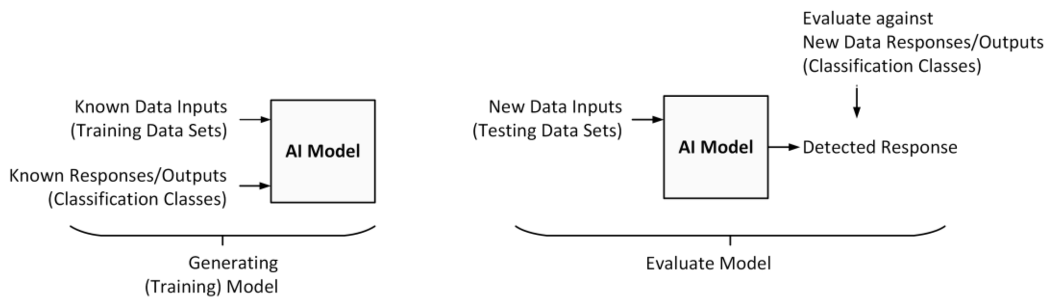


Figure 12-1. Development procedure of training and evaluation of an AI model and which data sets that are involved in the procedure

12.1 Generating Fall Detection AI Model

The rows E_{1-130} of $\boldsymbol{F}$ are randomly ordered using '*randperm()*' function in MATLAB, and $\boldsymbol{F}_{train}$ and $\boldsymbol{F}_{test}$ sets are extracted from the new order. The Neural Network and Classification Learner Toolboxes in MATLAB are utilized for generation and evaluation of the classifiers. Figure 12-2 describes the activities involved in the generation and evaluation of the classifiers using the $\boldsymbol{F}$. Table 12-1 lists the combinations of placement and numbers of wearable sensors the classifiers will be evaluated against. The purpose of this is to disclose if some combinations produce good accuracy, A, that can allow the possibility to eliminate a placement or sensor type from the experimental architecture, consisting two pairs of gyroscope and an accelerometer placed on the chest and wrist.

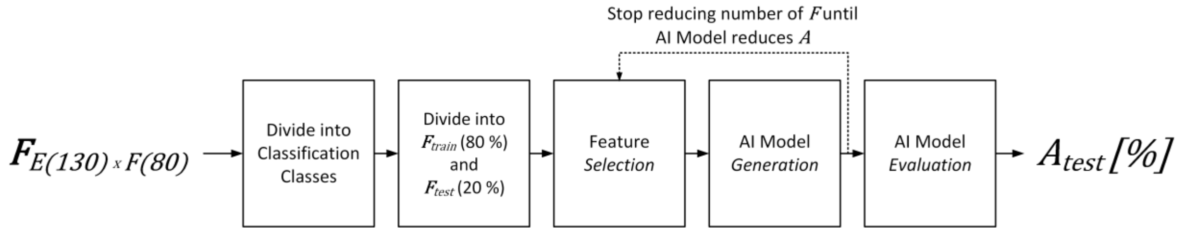


Figure 12-2. The activities involved in the generation and evaluation of AI models for fall detection. The feature sets are divided into classification classes (classes of responses), E_{fall} and $E_{non-fall}$. $\boldsymbol{F}$ is divided randomly into $\boldsymbol{F}_{train}$ and $\boldsymbol{F}_{test}$ sets despite classification classes. Number of F from the wearable sensors $\boldsymbol{F}$ are selected to reduce the input data sets based on feature selection of the $\boldsymbol{F}_{train}$ only. The AI models are so generated based on the $\boldsymbol{F}_{train}$ and evaluated using $\boldsymbol{F}_{test}$. Lastly, the accuracy, A_{test} produced by the AI Models based on the various combinations of sensor placement and number of sensors are compared to evaluate which model provided the best result

Table 12-1. Combination of sensor placement and number of sensors to be evaluated using the AI models

Combination No.	Evaluation	Placement on Body	Number of Sensors
1	Placement	Chest and Wrist	Gyroscope and Accelerometer
2		Chest	Gyroscope and Accelerometer
3		Wrist	Gyroscope and Accelerometer
4	Sensor	Chest and Wrist	Gyroscope
5		Chest and Wrist	Accelerometer

12.1.1 Feature Selection Procedure

All the features of $\boldsymbol{F}_{train}$ of the different combinations listed in Table 12-1 is ranked by importance using the *'ReliefF'* [73] feature filtering algorithm in the feature selection procedure. The columns F of $\boldsymbol{F}_{train}$ and $\boldsymbol{F}_{test}$ is so rearranged based on this rank list from highest to lowest importance. The minimum or optimal feature subset left to obtain satisfactory classification are found by training the SVM and kNN classifiers in classification learner in MATLAB and removing lower ranked Fone after one and at the same time updating the classifiers until accuracies, A_{train}, decrease. When the A_{train} has decreased, the last F removed are then added to the feature subset. Further, the SVM and kNN classifiers are exported for evaluation by running $\boldsymbol{F}_{test}$ through the classifiers. The optimal feature subset is also evaluated when generating an ANN model based on all E_{1-130}. This is because the Neural Network Toolbox in MATLAB do the separation of $\boldsymbol{F}_{train}$, $\boldsymbol{F}_{test}$ and $\boldsymbol{F}_{validation}$ itself. Figure 12-3 is a flow diagram that shows the steps involved in the feature selection procedure graphically.

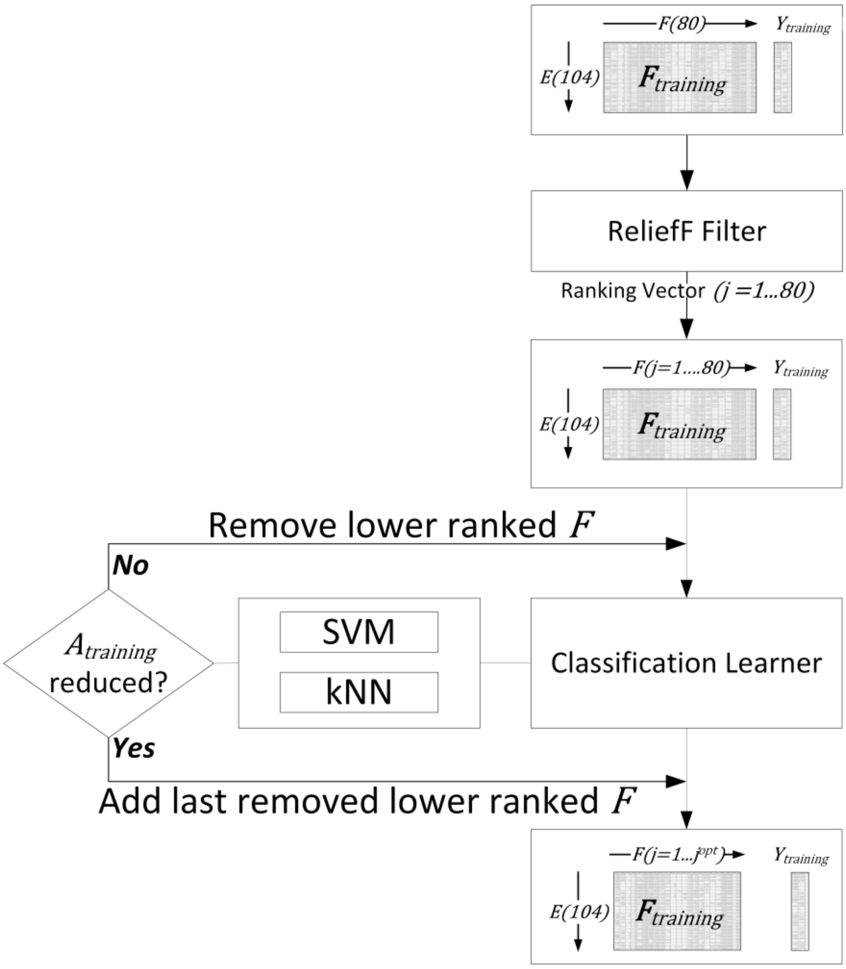


Figure 12-3. Flow diagram showing the steps involved in the Feature Selection Procedure. The optimal features $F(j \ldots j^{opt})$ for each evaluation scenario is obtained by always evaluating the accuracy of SVM and kNN models in Classification Learner until the lowest number of features possible j^{opt} for optimal performance of the classifiers is obtained

ReliefF Filter Algorithm

Using statistical measures, filter feature selection methods assign ranks to each feature of a data sets [74] . The ReliefF filter algorithm [73] is applied to all $\boldsymbol{F}_{train}$, which outputs two vectors, rank and weight. The order of the rank vector represents the most important to the least important F, the first index of the rank vector being the most important feature.

The ReliefF algorithm takes three inputs; the feature set $\boldsymbol{F}_{train}$, responses Y_{train}, and a K-parameter, decided through trial and error. Too low k, for example $k = 1$, will provide noisy and unreliable ranks while high k, around number of observations will fail to find the important F. In the feature selection during the present work, k was set to 50. The evaluation behind this was by comparing the ranks of k equal to 10, 50 and 100 for some $\boldsymbol{F}_{train}$ sets, $k = 50$ included many of the same ranks for k equal to 10 and 100.

ReliefF ranks the F by discovering strong conditional dependencies between them by evaluating the nearest hit and nearest miss of the kNN's per classification class of the specific attribute [73, 75].

12.1.2 Classification Algorithms for Generating Fall Detecting AI Models

ANN, SVM and kNN algorithms are used to create AI models or classifiers. The following sections describes these algorithms in more details.

The ANN Algorithm

An ANN consists of an input layer, one or more hidden layers and an output layer, were each layer consists of one or more neurons, as illustrated in Figure 12-4. The layers are connected through lines between the neurons, which indicates the flow of information through the network layers. The ANN classifier is generated by updating a weight value W in each neuron and comparing the output against an updated error e during the training process, zero e demonstrates that the network structure with its updated weights managed to classify the given inputs properly.

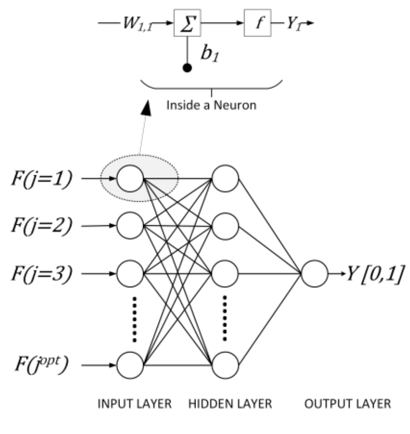


Figure 12-4. Illustration of the ANN structure showing the optimal feature subset $F(j = 1..j^{opt})$, in the input layer, hidden neurons in the hidden layer and output neurons in the output layer. The illustration also shows the elements involved in each neuron where weights W from all neuron inputs are summed up and the neuron output y is determined by applying an activation function $f(W + b)$

The output of a neuron Y is determined by equation (12-1) :

$$Y = f(W + b) \tag{12-1}$$

Where f is an activitation function, W is the weight from the specific input $F(j)$ and b is the bias. The output Y_1 of the output of the neuron transferred to next neuron or is the determined output Y.

SVM Algorithm

SVM can be applied as a classifier when having two classification classes, as fall and non-fall. A SVM creates the best suited hyperplane to separate the two classes [76]. The hyperplane best suited is the one with the largest margin, maximal width of the *slab* parallel to the hyperplane that does not have interior data points, between the two classes as shown in Figure 12-5. The data points closest to the hyperplane are called support vectors and are the *slab* boundary [76]. Figure 12-5 shows the separating hyperplane, the margin between the slab (--), the support vectors, and the data points of the two classes (+) and (-).

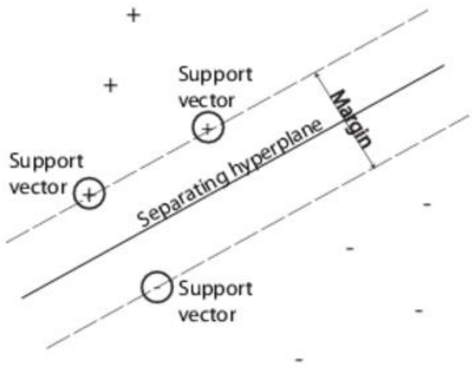


Figure 12-5. Illustration of the SVM separating hyperplane, boundary between the slab , support vectors and the data points of the two classes (+) and (-) [76]

Considering training data points (vectors) E_l along the classification class vector Y_n. For a dimension d, the $F_n \in \mathbb{R}^d$ and Y_n $[-1,1]$. The equation of a hyperplane is described by (12-2) [76]:

$$f(F) = F'\beta + b \tag{12-2}$$

Where $\beta = \mathbb{R}^d$ is the normal vector to the hyperplane and b is a real number.

The problem that defines the best separating hyperplane is to find the value of β and b that minimize $\|\beta\|$ (Euclidean norm) such that for all data points (F_n, Y_n):

$$Y_n f(F_n) \geq 1 \tag{12-3}$$

Where the support vectors are the F_l on the boundary defined by (12-4):

$$Y_n f(F_n) = 0 \tag{12-4}$$

kNN Algorithm

The kNN algorithm is a distance based classifier, that classify classes into groups of their neighbouring data points. The kNN algorithm classifies a new data point E_l based on the eucledan distance of the closest kNN's. If most of the kNN belongs to a class, the new data point will be assigned or classified into this class. kNN is then very sensitive to poor features and outliers, so a proper outlier removal and feature selection procedure before generating the kNN classifier is key [77]. Figure 2-7 Illustrates how a new data point are classified using the Euclidean distance of 5-NN. The Euclidean distance is defined as the distance between two points, input $\boldsymbol{E}(F(j = 1 \dots j^{opt})$ and the kNN's of the $\boldsymbol{Y}$ classes in feature space. Figure 12-6 shows how a new point E is placed in feature space and how Euclidean distance of the 5-NN are calculated and the new data point is then classified as the $Y = 1$ category.

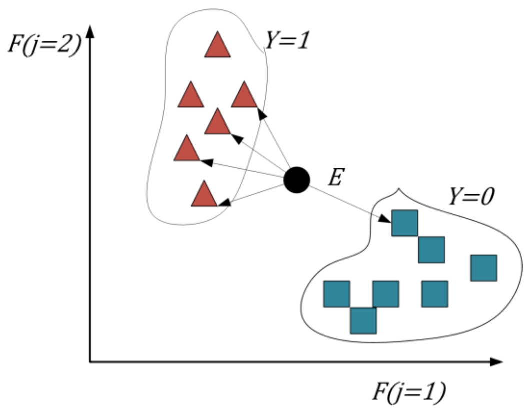


Figure 12-6. Showing a new point E to be classified by calculating the Euclidean distance of the k=5 NN of the classification classes. 4 of the NN belongs to Y=1 (fall) category, so the new point E will be classified as a fall

12.1.3 Toolboxes for Generation and Evaluation of AI Models for Fall Detection

Neural Networks and Classification Learner toolboxes are included in the MATLAB software and are applied when generating and evaluating the AI models. Classification Learner can train several types of classifiers, which makes it easy to choose the classifiers suited for the problem at hand. The following sections explains the steps involved in the Neural Network and Classification Learner toolboxes. Figure 12-7 illustrates that the $\boldsymbol{F}_{training}^{opt}$is used to generate the AI Models in the toolboxes, and the $\boldsymbol{F}_{test}^{opt}$ is used to evaluate the AI model performance by studying the accuracies A_{test}.

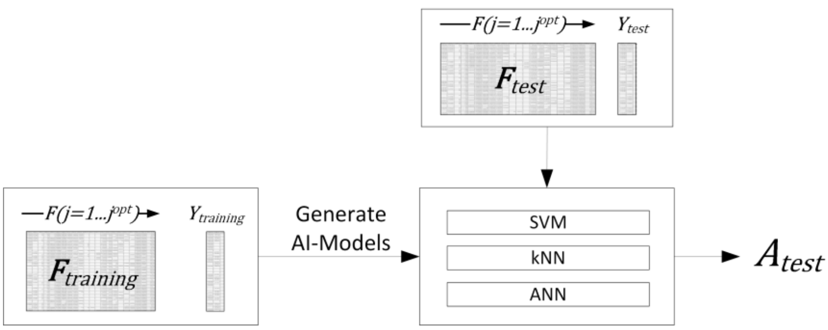


Figure 12-7. Graphical overview showing how the $\boldsymbol{F}_{training}^{opt}$ is used for generating the SVM, kNN and ANN AI models, and $\boldsymbol{F}_{test}^{opt}$ is used for evaluating the models by studying the accuracies A_{test}

Neural Network Toolbox

$\boldsymbol{F^{opt}}$ are fed into the *Neural Network Pattern Recognition tool* in the *Neural Network Toolbox*, which classifies inputs into a set of classification classes (denominated *targets* in the toolbox). $\boldsymbol{F}$ are divided into $\boldsymbol{F}_{train}$ (80 %), $\boldsymbol{F}_{test}$ (10 %) and validation $\boldsymbol{F}_{validation}$ (10 %) in the toolbox. Each ANN classifier is generated using a 5-neurons in the hidden layer. The ANNs are trained using the scaled conjugate gradient backpropagation. Figure 12-8 shows the result window in the Toolbox displaying the Cross Entropy CE and error e of the trained network.

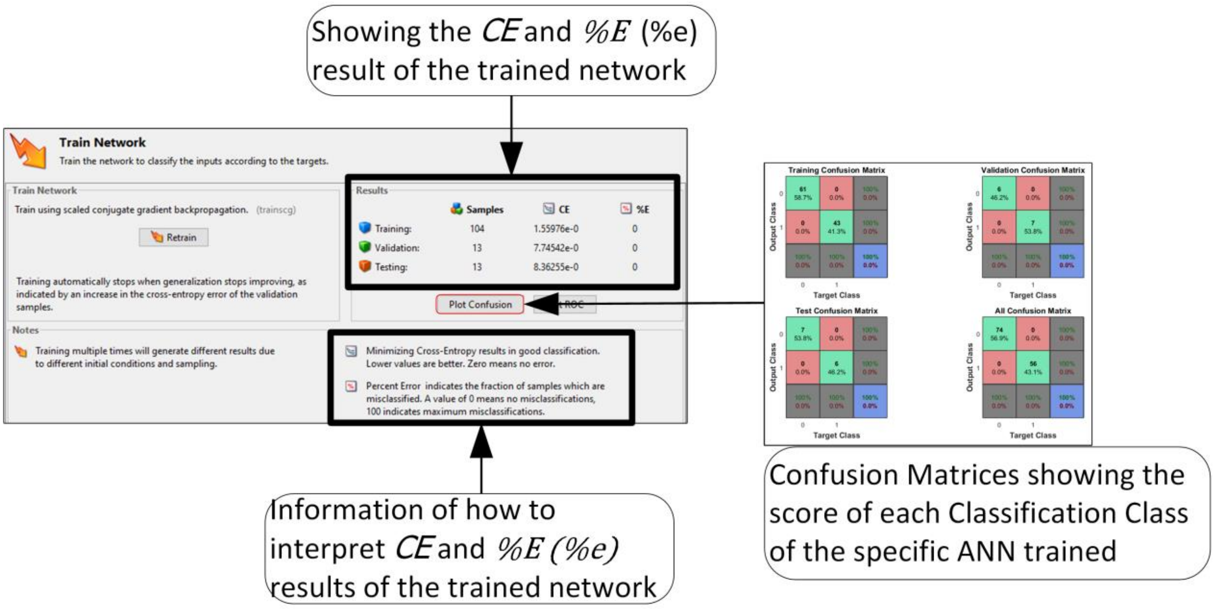


Figure 12-8. Evaluating the ANN performance in this window of the pattern recognition app. CE means (minimizing cross entropy), %E means fraction of samples ($\boldsymbol{E}(F(j = 1 \dots j^{opt}))$ which are misclassified. The figure stems from the training an ANN based on the combination 1 described in Table 12-1. A Confusion Matrix is also displayed in the figure, showing the score of each Classification Class during training, testing and validation of the specific ANN

Classification Learner Toolbox

The *Classification Learner Toolbox* calculates the A of a vast number of different classifiers to compare them. The $\boldsymbol{F}$ of the ranked feature subsets determined using *ReliefF* is imported from the MATLAB workspace. In the toolbox, the F are defined as predictors, and the last column in feature data being the classification classes $\boldsymbol{Y}(Y_{fall} = 1, Y_{non-fall} = 0)$, is defined as responses. Before training the classifiers, 5-folds Cross-Validation is chosen as validation method, to protect protects against classifier overfitting. 5- folds means that the training predictors are divided into 5 folds, where each of the folds represents a set of training data used during 5 iteration of training the specific classifier. Number of folds must be chosen based on size of the data set in at hand. Figure 12-9 shows a screen dump of the training window in the Classification Learner Toolbox. Features can be removed and added, different classifiers can be trained at the same time, scatter plots can help distinguish features that separates classes well. A trained classifier can be imported to workspace from the toolbox for evaluation using a training set.

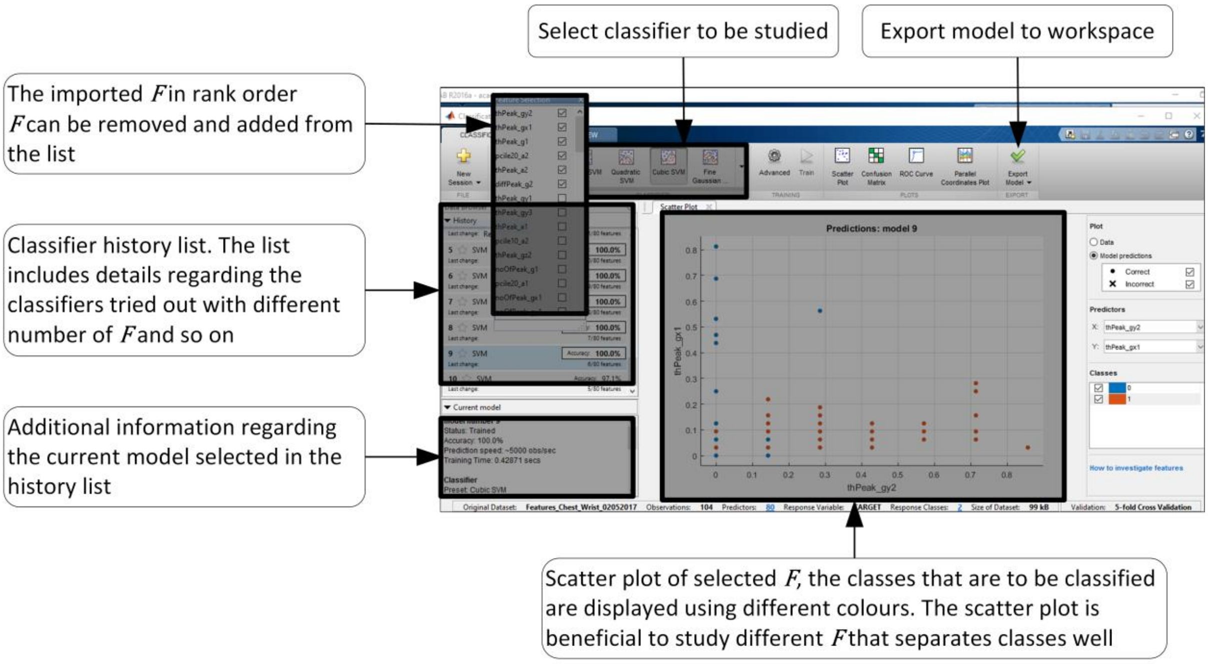


Figure 12-9. Showing the training window in the Classification Learner toolbox. The window displays a good overview of history list all trained classifiers and their training accuracies A, scatter plots of the F that are beneficial for studying F that separates classes well

12.2 Generating Geofence Threshold AI Model

The geofence threshold AI model for classification if a user has not come home after a pre-set time is simulated. Time dependent variable vectors, door simulation opening, δ position measurement based on E_{pos} signals, are used when generating and testing the geofence threshold AI Model.

The model inputs for simulation, which in a real-world situation will be established (inserted into an application) by the user/actor of this system is:

- $(\varphi, \lambda)^{\text{home}}$: $10.438°\ O$, $59.819°\ N$
- $\emptyset_{thresh}$: $0.001°$
- $t^{pre-set}$: 120 min ($2\ h$)

The position measurements are recorded every minute as described in 8.2. The generated simulation vectors can be studied in the plots of Figure 12-10, where the user steps out of the living area after $4\ min$ into the simulation. The code for simulating the geofence threshold model can be studied in detail in Appendix B.5.

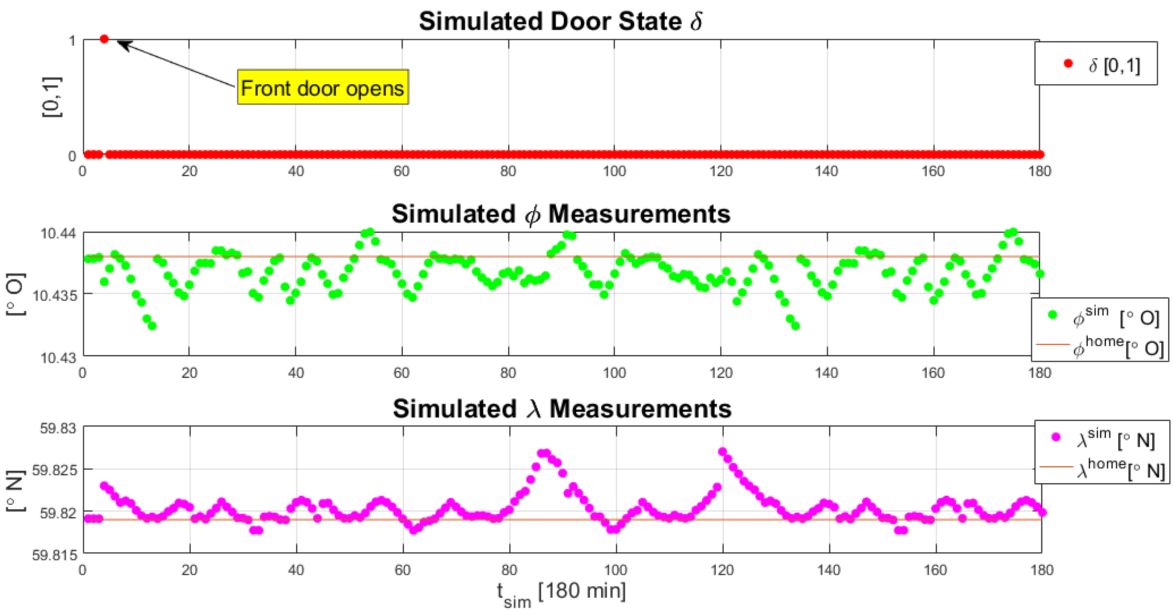


Figure 12-10. Plots of variables δ, φ and λ during a $180\ min$ ($3\ h$) generated simulation. The home coordinates $(\varphi, \lambda)^{home}$ are also plotted as a reference. These vectors will be the inputs to the model which will detect if the user has not come home after a pre-set time, $t^{pre-set}$, $2\ h$, for simulation purposes

13 Common Features in the AI Models used with Possible Extension to a Smartphone and a Smartwatch

This chapter presents the common features granted a high score of the ranking feature filter algorithm ReliefF (see chapter 12). This chapter will present a proposed system architecture and working principle of the sensor types and sensor fusion strategy used in the experimental procedure conducted in the present work including AI models for detecting critical events. This system architecture can be applied in an unobtrusive system for monitoring and alerting of ELIAH.

13.1 Common Features in Comparing Different Placement and Sensors based on the Wearable Feature Data Set

Table 13-1 lists the optimal feature subsets $\boldsymbol{F}(j = 1..j^{opt})$ obtained for each evaluation, combination of placement and sensor type, described in Table 12-1. The plot in Figure 13-1 shows the ranks compared to the weights given by feature filter ReliefF of the features subset $F(j = 1 \dots j^{opt})$ for all the evaluation combinations 1-5. The peak features (red) is clearly the feature type that involves the most important features, the percentile features (green) is the second most important features. Wavelet features (pink) are among the highest ranked features feature subset of combination number 2 (chest) and 3 (wrist). The cross-correlation features are only contributing the feature subset of combination 4 (gyroscope), but are not among the features that are most important in this combination either. The peak features $F_{noOfPeaks}$ are not included in any of the optimal feature subsets listed in Table 13-1, and are not significant in contributing to separate fall from non-fall events. Considering the feature subset of each combination, peak-features may be connected to gyroscope, or ω measurements since almost all feautures in feature subset of combination 1 (wrist and chest) and combination 4 (gyroscope) contains mostly peak-gyroscope combined features.

Table 13-1. Overview of the optimal feature subsets $F(j = 1 \dots j^{opt})$ of the combination listed in Table 12-1

Combination No.	**No. of $\boldsymbol{F(j^{opt})}$**	**Optimal Feature Subset $\boldsymbol{F}(\mathbf{j = 1 \dots j^{opt}})$**
1	6	$F_{thPeak}(\omega_y^{wrist})$, $F_{thPeak}(\omega_x^{chest})$, $F_{thPeak}(\omega^{chest})$, $F_p^{20}(a^{wrist})$, $F_{thPeak}(a^{wrist})$, $F_{diffPeak}(\omega^{wrist})$,
2	5	$F_c(a^{chest})$, $F_p^{20}(a^{chest})$, $F_p^{10}(a^{chest})$, $F_p^{35}(a^{chest})$, $F_{diffPeak}(\omega^{chest})$
3	13	$F_p^{20}(a^{wrist})$, $F_p^{10}(a^{wrist})$, $F_c(a^{wrist})$, $F_{thPeak}(\omega_y^{wrist})$, $F_{diffPeak}(\omega^{wrist})$, $F_{thPeak}(a^{wrist})$, $F_{diffPeak}(a^{wrist})$, $F_{thPeak}(\omega_z^{wrist})$, $F_{thPeak}(\omega_x^{wrist})$, $F_{thPeak}(\omega^{wrist})$, $F_p^{35}(a^{wrist})$, $F_{xcorr}(a_{xy}^{wrist})$, $F_p^{10}(\omega^{wrist})$
4	9	$F_{thPeak}(\omega_y^{wrist})$, $F_{thPeak}(\omega_z^{wrist})$, $F_{diffPeak}(\omega^{wrist})$, $F_{diffPeak}(\omega^{wrist})$, $F_{diffPeak}(\omega^{chest})$, $F_{diffPeak}(\omega_x^{chest})$, $F_p^{95}(\omega^{chest})$, $F_{thPeak}(\omega_y^{chest})$, $F_{thPeak}(\omega_z^{chest})$
5	5	$F_{thPeak}(a^{wrist})$, $F_p^{20}(a^{wrist})$, $F_p^{10}(a^{wrist})$, $F_c(a^{chest})$, $F_{thPeak}(a^{chest})$

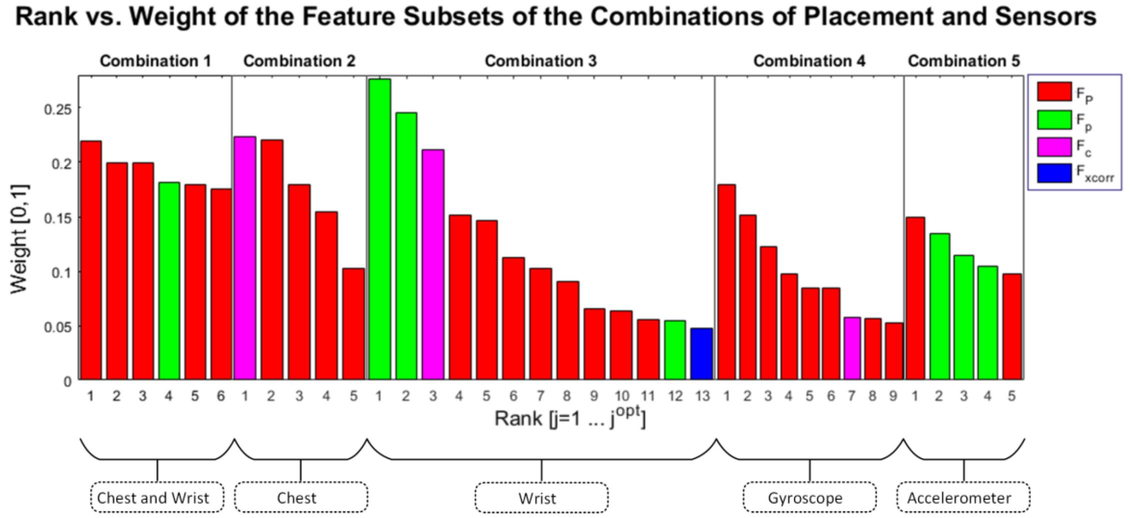


Figure 13-1. Showing the optimal ranked feature subset $F(j = 1 \dots j^{opt})$ for each combination of placement and sensors described in Table 12-1. The most common or important features are F_P(red) which is highly involved in approximately all feature subsets. F_p(green) are considered the second most important feature type. F_c have received a high score in combination 2 (chest) and combination 3 (wrist). F_{xcorr} is considered the least important feature type, which is only involved in combination 3 (wrist) and are among the lower ranked feature in the subset

13.2 Proposed Extension to a Smartphone and Smartwatch

Using a bluetooth compatible chest strap including gyroscope and accelerometer, a wireless compatible door sensor or a cloud service (or IoT platform) that both transmits sensor data to a smartwatch located on the left wrist. The smartwatch includes the application software including the fall detection and geofence threshold models for monitoring and alerting relevant actors in case of critical events. The proposed system structure including sensor fusion strategies can be studied in detail in Figure 13-2, where the relevant actors are responsible for software setup etc. The software application can utilize windowing techniques and handle each 10 s time intervals of the fall data, ω and a, every 1 s in separate software threads or processes.

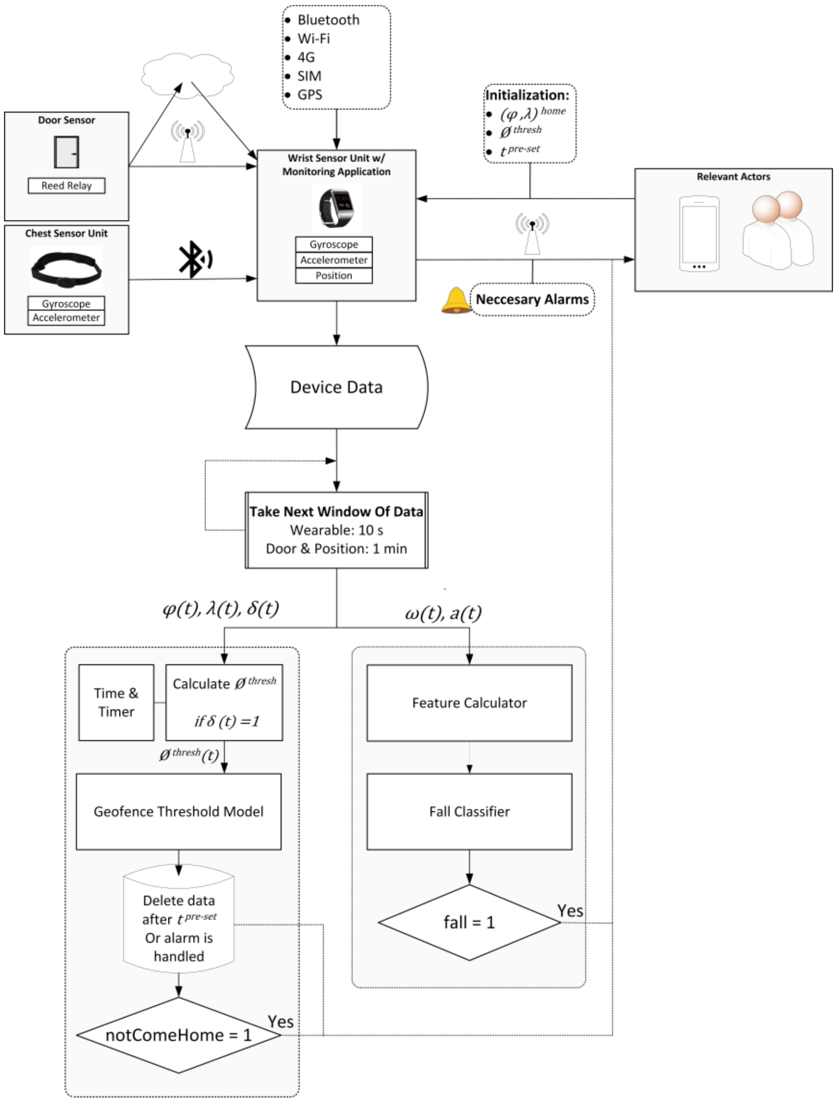


Figure 13-2. Proposed system structure of the system including the system software on a smartwatch where the model algorithms for detecting critical events are implemented on. The relevant actors have a connected application installed in their smartphones and would receive notification with the possibility to contact the user through the smartwatch that includes a SIM card. If the relevant actors do not get contact with the user, position data would be shared to localize the person in the case of not coming home after pre-set time, or would send an emergency unit to the user's home or location in the case of a fall. Smartwatch figure [78] chest strap figure [79] bluetooth icon [60]

14 Results

This chapter lists the results from the task description. The purpose of the present work is to categorize different scenarios, identifying sensors and their proper placement on the body and ambient environment and developing data fusion strategies.

14.1 Resulted AI Model Performances for Fall-Detection using Wearable Sensors

Table 14-1 lists resulting A_{test} of ANN, SVM and kNN AI models using different features F extracted from both gyroscope and accelerometer sensor measurement of different body placements. Table 14-2 lists resulting A_{test} of ANN, SVM and kNN AI models using different features F extracted from only gyroscope or only accelerometer sensor measurements, respectively.

Table 14-1. Performance of the fall detection AI models based on the resulting values of A_{test} *using the optimal feature subset* F_{test}^{opt} *for each placement on the body including both gyroscope and accelerometer sensors*

All Sensors			
AI Model	**Chest and Wrist**	**Chest**	**Wrist**
ANN	100 %	96 %	100 %
SVM	100 %	96 %	96 %
kNN	100 %	92.3 %	100 %

Table 14-2. Performance of the fall detection AI models based on the resulting values of A_{test} *using the optimal feature subset* F_{test}^{opt} *for each sensor type, on both chest and wrist placement*

All Placements		
AI Model	**Gyroscope**	**Accelerometer**
ANN	92 %	92 %
SVM	96 %	96 %
kNN	92 %	100 %

The optimal AI models generated in Classification Learner Toolbox are extracted to the MATLAB workspace, where the $\boldsymbol{F}_{test}^{opt}$ is fed into the models and its performance is evaluated where misclassified inputs E can be studied. The Neural Network Toolbox does not have this functionality, as the training and validation data set must be included when training a network. The misclassified E's of the F_{test}^{opt} are listed in Table 14-3 from SVM and kNN model evaluation. It is expected that the majority of the misclassified input categories involve E_{fall} and $E_{transition}$, since most of the input samples is extracted from these measurements.

Table 14-3. Listing the misclassified Es when evaluating the SVM and kNN algorithms

Description of misclassified inputs *E*		
	SVM	**kNN**
Combination 2: Chest	E_{101} of category $E_{transition}$. Stand in front of sofa, sit, then lie, slow transition	E_{71} of category $E_{transition}$. Sit on kitchen chair to stand (SiSt), fast transition
Combination 3: Wrist	E_{55} of category E_{fall}. Walking, fall forward, land on right side in lying position, movement after fall	-
Combination 4: Gyroscope	E_{105} of category $E_{transition}$. Stand in front of sofa, sit, then lie, fast transition	• E_{105} of category $E_{transition}$. Stand in front of sofa, sit, then lie, fast transition • E_{22} of category E_{fall}. Forward fall, from standing, knee flexion to seating on ground (matrass) landing with left arm
Combination 5: Accelerometer	E_{13} of category E_{fall}. Backward fall from seating on kitchen chair to lying positon landing with right arm	

14.2 Simulation Results of the Geofence Threshold AI Model using Ambient and Smartphone DCUs

The geofence threshold AI model for detection if a user has not come home after $t^{pre-set}$, is tested with simulated test data that includes the setup described in chapter 12.2.

Figure 14-1 shows the various input and output parameters of the geofence threshold model during a 180 min (3 h) simulated time sequence. The user can be detected inside the geofence boundary after the front door is detected open. But since the front door has not been opened again after this, and not inside the $t^{pre-set}$, will indicate that the user not come home for some reason, which could be critical. The simulation code of the geofence threshold AI model can be studied in detail in Appendix B.5, where fictive measured parameters are generated to verify the AI model.

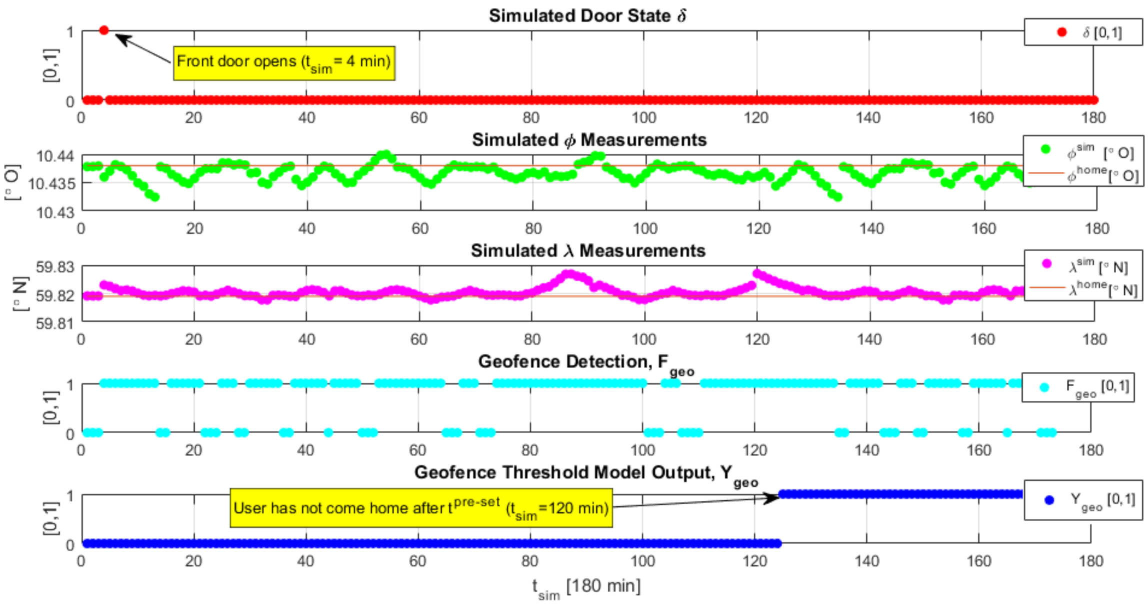


Figure 14-1. The five subplots show the parameters of the geofence threshold AI model on the same timeline (3 hours). The $t^{pre-set}$ before the geofence threshold model activates is 2 hours, for simulation purposes. Even if the user may be inside the geofence as shown in the 'Geofence Detection, F_{geo}' plot, the front door has not yet been opened, which indicate that the user has not entered the living area for some reason, which could be critical

15 Discussion

In the present work, several AI models are generated and evaluated using various wearable and ambient sensors, as well as position data recorded by a smartphone, for detection of possible critical events of ELIAH. Two critical events were adressed to be detected by two AI models; (1) fall of the user and (2) if the user has not come home after a pre-set time. Different AI algorithms were evaluated to find the one with the best performance. The AI models were evaluated using test sets and generated simulation test sets to distinguish proper (1) number- and (2) placement- of sensors using the least possible number of features as feature calculation in a real-time system can be time consuming if too many features are to be calculated in each time window.

Considering the evaluation of the fall detection AI models. The AI models generated using both gyroscope and accelerometer sensors placed on the chest and left wrist all showed the best accuracies of 100 % in distinguishing fall from non-fall events. The high accuracies are most likely because the test sets were relatively small (26 inputs) and the experiments were only performed by one subject. Performing experimental procedures, the motion can become very distinct, compared to a real-world fall or non-fall event, were more complex motions may be involved. Possible improvement and more certain AI models may have more potential if generating models based on a larger data set including several subjects of different heights and physiques as carried out in related research [9, 22]. Other time intervals of the input signals were not tried out and could possibly contribute to other outcomes, especially since some of the features extracted are reflected by the relatively long time interval of 10 s. But a fall often occurs in a couple of seconds. The experiment categories that endured most misclassification when evaluating SVM and kNN AI models was fall and transition, which were expected, as the majority of the inputs was extracted from these experiment categories. The misclassified fall inputs involved using sofa as landing ground and landing on the right side, where the sensors were placed left wrist and possibly did not involve sufficient motion for the models to proper identify. Many of the transition misclassified inputs involved reverse motion, as lying to standing (LySt) and sitting to standing (SiSt) compared to a fall. All the fall detection models obtained an overall accuracy of >92 %. Promising results were obtained in the evaluation combination using only inputs based on left wrist measurements. It would be beneficial to only use sensors placed on the wrist, as this would involve a more comfortable wearable system and the user would experience less stigmatisation which is addressed as an important ethical aspect of AT. Implementing the fall detection model only using wrist-based sensor measurements would enable the user to more comfortable by not having to wear a chest band constantly. Using only one pair of gyroscope and accelerometer sensors together on the wrist will greatly simplify the recording protocol and preparing features for the fall detection model, because a possible software application does not have to handle data from the chest band transmitted by for example bluetooth. However, the wrist-based AI model obtained the best accuracy but involving the highest amount of F^{opt}, compared to the other evaluation combinations. The common features with highest rank using the feature selection algorithm ReliefF was peak features, primarily the F_{thPeak} and $F_{diffPeak}$. Percentile features F_p, mainly the lower and highest ranged $F_p^{10-35,95}$, also showed promising results in the evaluation involving only wrist-based features. The SVM and kNN was the AI algorithms that gave the overall best accuracies in detecting fall from non-fall events, even though ANN classifiers demonstrated very promising results as well.

The geofence threshold-based model is simple, and requires initialization parameters, home coordinates $(\varphi, \lambda)^{home}$, size of geofence around the home $\emptyset_{thresh}$, and length of the pre-set time,$t^{pre-set}$ before the model will alert relevant actors that the user has not come home. The geofence threshold model has a simple structure, requires minimum feature preparing, and could be implemented into all smartphones and smartwatches in form of a software application, that is normally equipped with positioning functionality to extract these values. Many smartwatches are also compatible with mobile network communication, which is beneficial as network determined position has better performance when the device is located inside buildings as described in chapter 7.3.

The proposed system architecture is a basic solution that reaches the user group, enabling elderly to live longer at home independently while still being provided necessary help and feel safe. The solution, described in chapter 13.2, which includes the two AI models for detection of fall and if the user has not come home after a pre-set time. AI models that detects essential critical events, is manageable for relatives or care services to unobtrusively monitor elderly with mild physical and/or cognitive disabilities. This solution could be a category of AT that enables elderly to becomfortable with using technology and having relevant actors receiving just the most necessary alarms. Such initiative showed good result in Scotland where all elderly over 60 years old were handed technology packets consisting of alarms and other supporting technology to prepare the early on to accept this category of technology before the need emerges.

16 Conclusions and Future Work

Final Conclusions

The motivation behind the study in the present work was to apply AI techniques for detecting critical events for an unobtrusive monitoring system of elderly, enabling them to live longer in their home independently, while still feeling safe and being provided necessary help and care.

The goal was to address critical events to be monitored by a comprehensive background study, create experimental procedures for recording data sets that would form the basis of generating and evaluating AI models for detecting the critical events. The main goal was to identify proper placement and number of sensors to be used in such a system. In this report, two critical events were to be detected; (1) fall and (2) if the user had not come home after a pre-set time.

Several AI models was generated an evaluated to detect fall from non-fall events based on different evaluation combinations involving placement and number of wearable sensors. All the fall detection models obtained an overall accuracy of >92 %. All the AI models managed to detect fall from non-fall events with accuracy of 100 % in the evaluation combination using two pairs of sensors, consisting of gyroscopes and accelerometers, placed on the chest and the left wrist. Other evaluation combination also achieved 100 % accuracy, the most applicable one, ANN or kNN AI model only based on one pair of gyroscope and accelerometer sensors placed on the left wrist, which could easily be implemented into a software application of a smartwatch. Using only wrist-based sensors, is also the most comfortable and less stigmatizing solution for a body worn sensor system which must be worn at all times. Also, using only sensors from one placement will greatly simplify the recording protocol for preparing feature for the fall detection model as less wireless data transmission is needed.

The geofence threshold model for detecting if the user has been outside the home environment for over a pre-set time period also works well when establishing a radius Ø_{thresh} for the model. This model is a simple algorithm that can easily utilize GPS and other position estimation features commonly included in smartwatches' functionality together with a door sensor fixed on the front door. A proposed geofence radius is used in the evaluation of the model.

A proposed architecture for extending the models for detecting the critical events and be applied on a mobile platform, preferably alerting relevant actors, like relatives or care personnel in case of emergency, is also elaborated. The proposed system involves primarily the elderly user wearing a smartwatch including the software that handles sensor data, feature calculation and detecting of critical events as well as alerting procedures. The smartwatch software application will alert relevant actors, on a smartphone including another software application that receives the alarms if critical events occurs.

Future Work

Finally, this study during the present work has some limitations. First, it would be advisable to record more experiments from several subjects as height and physique can provide other accuracies of the fall detection AI model. Further evaluation using windowing-techniques and trying different time windows of the signal trains before extracting features could also give other results as only one time interval is handled in this study. Issues regarding power management of the wireless sensor units are not elaborated in the present work, but is of importance to investigate in case of working with this particular solution further or other related work.

Two pulse sensors are installed on the experimental wearable DCUs, and could be of interest to use in combination with detecting if the user has not woken up or not after a pre-set time during the day (detected lying for too long period).

References

[1] (12.05.2017). *Assistive technology*. Available: https://en.wikipedia.org/wiki/Assistive_technology

[2] (12.05.2017). *Feature (machine learning)*. Available: https://en.wikipedia.org/wiki/Feature_(machine_learning)

[3] *unobtrusive*. Available: https://www.merriam-webster.com/dictionary/unobtrusive

[4] S. Patel, H. Park, P. Bonato, L. Chan, and M. Rodgers, "A review of wearable sensors and systems with application in rehabilitation," *Journal of NeuroEngineering and Rehabilitation,* vol. 9, 2012.

[5] J. B. Melting, "Andre gevinstrealiseringsrapport med anbefalinger. Nasjonalt velferdsteknologiprogram," Helsedirektoratet01.2017 2017.

[6] B. Nordtug, H. M. Aasan, S. Myrum, Gunn Eva, and S. f. Omsorgsforskning, "Implementering av Velferdsteknologi - En kvalitativ studie: Hvilken nytte og hvilke utfordringer erfarer ansatte i kommunal helsetjeneste?," 2015.

[7] A. Agah, *Medical Applications of Artificial Intelligence*: CRC Press, 2013.

[8] Teknologirådet. (2009, 15.03.2016). *Fremtidens alderdom og ny teknologi* Available: https://teknologiradet.no/wp-content/uploads/sites/19/2013/08/Rapport-Fremtidens-alderdom-og-ny-teknologi.pdf

[9] X. Garcia-Massó, P. Sierra-Añó, L. Gonzales, Y. Ye-Lin, G. Prats-Boluda, and J. Garcia-Casado, "Identifying physical activity type in manual wheelchair users with spinal cord injury by means of accelerometers," *Spinal Cord,* vol. 53, pp. 772-777, 2015.

[10] A. Dementyev, S. Hodges, S. Taylor, and J. Smith, "Power Consumption Analysis of Bluetooth Low Energy, ZigBee and ANT Sensor Nodes in a Cyclic Sleep Scenario " presented at the Wireless Symposium (IWS), Beijing, China, 2013.

[11] J.-V. Lee, Y.-D. Chuah, and K. T. H. Chieng, "Smart Elderly Home Monitoring System with an Android Phone," *International Journal of Smart Home,* vol. 7, 2013.

[12] (27.04.2017). *Smartwatch*. Available: https://en.wikipedia.org/wiki/Smartwatch

[13] Telia. (2017, 20.04.2017). *Slik kan norske Contact varsle fall*. Available: https://telia.no/magasinet/slik-kan-norske-contact-varsle-fall

[14] N. Siddique and H. Adeli, *COMPUTATIONAL INTELLIGENCE - SYNERGIES OF FUZZY LOGIC, NEURAL NETWORKS AND EVOLUTIONARY COMPUTING*. United Kingdom: John Wiley & Sons, Ltd, 2013.

[15] J. Boyle, M. Karunanithi, and T. Wark, "Quantifying Functional Mobility Progress for Chronic Disease Management," presented at the EMBS '06. 28th Annual International Conference of the IEEE, 2006.

[16] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, D. Howard, K. Meijer, and R. Crompton, "Activity identification using body-mounted sensors—a review of classification techniques," 2009.

[17] K. Zhang, M. Sun, D. K. Lesterc, F. X. Pi-Sunyer, C. N. Boozer, and R. W. Longmand, "Assessment of human locomotion by using an insole measurement system and artificial neural networks," *Journal of Biomechanics,* 2005.

[18] J. Y. G. Stephen J. Preece, Laurence P J Kenney, Dave Howard, Kenneth Meijer, Robin Crompton,, "Activity identification using body-mounted sensors—a review of classification techniques," 2009.

[19] H. E. Jian and H. U. Chen, "A Portable Fall Detection and Alerting System Based on k-NN Algorithm and Remote Medicine," *China Communications,* vol. 12, 2015.

[20] Q. Ni, A. B. G. Hernando, and I. P. d. l. Cruz, "The Elderly's Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development," *Sensors 2015,* 22.04.2015 2015.

[21] M. Røhne, I. Svagård, D. Ausen, A. B. Fossberg, I. Husebø, and T. Øverli, "Bo lenger hjemme med mobil trygghetsalarm?," SINTEF IKT, Bærum Kommune2015.

[22] D. W. Kang, J. S. Choi, J. W. Lee, S. C. Chung, S. J. Park, and G. R. Tack, "Real-time elderly activity monitoring system based on a tri-axial accelerometer," 2010.

[23] A. Krause, D. P. Siewiorek, A. Smailagic, and J. Farringdon. (2003). *Unsupervised, Dynamic Identification of Physiological and Activity Context in Wearable Computing*. Available: http://ieeexplore.ieee.org/document/1241398/

[24] B. Dolan. (2014, 22.04.2017). *iHealth unveils wearable ECG, pulse ox, BP devices*. Available: http://www.mobihealthnews.com/28547/ihealth-unveils-wearable-ecg-pulse-ox-bp-devices

[25] G. Krishnamurthy. (2014, 15.03.2017). *iHealth Launches New Wristworn Pulse Oximeter, Ambulatory Heart and Blood Pressure Monitors at CES 2014*. Available: https://www.medgadget.com/2014/01/ihealth-launches-new-wristworn-pulse-oximeter-ambulatory-heart-and-blood-pressure-monitors-at-ces-2014.html

[26] M. Kanamaru, H. Kawai, H. Kobayashi, J. Tatsuno, H. Mochiyama, and N. Kobayashi, "Networked Cellular Motion Detection System by Using Pyroelectric Infrared Sensor for Surveillance," presented at the 36th Annual Conference on IEEE Industrial Electronics Society, 2010.

[27] Telenor Objects. *Digitale trygghetsalarm og sensorer*. Available: http://www.telenorobjects.com/velferdsteknologi/sensorer/

[28] J. Thunqvist. (2017) Robothunden gir henne trygghet. *Teknisk Ukeblad*.

[29] I. Marlow. (2017, Japan's bold step. Available: http://www.theglobeandmail.com/globe-investor/retirement/retire-planning/how-japan-is-coping-with-a-rapidly-aging-population/article27259703/

[30] P. R. U.S. (01.04.2017). *PARO Therapeutic Robot*. Available: http://www.parorobots.com

[31] ÆldreForum. (2010). *Velfærdsteknologi – nye hjælpemidler i ældreplejen*.

[32] Dennis C. Søndergård. (2017). *Etikk og Velfærdsteknologi*. Available: http://www.ks.no/globalassets/nordens-velferdssenter.pdf

[33] J. Ivarson. (15.04.2017). *Velferdsteknologi i eldreomsorgen*. Available: http://telemedicineconsult.com/wp-content/uploads/2013/03/nr-11-Telenor-Objects-Gardermoen-17juni2013.pdf

[34] Bjørn Hofmann. (2010, 06.03.2017). *Etiske utfordringer med velferdsteknologi*.

[35] Statistics Norway. (2013, 26.01.2017). *Eldres bruk av helse- og omsorgstjenester*. Available: https://www.ssb.no/helse/artikler-og-publikasjoner/_attachment/125965?_ts=13f8b5b6898

[36] S. Hårstad. (25.01.2017). *BO LENGRE HJEMME «ØKT SELVHJULPENHET OG STØRRE TRYGGHET»*. Available: https://www.varnesregionen.no/fellestjenester/vrhelse/prosjekter/bolengrehjemme/Documents/Prosjektbeskrivelse%20Bo%20lengre%20hjemme.pdf

[37] Helse -og Omsorgsdepartementet. (2013). *Morgendagens omsorg. Meld. St. 29 (2012-2013)*. Available: https://www.regjeringen.no/contentassets/34c8183cc5cd43e2bd341e34e326dbd8/no/pdfs/stm201220130029000dddpdfs.pdf

[38] SSB. (1999). *Eldre i Norge*. Available: http://www.ssb.no/a/publikasjoner/pdf/sa32/sa32.pdf

[39] Kåre Hagen, "Innovasjon i Omsorg," D. s. Informasjonsforvaltning, Ed., ed. https://www.regjeringen.no, 2011.

[40] Helsedirektoratet - Avdeling Statistikk, "Helse-, omsorgs- og rehabiliteringsstatistikk - Eldres helse og bruk av kommunale helse- og omsorgstjenester," 2.2016 2016.

[41] M. Holm. *Kognitiv svikt hos eldre*. Available: https://www.nsf.no/Content/641533/kognitiv

[42] N. f. folkehelsen. (2016). *Hva er demens?* Available: http://nasjonalforeningen.no/demens/hva-er-demens/

[43] J. Kjelvik, S. M. Herbern, M. C. Kaurin, B. K. Grønnestad, and T. H. Johansen, "Diagnosestatistikk for kommunale helse- og omsorgstjenester," 2015.

[44] V. Skirbekk, B. H. Strand, and K. Tambs. (2015, 03.02.2017). *Demens*. Available: https://www.fhi.no/nettpub/hin/helse-og-sykdom/demens---folkehelserapporten-2014/

[45] A. B. Fossberg, E. J. Gottschal, D. Ausen, and M. Røhne. (2015, 07.02.2017). *Erfaringer med mobil trygghetsalarm i Bærum og Skien*. Available: https://www.drammen.kommune.no/Documents/Helse/UHT%20Buskerud/Mobil%20trygghetsalarm_erfaringer%20B%C3%A6rum%20og%20Skien.pdf

[46] Vakt og Alarm AS. (04.02.2017). *TMA Fallalarm*. Available: http://www.vaktogalarm.no/tma-fallalarm.301803.no.html

[47] SINTEF. (2016, 15.03.2017). *Trykkmålinger gjør at en nyutviklet fallsensor "ser" fall som dagens produkter ikke registrerer. Det vil øke sikkerheten for eldre som bor hjemme.* Available: http://www.sintef.no/siste-nytt/ny-sensor-vil-oke-tryggheten-for-eldre/

[48] L. G. Gravdal. (2014, 24.01.2017). *Slik kan teknologien hjelpe demente*. Available: http://forskning.no/2014/09/sann-kan-teknologien-hjelpe-demente

[49] Telenor. (15.02.2017). *Omsorgs-pionerene på Kongsvinger*. Available: https://www.telenor.no/om/samfunnsansvar/artikler/omsorg.jsp

[50] D. A. Abbott. (2015, 22.03.2017). *Ethics of passive wellbeing monitoring and focus group report*. Available: http://www.kemurisense.com/papers/Kemuri-Ethics2.pdf

[51] A. J. Wheeler and A. R. Ganji, *Introduction to engineering experimentation*, 3rd ed. ed. Boston: Pearson, 2010.

[52] *The Mechatronics Handbook*, 2 ed.: CRC Press, 2008.

[53] S. Electronics. (23.03.2017). *What is a Gyroscope*. Available: https://learn.sparkfun.com/tutorials/gyroscope

[54] A. Wawoo. (2014, 03.03.2017). *Mems gyroscope working, principle of operation of disc resonator gyroscope*. Available: https://www.slideshare.net/ankushwawoo/mems-gyroscope-working

[55] S. Electronics. (23.03.2017). *Accelerometer Basics*. Available: https://learn.sparkfun.com/tutorials/accelerometer-basics

[56] R. O'Reilly, K. Harney, A. D. Inc., and A. Khenkin. (2009, 24.03.2017). *Acceleration/Vibration. Sonic Nirvana: MEMS Accelerometers as Acoustic Pickups in Musical Instruments*. Available: http://www.sensorsmag.com/sensors/acceleration-vibration/sonic-nirvana-mems-accelerometers-acoustic-pickups-musical-i-5852

[57] A. Otero, "Operation of a reed switch," ed, 2010.

[58] (23.04.2017). *Mobile phone tracking*. Available: https://en.wikipedia.org/wiki/Mobile_phone_tracking

[59] "Geographical Coordinate System," ed.

[60] "Bluetooth icon," ed.

[61] "Dropbox icon."

[62] Arduino AG. (23.02). *Arduino Pro Mini*. Available: https://www.arduino.cc/en/Main/arduinoBoardProMini

[63] "Arduino Pro Mini," ed. Snapdeal.

[64] "HC-06 Bluetooth Module," ed.

[65] Arduino. (22.02.2017). *MPU-6050 Accelerometer + Gyro*. Available: http://playground.arduino.cc/Main/MPU-6050

[66] InvenSense Inc. (2013, 01.03.2017). *MPU-6000 and MPU-6050 Product Specification Revision 3.4*. Available: https://www.invensense.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf

[67] (27.03.2017). *I²C*. Available: https://en.wikipedia.org/wiki/I%C2%B2C

[68] EEEnthusiast, "Ep. 57 Arduino Accelerometer & Gyroscope Tutorial MPU-6050 6DOF Module," ed: Youtube, 2016.

[69] (19.03.2017). *GPS Logger for Android*. Available: https://play.google.com/store/apps/details?id=com.mendhak.gpslogger&hl=no

[70] K. Devleker. (15.04.2017). *Understanding Wavelets, Part 1: What Are Wavelets*. Available: https://se.mathworks.com/videos/understanding-wavelets-part-1-what-are-wavelets-121279.html

[71] MathWorks Documentation. (16.04.2017). *wavedec*. Available: https://se.mathworks.com/help/wavelet/ref/wavedec.html

[72] S. Raschka. (2014, 22.04.2017). *About Feature Scaling and Normalization*. Available: http://sebastianraschka.com/Articles/2014_about_feature_scaling.html

[73] Mathworks Documentation, "relieff."

[74] J. Brownlee. (2014, 25.04.2017). *An Introduction to Feature Selection*. Available: http://machinelearningmastery.com/an-introduction-to-feature-selection/

[75] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, "Overcoming the Myopia of Inductive Learning Algorithms with RELIEFF," *Applied Intelligence*, vol. 7, 1997.

[76] Mathworks documentation. (05.04.16). *Support Vector Machines for Binary Classification*. Available: http://se.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html?requestedDomain=www.mathworks.com

[77] Wikipedia. (05.05.2017). *k-nearest neighbors algorithm*. Available: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[78] "Samsung Galaxy Gear Smartwatch," ed. Amazon.

[79] "Smart Sensor Chest Strap," ed. Ali Express.

Appendices

Appendix A

A.1 Project Description

A.2 Gantt Chart

FMH606 Master's Thesis

Title: AI Techniques in assisting elderly people at home with unobtrusive supervision of events related to health and safety

USN supervisor: Saba Mylvaganam and Alexander Jonsaas

External partner: Professor Hubert Roth, University of Siegen, Germany

Task background:

Sensor networking and data fusion have been used in different contexts from the beginning 90s at USN. USN has done work on process tomography, sensor networks, soft sensing during the last 20 years and has the latest equipment and sensors. The hardware and software for this project are available. Possibilities are there for visiting other organizations working with similar issues. As to the elderly care, close interaction with health authorities and organizations are foreseen. There is an ongoing project on elderly care at USN in collaboration with local health authorities.

Task description:

There are many systems for supervising elderly in their own homes without moving them to institutions. A dedicated system preferably linked to the mobile phone alerting relevant personnel for swift and relevant response will be very useful. In addition, wearable systems for alerting health-care personnel on any unusual events will also be studied.
In assisting needy persons, such as persons in wheel chair after accidents to reduced functionality, elderly and sick people, in recent years, there has been considerable technological advance involving dedicated sensors and associated data fusion. A recent paper describes the usage of different sensors mounted on the person and logging in the data for analysis.
This project has focus on AI-techniques relevant for assisting elderly as well as alerting relevant personnel for circumventing acute problems. Focus is on categorizing different scenarios, identifying sensors and their proper placement on the body and the ambient environment and developing data fusion strategies. The planned activities in this project involve

1. Literature study on existing techniques in the field under focus as well as related fields
2. Brief survey of sensors used in assistance to elderly and supervision of events related to their health and safety
3. Brief survey on assistance technology to elderly especially in Japan and other countries
4. Survey regarding the use of welfare technology in Norwegian elderly healthcare services from the perspective of relevant actors (users and service providers)
5. Ethical aspects of sole dependence on technology for elderly assistance
6. Identify the measurands needed for monitoring and alerting in the elderly care sector
7. Select necessary sensors, including wearables and ambient sensors. Some of these sensors may be already available in smart phone used by the assisted person.

8. Studying the signal trains in different scenarios and observing their characteristics.
9. Use of different AI-techniques in the analysis of the body and ambient sensors. This part is a data fusion exercise and should use the steps needed to guarantee data quality before launching on the data fusion.
10. Presenting common features in the AI-techniques used with possible extension to a smart phone
11. Submitting a project report following the guidelines of USN with necessary programs and including a well documented data and software used

Student category: IIA students

Practical arrangements:

Necessary hardware and software will be provided by USN. Work will be performed in Sensor Lab and possibly with visits to health-related units. However, possible interaction with experts in relevant fields from the sensor market as well as the health sector are foreseen.

Signatures:

Student (date and signature):
06.02.2017
Karina Kaspersen

Supervisor (date and signature):
M. Kaus

Appendix A.2

			Week																			
No.	Activity description	Progress	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Literature study	100 %																				
2	Survey of sensor used in the field	100 %																				
3	Brief survey on assistance technology of elderly in Japan and other countries	100 %																				
4	Survey of current situation on velfare tecnology in Norway	100 %																				
5	Ethical Aspects	100 %																				
6	Identify the Measurands	100 %																				
7	Obtain the modules to use in the data collection system	100 %																				
8	Develop the data collection system	100 %																				
9	Collect data	100 %																				
10	Studying the signal trains and characteristics of different scenarios	100 %																				
11	Use of different AI-techniques in analysis of the sensors	100 %																				
12	Presenting common features in the AI-techniques and possible Extension to a Smartphone	100 %																				
13	Submitting project report	100 %																				
			8.1	15.1	22.1	29.1	5.2	12.2	19.2	26.2	5.3	12.3	19.3	26.3	2.4	9.4	16.4	23.4	30.4	7.5	14.5	15.5:Delivery

Appendix B

B.1 *MPU-6050 Code:* Arduino code transmitting gyroscope and accelerometer sensor measurement with bluetooth serial communication

B.2 *doorSensor Code:* Arduino code transmitting reed switch with bluetooth serial communication

B.3 *Logging Application for Wireless Data Collection Units:* LabVIEW Code for Wireless DCUs (Front Display and Block Diagram)

B.4 *featureExtraction Code*: MATLAB Code for extracting the features of all experiments (featureExtraction.m.)

B.5 *geofenceSimulation Code*: MATLAB Code for simulating the Geofence Threshold AI model *(geofenceClassifierSimulation.m)*

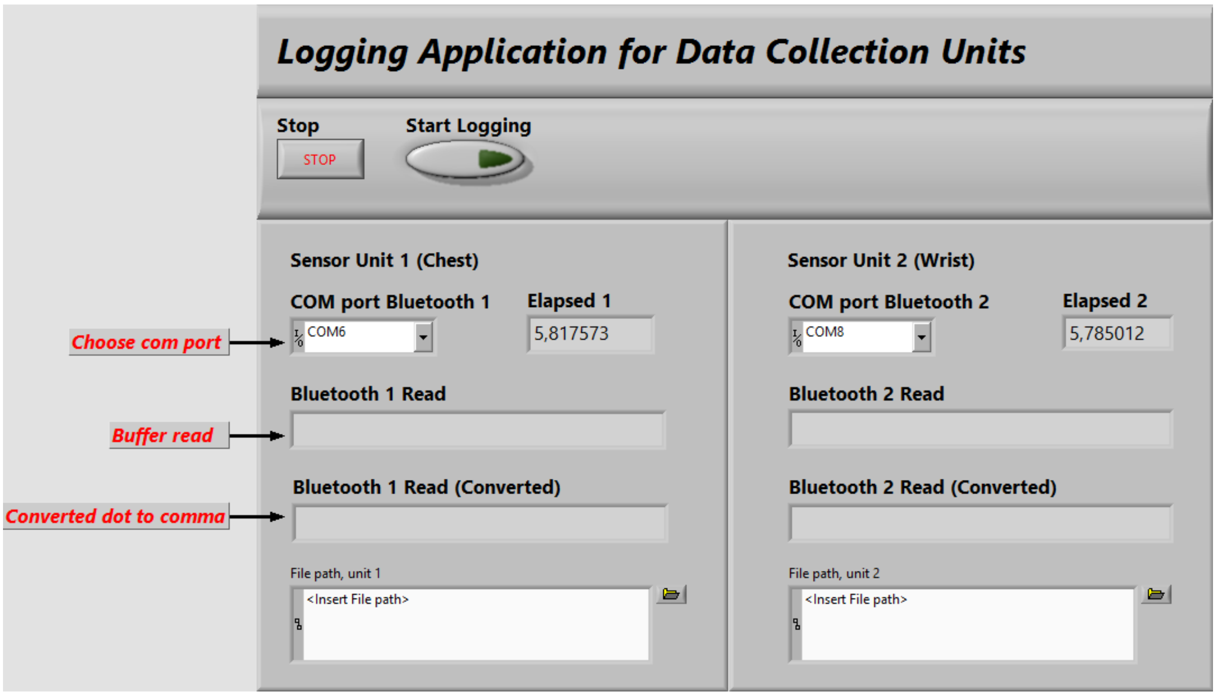
********************************** APPENDIX B.1: MPU-6050 CODE **

```
//This sketch will communicate with the MPU over the I2C bus and transmit the sensor values with BT communication
#include <SoftwareSerial.h>
#include <Wire.h> //Enable I2C communication
/*
===MPU6050 information:===
- Latest Software: https://github.com/VRomanov89/EEEnthusiast/tree/master/MPU-6050%20Implementation/MPU6050_Implementation
- Arduino IDE v1.6.9
- Arduino Wire library
===Terms of use===
The software is provided by EEEnthusiast without warranty of any kind. In no event shall the authors or
copyright holders be liable for any claim, damages or other liability, whether in an action of contract,
tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in
the software.
*/
SoftwareSerial BTSerial(8, 9); //Connect HC-06. TX: Digital Pin 8, RX: Digital Pin 9
long accelX, accelY, accelZ; //storing acc values read from sensor
float gForceX, gForceY, gForceZ; //for calculatng g forces
long gyroX, gyroY, gyroZ; //storing gyro values read from sensor
float rotX, rotY, rotZ; //storing the rotational speed around those axis
void setup() {
BTSerial.begin(9600); //for starting the bluetooth serial at BR 9600
Wire.begin(); //command initialized I2C communication
setupMPU(); //initialize this function
}
void loop() {
recordAccelRegisters(); //Determine accel measurement
recordGyroRegisters();  //Determine gyro measurement
printData();
delay(20);
//}
}
void setupMPU(){
//The main purpose of this function is to
//1) establish communication with MPU,
//2) set up all the registers we will be using in order to read the data back from the MPU into the Arduino
Wire.beginTransmission(0b1101000); //This is the I2C address of the MPU (b1101000/b1101001 for AC0 low/high datasheet sec. 9.2)
Wire.write(0x6B); //Accessing the register 6B - Power Management (Sec. 4.28)
Wire.write(0b00000000); //Setting SLEEP register to 0. (Required; see Note on p. 9)
Wire.endTransmission();
Wire.beginTransmission(0b1101000); //I2C address of the MPU
Wire.write(0x1B); //Accessing the register 1B - Gyroscope Configuration (Sec. 4.4)
Wire.write(0x00000000); //Setting the gyro to full scale +/- 250deg./s
Wire.endTransmission();
Wire.beginTransmission(0b1101000); //I2C address of the MPU
Wire.write(0x1C); //Accessing the register 1C - Acccelerometer Configuration (Sec. 4.5)
Wire.write(0b00000000); //Setting the accel to +/- 2g
```

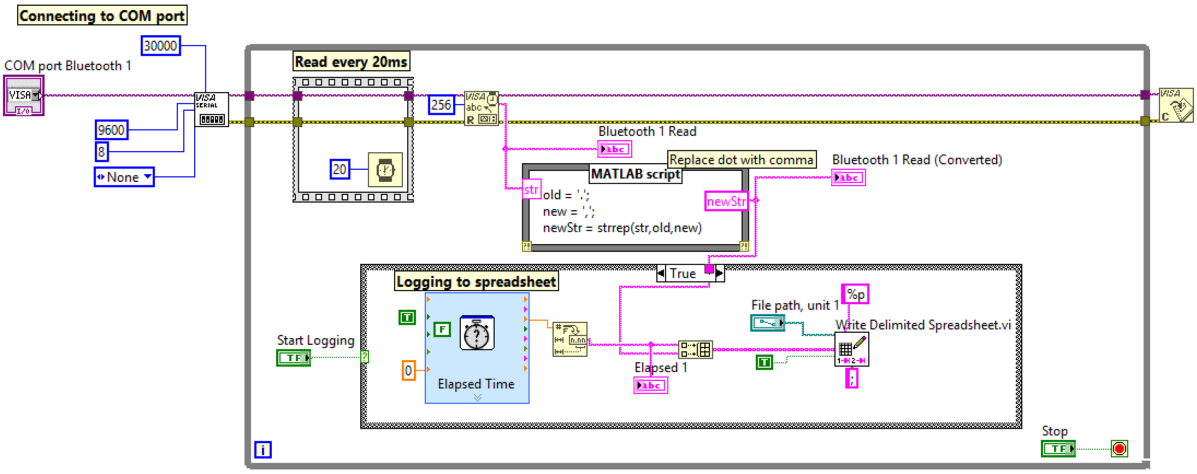
```
Wire.endTransmission();
}
void recordAccelRegisters() {
Wire.beginTransmission(0b1101000); //I2C address of the MPU
Wire.write(0x3B); //Starting register for Accel Readings
Wire.endTransmission();
Wire.requestFrom(0b1101000,6); //Request Accel Registers (3B - 40)
while(Wire.available() < 6);
accelX = Wire.read()<<8|Wire.read(); //Store first two bytes into accelX
accelY = Wire.read()<<8|Wire.read(); //Store middle two bytes into accelY
accelZ = Wire.read()<<8|Wire.read(); //Store last two bytes into accelZ
processAccelData(); //function to get something meaningful
}
void processAccelData(){ //divide with 16384 based on accel sensitivity config +/-2g
gForceX = accelX / 16384.0;
gForceY = accelY / 16384.0;
gForceZ = accelZ / 16384.0;
}
void recordGyroRegisters() {
Wire.beginTransmission(0b1101000); //I2C address of the MPU
Wire.write(0x43); //Starting register for Gyro Readings
Wire.endTransmission();
Wire.requestFrom(0b1101000,6); //Request Gyro Registers (43 - 48)
while(Wire.available() < 6);
gyroX = Wire.read()<<8|Wire.read(); //Store first two bytes into gyroX
gyroY = Wire.read()<<8|Wire.read(); //Store middle two bytes into gyroY
gyroZ = Wire.read()<<8|Wire.read(); //Store last two bytes into gyroZ
processGyroData();
}
void processGyroData() { //divide with 131 based on gyro sensitivity config +/-250deg/s
rotX = gyroX / 131.0;
rotY = gyroY / 131.0;
rotZ = gyroZ / 131.0;
}
void printData() { //Transmit the gyro and accel values over BT serial com
BTSerial.print(rotX);
BTSerial.print(";");
BTSerial.print(rotY);
BTSerial.print(";");
BTSerial.print(rotZ);
BTSerial.print(";");
BTSerial.print(gForceX);
BTSerial.print(";");
BTSerial.print(gForceY);
BTSerial.print(";");
BTSerial.println(gForceZ);
}
```

```
//*********************************   APPENDIX B.2: DOOR SENSOR CODE *****************************************

#include <SoftwareSerial.h>

const int switchPin = 2;  //Digital Pin 2 reads status of the Reed Switch
bool delta = false; //Delta is Door activity is false=0, when door is locked

SoftwareSerial BTSerial(9, 10); //Connect HC-06. TX - Digital Pin 9, RX - Digital Pin 10

void setup() {
  BTSerial.begin(9600);   //Communicating Through serial using BR of 9600
  pinMode(switchPin, INPUT); //Declaring the switchPin as a INPUT pin
  attachInterrupt(digitalPinToInterrupt(switchPin),doorEvent,CHANGE); //If the state of doorActivity changes, doorEvent function will activate
  //See details: https://www.arduino.cc/en/Reference/AttachInterrupt
}

void loop()
{
    BTSerial.println(delta);          //For demonstration
    delay(500);                       //For demonstration, the state of delta will be transmitted via BT every 500 ms
}

void doorEvent()
{
  delta = !delta;   //Changes state to true=1, when door opens
}
```

Appendix B.3.1



Appendix B.3.2



```
% APPENDIX B.4: Feature Extraction
%%  featureExtraction.m
%   KARINA KASPERSEN, MASTERHESIS SPRING17, INDUSTRIAL IT AND AUTOMATION
%   CODE DECRIPTION:
%   1.   TAKES EACH VARIABLE (EXPERIMENT) IN WORKSPACE AND APPLIES FEATURES TO
%        THE DATA
%   4.   THEN SAVES IN A MATRIX IN A Mat FILE(feature_calculated)

%   Note on notation: variable names including a '1' or '2' corresponds to
%   CHEST (1) and WRIST (2)
clear
clc
load('workspace_experiments.mat')
%Workspace with all fall and non-fall experiment data sets
%ex 1 - 10: E_fall, ex 11 - 16: E_transition, ex 17 - 18: E_sedentary, ex 19:
E_walking, ex 20: E_running

filename='featureCalc.mat';
%Create a .mat file the feature matrix F will be loaded into
feature_calculated = matfile(filename, 'Writable', true);
%Note: writable is true by d.

for l = 1:length(E_list)
%One loop per experiment. E_list is the list (cell array) of all variables in
the workspace found by (E_list = who)
        E = eval(num2str(E_list{l,1}));
%Selcting the data set in the workspace to apply features to (a new every loop
from the list of variables (exp_list)
        [rowNo,ColumnNo] = size(E);

        numberOfFeatures=200;
%Create enough columns (less than number of total features
        featureMatrix = (ones(1,numberOfFeatures))*999999;
%999999, so it easy extract the features after the coderun is done, also
helpful for finding available cells to apply features in
        %% 1. F_noOfPeak: NUMBER OF PEAKS ABOVE OR UNDER A THRESHOLD VALUE

        n1_1 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS GYRO CHEST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n),E(:,1),
'MinPeakProminence',50);       %Finds max peaks
                [pksMin,locsMin]= findpeaks(-(E(:,n)),E(:,1),
'MinPeakProminence',50);    %Finds min peaks
                noOfPeak = length(pksMax)+length(pksMin);
%The number of peaks that drops over and under threshold on either side before
the signal attains a higher value
                featureMatrix(1,n1_1+(n-2)) = noOfPeak;
%Calculating standard deviation of all measurements of this experiment
        end

        n1_2 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS ACC CHEST
```

```
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,(n+3)),E(:,1),
'MinPeakProminence',0.5);
                [pksMin,locsMin]= findpeaks(-(E(:,(n+3))),E(:,1),
'MinPeakProminence',0.5);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n1_2+(n-2)) = noOfPeak;
        end

        n1_3 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS GYRO WRIST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n+6),E(:,1),
'MinPeakProminence',50);
                [pksMin,locsMin]= findpeaks(-(E(:,n+6)),E(:,1),
'MinPeakProminence',50);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n1_3+(n-2)) = noOfPeak;
        end

        n1_4 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS ACC WRIST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n+9),E(:,1),
'MinPeakProminence',0.5);
                [pksMin,locsMin]= findpeaks(-(E(:,n+9)),E(:,1),
'MinPeakProminence',0.5);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n1_4+(n-2)) = noOfPeak;
        end

        %% 2. F_S(noOfPeak): NUMBER OF PEAKS IN ALL CHEST AND ALL WRIST
MEASUREMENT
        % Finds number of peaks in each experiment measurement
            n2_1 = find(featureMatrix>=999999,1);

            k=1;
            for n=1:3:(ColumnNo-1)
                x_val_sumPeaks = featureMatrix(1,n1_1 + n-1);
                y_val_sumPeaks = featureMatrix(1,n1_1 + n);
                z_val_sumPeaks = featureMatrix(1,n1_1 + n+1);
                featureMatrix(1,n2_1+(k-1)) =  x_val_sumPeaks + y_val_sumPeaks
+ z_val_sumPeaks;   %The number of peaks that drops over and under threshold
on either side before the signal attains a higher value
                k = k+1;
            end


        %% 3. F_thPeak: PEAK THRESHOLD, NUMBER OF PEAKS THAT DROPS A CERTAIN
THRESHOLD VALUE FROM NEIGHBOURING VALUES
        n3_1 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS GYRO CHEST
```

```
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n),E(:,1), 'Threshold',50);
%Finds max peaks
                [pksMin,locsMin]= findpeaks(-(E(:,n)),E(:,1), 'Threshold',50);
%Finds min peaks
                noOfPeak = length(pksMax)+length(pksMin);
%The number of peaks that drops over and under threshold on either side before
the signal attains a higher value
                featureMatrix(1,n3_1+(n-2)) = noOfPeak;
%Append to feature matrix
        end

        n3_2 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS ACC CHEST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,(n+3)),E(:,1),
'Threshold',0.5);
                [pksMin,locsMin]= findpeaks(-(E(:,(n+3))),E(:,1),
'Threshold',0.5);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n3_2+(n-2)) = noOfPeak;
        end

        n3_3 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS GYRO WRIST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n+6),E(:,1), 'Threshold',50);
                [pksMin,locsMin]= findpeaks(-(E(:,n+6)),E(:,1),
'Threshold',50);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n3_3+(n-2)) = noOfPeak;
        end

        n3_4 = find(featureMatrix>=999999,1);

        %NUMBER OF PEAKS ACC WRIST
        for n=2:((ColumnNo-1)/3)
                [pksMax,locsMax]= findpeaks(E(:,n+9),E(:,1), 'Threshold',0.5);
                [pksMin,locsMin]= findpeaks(-(E(:,n+9)),E(:,1),
'Threshold',0.5);
                noOfPeak = length(pksMax)+length(pksMin);
                featureMatrix(1,n3_4+(n-2)) = noOfPeak;
        end

        %% 4. F_S(thPeak): SUM NUMBER OF PEAKS THAT DROPS A CERTAIN THRESHOLD
VALUE FROM NEIGHBOURING VALUES
        n4_1 = find(featureMatrix>=999999,1);

        for n=2:((ColumnNo-1)/4)
                peaksGyro1 = featureMatrix(1,n3_1+(n-2));
%The number of peaks that drops over and under threshold on either side before
the signal attains a higher value
```

```
                peaksAcc1 = featureMatrix(1,n3_2+(n-2));
%The number of peaks that drops over and under threshold on either side before
the signal attains a higher value
                featureMatrix(1,n4_1+(n-2)) = peaksGyro1+peaksAcc1;
%Append to feature matrix
        end

        n4_2 = find(featureMatrix>=999999,1);

        for n=2:((ColumnNo-1)/4)
                peaksGyro2 = featureMatrix(1,n3_3+(n-2));
                peaksAcc2 = featureMatrix(1,n3_4+(n-2));
                featureMatrix(1,n4_2+(n-2)) = peaksGyro2+peaksAcc2;
        end

        %% 5. F_p: PERCENTILES

        % 10th PERCENTILE
        n5_1 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)

            pcile10_x = prctile(E(:,n+1),10);
            pcile10_y = prctile(E(:,n+2),10);
            pcile10_z = prctile(E(:,n+3),10);

            featureMatrix(1,n5_1+(k-1)) =
sqrt((pcile10_x)^2+(pcile10_y)^2+(pcile10_z)^2);  %MAGNITUDE
            k=k+1;
        end

        % 20th PERCENTILE
        n5_2 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile20_x = prctile(E(:,n+1),20);
            pcile20_y = prctile(E(:,n+2),20);
            pcile20_z = prctile(E(:,n+3),20);

            featureMatrix(1,n5_2+(k-1)) =
sqrt((pcile20_x)^2+(pcile20_y)^2+(pcile20_z)^2);
            k=k+1;
        end

        % 35th PERCENTILE
        n5_3 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile35_x = prctile(E(:,n+1),35);
            pcile35_y = prctile(E(:,n+1),35);
            pcile35_z = prctile(E(:,n+1),35);
```

```
            featureMatrix(1,n5_3+(k-1)) =
sqrt((pcile35_x)^2+(pcile35_y)^2+(pcile35_z)^2);
            k=k+1;
        end

        % 50th PERCENTILE
        n5_4 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile50_x = prctile(E(:,n+1),50);
            pcile50_y = prctile(E(:,n+1),50);
            pcile50_z = prctile(E(:,n+1),50);

            featureMatrix(1,n5_4+(k-1)) =
sqrt((pcile50_x)^2+(pcile50_y)^2+(pcile50_z)^2);
            k=k+1;
        end

        % 70th PERCENTILE
        n5_5 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile70_x = prctile(E(:,n+1),70);
            pcile70_y = prctile(E(:,n+1),70);
            pcile70_z = prctile(E(:,n+1),70);

            featureMatrix(1,n5_5+(k-1)) =
sqrt((pcile70_x)^2+(pcile70_y)^2+(pcile70_z)^2);
            k=k+1;
        end

        % 80th PERCENTILE
        n5_6 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile80_x = prctile(E(:,n+1),80);
            pcile80_y = prctile(E(:,n+1),80);
            pcile80_z = prctile(E(:,n+1),80);

            featureMatrix(1,n5_6+(k-1)) =
sqrt((pcile80_x)^2+(pcile80_y)^2+(pcile80_z)^2);
            k=k+1;
        end

        %95th PERCENTILE
        n5_7 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1)
            pcile95_x = prctile(E(:,n+1),95);
            pcile95_y = prctile(E(:,n+1),95);
```

```
            pcile95_z = prctile(E(:,n+1),95);

            featureMatrix(1,n5_7+(k-1)) =
sqrt((pcile95_x)^2+(pcile95_y)^2+(pcile95_z)^2);
            k=k+1;
        end

         %% 6. F_c: WAVELET COEFFICIENTS
        n6_1 = find(featureMatrix>=999999,1);

        %Using the wavelet decomposition vector C
        %GYRO CHEST
        [Cx_g_1,Lx_g_1]=wavedec(E(:,2),2,'db2');   %first row corresponds to
g_chest, second row is a_chest and so on
        [Cy_g_1,Ly_g_1]=wavedec(E(:,3),2,'db2');
        [Cz_g_1,Lz_g_1]=wavedec(E(:,4),2,'db2');

        gx_1_wlt = sum(Cx_g_1); gy_1_wlt = sum(Cy_g_1); gz_1_wlt =
sum(Cz_g_1);  %Summing up all wavelet dec. vectors C
        sum_sqrt_wlt_g_1 = sqrt((gx_1_wlt)^2 + (gy_1_wlt)^2 + (gz_1_wlt)^2);
%Magnitude of the sums

        %ACC CHEST
        [Cx_a_1,Lx_a_1]=wavedec(E(:,5),2,'db2');
        [Cy_a_1,Ly_a_1]=wavedec(E(:,6),2,'db2');
        [Cz_a_1,Lz_a_1]=wavedec(E(:,7),2,'db2');

        ax_1_wlt = sum(Cx_a_1); ay_1_wlt = sum(Cy_a_1); az_1_wlt =
sum(Cz_a_1);
        sum_sqrt_wlt_a_1 = sqrt((ax_1_wlt)^2 + (ay_1_wlt)^2 + (az_1_wlt)^2);

        %GYRO WRIST
        [Cx_g_2,Lx_g_2]=wavedec(E(:,8),2,'db2');
        [Cy_g_2,L_g_2]=wavedec(E(:,9),2,'db2');
        [Cz_g_2,Lz_g_2]=wavedec(E(:,10),2,'db2');

        gx_2_wlt = sum(Cx_g_2); gy_2_wlt = sum(Cy_g_2); gz_2_wlt =
sum(Cz_g_2);
        sum_sqrt_wlt_g_2 = sqrt((gx_2_wlt)^2 + (gy_2_wlt)^2 + (gz_2_wlt)^2);

        %ACC WRIST
        [Cx_a_2,Lx_a_2]=wavedec(E(:,11),2,'db2');
        [Cy_a_2,L_a_2]=wavedec(E(:,12),2,'db2');
        [Cz_a_2,Lz_a_2]=wavedec(E(:,13),2,'db2');

        ax_2_wlt = sum(Cx_a_2); ay_2_wlt = sum(Cy_a_2); az_2_wlt =
sum(Cz_a_2);
        sum_sqrt_wlt_a_2 = sqrt((ax_2_wlt)^2 + (ay_2_wlt)^2 + (az_2_wlt)^2);

        wlt_feature = [sum_sqrt_wlt_g_1, sum_sqrt_wlt_a_1, sum_sqrt_wlt_g_2,
sum_sqrt_wlt_a_2]; %A vector with all magnitude vectors of gyro and acc

        for n=1:(length(wlt_feature));
            featureMatrix(1,n6_1+(n-1)) = wlt_feature(1,n);
%Append the wavelet decomposition coefficients to the feature matrix
```

```
        end

    %% 7. F_xcorr: CROSS CORRELATION
        n7_1 = find(featureMatrix>=999999,1);

        % Calculating the cross correlation between all axis of all sensors on
        % Chest and Wrist
          k=1;
            for n=1:(ColumnNo-1)/3;    %4 iterations (gt, at, gw, aw)
                x_y(n,:)=xcorr(E(:,n+(1+k)),E(:,n+(2+k)),0,'coef');
                x_z(n,:)=xcorr(E(:,n+(1+k)),E(:,n+(3+k)),0,'coef');
                y_z(n,:)=xcorr(E(:,n+(2+k)),E(:,n+(3+k)),0,'coef');
                k=k+1;
            end

        CrossCorrFeature = [x_y, x_z, y_z];                                %4
rows and 3 columns
        CrossCorrFeature2 = reshape(CrossCorrFeature',12,1);

            for n=1:length(CrossCorrFeature2);
                    featureMatrix(1,n7_1+(n-1)) = CrossCorrFeature2(n,1);
%Append to feature matrix
            end

        %% 8. F_S(diffPeak): VALUE BETWEEN HIGHEST AND LOWEST PEAK IN EACH
SENSOR MEASUREMENT

        % Calculates the difference between highest and lowest peak in each
        % x,y and z measurement
        n8_1 = find(featureMatrix>=999999,1);

        k=1;
        for n=1:3:(ColumnNo-1);    %4 iterations (g1, a1, g2, a2)
            high_x = max(E(:,n+1));
            low_x = min(E(:,n+1));
            high_y = max(E(:,n+2));
            low_y = min(E(:,n+2));
            high_z = max(E(:,n+3));
            low_z = min(E(:,n+3));

            max_min_vector= [high_x low_x high_y low_y high_z low_z];

            high = max(max_min_vector);
            low = min(max_min_vector);

            peak_diff = abs(high)+abs(low);

            featureMatrix(1,n8_1+(k-1)) = peak_diff;
            k=k+1;
        end

        %% APPEND THE FEATURES OF PRESENT SUB EXPERIMENT (E) TO
FEATURE_CALCULATED_MATRIX
        feature_calculated.featureMatrix(l, 1:200) = featureMatrix;
end
```

```
% APPENDIX B.5: Geofence Threshold Model Simulation

%%  geofenceSimulation.m
%   KARINA KASPERSEN, MASTERHESIS SPRING17, INDUSTRIAL IT AND AUTOMATION
%   CODE DECRIPTION:
%   1.   USES GENERATED VALUES (VECTORS) IN TIME (3h) TO SIMULATE A
%   THRESHOLD GEOFENCE AI-MODEL FROM A WORKSPACE THAT INCLUDES
            % a) Time column, time, with values representing minuties 1-180
            % b) Latitude and Longitude values inside and outside living
            % area
            % c) When door opens and user goes outside living area
            % rad_thresh or radius of the geofence = 0.001 deg
%   4.   THEN PLOTS THE DOOR SENSOR, DETECTED INSIDE OR OUTSIDE GEOFENCE
%   BOUDARIES AND AI-MODEL OUTPUT

%   Note on notation: variable names including a '1' or '2' corresponds to
%   CHEST (1) and WRIST (2)

% This script simulates the Geofence classifier based on several inputs the
% classifier need to crate the geofence boundaries

load('workspace_geofenceSimulation.mat')

preset_time = 120; % 3 h, for simulation purposes
home_lon = 10.438; %Initial value
home_lat = 59.819; %Initial value
rad_thresh = 0.001; %For simulation purposes, can be larger. If smaller is too
close, shown in position signal train
detected_outside = time*0; %Make vector

for i = 1:length(time)
    if (((abs(home_lon-lon_measured(i))) > rad_thresh) || ((abs(home_lat-
lat_measured(i))) > rad_thresh))  %F_geo feature
        detected_outside(i) = 1;
    else
        detected_outside(i) = 0;
    end
end

classifier_output = time*0; %Make Vector
time_detected_notHome = time*0; %Make Vector

%classifier
if((find(detected_outside==1,1)>1)==true) %%If detected outside
    for j=(find(detected_outside==1,1)):(length(time)) %from detected outside
geofence
        door_time= find(door_state==1,1); %the time(s) from when door was
opened and user went outside geofence
        time_detected_notHome(j,1) = abs(door_time - time(j,1));  %Calculating
how long outside
        if (time_detected_notHome(j,1) > preset_time) %True when not home
after preset time
            classifier_output(j,1)= 1; %Classfier will change status to 1= not
home after preset Time
```

```
        elseif (time_detected_notHome(j,1) < preset_time) %Classifier is 0
until time before door opens and detected inside geofence
            classifier_output(j,1) = 0;
        end
    end
end
```