

Analysis of Derived Features for the Motion Classification of a Passive Lower Limb Exoskeleton

**Master's Thesis
of**

Albert Costa Ruiz

**KIT Department of Informatics
Institute for Anthropomatics and Robotics (IAR)
High Performance Humanoid Technologies Lab (H²T)**

Referees: Prof. Dr.-Ing. Tamim Asfour

**Advisors: M.Sc. Isabel Patzer
Dr.-Ing. Jonas Beil**

Duration: 1st April 2019 – 30th September 2019

Erklärung:

Ich versichere hiermit, dass ich die Arbeit selbstständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht habe und die Satzung des Karlsruher Instituts für Technologie zur Sicherung guter wissenschaftlicher Praxis beachtet habe.

Karlsruhe, den 30th September 2019

Albert Costa Ruiz

Abstract:

The recognition of human motion intentions is a fundamental requirement to control efficiently an exoskeleton system. The exoskeleton control can be enhanced or subsequent motions can be predicted, if the current intended motion is known.

At H²T research has been carried out with a classification system based on Hidden Markov Models (HMMs) to classify the multi-modal sensor data acquired from a unilateral passive lower-limb exoskeleton. The training data is formed of force vectors, linear accelerations and Euler angles provided by 7 3D-force sensors and 3 IMUs. The recordings consist of data of 10 subjects performing 14 different types of daily activities, each one carried out 10 times.

This master thesis attempts to improve the motion classification by using physical meaningful derived features from the raw data aforementioned. The knee vector moment and the knee and ankle joint angles, which respectively give a kinematic and dynamic description of a motion, were the derived features considered. Firstly, these new features are analysed to study their patterns and the resemblance of the data among different subjects is quantified in order to check their consistency. Afterwards, the derived features are evaluated in the motion classification system to check their performance. Various configurations of the classifier were tested including different preprocessors of the data employed and the structure of the HMMs used to represent each motion. Some setups combining derived features and raw data led to good results (e.g. norm of the moment vector and IMUs got 89.39% of accuracy), but did not improve the best results of previous works (e.g. 2 IMUs and 1 Force Sensor got 90.73% of accuracy).

Although the classification results are not improved, it is proved that these derived features are a good representation of their primary features and a suitable option if a dimensional reduction of the data is pursued. At the end, possible directions of improvement are suggested to improve the motion classification concerning the results obtained along the thesis.

Kurzzusammenfassung:

Die Erkennung menschlicher Bewegungsabsichten ist eine grundlegende Voraussetzung für die effiziente Steuerung eines Exoskelettsystems. Die Exoskelettkontrolle kann verstärkt oder nachfolgende Bewegungen vorhergesagt werden, wenn die aktuell beabsichtigte Bewegung bekannt ist.

Bei H²T wurde mit einem Klassifizierungssystem auf Basis von Hidden Markov Models (HMMs) geforscht, um die multimodalen Sensordaten zu klassifizieren, die von den unteren Extremitäten eines einseitigen passiven Exoskeletts gewonnen wurden. Die Trainingsdaten bestehen aus Kraftvektoren, Linearbeschleunigungen und Euler-Winkeln, die von 7 3D-Kraftsensoren und 3 IMUs bereitgestellt werden. Die Aufzeichnungen bestehen aus Daten von 10 Personen, die 14 verschiedene Arten von täglichen Aktivitäten ausüben, von denen jede 10 mal durchgeführt wurde.

Diese Masterarbeit versucht, die Bewegungsklassifizierung zu verbessern, indem sie physikalisch sinnvolle, abgeleitete Merkmale aus den oben genannten Rohdaten verwendet. Das Knievektormoment und die Knie- und Sprunggelenkwinkel, die jeweils eine kinematische und dynamische Beschreibung einer Bewegung liefern, waren diese abgeleiteten Merkmale. Zunächst werden diese neuen Merkmale analysiert, um deren Muster zu untersuchen. Danach wird die Ähnlichkeit der Daten zwischen den verschiedenen Probanden quantifiziert, um ihre Konsistenz zu überprüfen. Anschließend werden die abgeleiteten Merkmale im Bewegungsklassifizierungssystem ausgewertet, um ihre Leistung zu überprüfen. Diverse Konfigurationen des Klassifikators wurden getestet, darunter verschiedene Präprozessoren der verwendeten Daten und die Struktur der HMMs, die zur Darstellung jeder Bewegung verwendet werden. Einige Setups, die abgeleitete Merkmale und Rohdaten kombinierten, führten zu guten Ergebnissen (z.B. Norm des Momentvektors und IMUs erreichen eine Genauigkeit von 89.39%), die allerdings knapp unter den besten Ergebnissen früherer Arbeiten liegen (z.B. 2 IMUs und 1 Kraftsensor erhielten 90,73% der Genauigkeit).

Obwohl die Klassifikationsergebnisse nicht verbessert werden, wurde nachgewiesen, dass diese abgeleiteten Merkmale eine gute Darstellung ihrer primären Merkmale sind und eine geeignete Option darstellen, wenn eine Dimensionsreduktion der Daten angestrebt wird. Am Ende werden Verbesserungsvorschläge vorgestellt, die Potenzial zur Verbesserung der Bewegungsklassifizierung unter Berücksichtigung der Ergebnisse dieser Arbeit besitzen.

Contents

1. Introduction	1
2. State of the Art	4
2.1. Applications of Motion Recognition	4
2.2. Motion Data	4
2.3. Derived Features	5
2.4. Motion Classification Methods	6
2.5. Motion Recognition at H ² T	7
2.5.1. Whole-Body Motion	7
2.5.2. Motion Classification for LLE	7
3. Basics	9
3.1. Passive Exoskeleton	9
3.1.1. Structure and Design	9
3.1.2. Sensors	10
3.2. Motion Data	11
3.3. HMM-Based Motion Classification	12
3.3.1. Hidden Markov Model	12
3.3.2. HMM with Continuous Observations	14
3.3.3. Types of HMMs	14
3.3.4. Fundamental Problems for HMMs	15
3.4. Model Validation	17
3.5. Evaluation of Results	18
3.5.1. Accuracy	18
3.5.2. Precision and Recall	18
3.5.3. F ₁ Score	18
3.5.4. Results in Multi-Class Classification	19
4. Derived Features	20
4.1. Joint Moments	20
4.1.1. Calculation of the Moments	21
4.1.2. Expressions of the Moments	24
4.2. Leg Joint Angles	25
5. Evaluations	27
5.1. Moment Feature Analysis	27
5.1.1. Resemblance Analysis	27
5.1.2. Intra-Subject Analysis	29
5.1.3. Inter-Subject Analysis	29
5.2. Motion Classification	29
5.2.1. Data Preprocessing	29
5.2.2. Motion Classification System	33
5.2.3. Model Validation and Evaluation of Results	34

6. Results	36
6.1. Moment Feature Analysis	36
6.1.1. Resemblance Analysis	36
6.1.2. Data of the Resemblance Analysis	42
6.1.3. Main Results of the Resemblance Analysis	43
6.2. Joint Angles Analysis	46
6.3. Motion Classification	48
6.3.1. Results with Raw Data	48
6.3.2. Corrupted Data	49
6.3.3. Comparison of Derived and Primary Features	51
6.3.4. Performance of Filtered Data	51
6.3.5. Comparison of Topologies	52
6.3.6. Effect of the Correlations	55
6.3.7. Best Evaluations with Derived Features	55
7. Conclusion and Outlook	59
A. Appendix	61
A.1. Derived Features	61
A.2. Correlation Analysis	63

1. Introduction

The term *exoskeleton* comes from a biological field and it is described by the Cambridge Dictionary as: *A hard outer layer that covers, supports, and protects the body of an invertebrate animal such as an insect or crustacean.* The main concept of this description was the reason to name in the same way a person-oriented wearable robot that supplements or replaces the function of a limb. Therefore, an exoskeleton could be described as a wearable robot that allows for extending, complementing, substituting or enhancing the human limb where it is worn (Pons, 2008).

The amount of people with mobility problems has increased enormously in the last decades (e.g stroke or spinal cord injury). For instance, every year between 250000 and 500000 people suffer a spinal cord injury according to the World Health Organization (2019). The development of devices in the area of motion assistance have become really relevant in order to enhance the quality of life of these people. By contrast, devices that augment physical abilities of humans to improve their performances in industrial and military work are also in demand. In the past decade, robotic assistance devices such as exoskeletons have made significant progress, and some products have already been commercialised ((Chen et al., 2016a)).

The main classification of exoskeletons divides them according to the performance in cooperation with the human actor: rehabilitative, assistive or augmentative systems. ((Viteckova et al., 2018)). The first two have medical purposes, and meanwhile exoskeletons for rehabilitation have as goal the assistance with the motion recovery after accidents or illnesses, the other one pursues to design robotic devices that provide a permanent support to the wearer. These two types of exoskeletons are oriented to patients suffering from stroke, spinal cord injury (SCI) or similar problems (some examples can be found in Lokomat (2019) and UniExo (2019)).



(a) Medical Robot Suit HAL 3 system of the Japanese robotics firm Cyberdyne (Hybrid Assistive Limb, 2013)



(b) Toyota Motor Corp's rehabilitation robot Welwalk WW-1000 (Toyota, 2017)

Figure 1.1.: Examples of exoskeletons with medical purposes.

Several projects in this area can be found currently in development. The Hybrid Assistive Limb (HAL) is an example of a powered exoskeleton suit developed by the Tsukuba University in Japan. It was designed for people with physical disabilities and among other achievements it was certified in Europe as the first non-surgical medical treatment robot. Figure 1.1(a) shows the medical Robot Suit Hal 3 leg-only device. There is also a HAL 5 full body device that works also with arms and torsos. Another example of project, this time only focused on rehabilitation, is Toyota Motor Corp's rehabilitation robot Welwalk WW-1000, designed to aid in the rehabilitation of individuals with lower limb paralysis (Figure 1.1(b)).

Exoskeletons directed towards a human performance augmentation are aimed this time at healthy subjects and it pursues a limb movement with an increment in strength and endurance. This kind of exoskeleton can be found in different areas. The military sector could be an example, in which some projects have been carried out to decrease fatigue and increase productivity (The three million suit (2014) and Eric Adams (2018)). In industry there is a willingness towards the use of passive exoskeleton technology with the target of reducing injuries and fatigue, the main cause of errors and low productivity. A clear example is the exoskeleton of the company Hyundai with the *Hyundai Chairless Exoskeleton (H-CEX)*. It is designed for workers who have to stay in a seated position throughout the day and it helps workers lessen safety risks and decrease fatigue since it decreases the use of waist and lower body muscles by 80% (Figure 1.2). Chen et al. (2016b) presents a survey of the current situation in the field of Exoskeletons devices which explains current issues related to them like the control strategies used for lower extremity exoskeletons or their limitation, and gives hints of possible directions that can be taken in future research.



Figure 1.2.: Hyundai Chairless Exoskeleton (H-CEX). It is aimed to workers who have to stay in a seated position throughout the day (Hyundai, 2018).

Getting more in deep in the exoskeleton's control area, all the powered exoskeletons need a complex control system (electric motors, pneumatics, levers and/or hydraulics). With the exception of rehabilitative exoskeletons, in which almost all the motions are predefined, an exoskeleton should be capable to adapt to different motion types in the minimum procedure time (Viteckova et al. (2018) and Beil et al. (2018)). Each motion can be defined by its kinematic and dynamic description, and if these properties are contextualised in the exoskeleton's control, the amount of variables to take into account is large (moments, velocities, accelerations, forces, etc). Accordingly, in order to adapt, guide and support the exoskeleton's control to the motion intended to perform, it is crucial to estimate and consider it in real-time (Jang et al., 2017). In that way, the control can be optimised, which allows an intuitive control of the exoskeleton and an improvement of its acceptance and wearing comfort (Beil et al., 2018).

Once the main target is set, the necessity to carry out a motion classification during the exoskeleton's operation turns out to be an essential request. This classification has to be done using data that, as much as possible, should describe that motion in an unequivocal way. Kinematic and dynamic properties char-

acterise the nature of a motion, therefore data gathered with this kind of information should become one of the best ways to represent human movements (Senanayake and Senanayake, 2010). Gait analysis have been using different types of motion sensors and systems, such as accelerometer, gyroscope, magnetoresistive sensors, electromagnetic tracking system (ETS), force sensor and sensors for electromyography (EMG). Based on these sensors, a single type or a combined sensor system of multiple types of sensors may be used for various gait analysis applications.

The passive exoskeleton considered in this thesis equipped sensors that give kinematic and dynamic data, but it is not directly related to the leg's physics. This thesis should investigate if the usage of meaningful derived features extracted from this data improves the classification system. Beside, this idea was one of the suggestions in the conclusions of Patzer, Isabel and Asfour, Tamim (2019). Additionally, a dimensional reduction of the current data set might be obtained since the derived features calculated from the primary features should present a lower dimension and, at the same time, be more representative and purposeful.

This thesis is organized as follows: Chapter 2 presents an overview of the work done until now in the field of motion classification, containing a specific section for the projects carried out in H²T about whole-body motion classification and more specifically for lower limb exoskeletons. Next, in Chapter 3, the basics used along this thesis are introduced, like the database of motion recordings with which the motion classification system will be tested or the theory behind the machine learning technique used for the classification. Afterwards, Chapter 4 presents the new derived features that will be used in the motion classification. This chapter also explains how were these features calculated from the sensor data, and which assumptions were taken. Chapter 5 describes the basics of the two main issues addressed: a resemblance analysis carried out using derived features and the classification system based on Hidden Markov Models to identify motions. The results, conclusions and comments of the issues introduced in Chapter 5 can be found in Chapter 6. Finally. Chapter 7 will do a summary of all the conclusions and ideas extracted from the results and some future work is suggested.

2. State of the Art

This section gives an overview of the main topics that will be addressed during this thesis: motion classification, motion data and derived features. The last section focuses on the projects carried out in H²T about motion classification for lower limb exoskeletons.

2.1. Applications of Motion Recognition

Human activity recognition has taken a really important role in a wide range of areas. In medicine, for instance, reliable gait-phase classification is used in locomotion analysis and in identifying abnormalities (Pappas et al., 2001). In gesture recognition, more focused on the motions and gestures of the face and hands, pursues control devices with a minimum physical interaction with them. Fields like remote control (Stern et al., 2010) or sign language translation (Bhatnagar and Agrawal, 2015) are some examples. In the entertainment industry, motion devices like the Microsoft's camera *Kinect*, opened up a range of possibilities in terms of interaction and control without the need for a game controller (Lun and Zhao, 2015). Although this device was created with entertainment purposes, other application like physical therapy and rehabilitation, robotics control or education area, are starting to be consumers of this technology. Furthermore, more common cameras are used for motion identification and tracking in robotics (Dondrup et al., 2015), in security (He et al., 2012) and in human motion representation and imitation with the use of motion capture techniques.

2.2. Motion Data

To carry out this motion recognition, descriptive data from the motion has to be extracted. Depending on the environmental circumstances and type of motion considered, this data should provide information of a different constitution and type. The common data used to recognise motions can give kinematic (e.g. accelerometers or gyroscopes), dynamic (e.g. torques on articulations or reaction forces), bioelectric (e.g. electrical activity of the muscles) or visual (e.g. range cameras or RGB cameras) information.

A common motion capture sensor used to record kinematic data is the Inertial Measurement Unit (IMU). It equips accelerometers, gyroscopes and sometimes magnetometers to report and measure kinematic data of a body like orientations or accelerations. IMUs are used in different areas such as inertial navigation (Tan and Park, 2005), vehicle applications (Händel et al., 2010), behavior analysis (Szemes et al., 2005) and in the biomedical field (Senanayake and Senanayake, 2010).

In human locomotion analysis, IMU sensors have been used in human locomotion analysis like human activity monitoring (Yang and Hsu, 2010), fall detection (Wan et al., 2007) and gesture recognition (Akl et al., 2011). Its light weight and cheap components are the reasons of its widespread use. Taborri et al. (2015) detects the current gait phase of a lower limb orthosis by retrieving data of force and IMU sensors and in Jang et al. (2017) a human gait recognizer is executed at the moment of foot contact as estimated by an IMU on the pelvis. Jang et al. (2017) proposes a two-tier gait recognizer in a lower limb exoskeleton that recognises different actions during five gaits by using two wireless IMUs integrated in a hip exoskeleton. A last IMU setup example that will be mentioned is the one in the multisensor system of the CUHK-EXO (Chen et al., 2017), where the IMUs are mounted on the backpack to obtain the wearer's trunk posture and pressure sensors are designed in the insoles and smart crutches to detect the ground contact. With this information, the wearer's center of gravity can be calculated in real-time and the wearer's motion intention can be estimated by detecting the change in motion data.

Electromyographic Signals (EMG) is bioelectric data that measures the electrical activity of a muscle. For instance, in the sensing system of the HAL-5 exoskeleton (a variant of the exoskeleton of the Figure 1.1(a)), EMG sensors are attached on the wearer's skin to detect the extensor and flexor muscle activities of the knee and hip joints. Another example of use is in Matsubara and Morimoto (2013), where it is introduced a bilinear modeling of EMG signals used to recognise five hand gestures for robotic hand control. Another sensor that measure bioelectric data is the Electroencephalography (EEG) sensor which records electrical activity of the brain. Unfortunately for motion recognition purposes these two sensors present some disadvantages. In the case of EMGs, environmental factors such as user fatigue, sweating, and electrode shift can change surface EMG patterns and degrade classification performance over time and electrode channels may also fail or become noisy due to loss of skin contact (Young et al., 2014). Both sensors have the problem of poor signal quality during long-term use, which is aggravated by low wearing comfort and the quality of the measured values can widely differ concerning the environmental circumstances (Khalili Moghaddam and Lowe, 2019).

In the field of visual data, RGB cameras provide a lot of information about the situation, but are dependent on the focus, illumination and orientation (Chen et al., 2015). Deep cameras, which show distances to points from a specific reference, are not dependent on environmental lighting, but still the angle of view is crucial and they can not offer information about the colours of the objects. 3D cameras, like *Kinect*, combines an infrared projector and camera to generate a grid that locates objects in a certain range. Some of its functionalities and fields of application can be found in Lun and Zhao (2015).

2.3. Derived Features

Once the motion data has been recorded, this set of features can be redundant, not representative enough and too large to be managed. Therefore, in some cases a previous step in applications of machine learning and pattern recognition is carried out consisting of constructing a derived and reduced set of features to facilitate learning, and to improve generalisation and interpretability (Bishop, 2006). In lots of areas we can find the use of derived features. As an example, in computer vision points, edges and objects are extracted from the images and optical character recognition uses line directions, closed loops and line intersection as features.

The extraction of features can become a really complex process in which is needed the use of the domain knowledge of the data to create new features with the target of achieving good results in machine learning algorithms.

A large amount of automation techniques can facilitate this process and, as Freeman (2014) argues, represent a generation stage that expands the number of available features. Linear techniques would be an example where new features that are linear combinations of the inputs are created (Zhu, 2005). In the meanwhile, non-linear techniques use an arbitrary function of the original feature set to extract new values. Variance-based feature extraction is an other technique that aims to create a projection that preserves the variance in the data. As examples we have: Principal Component Analysis (PCA) (Jolliffe and Cadima, 2016), Probabilistic PCA (PPCA) (Park et al., 2017) or Sparse PCA (SPCA) (Zou et al., 2012). Distance-based feature extraction is an other approach to face this issue and aims to preserve or optimize some property of the distance between points, as examples we have Fiser's discriminant analysis (FDA) and Multidimensional scaling (MDS) (Borg and Sireci, 1997). Classification grouping techniques, like clustering, can be adapted to carry out feature extraction. Self-organizing maps (SOM) and spectral clustering (Tsuruta and Aly, 2006) can be used in this instance. Presented as a stochastic optimization problem, in Guo et al. (2010) a method for non-linear feature extraction is introduced using genetic programming and an expectation maximization algorithm (GP-EM). The genetic program creates a tree with input features at the leaf nodes and one feature at the top level of the tree is built from the base features in the set. In the area of connectionist systems, autoencoders are regression neural networks trained such that the output of the system is equal to the input. A layer with fewer nodes than the input creates a bottleneck where node values are a good approximation of the inputs. Features can be extracted by feeding inputs to the system and taking the outputs of the hidden nodes (Holden et al., 2006). Domain-specific tools are usually used to design features. For example, in speech recognition applications often

employ signal processing techniques to generate a set of features from the initial audio signal and that are often more meaningful and trackable. Some features that are commonly used in this area include frequencies, Zero-Crossing Points (ZCP) (Peltonen et al., 2002) and mean frequency (Tchorz et al., 2017).

2.4. Motion Classification Methods

This section explains the different methods that are used in human motion classification. Firstly, we have to divide them in two categories: heuristic rule-based classifiers and automated pattern recognition.

In the heuristic category, not exemplar training data is used, but the motion is specified by a set of rules. For example, these rules can be based on the values of some joint angles of the body (Clark et al., 2013). Afterwards, some parameters of the rules have to be tuned to do the classification of unknown motions. For instance the starting and ending pose of a motion can be used to identify an iteration of this motion.

In automated pattern recognition approaches, a captured template previously registered is compared with the unknown motion. If the template motion is compared directly with the unknown motion to be classified, the approach is called Direct Matching. Dynamic Time Warping (DTW) is the most famous algorithm (Bemdt, 1994). The unknown motion is compared with a template sequence, but first they are aligned to match time and speed of both sets of data (Myers et al., 1979). The differences between the two sequences can be expressed in terms of distance or differences of the correlation coefficients (Wu et al., 2012).

If the template is just used to determine parameters for the model training, then the method belongs to the class of Modeled Based Matching. The model trained is fitted with the data and used to carry out the classification of samples from unknown motions. In this sub-classification, the machine learning methods and artificial neural networks are techniques used. This approach is characterised by the use of large data sets, high compute times and difficulties to carry out an online implementation. Additionally, the set of features has to be carefully selected to extract features with meaningful information about the motion and, additionally an specific tuning is needed in order to train proper models. Machine learning based on motion recognition is usually framed as a classification problem and the trained model is usually referred to as a classifier. The models used with these methods consist of a large number of parameters, which have to be determined in a training step based on pre-labeled motion data. The output of the classifier is a discrete value related to the class in which the unknown motion got the highest probability.

A Hidden Markov Model (HMM) is a model frequently used for motion recognition for a number of reasons. Motion data naturally has a time-based component, as the demonstrator moves through various poses. The model representation of the motions has an intuitive representation, where each state is a primitive motion or pose, and the state transition matrix describes the flow of the motion through these various primitives (Freeman, 2014). Large sets of training data are usually required for HMM to be accurate for human motion recognition (Lun and Zhao, 2015). To give some examples, in Panahandeh et al. (2013) a classification of pedestrian activity and gait analysis based on IMUs is carried out. The pedestrian motion is modeled with a continuous hidden Markov model where the hidden states are in different gait phases. In Gehrig et al. (2009) real-world kitchen task are classified using HMMs based on features consisting of optical flow gradient histograms. For each primitive motion unit they trained one HMM and then concatenated these primitive motion units to form complex motion sequences.

Artificial neural networks (ANNs) refer to a collection of statistical learning algorithms based on biological neural networks. An ANN models the system as a network of neurons with several layers. Among the ANN models that can be found, multi-layer perception (MLP) has been used with motion data (Anjo et al., 2012). NN-based additive nonlinear auto regressive exogenous (NN-ANARX) is an other model that has been used to determine the quality of a rehabilitation exercise in terms of the difference between the observed motion and the predicted motion using the trained model (Nomm and Buhhalko, 2013).

Support Vector Machines (SVMs) have also been used in motion classification in supervised learning models. Madeo et al. (2013) proposed to segment a gesture into a sequence of units and formulate the

gesture analysis problem into a classification task using SVM. Miranda et al. used SVM to identify key poses in a sequence of body motion where the joint angles are used as features. The actual gesture recognition was accomplished via a decision forest. Decision Forests constructed using pre-classified training data have been also used in motion classification by modeling a gesture as a sequence of key poses. Each path from a leaf node to the root was considered a gesture. Miranda et al. (2012) used the decision forest algorithm to identify gestures in real-time on Kinect motion data. Also Randomized Forests have been used in fall detection with the target of recognising the skeleton shape deformation caused by the human body falling (Bian et al., 2012). The meta-algorithm Adaptive Boosting (Adaboost), using decision dumps classifiers as the low-level classifiers, has been used to provide categorical gesture recognition (Filipe and Henriques, 2017).

2.5. Motion Recognition at H²T

In H²T different projects concerning whole-body motion classification and more specifically for Lower Limb Exoskeletons (LLE) have been carried out. Since these works are the baseline of this thesis, they will be explained below in more detail.

2.5.1. Whole-Body Motion

In H²T a reference model of the human body to capture human motions was designed and implemented named Master Motor Map (MMM) (Terlemez et al., 2015). Its main target is the standardisation of human motion representation and mapping to humanoid robots. It pursues the representation of the kinematics and dynamics of the different motions in order to capture and represent interactions of the humans with the environment. As a part of this project, a KIT Whole-Body Human Motion Database (KIT H2T, 2014) has been developed to create a database of high quality human whole-body motion capture (MoCap) recordings.

In Mandery et al. (2016) a dimensionality reduction for the whole-body human motion recognition was carried out. They used a marker-based human motion capture system consisting on 56 markers placed at characteristic anatomical landmarks of a human. The most interesting part regarding this thesis was the extraction of derived features from the Cartesian coordinates of the markers and the recording time steps associated. A total number of 29 features with a total of 702 dimension was extracted to describe the human motion. The main features extracted were: Cartesian velocity vectors of the markers, joint angles and velocities of the 40 joints of the MMM reference model, the location and rotation of the root pose, the whole-body Center of Mass and different whole-body angular momentums. An algorithm named N-Best Feature Subset Exploration was used to search the space of all possible feature subsets for the subsets with best performance. A subset of 4 derived features achieved an accuracy of 94.76% and a subset of 8 features got 95.81%. It was used a motion recognition approach with HMMs and a set used for evaluation of 353 different motion recordings captured from 9 different subjects.

2.5.2. Motion Classification for LLE

H²T introduced in 2013 the exoskeleton KIT-EXO-1 which was created to augment human capabilities and for rehabilitation of the musculoskeletal system. In this first version there are force sensors on the physical human-robot interface that measure the forces between user and exoskeleton during operation. The force data is used to generate an intuitive device control approach, which allows the generation of motion pattern based on interaction force pattern between the exoskeleton and the human (Beil et al., 2015). The first prototype consists of two active DOFs in the knee and the ankle. It can be seen in Figure 2.1.

The work done at H²T about exoskeletons has been oriented towards the motion classification in order to improve intuitive control for exoskeletons. In Beil et al. (2018) a passive exoskeleton was used to create a motion classification method based on Hidden Markov Models. In this first approach three IMUs and seven 3D-force sensors were used to record the data and classify the motions using a training and testing



Figure 2.1.: Exoskeleton KIT-EXO-1 of the H²T created for augmenting human capabilities and to use it with rehabilitation purposes. Picture taken from Beil et al. (2015).

set consisting on 10 subjects performing 13 different motion tasks. Each motion was represented by a fully-connected HMM with 14 states. An accuracy of 92.8% was obtained using a data window size of 300 ms and a stratified 5-fold cross validation.

In Patzer, Isabel and Asfour, Tamim (2019), as a continuation of the aforementioned work, a systematic exploration of the feature space was carried out in order to simplify the sensor setup and consequently, the amount of data used and the training time needed. A trade off between the number of sensors used for the classification and the accuracy obtained was found. With the combination of 3 3D-force sensors and 3 IMUs an accuracy of 92.20% was achieved, which is similar to the one got in Beil et al. (2018) using 3 IMUs and 7 force sensors. The same setup of the motion classification system of the previous work was used to carry out the evaluations.

3. Basics

In this chapter the basics for this thesis are explained more in detail. First, the passive exoskeleton used at H²T is described. Afterwards, the data recorded by the sensors used to train the Hidden Markov Models is introduced. Finally, all the characteristics of the HMMs employed and the possible approaches used to address different problems are explained.

3.1. Passive Exoskeleton

In this section the passive Exoskeleton of H²T will be described as well as the sensors used to capture the data. This passive exoskeleton was introduced in Beil et al. (2018). Figure 3.1 show two pictures of the rendering of the passive exoskeleton's CAD model.



Figure 3.1.: Views of the rendering of the passive exoskeleton's CAD model.

3.1.1. Structure and Design

The passive exoskeleton for the lower left limb was designed for the motion data recording's setup allowing at the same time a natural motion of the user. It consists of three aluminum frame parts for the thigh, the shank and the foot which are connected by orthotic revolute joints at the knee and ankle. The main purpose of the creation of this exoskeleton is to investigate which disposition, amount and type of sensors a functional exoskeleton should bring in order to carry out the best motion classification possible.

The aluminum (EN-AW 5083) that constitutes its structure allows a tightened adjustment to the subject's leg and provides slight compensation of the missing degrees of freedom at the ankle and knee joint during the motion. Velcro straps are used to fix the exoskeleton to the wearer via the anterior thigh and shank and by a shoe at the foot.

3.1.2. Sensors

Force Sensors

A 3D-force sensor is a device with capacitive readout designed and realised for the measurement of mechanical power. The 3D term means that it can measure the normal force and two forces in shear directions and perpendicular between them (Brookhuis et al., 2012).

In our case, seven force sensors model OMD-30-SE-100N are located in strategic locations to gather force interactions by the main important muscles involved in the locomotion. It measures the magnitude and the directions F_x , F_y and F_z using only optical principles. The measurement characteristics of the sensors are not the same in normal and shear directions (Optoforce Ltd., 2016). Nominal force, resolution and deflection are 100N, 6.25mN and 3mm in the normal direction, and 25N, 7mN and 2.5mm in the shear direction. Up to four force sensors can be connected to one data acquisition unit, which records the measured values at a maximum of 100 Hz. Figure 3.2(a) shows the location of the seven force sensors (in red) regarding the muscles on where they are: *m. rectus femoris* (Figure 3.2(a) (a)), *m. biceps femoris* (Figure 3.2(a) (b)), *m. tibialis anterior* (Figure 3.2(a) (c)) and *m. gastrocnemius* (Figure 3.2(a) (d)). In Figure 3.2(b) we can find the distribution of the seven force sensors on the passive exskeleton.

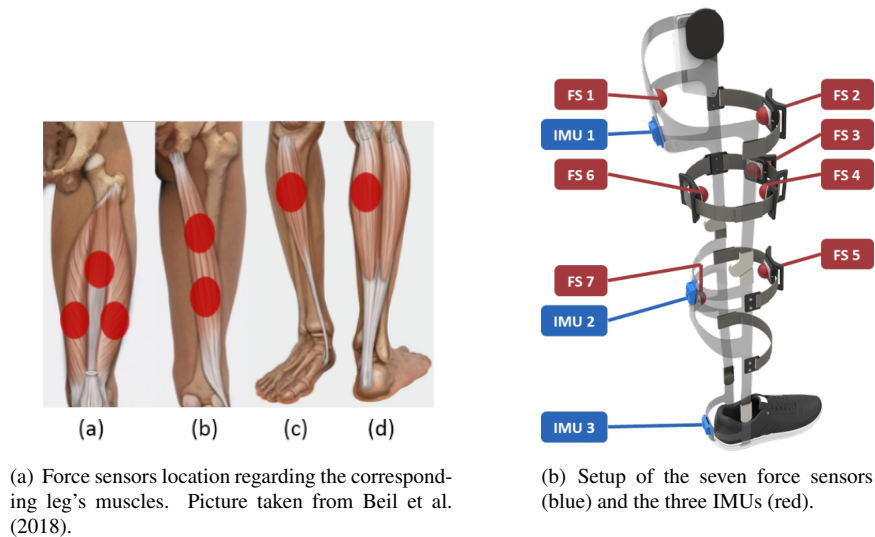


Figure 3.2.: Pictures that relates the sensors' distribution in the passive exoskeleton with the muscles on where they are located

IMUs Sensors

An Inertial Measurement Unit (IMU) is an electronic device that includes a combination of gyroscopes and accelerometers used to record a body's orientation and linear accelerations. The usual configuration is such that the reference system is set using the pitch, roll and yaw axes. A magnetometer is also usually included as a heading reference.

The passive exoskeleton equips three IMUs located in different segments of the leg (thigh, shank and foot). The model of these IMUs is BNO055 from Bosch Sensortec GmbH (Bosch Sensortec GmbH, 2014). This device is a System in Package (SiP) that integrates a triaxial 14-bit accelerometer and a

triaxial 16-bit gyroscope. From this sensors the linear accelerations in Cartesian coordinates and angles represented by quaternions are obtained as raw data. Since quaternions represent an angle in a way that can set two angular configurations, they were transformed to Euler angles (roll, pitch and yaw). Therefore, from all the IMUs, 18 values are obtained: three accelerations and three angles from each IMU sensor. Figure 3.2(b) shows the distribution of the three IMUs located each one on each segment of the leg.

3.2. Motion Data

In this section, the motion data used for training and testing the motion classification system is explained. The data was recorded by three IMUs and seven force sensors while a subject was performing an specific motion. For each motion's recording, data samples of each sensor are registered every 10 ms. Each force sensor captures a force vector which means three force values F_x , F_y and F_z in Newtons. Meanwhile three IMUs register a linear acceleration vector a_x , a_y a_z in m/s and three Euler angles ϕ , θ and ψ in radians. In total we have 21 primary features from the force sensors and 18 primary features from the IMUs. Therefore 39 features are available if only the raw data from the sensors is considered.

For the recordings, ten healthy subjects were chosen to carry out different daily motions which were repeated ten times by each subject. In KIT H2T (2014) we can find the *KIT Whole-Body Human Database* where further information of all the subjects can be found. The main characteristics of each subject are shown in the Table 3.1).

Subj.	Age	Height[cm]	Weight[Kg]	UL circ.[cm]	LL circ.[cm]	UL len.[cm]	LL len.[cm]
674	27	170	63	57.5	39.0	44.0	41.0
917	31	172	70	54.5	38.0	44.5	42.0
1717	22	178	74	54.5	35.0	45.5	41.5
1718	17	175	60	50.5	35.0	43.0	41.5
1719	24	172	66	59.0	36.5	45.0	45.0
1720	26	170	56	51.5	36.0	41.5	43.5
1722	25	171	64	61.0	37.0	42.0	37.0
1723	22	168	62	56.5	38.5	37.0	37.0
1724	29	175	75	55.0	36.5	44.0	39.5
1725	28	178	70	52.0	36.0	41.0	42.5
Avg.	25.1	172.9	66.0	55.2	36.8	42.8	41.1
Dev.	3.9	3.0	5.9	3.2	1.3	2.6	2.3

Table 3.1.: Overview of subject characteristics and physical parameters. UL denotes Upper Leg and LL denotes Lower Leg.

Each of the ten subjects repeated ten times different daily motions wearing the passive exoskeleton. The basic motions performed are the ones that follow (with its abbreviation in brackets): *Walking Forward* (WF), *Walking Backward* (WB), *Turn Left* (TL), *Turn Right* (TR), *Sidesteps Right* (SR), *Sidesteps Left* (SL), *Going Upstairs* (GU), *Going downstairs Backwards* (GB), *Going Downstairs* (GD), *Lift Object* (LO), *Drop Object* (DO), *Stand Up* (SU), *Sit Down* (SD) and *Stand* (ST).

Except for the motions TR, SR, DO, LO, SU, SD and ST, each motion had to be started with the leg on which the exoskeleton is worn. Before and after each repetition, the subjects stood still (not recorded). Also we have to inform that from the recordings, three repetitions of one IMU of the person ID1722 are missing for the motion SU and in all the repetitions of the motions SD. Likewise, the data of the motion GU execution is missing for subject ID1725.

3.3. HMM-Based Motion Classification

This section introduces the method that will be used to carry out the motion classification. Its parameters and other properties are explained in the subsequent sections. The notation has been adopted from Stamp (2018) and some parts of the explanation from Rabiner (1989).

3.3.1. Hidden Markov Model

A Markov Model represents a system that can be described at any time as being in one of a set of N different states S_1, S_2, \dots, S_N and that evolves from one state to another (possibly back to the same state) according to a set of probabilities associated with the state (Rabiner, 1989). Figure 3.3 shows the description of this random problem represented by a *Markov chain*. In this figure a model is represented by five states each one being a node of the graph. All the nodes are interconnected using oriented edges and each one has associated a transition probability a_{ij} representing the probability of the Markov process of changing from one state to another state.

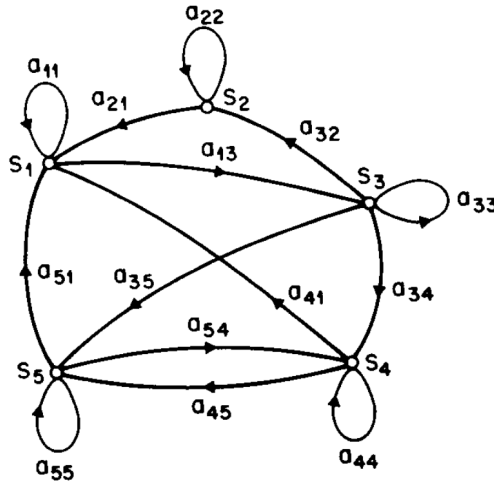


Figure 3.3.: A Markov chain with five states (labeled from S_1 to S_5) with selected state transitions. Picture from Rabiner (1989).

The particularity of a Hidden Markov Model is that the states are unobserved (hidden states). That means that these unobservable states are those that we want to predict but at the same time they are not possible to observe. Therefore we use other variables named *Observations* ($O_0, O_1, O_2, \dots, O_{T-1}$), which are possible to measure, correspond to the physical output of the system being modeled and are related to the states. The individual symbols of the observation is denoted as $V = V_0, V_1, V_2, \dots, V_M$.

The probability to pass from one state to an other (or to remain in the same one) along the time, is represented using a state transition probability matrix \mathbf{A} , which is quadratic and its dimensions are equal to the number of states N in the model (Expression 3.1). Each element a_{ij} of \mathbf{A} represents the probability to pass from one state to another. To represent the probability of being in the actual state at a specific time t is denoted as q_t .

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{N1} & \dots & \dots & a_{NN} \end{bmatrix} \quad a_{ij} = P(\text{state } q_j \text{ at } t+1 \mid \text{state } q_i \text{ at } t) \quad (3.1)$$

It is also needed to know the probabilities between observation and states. Each observation has a probability that denotes how likely is that a certain state takes place when it is measured that observation. These probabilities are grouped in a observation probability matrix \mathbf{B} where M is the number of distinct observation symbols per state, implying a rectangular matrix of dimensions $N \times M$ (Expression 3.3). The probabilities $b_j(k)$ are independent of t .

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1M} \\ b_{21} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ b_{N1} & \dots & \dots & b_{NM} \end{bmatrix} \quad b_j(k) = P(v_k \text{ at } t \mid \text{state } q_j \text{ at } t) \quad (3.2)$$

To estimate the first state it is needed a vector that shows the probability of starting in each state. This vector is named *initial state distribution* and it represented as π (Expression 3.3)

$$\pi = [\pi_1 \quad \pi_2 \quad \dots \quad \pi_N] \quad \pi_i = P(q_1 = S_i) \quad (3.3)$$

The transition from one state to the next is a *Markov process* of order 1, meaning that the probability of the subsequent state depends only on the current state and the probabilities of the transition matrix \mathbf{A} are fixed. This idea can be expressed with

$$P(s_i^t | s_i^1, s_i^2, \dots, s_i^{t-1}) = P(s_i^t | s_i^{t-1}) \quad (3.4)$$

All these matrices define the Hidden Markov Model $\lambda = (A, B, \pi)$ and they obey standard stochastic constraints, which means that the sum of probabilities of each row is always 1 and its values follow a probability distribution.

$$a_{ij} \geq 0 \quad b_j \geq 0 \quad \pi_i \geq 0 \quad (3.5)$$

$$\sum_{j=1}^N a_{ij} = \sum_{j=1}^N b_j = \sum_{i=1}^N \pi_i = 1$$

Figure 3.4 shows a generic hidden Markov model in which $X_0, X_1, X_2, \dots, X_{T-1}$ represents the hidden state sequence that is determined by the current state and the *Transition matrix* \mathbf{A} . The sequence of observation variables $O_0, O_1, O_2, \dots, O_{T-1}$ are the only ones that are possible to observe and they are related to the hidden states by the *Observation probability matrix* \mathbf{B} . The dashed line separates the hidden part from the one that can be observed.

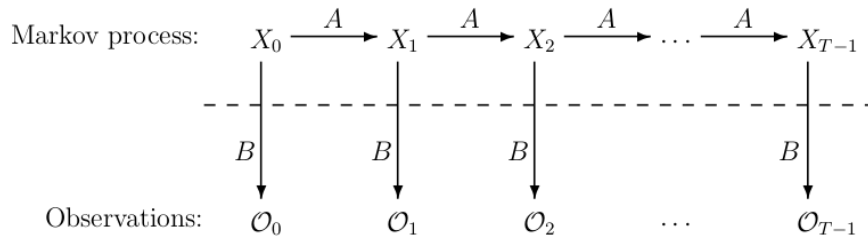


Figure 3.4.: Diagram of a generic Hidden Markov Model. Picture from Stamp (2018).

Once a Hidden Markov Model is defined $\lambda = (A, B, \pi)$, it can be used to give an observation sequence $O_0, O_1, O_2, \dots, O_{T-1}$ as follows:

1. An initial state is chosen $q_1 = S_i$ using the Initial state distribution π
2. It is chosen $O_t = v_k$ depending on the symbol probability distribution of the current state S_i , $b_i(k)$
3. Transition to a new state $q_{t+1} = S_j$ considering the transition probability matrix A and the corresponding element a_{ij} .
4. It is set $t = t + 1$, and then return to step 2 if $t < T$. Otherwise the process is over.

3.3.2. HMM with Continuous Observations

The previous section considered that observations were characterised as discrete values taken from a finite set of data with which a discrete probability density can be used for each state. However, as it will be introduced in next chapters, the observation used in this thesis and in other application of HMMs are continuous data vectors, that require a continuous probability density within each state of the model. To be able of using a continuous observation space instead of a finite number of discrete values, the components of the observation probability matrix \mathbf{B} (Expression 3.3) must be replaced with a continuous probability distribution.

For continuous observation, the probability density function of the D-dimensional observation O_t , related to the state i , using a Gaussian distribution with mean value vector μ and Σ (Bishop, 2006) is

$$b_i(O) = (2\pi)^{-\frac{D}{2}} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(O_t - \mu)^T \Sigma^{-1}(O_t - \mu)\right) \quad (3.6)$$

3.3.3. Types of HMMs

In Beil et al. (2018) and in Patzer, Isabel and Asfour, Tamim (2019), only a fully connected topology was used in the motion classification system. This kind of model, also called ergodic, has the property that every state can be reached from every other state of the model. That supposes that in the Transition Matrix (Expression 3.1) every single component is such that $a_{ij} > 0$ for all i, j . Figure 3.3 shows a graph that represents this kind of topology using 5 states.

Different types of HMMs can be used depending on the observed properties of the signal being modeled. As an example, Figure 3.5 shows a Left-Right model or Bakis model (Saerens, 1993), in which the state sequence associated with the model has the characteristic that as time increments the state index increases or stays in the same state. In the example shown the states proceed from left to right. This kind of topology is really suitable to model signals whose properties and constitution change over time. A known application is the use of this model for speech recognition. In our case of motion recognition, where each motion changes over time and has a continuity, this topology might improve the representation of the HMM. In this model the transition matrix is triangular with all the elements under the diagonal equal to 0. If we consider the Expression 3.1, the state transition properties of a left-right model have the characteristic that $a_{ij} = 0$ if $j < i$.

Another type of HMM is the Cyclic linear topology. This one is characterised by having the same structure as the Left-Right, but in this case it is possible to proceed from the last state to the first one creating a cycle. Concerning the transition matrix, this topology consists also in a triangular transition matrix with 0s under the diagonal with the exception of the element a_{N1} which is greater than 0. The graph of this model is similar to the one of Figure 3.5, but with a transition with direction from state 4 to state 1. This kind of HMM was used in Panahandeh et al. (2013) in a joint activity and gait-phase classification. One great advantage of topologies with transition matrix where some a_{ij} are 0s is the reduction of calculations to train the model.

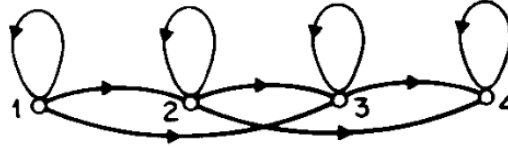


Figure 3.5.: Left-Right model with 4 states. The states proceed from left to right. In this example there is the exception that state 4 is not reachable from state 1. Picture taken from Stamp (2018).

3.3.4. Fundamental Problems for HMMs

In this section some application of HMMs are presented to face different issues. The Evaluation problem and the Learning problem approaches will be the application of the HMMs that will be used in this thesis. The Learning problem will be used to train a specific model for each motion and the Evaluation problem to classify each new data sample in its corresponding motion.

Evaluation Problem

In this problem it is pursued to check if an observation sequence O_1, O_2, \dots, O_T belongs to a given HMM model $\lambda = (A, B, \pi)$. A probability is obtained that informs about how likely was that this model generated that specific sequence O . The probability of the observation sequence given the model is expressed as $P(O|\lambda)$.

This kind of problem can be solved by considering every state sequence of length T (number of observations) and then calculate the probability of the observation sequence O for each state sequence $Q = q_1, q_2, \dots, q_T$. If we consider statistical independence of observations, then we get that

$$P(O|Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T) \quad (3.7)$$

The probability of O is got by summing a joint probability over all possible sequences q such that

$$P(O|\lambda) = \sum_{all Q} P(O|Q, \lambda) P(Q|\lambda) = \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \cdots a_{q_{T-1} q_T} b_{q_T}(O_T) \quad (3.8)$$

To solve this problem a forward-backward procedure is required since otherwise the calculations are computationally unfeasible. It considers the forward variable $\alpha_t(i)$ defined as

$$\alpha_t(i) = P(O_1, O_2 \cdots O_t, q_t = S_i | \lambda) \quad (3.9)$$

which gives the probability of the partial observation sequence $O_1, O_2 \cdots O_t$ and state S_i at time t given the model λ . And the backward variable $\beta_t(i)$ such

$$\beta_t(i) = P(O_{t+1}, O_{t+2} \cdots O_T | q_t = S_i, \lambda) \quad (3.10)$$

which gives the probability of the partial observation sequence from $t + 1$ to the end, given state S_i at time t . For more details about it, please consult Rabiner (1989).

Decoding Problem

In this problem it is pursued to find the correct state sequence. Taken into account an HMM model $\lambda = (A, B, \pi)$ and an observation sequence $O = O_1, O_2, \dots, O_T$, it is calculated the most likely sequence

of hidden states $Q = q_1, q_2, \dots, q_T$ produced by the observation sequence O .

Finding the correct state sequence means looking for the optimal one, which can differ depending on the searching criteria (Rabiner, 1989). If we want maximise the expected number of correct individual states, we would choose the states q_t that are individually more likely. To solve this problem the following variable is defined

$$\gamma(i) = P(q_t = S_i | O, \lambda) \quad (3.11)$$

which defines the probability of being in state S_i at time t given the observation sequence O and the model λ . This equation can be expressed in terms of the forward-backward variable

$$\gamma(i) = \frac{\alpha_t(i) \cdot \beta_t(i)}{\sum_{i=1}^N \alpha_t(i) \cdot \beta_t(i)} \quad (3.12)$$

Using $\gamma(i)$, the next optimization problem should be solved to obtain the individually most likely state q_t at time t as

$$\arg \max_{1 \leq i \leq N} [\gamma(i)] \quad (3.13)$$

For more details about it, please consult Rabiner (1989).

Learning Problem

This problem attempts to optimize the model parameters in such way that the output of a given observation sequence is predicted using a specific criterion. Therefore, this problem tries to define an HMM model with all its parameters $\lambda = (A, B, \pi)$. To do so, it is given some training observation sequences O_0, O_1, \dots, O_{T-1} and the definition of its structure (the number of hidden and visible states). The model parameters $\lambda = (A, B, \pi)$ are chosen in order to maximise $P(O|\lambda)$.

The Baum-Welch method or gradient techniques (Levinson et al., 1983) are used in order to chose $\lambda = (A, B, \pi)$ such that $P(O|\lambda)$ is a global maximum, although it can only provide a local maximum. The HMM parameters are reestimated and updated in each iteration of the procedure looking for the best representative model. For that purpose, the probability of being in state S_i at time t , and state S_j at time $t + 1$, given the model and the observation sequence is

$$\xi_t(i, j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda) \quad (3.14)$$

As it was defined in the *Decoding Problem* case, $\gamma(i)$ is defined as the probability of being in state S_i at time t , given the observation sequence and the model. It is possible to relate $\gamma(i)$ with $\xi_t(i, j)$ in such way that

$$\gamma(i) = \sum_{j=1}^N \xi_t(i, j) \quad (3.15)$$

If we do a sum of $\gamma(i)$ over the time indexes, we would get the expected number of times that state S_i is visited. If the summation is done with $\xi_t(i, j)$, the number of transitions from S_i to S_j are calculated. Then, the set of reestimation formulas for π , **A** and **B** are the following ones:

$$\bar{\pi}_i = \gamma_1(i) \quad \bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad \bar{b}_j(k) = \frac{\sum_{t=1}^T 1_{(s.t. O_t = v_k)} \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)} \quad (3.16)$$

If the current model is $\lambda = (A, B, \pi)$ and the reestimated model is $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$, model $\bar{\lambda}$ is more likely than model λ since $P(O|\bar{\lambda}) > P(O|\lambda)$. That means that the observation sequence considered is more probable to have been produced with this new model. For more details about the Learning Problem approach, please consult Rabiner (1989).

3.4. Model Validation

In order to estimate the accuracy and predict the performance in practice of a classification system, a model validation technique is carried out on a data set. A *Cross-validation* technique is the one chosen in this work as it was already employed in previous works on which this thesis is based (Beil et al. (2018), Patzer, Isabel and Asfour, Tamim (2019) and Mandery et al. (2016)). It targets to assess how the results of a statistical analysis would be for an unrelated data set. In our case, the main reason behind the use of a *Cross-validation* is the limited data available to evaluate the models. The main idea of this technique is training a model with all the data but for a portion that is kept aside to use it later for testing.

The type of *Cross-validation* techniques is named *K-Folds Cross Validation* and its main characteristic is that every observation from the whole dataset is used in the training and test set. More specifically in this work, a *Stratification* in the validation is used, and it consists of keeping the proportion of data points for each class (type of motion) in every fold, in such way that the data is divided equally among training and test folds. Figure 3.6 shows a *5-Folds Cross Validation* and a summary of the process. The train and testing process follows the following steps:

1. The entire data set is randomly split into k folds, ideally between 5 and 10 considering the data size (Ohannessian, 2017). The higher is the k value, less biased would be the model. However the *bias-variance trade-off* should be such that keeps the variance in a low value.
2. The model is fitted using $K - 1$ folds and the isolated fold is used to test it.
3. Points 1 and 2 are repeated until each fold has been employed as a testing fold. Each iteration is call a *Round* and for each one the results should be stored to be considered once the process is over.
4. Evaluate the results of each *Round* to extract the descriptive results of the process. The evaluations that will be introduced in Section 3.5 are some possibilities to extract information about the performance of the models.

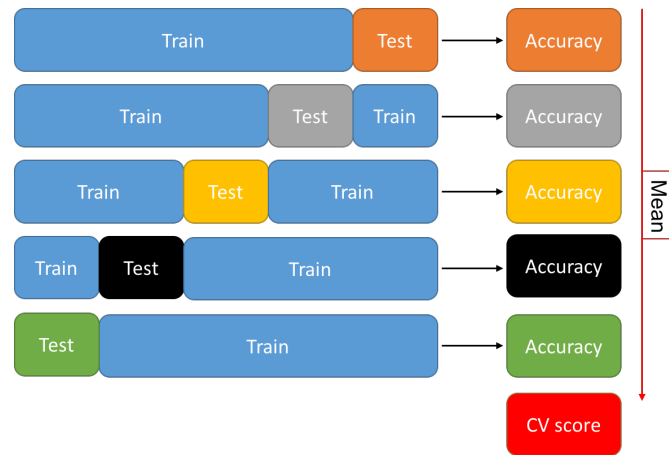


Figure 3.6.: Diagram of a *5-Folds Cross Validation* in which the accuracy metric is extracted in each iteration in order to calculate a score associated with the *Cross-validation* at the end of the process. Picture taken from Drouin, Alexander and Laviolette, François (2019).

3.5. Evaluation of Results

This section explains how the success of the classification system is evaluated. Below the definitions of the quantitative methods to represent this performance are presented. First the definition of the term *Accuracy* is explained. Following the *Recall* and *Precision* are introduced to explain afterwards the idea behind the *F₁ score* measurement.

During this section the following notation concerning the classification of motion samples are used: *TP* are true positives, *FP* are false positives, *FN* are false negatives and *TN* are true negatives.

3.5.1. Accuracy

The *Accuracy* can be used to evaluate the performance of our model. In simple words, it is the ratio of correct predictions to the total number of samples predicted. In the context of a supervised learning approach, as in the case of our motion classification, it can be defined as the number of data points correctly classified as positive *TP* or negative *TN* divided by the total number of samples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.17)$$

We have to keep in mind that this metric gives a meaningful value, only if the number of samples belonging to each class is equal or at least similar. If there are more samples of one class comparing to others, a high accuracy for that class might not be a good result. If the cost of misclassification for the minor classes is high, this metric would not be the most suitable evaluation expression to use.

3.5.2. Precision and Recall

Precision is an indicative that attempts to inform about the proportion of positive identifications that was actually correct. In our case it quantifies the percentage of motion samples classified as an specific class that actually belong to that class.

$$Precision = \frac{TP}{TP + FP} \quad (3.18)$$

The statistical measure *Recall*, also called *Sensitivity*, is the measurement of the proportion of positive identifications that are correctly identified as this type. In the motion classification system, the *Recall* gives the percentage of motion samples of an specific type that have been identified as such and not as another motion one. In this case it is necessary to take into account the classifications carried out for the other motions.

$$Recall \text{ or } Sensitivity = \frac{TP}{TP + FN} \quad (3.19)$$

3.5.3. F₁ Score

This statistical measure is the weighted average of *Precision* and *Recall* (Van Rijsbergen, 1979). It gives a valuable information if the cost of getting false positives is similar to the cost of getting false negatives. Otherwise it will be also necessary to take a look at the *Recall* and the *Sensitivity*. Depending on if it is pursued to reduce false negatives or false positives, the precision term can be weighted. For example an *F₂* would allow to push the reduction of false negative. The Expression 3.21 shows how could be possible to weight this measure. In the previous example β would take 2.

$$F_1score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (3.20)$$

$$F_\beta = (1 + \beta^2) \cdot \frac{Precision \cdot Recall}{(\beta^2 \cdot Precision) + Recall} \quad (3.21)$$

3.5.4. Results in Multi-Class Classification

When the classification has to be carried out among multiple classes, the *Accuracy* can be defined as a percentage of the correctly classified data points among all data points. In our approach, each observation is evaluated in each HMM of the motions considered and the one that got the highest likelihood is classified as that motion.

Meanwhile for *Sensitivity*, *Precision* and *F₁ Score*, one class is considered and the remaining classes are summarised as another global class. The average of the *F₁* scores of all the classes is calculated in order to calculate the global *F₁* score of the whole classification. This average measure was already used in Mandery et al. (2016) in which the number of data points of each class was considered to calculate the *F1* of the classification. The following expression shows this weighing

$$F_{1,Cassifier} = \frac{\sum_{i=1}^N n_i \cdot F_{1,i}}{\sum_{i=1}^N n_i} \quad (3.22)$$

where n_i is the number of data points for the class i , $F_{1,i}$ is the score obtained on the class i and N is the number of classes (in our case the number of motions considered).

4. Derived Features

This chapter introduces the new derived features used to carry out the Hidden Markov Model (HMM) based motion classification. These features are calculated from the raw data and the intention is to use more significant data regarding the human motion and reduce the dimensionality needed without affecting the performance.

4.1. Joint Moments

On the basis of using derived features from the raw data, it was decided to use the knee and ankle moments as new derived features due to the dynamic relevance that they mean for an human motion. Mansur et al. (2013) suggests to use the dynamic features instead of kinematic features for human action recognition using HMM. It is argued that as they are derived from the physics-based representation of the human body, such as the torques from some joints, they have a lower dimension than kinematic features and less motion data is required to train the HMMs. In this section it is explained the calculation of the knee moments and the ankle moments based on the force sensor values.

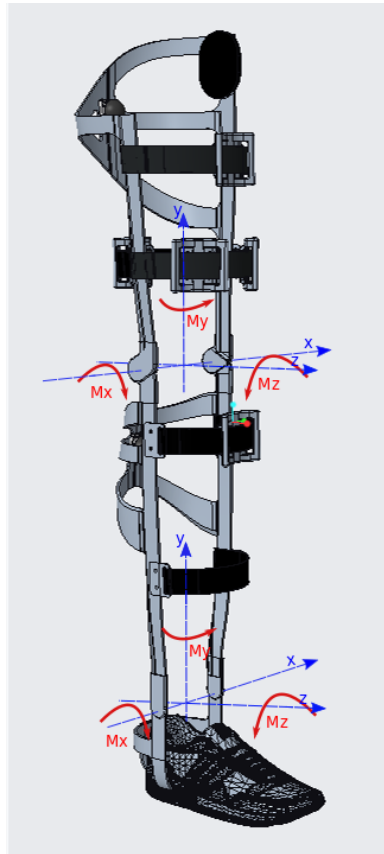


Figure 4.1.: Representation of the knee and ankle moments on the common reference systems

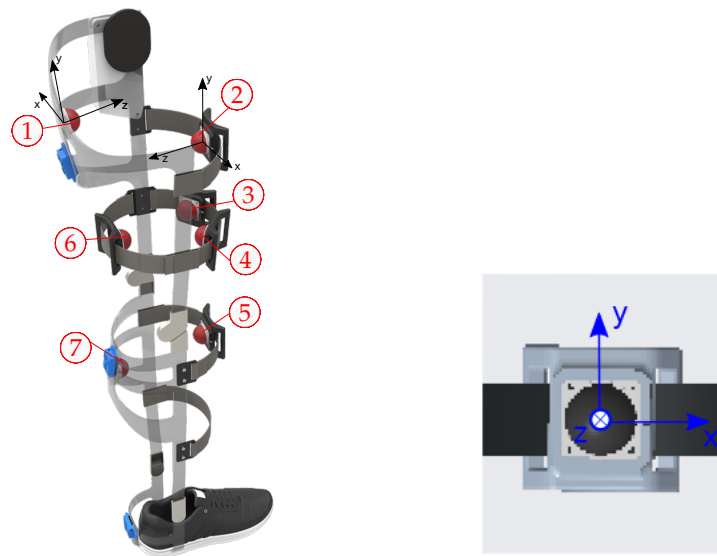
4.1.1. Calculation of the Moments

To calculate the knee moment vector was considered a common reference system set up in the middle of the rotation knee axis, in such way that Y-axis is vertical, X-axis goes along the rotation knee axis and Z-axis direction is positioned forming a right-hand rule coordinate system in combination with the other axes. Figure 4.1 shows a representation of the knee moments on the new common reference system. The same oriented common reference system was used, but with the X-axis going along the rotation ankle axis, to calculate the ankle moments. Using the dimensions of the exoskeleton CAD model and considering the reference system of each force sensor, each force vector was transformed into the common reference system aforementioned. We should point out that the moments calculated are not the real moments that the knee or ankle are doing, but how much are the muscles, where the force sensors are situated, contributing to create these moments. The dynamics of the exoskeleton, the weights of different parts of the leg, the force reaction between ground and foot and other factors should be taken into account to calculate a better approximation of the moment vector.

In this section it is explained only the calculation of the knee moments based on the force sensor values. The procedure is analogous with the calculation of the ankle moments. The expression of both moments are shown in the next section.

Coordinate axes of the force sensors

The coordinate axes of each sensor are oriented as it is shown in Figure 4.2(a). Each sensor has its own orthogonal coordinate system. The axes orientation is always such that y axis is upright oriented, z axis points towards the inside of the leg and x axis follows Right-hand rule with the others. Figure 4.2(b) shows these axes for one generic force sensor.



(a) Force sensors distribution in the exoskeleton with reference numbers. Reference axes of sensors 1 and 2 are shown.

(b) Orthogonal coordinate system of one force sensor.

Figure 4.2.: Orthogonal coordinate system of the force sensors.

Transformation matrices

In order to calculate more easily the knee moments, the reference systems of each sensor is transformed in a way that all of them have the same orientation. The global basis chosen B' to use is the one that appears in the Figure 4.3(a), in blue color with the x axis as the knee rotational axis.

To find the change of coordinates matrix from the basis of each sensor B to the new global basis B' , the basis vector of B is expressed as linear combination of the basis vectors of B' .

Being $B = (\vec{u}_1, \vec{u}_2, \vec{u}_3)$ a basis of one of the sensors and $B' = (\vec{v}_1, \vec{v}_2, \vec{v}_3)$ the new basis, the coordinates of \vec{u} relative to the new basis B' can be expressed uniquely as a linear combination of \vec{v} :

$$\begin{aligned}\vec{u}_1 &= a_1 \cdot \vec{v}_1 + a_2 \cdot \vec{v}_2 + a_3 \cdot \vec{v}_3 \\ \vec{u}_2 &= b_1 \cdot \vec{v}_1 + b_2 \cdot \vec{v}_2 + b_3 \cdot \vec{v}_3 \\ \vec{u}_3 &= c_1 \cdot \vec{v}_1 + c_2 \cdot \vec{v}_2 + c_3 \cdot \vec{v}_3\end{aligned}\tag{4.1}$$

with \vec{a} , \vec{b} and \vec{c} the coordinates of \vec{u} relative to the basis B' . For each sensor the following expressions are obtained. The number of each sensor corresponds to the number scheme of Figure 4.2.

<i>Sensor 1 :</i>	<i>Sensor 2 :</i>	<i>Sensor 3 :</i>
$\vec{u}_1 = \vec{v}_1$	$\vec{u}_1 = -\vec{v}_1$	$\vec{u}_1 = -\vec{v}_1 \cdot \cos(\alpha) + \vec{v}_3 \cdot \sin(\alpha)$
$\vec{u}_2 = \vec{v}_2$	$\vec{u}_2 = \vec{v}_2$	$\vec{u}_2 = \vec{v}_2$
$\vec{u}_3 = \vec{v}_3$	$\vec{u}_3 = -\vec{v}_3$	$\vec{u}_3 = -\vec{v}_1 \cdot \sin(\alpha) - \vec{v}_3 \cdot \cos(\alpha)$

(4.2)

<i>Sensor 4 :</i>	<i>Sensor 5 :</i>	<i>Sensor 6 :</i>
$\vec{u}_1 = -\vec{v}_1 \cdot \cos(\alpha) - \vec{v}_3 \cdot \sin(\alpha)$	$\vec{u}_1 = -\vec{v}_1$	$\vec{u}_1 = \vec{v}_1$
$\vec{u}_2 = \vec{v}_2$	$\vec{u}_2 = \vec{v}_2$	$\vec{u}_2 = \vec{v}_2$
$\vec{u}_3 = \vec{v}_1 \cdot \sin(\alpha) - \vec{v}_3 \cdot \cos(\alpha)$	$\vec{u}_3 = -\vec{v}_3$	$\vec{u}_3 = \vec{v}_3$

(4.3)

<i>Sensor 7 :</i>
$\vec{u}_1 = \vec{v}_1$
$\vec{u}_2 = \vec{v}_2$
$\vec{u}_3 = \vec{v}_3$

(4.4)

The change of basis matrix to change from B to the basis B' regarding the expressions 4.1 is

$$M = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix}$$

This allows to find the force vectors \vec{w} of the sensor i in the common reference system defined by B' such that

$$|\vec{w}_i|_{B'} = M_i |\vec{w}_i|_B$$

Each reference system of each sensor i has its own change of basis matrix

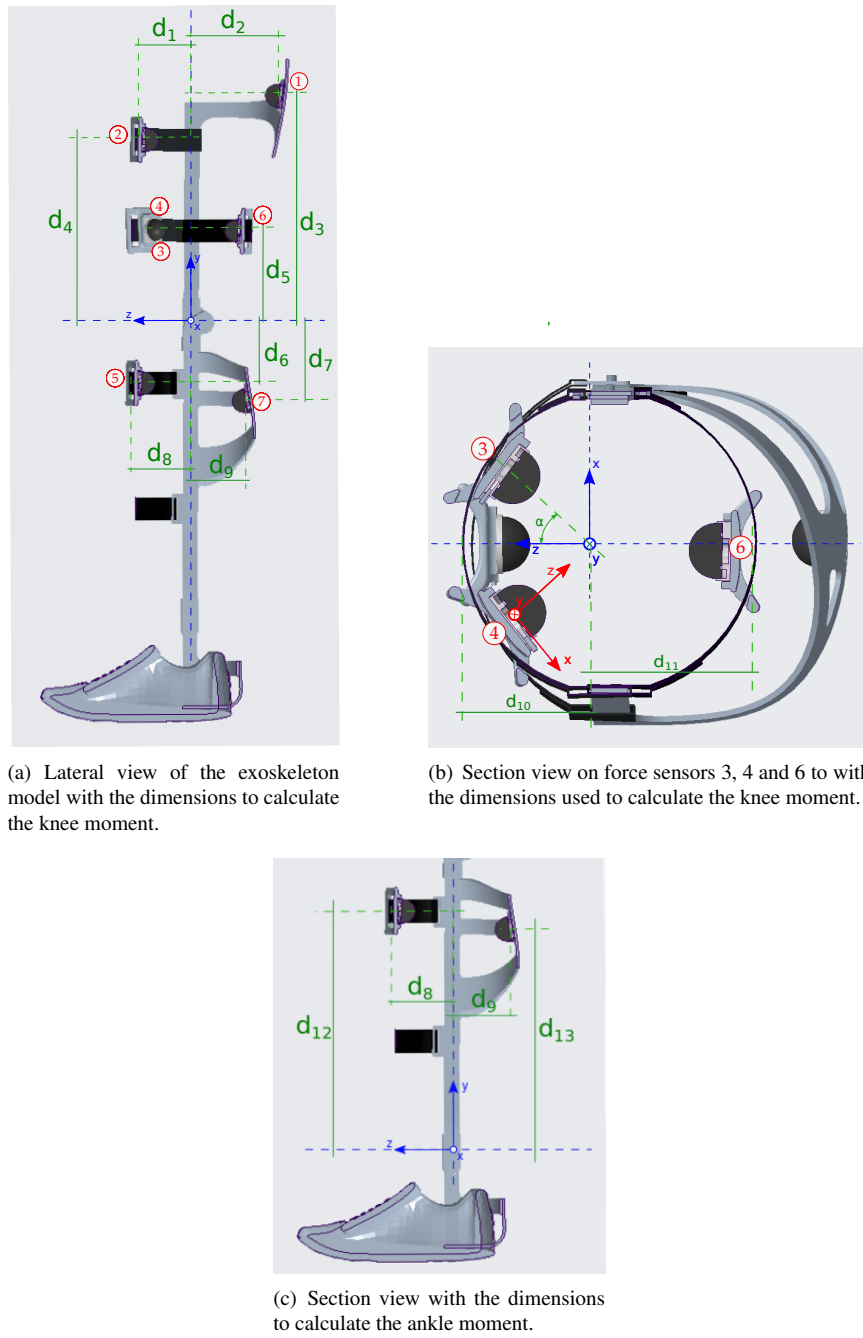


Figure 4.3.: Main dimensions of the exoskeleton model used to compute the moments.

$$M_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad M_2 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad M_3 = \begin{pmatrix} -\cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & -\cos(\alpha) \end{pmatrix} \quad (4.5)$$

$$M_4 = \begin{pmatrix} -\cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & -\cos(\alpha) \end{pmatrix} \quad M_5 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad M_6 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.6)$$

$$M_7 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.7)$$

4.1.2. Expressions of the Moments

Knee Moments

The expression of the moments are computed using the vector values in the new basis B' . The total moment is split in the one produced by the upper leg Γ_{ul} and the lower leg Γ_{ll}

$$\Gamma = \Gamma_{ul} + \Gamma_{ll} = \begin{bmatrix} \Gamma_{ul}|_x \\ \Gamma_{ul}|_y \\ \Gamma_{ul}|_z \end{bmatrix} + \begin{bmatrix} \Gamma_{ll}|_x \\ \Gamma_{ll}|_y \\ \Gamma_{ll}|_z \end{bmatrix} \quad (4.8)$$

The expressions of the upper leg and lower leg using the dimensions in Figure 4.3(a) and Figure 4.3(b) to calculate the knee moments are:

$$\begin{aligned} \Gamma_{ul}|_x &= FS1_y \cdot d_2 + FS1_z \cdot d_3 - FS2_y \cdot d_1 + FS2_z \cdot d_4 + FS3_z \cdot d_5 \\ &\quad - FS3_y \cdot d_{10} \cdot \cos(\alpha) + FS4_z \cdot d_5 - FS4_y \cdot d_{10} \cdot \cos(\alpha) + FS6_z \cdot d_5 + FS6_y \cdot d_{11} \\ \Gamma_{ul}|_y &= -FS1_x \cdot d_2 + FS2_x \cdot d_1 + FS3_x \cdot d_{10} \cdot \cos(\alpha) - FS3_z \cdot d_{10} \cdot \sin(\alpha) \\ &\quad + FS4_x \cdot d_{10} \cdot \cos(\alpha) + FS4_z \cdot d_{10} \cdot \sin(\alpha) - FS6_x \cdot d_{11} \\ \Gamma_{ul}|_z &= -FS1_x \cdot d_3 - FS2_x \cdot d_4 + FS3_y \cdot d_{10} \cdot \sin(\alpha) - FS4_y \cdot d_{10} \cdot \sin(\alpha) \\ &\quad - FS6_x \cdot d_5 - FS3_x \cdot d_5 - FS4_x \cdot d_5 \end{aligned} \quad (4.9)$$

$$\begin{aligned} \Gamma_{ll}|_x &= -FS7_z \cdot d_7 - FS5_z \cdot d_6 - FS5_y \cdot d_8 + FS7_y \cdot d_9 \\ \Gamma_{ll}|_y &= FS5_x \cdot d_8 - FS7_x \cdot d_9 \\ \Gamma_{ll}|_z &= FS5_x \cdot d_6 + FS7_x \cdot d_7 \end{aligned} \quad (4.10)$$

Ankle Moments

As mentioned before, the procedure to calculate the ankle moments is analogous. Therefore, only the expressions of the ankle moments will be shown in this section. Using the dimensions in Figure 4.3(c) the following expressions are obtained (note that in this case only the force sensors of the lower leg are used):

$$\begin{aligned}
\Gamma_{II}|_x &= FS5_z \cdot d_{12} - FS5_y \cdot d_8 + FS7_z \cdot d_{13} + FS7_y \cdot d_9 \\
\Gamma_{II}|_y &= FS5_x \cdot d_8 - FS7_x \cdot d_9 \\
\Gamma_{II}|_z &= -FS5_x \cdot d_{12} - FS7_x \cdot d_{13}
\end{aligned} \tag{4.11}$$

4.2. Leg Joint Angles

In this section two more derived features are introduced. These features are extracted from the Euler angles of the IMUs. These two new derived features are the joint angles of the knee and the ankle, which are representative for the kinematic description of a motion. The knee and ankle joint angles extracted from the recordings are relative angles, which provide a value relating to the orientation between two parts of the leg: the angle between the thigh and the shank (joint angle of the knee) and the angle between the shank and the foot (joint angle of the ankle). Figure 4.4(a) shows these two angles as β for the knee joint angle and γ for the ankle joint angle.

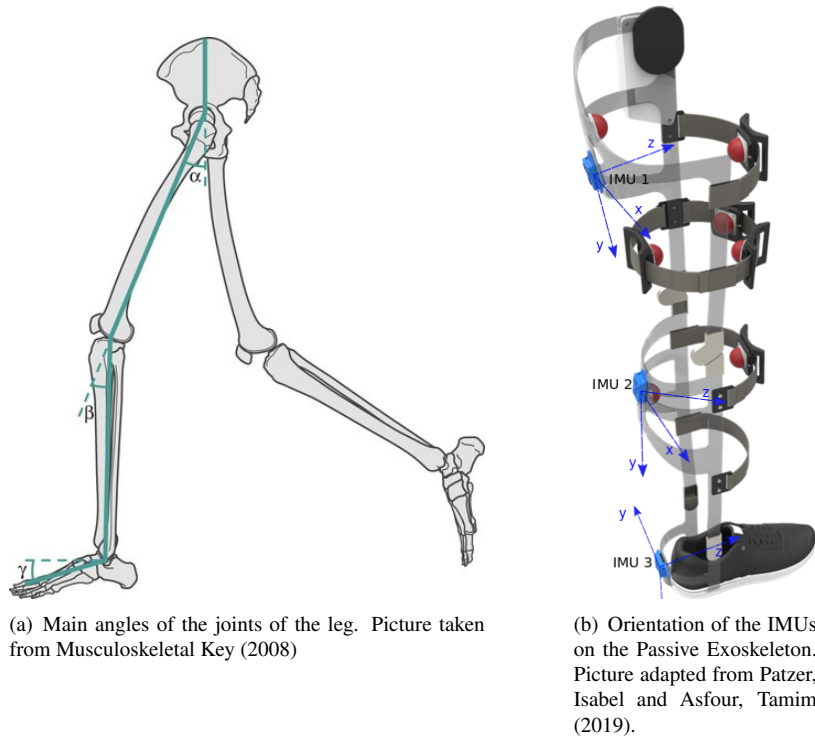


Figure 4.4.: Representation of the angles that are pursued to calculate based on the IMU values of the Passive Exoskeleton.

To calculate the knee joint angle it is considered the orientation of the two IMUs sensors that are located on the thigh (IMU 1) and on the shank (IMU 2). Then, taking into account the reference system of each IMU and the component of the Euler angle that describes a rotation about an axis parallel to the rotation axis of the knee, as can be seen on the Figure 4.4(b), it is assumed that

$$\Phi_{knee} = \Phi_{IMU1x} - \Phi_{IMU2x} \tag{4.12}$$

The same idea can be used to calculate the joint angle of the ankle, but this time the pair of IMUs considered is the one located on the shank (IMU 2) with the one located on the foot (IMU 3). Therefore the expression to calculate the joint angle of the ankle is

$$\varphi_{ankle} = \varphi_{IMU3y} - \varphi_{IMU2x} \quad (4.13)$$

It is important to stand out that the angles calculated are approximations to the joint angles of a human leg, fundamentally because the lower limb exoskeleton considered can only perform rotations around one axis, meanwhile the human knee can perform rotations around three axes.

5. Evaluations

In this chapter the main two topics of this thesis are introduced. Firstly, a feature analysis of the moments is carried out to examine the intra and inter subject motion resemblance. The results extracted will be used to analyse our data base and spot other issues regarding the motion classification. The second topic consists of using the derived features, previously introduced, in a motion classification system based on Hidden Markov Models. Different setups of the method for classification and data's preprocessors will be tested in order to find the best performance. In this chapter all the methods and procedures will be presented and the results obtained will be shown in Chapter 6.

5.1. Moment Feature Analysis

In order to study the dynamic meaning and quality of the moment derived features, it was decided to carry out an analysis to study the pattern, the consistence and other characteristics of these new features. Along this section the motion recordings off all the subjects introduced in the Section 3.2 will be analysed. The purpose of this analysis is study the motions execution among all the repetitions for a single subject and the similarities between different subjects.

Another goal of this part will be to identify if a Hidden Markov Model (HMM) of a motion related to a subject that tends to perform it differently in each repetition, affects negatively in the training of the model. Each time the motion classification method is executed, the accuracy in the classification for each subject is calculated. In that way, it is possible to know which are the subjects whose motions are usually more correctly classified. Once these results among the classification tests were examined, some tendencies were spotted on subjects that usually perform quite better or worse comparing with the other subjects considered. In this section the following hypothesis is established to explain this kind of tendency: "One reason of having a high accuracy in the classification for a subject (in comparison with others) is because that subject performs a motion always in a more similar way, which turns on a better model". The *Correlation*, which is a measure of dependence between two set of data, will be the indicator to quantify similarities between recordings of the same motion. It is important to underline that this motion study has been carried out using only the features of the moment vector which include all the force sensor values. It could have been also possible to study these motions using values from the IMUs sensors like the acceleration or the Euler angles.

5.1.1. Resemblance Analysis

The *Resemblance Analysis* pursues to quantify the resemblance among the execution of the same motion among different repetitions carried out by different subjects. If the data differs a lot among subjects, that feature won't be suitable to carry out a motion classification. But on the contrary, if data samples are too similar the variety existing in a real situation would not be represented. Something between these two situations would be appropriate.

To this objective, the motion values were plotted and correlations (the strength of the linear relationship between two samples of data) between pair of samples were calculated to quantify the similarities and spot singularities or irregular events among the motions.

Before calculating the correlation the data is filtered using a digital lowpass filter to reduce noise or outliers. In this way the data is smoothed and a clearer pattern can be distinguished. The parameters of the filter depicted in Table 5.1 are chosen to obtain the highest correlations of the knee moments between the motion recordings.

Type of Filter	Low Pass
Response	Infinite pulse
Order	10
Half Power Freq	2 Hz
Sample Rate	0.1 Hz

Table 5.1.: Parameters of the filter used for the Resemblance Analysis of moments.

Additionally, the two motion recordings are synchronized. Firstly, the delay between data signals is estimated. The delay is found by calculating the cross-correlation between each pair of signals at a set of possible lags, and then the estimated delay is given by the negative of the lag for which the normalized cross-correlation has the largest absolute value. Afterwards the pertinent signal is delayed by prepending zeros to align both. Continuing with the process, the output signals are trimmed considering the delay in such way that the leading zeros section of one of the signals is removed and the corresponding section of the other signal is also cut off. At the end, the sizes of both signals are matched by removing the last data section. Figure 5.1 shows this process. First the delay D is estimated, afterwards 0's are prepended to the signal $s1$ in order to add the delay and align it with $s2$, and finally the parts corresponding to the 0s addition and the matching of the final segments are cut off.

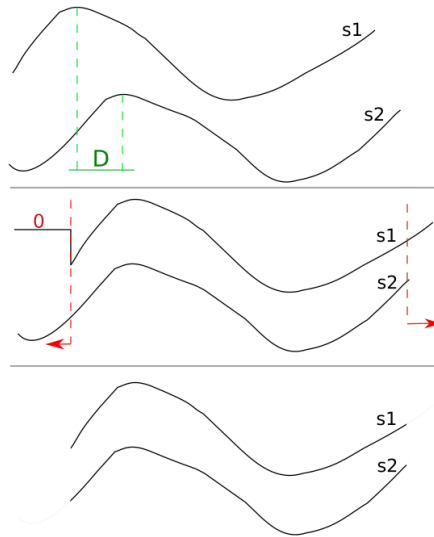


Figure 5.1.: Outline for an alignment process of two motion data sets.

Once the two signals are aligned it is possible to quantify the linear relation between them. Therefore the corresponding Pearson correlation coefficient can be computed after this synchronization as follows

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \cdot \sigma_Y} \quad (5.1)$$

where $cov(X,Y)$ is the covariance between two samples, and σ_X and σ_Y are the standard deviations of the values of each data sample. The Figure 6.1 from the Section 6.1.1 shows an example of this preprocessor.

5.1.2. Intra-Subject Analysis

This study lies in analyse the moments among all the repetitions of the same motion and subject. Through this study it is pursued to check the consistency of the knee moment feature. Also it is possible to see if a certain subject carries out the same motion each time in a similar way. A high variability between repetitions might be detrimental for the training of the model used to classify that motion.

After all intra-subject correlations have been calculated for one motion and subject, a heat map is generated. This heat map shows the correlations between repetitions of motions carried out by the same person.

5.1.3. Inter-Subject Analysis

In this second analysis it is pursued to spot similarities of motions between different subjects. High correlation values among the repetitions of two subjects will mean that they present a similar way of performing that motion. The results extracted can be useful to prove some results in the motion classification and connect these results with physical characteristics of the subjects.

The study consists of comparing the best pair of repetitions of each subject with the best pair of repetitions of the other subjects. Once all intra-correlations of all the subjects have been calculated, the pair of each subject with the highest correlation is chosen. The repetitions of this pair are selected as the ones with which the correlation analysis between subjects will be carried through.

To conclude, a last heat map is generated in which it is pursued to show the motion resemblance among subjects. Therefore it will be possible to distinguish which subjects act in a similar way when performing a certain motion. To generate this heat map the results of the *Inter-Subject* study are used. Each component of the resemblance matrix is the mean of the four correlation values computed for two subjects (Expression 5.2). We have to remember that only two repetitions of each subject were considered.

$$\bar{\rho}_{ID_i, ID_j} = \frac{\rho(ID_i^{r1}, ID_j^{r1}) + \rho(ID_i^{r1}, ID_j^{r2}) + \rho(ID_i^{r2}, ID_j^{r1}) + \rho(ID_i^{r2}, ID_j^{r2})}{4} \quad (5.2)$$

where ID_i and ID_j are two repetitions of that subject and the superscript $r1$ and $r2$ denote the two repetitions that got the highest correlations in the intra-subject analysis.

5.2. Motion Classification

In this section an outline of the motion classification system will be presented and the steps taken to carry out all the classification process will be explained.

5.2.1. Data Preprocessing

In this section it is explained how the motion data is processed before being used in the training of the Hidden Markov Models (HMMs).

Raw Data

All the raw data is stored in files and the force vectors, the linear accelerations and the Euler angle values are synchronized in time using a sampling period of 10ms. Therefore, there is a value of each feature each 10 ms which allows to have uniform data structures suitable for further processing.

During the recording of the data, the force sensors were connected to two different data acquisition systems (DAQ), force sensor 1 to 4 to the first and force sensor 5 to 7 to the second. Each DAQ acquired the raw analogue data using a frequency of 100 Hz. Meanwhile a micro-Controller (SAM3X8E ARM Cortex-M3, Microchip Technology Inc.) processed orientations and linear accelerations of every IMU with frequency of 80 Hz. Therefore, to get values on the same timestamps, the mesured values of

the IMUs were interpolated to 100 Hz. Also it is important to highlight that the Euler angles were calculated from the quaternion values recorded by the IMUs. It was rejected the used of quaternions in Beil et al. (2018), since quaternions are ambiguous (one angular configuration can be described by two quaternions). The quaternion sets of data were transform to Euler angles via the relations shown in Equations ??, where q_i are the quaternion components and ϕ , θ and ψ the Euler angles (Yaw, Pitch, and Roll respectively).

$$\phi = \arctan \left[\frac{2(q_0 \cdot q_1 + q_2 \cdot q_3)}{1 - 2 \cdot (q_1^2 + q_2^2)} \right] \quad (5.3)$$

$$\theta = \arcsin[2(q_0 \cdot q_2 - q_3 \cdot q_1)] \quad (5.4)$$

$$\psi = \arctan \left[\frac{2(q_0 \cdot q_3 + q_1 \cdot q_2)}{1 - 2 \cdot (q_2^2 + q_3^2)} \right] \quad (5.5)$$

Calculation of Derived Features

Each component of a derived feature is calculated using the raw data and considering only a single timestamp. In this way the new data is synchronized in time with the raw data and it is still possible to work with a rectangular data structure.

As it has been already shown in the previous sections 4.1 and 4.2, the features of the moment vector are calculated using the expressions 4.9 and 4.10 and the knee angles features are calculated with the expressions 4.12 and 4.13. For each time stamp where a set of force values were recorded, the resultant moment vector and norm of it are calculated. The knee and ankle angles are calculated likewise but using the corresponding Euler angles from the IMUs recordings. Table 5.2 shows the primary features from where the corresponding derived features were extracted. We have in total 6 derived features which means that in total we have 45 features available to carry out the motion classification.

Primary Features	Units	Number	Derived Features	Units
Forces (F_x, F_y, F_z)	N	21	M_x	$N \cdot mm$
			M_y	$N \cdot mm$
			M_z	$N \cdot mm$
			$ M $	$N \cdot mm$
Euler angles	rad	9	ϕ_{knee}	rad
			ϕ_{ankle}	rad

Table 5.2.: Derived features and primary features from where they were calculated.

Data Filtering

To aim an improvement of the motion classification results, a filtering of the knee moment vector was considered. The use of a filter that smooth the irregularities and reduce the noise can suppose a simplification of the outline of the data. If we take care of using this filter in a moderate way in order to not eliminate relevant data information, the motion representation can be clearer represented and the motion classification might improve. The smoother data set could reduce the number of states needed in the Hidden Markow Models, which could result in simpler models and lower training times.

It was considered to try a filter in a similar way as it was done in the *Moment Feature Analysis*. Unlike in the *Resemblance Analysis*, in which the full sequence motion was used, in this case the filter is applied in each window (e.g interval time of 300 ms). That supposes different conditions for the filtering, so its parameters have to be tuned again. The filter used is a low pass filter (high frequencies, like the ones belonging to the noise, will be attenuated) and the order and the edge of frequency w_n are the ones that can be seen in Table 5.3. Once the filter's parameters are calculated, the filter is applied twice, once forward and once backward, in such way that it has linear phase. The order of the filter is twice the original order. Figure 5.2 shows an example of the filter application for a data windows of 300 ms. The blue curve is the original data and the orange one is the filtered data. The windows belongs to a section of the motion Walking Forward.

Type of Filter	Low Pass
Response	Infinite pulse
Order	3
Edge of Freq.	0.15 rad/s
Type of Data	Digital

Table 5.3.: Parameters of the filter used for the preprocessing of the moment feature.

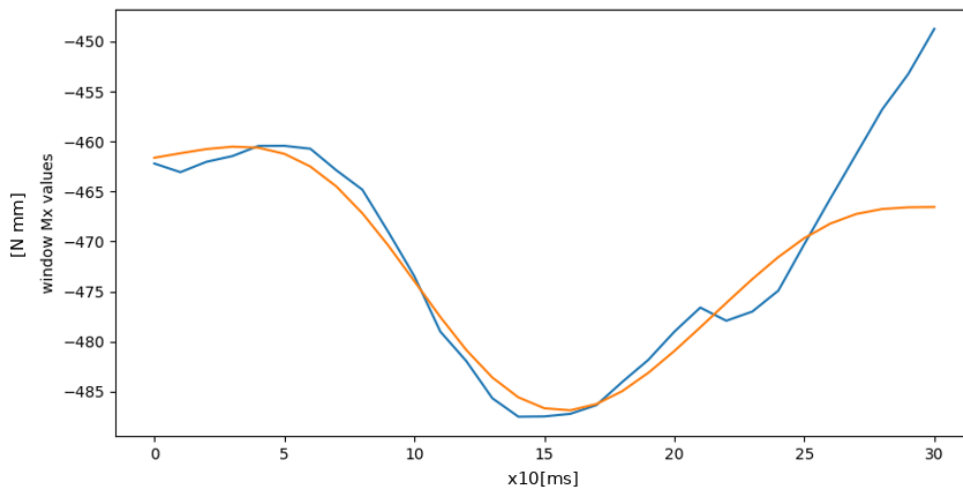


Figure 5.2.: Data windows of 300 ms of the feature M_x where the original data (blue curve) and the filtered data (orange curve) are plotted. The window belongs to one section of the motion Walking Forward.

Windows Creation

A sliding window approach was used to segment the whole measurement data stream of a motion. That means that after a time equal to the sampling distance a new window is generated containing all the data stream of a specific period of time (window size). In that way the data is split in different overlapping windows until complete the entire motion and using always a constant window size (e.g. 300 ms).

Figure 5.4 represents the process for a generic feature. We can observe that the last window created is such that its last timestamp coincides with the last timestamp of the whole data stream.

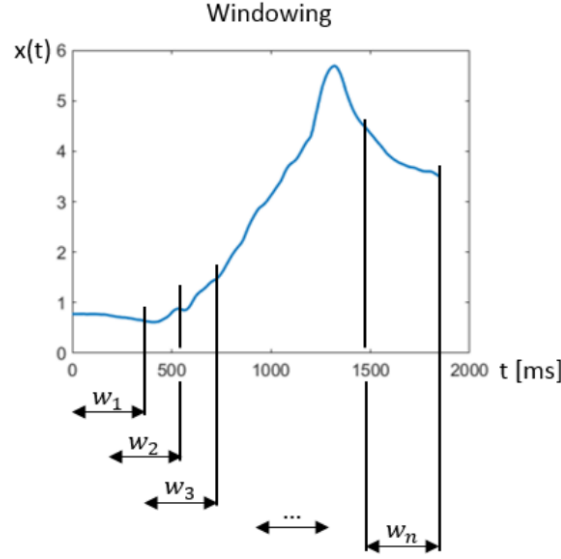


Figure 5.3.: Representation of the sliding window approach. Picture taken from Beil et al. (2018)

For smaller window sizes the amount of time needed for the classification is shorter, so an online performance of the classification system can be achieved. However, as it was tested in Beil et al. (2018), the window size is usually inversely proportional to the accuracy. For example, a stratified 5-fold cross validation obtained testing on all the subjects gave the following accuracies for each window size: 82.92% for $w = 100ms$, 92.80% for $w = 300ms$ and 97.45% for $w = 600ms$.

Data Normalisation

It is not possible to assume that the force sensors measure in the same way after each tightening of the exoskeleton for a certain person. Factors like the differences of the inter-subject characteristics or of tightness of the Velcro fasteners lead to different force values. Moreover concerning on the subject's gait style and the duration to carry out a specific motion, the linear accelerations and Euler angles can differ drastically among repetitions of the same subject and when being compared with other subjects. A solution to try to avoid these issues would be a calibration phase before the use of the exoskeleton, but since one of the requirements of these exoskeleton is to improve its acceptance and usability, this option must be dismissed.

Therefore, to make the classification robust against these influences, the differences of consecutive feature values are calculated such $F_t - F_{t-1}$ and used as an input for the classifying system. However in the following section 6.3 some tests were run without calculating these differences in order to study its influence.

Finally, taking into account that the data ranges are really different among the features used, a data normalisation is performed. Table 5.4 shows the approximate ranges for each feature in which we can notice that the orders of magnitude are quiet different.

To carry out the data normalisation, for each window and type of scalar values, the maximum and minimum values x_{max} and x_{min} are determined. Afterwards all individual values of each timestamp x_t are represented with the conversion to the interval $[-1; 1]$ as can be seen in the expression 5.6.

$$\hat{x}_t = 2 \cdot \frac{x_t - x_{min}}{x_{max} - x_{min}} - 1 \quad (5.6)$$

Feature	Units	Aprox. min	Aprox. max
Linear accelerations	m/s^2	-5	3
Euler angles	rad	-3	2
Forces	N	-20	15
Knee moments	$N \cdot mm$	-1000	500
Joint angles	rad	-4	-1

Table 5.4.: Approximate range of values for each feature.

5.2.2. Motion Classification System

In the context of our motion classification system, a *Learning problem* is presented with the target of determine the HMM's parameters that best fit our training data for each motion. Each model is defined by a matrix of transition probabilities A , a matrix of observation probabilities B and a vector of initial probabilities π . The observations are modeled using a Gaussian distribution (Expression 3.6) since they are continuous and the data samples are structured using a sliding windows approach.

For the data classification into motions, a *Learning Problem* is considered since we consider the case in which we are trying to choose among several competing models the one that best matches the observations. In other words, we will face a multi evaluation problem in which motion data windows are evaluated in each model's motion $M_i = (A_i, B_i, \pi_i)$. A score is obtained regarding how likely that data windows was generated by this model. Each window is labeled according to the model's motion in which the highest likelihood was obtained.

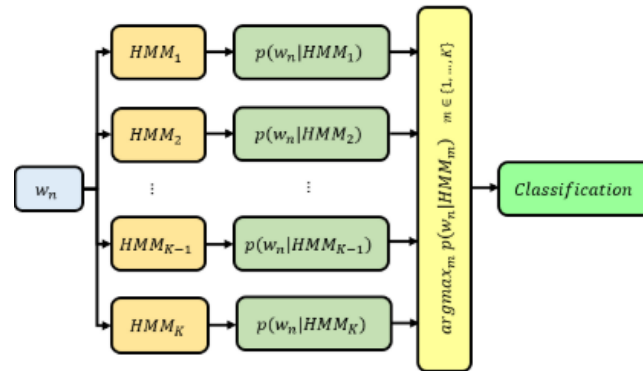


Figure 5.4.: Representation of multi evaluation problem to classify a data window. Picture taken from Beil et al. (2018).

Next paragraphs explain the main settings used to carry out the tests. Some decisions of the setup were chosen because of the results obtained in Beil et al. (2018) and Patzer, Isabel and Asfour, Tamim (2019). Firstly, how was the setup of the classification system that has been mostly used until now is explained. Secondly, the different setup possibilities are introduced looking for a better performance the motion classifier.

Basic Setup

Considering the results that will be compared are the ones obtained from Patzer, Isabel and Asfour, Tamim (2019), where an analysis of a dimensional reduction was carried out, only the settings for these

tests will be presented. Some of them were extracted from the conclusions of Beil et al. (2018). The results of these tests can be found in the following Section 6.3 in Table 6.6.

The data samples are processed using a sliding window approach using a window size of 300ms and vectors are concatenated using a time step of 10 ms. The window size value was proved in Beil et al. (2018) to have a good accuracy result of 92,80% while its corresponding latency of 389 ms allows an online use of the classifier. Regarding the model of the continuous data of the windows, the covariance matrix of the Gaussian distribution that represents the observations is diagonal. The values of the force sensors (21 features) and the linear accelerations and Euler angles from the IMUs (18 features) were used, meaning a total number of 39 featured to chose from.

For the training of the models, a fully-connected topology with 14 states to represent each motion is used. In Beil et al. (2018) it was proved that this number of states and type of HMM allow a good trade-off between accuracy and compute time. For the model training, 100 iterations are always run to recalculate the transition and observation probability matrices and the initial state distribution. To evaluate the results it was carried out a stratified 5-fold cross validation for all the configurations.

Adapted Setup

Different variations of the basic setup have been considered looking for better representations of the motions in order to obtain higher accuracies in the motion classification and lower compute times.

The total of features now available is 45, where 6 are the new derived features. During the motion classification tests a wide amount of combinations will be considered. First single features and derived features will be used alone to see their single performance. Afterwards, the number of features will increase and configuration mixing primary and derived features will be considered.

Regarding the preprocessor of the feature vectors, for some configurations the difference of two consecutive vectors ($F_t - F_{t-1}$) will be calculated to create the windows. This was already done in Beil et al. (2018), where it was argued that inter-subject characteristics caused a huge variability among the values captured each time the repetition of a motion was recorded. Therefore, calculating the differences among values this variability that was producing an specific offset can be removed. Furthermore, taking into account that each data sample was recorded using a constant sammple period, it could be considered that we are calculating a numerical differentiation of these value sequences as it was done in Mandery et al. (2016). In the tests carried out, for each set configuration of features, a version with the calculation of the differences and another without this calculation will be tested. In that way, we will be able to prove in which cases this preprocessor is better. Also in the data preprocessor, features will be filtered to improve the model's definition. The filter applied is the one that was explained in Section 5.2.1.

With respect to the model training, different types of HMM will be tested. *Left-Right* and *Cyclic* are HMM's topologies, which were introduced in Section 3.3, will be evaluated. These topologies are characterised by their simple structure since only subsequent states are connected. The models' complexity will be reduce in comparison with the fully-connected topology and consequently compute times will drop. Another parameter of the models tested will be the number of states employed. Less states will be used in combination with the aforementioned HMM topologies in order to find the model than represents in a better way the considered motions and with a low compute time. As it can be expected, the use of less number of states will lead to a reduction of the compute times of the classification system.

5.2.3. Model Validation and Evaluation of Results

This part referes to Sections 3.4 and 3.5 and it explains how is evaluated the performance of the motion classification system and how are the results displayed.

Concerning the Model Validation, the HMMs were trained and tested using a 5-fold cross validation over all the subjects and motions. It was also stratified to accommodate the varying number of generated windows per subject caused by the self-selected motion speed of subjects influencing the recording lengths (Beil et al., 2018).

Using the evaluation techniques introduced in Section 3.5, it will be explained how the classification performance is evaluated and how the results are addressed and displayed. While the classifier is running,

	precision	recall	f1-score	support
WalkingForward	0.96	0.86	0.91	7997
WalkingBackward	0.92	0.96	0.94	8381
TurnLeftSmall	0.87	0.88	0.87	4251
TurnRightSmall	0.86	0.92	0.89	4205
LiftObjectUP	0.87	0.92	0.90	2416
DropObject	0.76	0.94	0.84	2765
SidestepsLeft	0.98	0.97	0.98	7630
SidestepsRight	0.97	0.96	0.97	7719
GoingUpstairs	0.97	0.91	0.94	8014
GoingDownstairs	0.86	0.91	0.89	7120
GoingDownstairsBackwards	0.94	0.88	0.91	8202
SitDownArmsCrossed	0.94	0.85	0.89	3090
StandUpArmsCrossed	0.94	0.93	0.94	2756
stand	0.89	0.99	0.94	3864
avg / total	0.92	0.92	0.92	78410

Figure 5.5.: Display of results for a round in a train and testing process.

a log file records all the results and main events during one execution. Each time a round is launched the number of windows used for training and testing are shown, and also the windows of each subject used for testing. During a round, each time a motion is erroneously predicted, it is recorded on the log file reporting how was the erroneous classification: "predicted *wrong motion*, is *true motion*".

After a round's execution, the accuracy obtained for each subject and the global one is displayed. A table that summarises the results is found at the end of the round's report. Figure 5.5 is an example of the results displayed after one round. First column shows the corresponding motions, second column shows the precision calculated by using the Expression 3.18, third column refers to the recall metric (Expression 3.19), fourth column is the F_1score calculated using the Expression 3.20 and finally the last column displays the number of windows that were taken to train a model of a specific motion. The last row shows the average value of each column, but not for the F_1score 's column (highlighted using a red rectangle), whose value is calculated by using the Equation 3.22.

At the end of the log file, an overview of the results is shown. We can find there the number of wrong and right predictions, the total hit rate, the hit rate per each round and the total duration of the classifier execution.

6. Results

In this chapter the results of the presented analyses in Chapter 5 are exposed and described. On the first part, the work and conclusions from the resemblance analysed using the motion derived features are explained. On the second part we find a summary of the results obtained from the motion algorithm and an overview of all the setup configurations that were evaluated.

6.1. Moment Feature Analysis

This section contains the main results from Section 5.1.1 in which the knee moments were used to study the constitution of this new feature and analyse the motions. Firstly, we find an example for two specific cases where all the steps taken to do a resemblance analysis are detailed. Afterwards, the main results and conclusions from the *Intra-Subjects* and *Inter-Subjects* analyses are presented. Additionally, a section explains the constitution of the data base created during the resemblance analysis.

6.1.1. Resemblance Analysis

A resemblance analysis was done with the recordings of 16 motions and 10 subjects which repeated each motion 10 times, creating a data base that describes the constitution of each motion regarding the knee moment vector. This moment vector was calculated using the force sensor values as it was explained in Section 4.1. The data used was recorded by subjects which physical characteristics can be seen in the Table 3.1. The motions analysed are the ones that follow: *Walking Forward*, *Walking Backward*, *Turn Left*, *Turn Left Small*, *Turn Right*, *Turn Right Small*, *Sidesteps Right*, *Sidesteps Left*, *Going Upstairs*, *Going Downstairs Backwards*, *Going Downstairs*, *Lift Object*, *Lift Object Up*, *Drop Object*, *Stand Up* and *Sit Down*.

In this section, all the operations done during the resemblance analysis are shown. Two examples, each one of a specific motion and subject, will be shown during the explanation of the process compared. As we already mentioned in Section 5.1.1, this analysis targets to spot the similarities among repetitions performed by the same subject and among different subjects to see the consistency of the knee moment feature.

Firstly, a correlation analysis between pairs of samples is carried out. Figure 6.1 shows the preprocessor in a three time series plots for the subject ID1717, motion *Walking Forward* and repetitions 6 and 9. The first plot shows the knee moments data of two repetitions (repetition 6 in blue and 7 in red) directly calculated from the force values and using the Equations 4.9 and 4.10. The second one plots the filtered data using a low-pass filter with the parameters from the Table 5.1. The last one shows the two data signals synchronized using the method explained in Figure 5.1.

Figure 6.2 shows the preprocessor to calculate the correlation between repetitions 1 and 4 of the subject ID1719, motion *Going Upstairs* and component M_x . If we compare both examples shown, in the case of Figure 6.1 the two data signals are more coinciding when they were synchronized than in the Figure 6.2. That means that for the example of *Going Upstairs* both motions were not so similar performed, so a lower correlation value than in the case of *Walking Forward* will be obtained.

In Figure 6.3 the correlation value is represented using a scatter plot where the knee M_x values of the two repetitions considered of the motions *Walking Forward* are plotted once the data signals have been synchronized. We can appreciate in the plot a tendency towards a linear relationship among both knee moment samples. A high correlation value of 0.96 confirms this fact.

Figure 6.4 shows the scatter plot for the motion *Going Upstairs*. The linear relation is quite evident, but as we predicted in the time series plots, taking a look at the plot where the signals were synchronized,

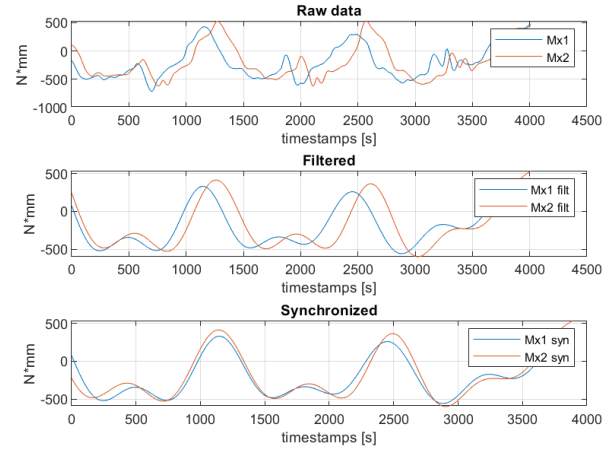


Figure 6.1.: Time series plots from the data of two motion samples. It is shown the three steps taken before the calculation of the correlation. Data from the repetition 6 and 9 of the motion *Walking Forward*, subject ID1717 and knee moment M_x .

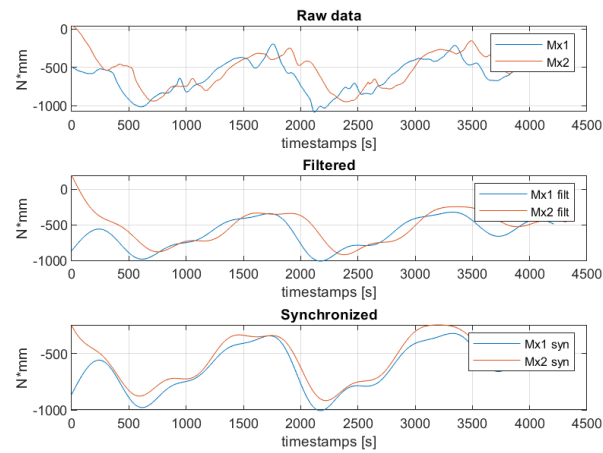


Figure 6.2.: Time series plots of the preprocessor to calculate the correlation between two samples. Data from the repetition 1 and 4 of motion *Going Upstairs*, subject ID1719 and knee moment M_x .

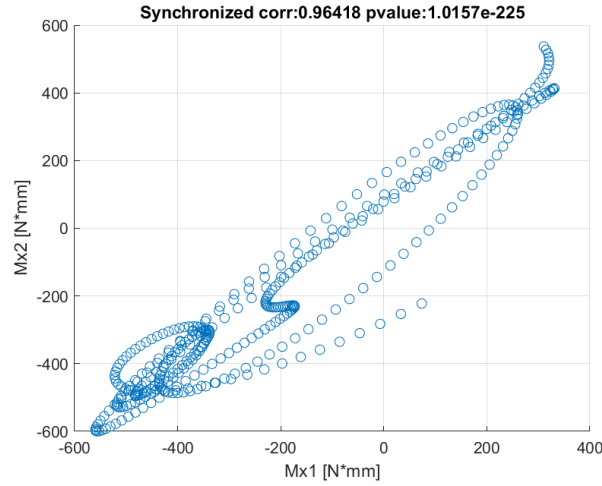


Figure 6.3.: Scatter plot of two repetitions once the data was filtered and synchronized. The correlation result is on the top. Data from the repetition 6 and 9 of the motion *Walking Forward*, subject ID1717 and knee moment M_x .

the correlation is lower than the other case with less than 0.93. More examples of correlation analyses carried with the motions *Walking Backward*, *Going Downstairs Backward*, *Going Upstairs* and *Stand Up* can be found in the Section A.2 of the Appendix.

Once all intra-subject correlations have been calculated for one motion, subject and component of the moment vector, a heat map is generated. The values that this heat map contains are percentages of correlations between pairs of repetitions. Figure 6.5 shows the heat map corresponding to the knee moment M_x , subject ID1717 and motion *Walking Forward*. We can observe a high mean correlation among all the repetitions since, regarding the blue-coloured vertical bar located next to the heat map, the minimum correlation is 86%. Meanwhile for the equivalent heat map of the motion *Going Upstairs* and subject ID1719, the correlations values are a bit lower and the minimum one is about 80%.

Right after, the intra-subject analysis starts in which the similarities among different subjects will be studied. The best pair of repetitions of each subject (the pair that got the highest correlation) with the best pair of the other subjects (two repetitions for each subject are used) will be compared, as it was explained in Section 5.1.3. The repetitions of this pair are selected as the ones to carry out the correlation analysis between subjects.

In Figure 6.7 the correlations between the pair with highest correlation of the subject ID1717 and the other pairs with the highest correlations of other subjects are shown. For subject ID1717 the pair of the repetitions 1st and 8th were those that gave the highest correlation (98.72% in Figure 6.5), therefore these are the two repetitions that will be compared with other subjects. We can appreciate that the correlations of the repetitions belonging to the same subject, the values obtained are similar, when they are contrasted with the same repetition of the other subject. For instance, the repetitions 1 and 3 of the subject ID720 with the repetition 1 from ID1717 are quite low in both cases. This low value might be caused by a bad synchronization process.

Figure 6.5 shows again the correlation among repetitions from different subjects, but this time for the motion *Going Upstairs*. Compared with the other case, higher correlation were obtained reaching almost a 86%. However some really lower values appear with a minimum of a 5%, probably due to a bad synchronization between the data signals.

To conclude, a last heat map is generated that shows the motion resemblance among subjects. Each component of the resemblance matrix is calculated using the Expression 5.2. Figure 6.9 shows the heat map of the comparison among all the subjects for the motion *Walking Forward* and the moment about knee moment M_x . For this specific motion this matrix shows that subjects ID1723 and ID674 gave the

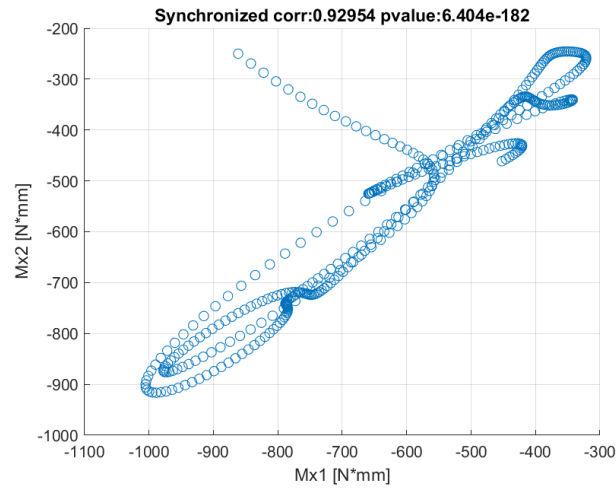


Figure 6.4.: Scatter plot of two repetitions once the data was filtered and synchroninized. The correlation result is on the top. Data from the repetition 1 and 4 of motion *Going Upstairs*, subject ID1719 and knee moment M_x .

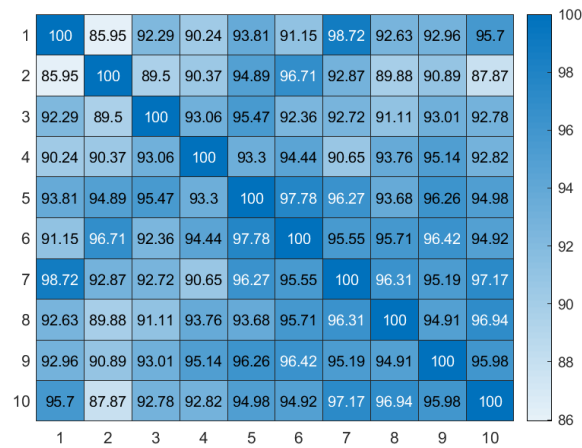


Figure 6.5.: Heat map of subject ID1717, motion *Walking Forward* and knee moment M_x .

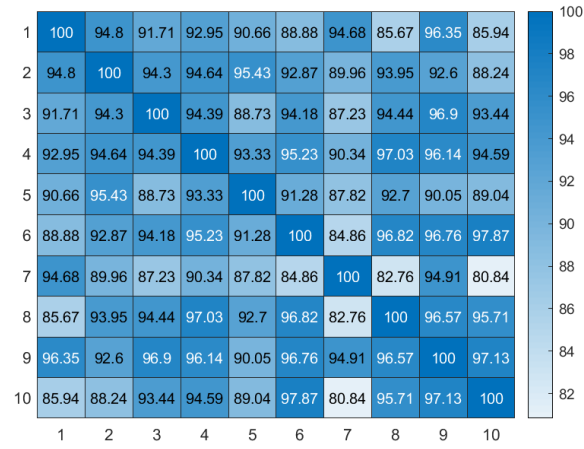


Figure 6.6.: Heat map of subject ID1719, motion *Going Upstairs* and knee moment M_x .

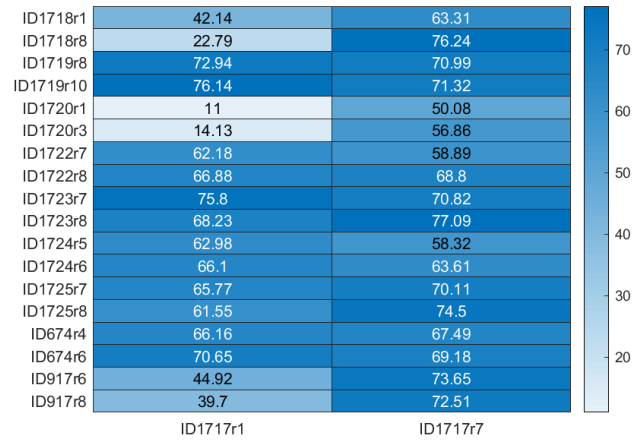


Figure 6.7.: Correlations between the repetitions with highest correlations among different subjects. Motion *Walking Forward* and knee moment M_x .

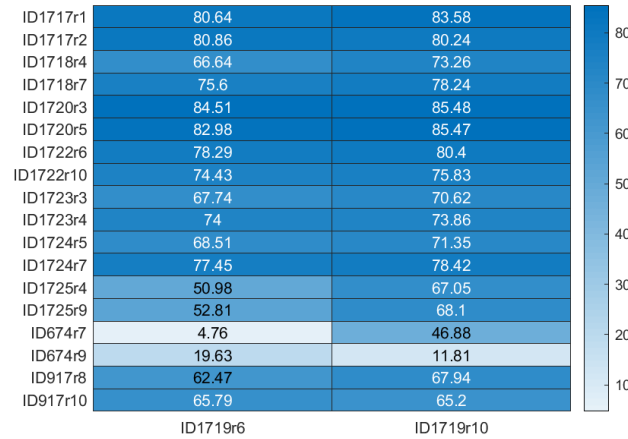


Figure 6.8.: Correlations between the repetitions with highest correlations among different subjects. Motion *Going Upstairs* and knee moment M_x .

highest correlation of nearly 84%, which would mean that these two persons present a similar way of performing the motion *Walking Forward*. Additionally, low values are spot in some cases where ID1722 is compared with other subjects.

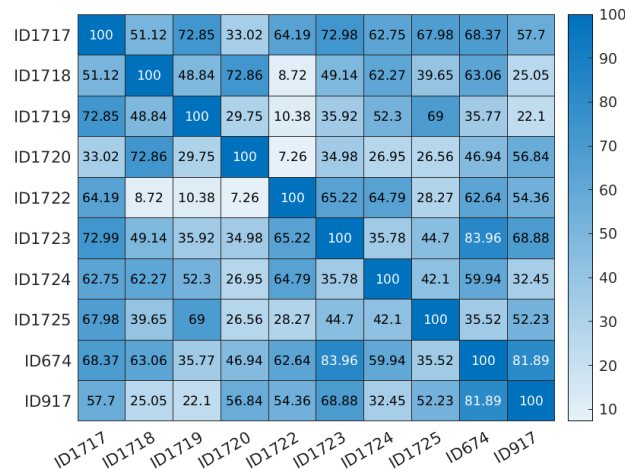


Figure 6.9.: Correlations of samples among different subjects. Motion *Walking Forward* and knee moment M_x .

In Figure 6.10 can be see the resemblance matrix of the motion *Going Upstairs*. In that case the correlation of subjects ID674 and ID917 gave really low values. The samples chosen for these two subjects present a pattern with a big delay when compared with others samples, which caused that the synchronization was not done properly.

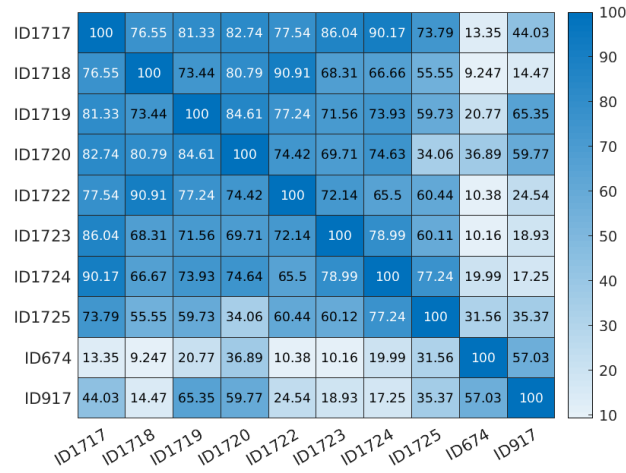


Figure 6.10.: Correlations of samples among different subjects. Motion *Going Upstairs* and knee moment M_x .

6.1.2. Data of the Resemblance Analysis

A resemblance analysis was done for all motions and subjects creating a data base that describes the constitution of each motion regarding the knee moment vector. This moment vector was calculated using the force sensor values as it was explained in the Section 4.1. The data recorded used was recorded by 10 subjects which physical characteristics can be seen in the Table 3.1. The motions analysed are the ones that follow: *Walking Forward*, *Walking Backward*, *Turn Left*, *Turn Left Small*, *Turn Right*, *Turn Right Small*, *Sidesteps Right*, *Sidesteps Left*, *Going Upstairs*, *Going Downstairs Backwards*, *Going Downstairs*, *Lift Object*, *Lift Object Up*, *Drop Object*, *Stand Up* and *Sit Down*.

There is a time series plots and scatter plots like the ones in Figures 6.1 and 6.3 between all the pairs of samples for each subject and vector moment's component (M_x , M_y and M_z). Also, a heat map showing all the intra-subjects correlations and a file with the means and standard deviations of the correlations calculated.

Regarding the *Inter-Subject* analysis, we can find more plots like the one in Figures 6.1 and 6.3, but this time comparing repetitions from different subjects, but only the ones with highest intracorrelations, as it was already mentioned in Section 5.1.3. The corresponding heat map puts together these correlations to show an overview of the results.

Apart from these plots, we can find in this data base heat maps and files that summarise the results. A heat map gives the means of intra-correlations for each component of the moment vector and for each each subject. A similar heat-map was created also, but this time with the means of the inter-correlations. Finally, to represent similarities of motion performance among different subjects, a last heat map shows the intra-correlations. High values in this heat map mean that the studied motion was performed in a similar way by the two corresponding subjects.

In order to outline a bit the data base, the following points define its structure and evaluations:

- For each subject and for each moment's component M_k (M_x , M_y and M_z):
 - **Subject analysis:** analysis between all the repetition of one subject.
 - * For each pair of repetitions (Sample i and Sample j):
 - **Time series plot:** (Sample i , Sample j) vs Time.
 - **Scatter plot:** Sample i vs Sample j .

- * **Heat Map**: matrix of correlations among all the repetitions.
- * Mean and standard deviations of the correlations.
- **Analysis between subjects**: analysis between pair of repetitions with the highest correlations from different subjects
 - * For each pair of repetitions (Sample i of Subject A and Sample j of Subject B):
 - **Time series plot**: (Sample i , Sample j) vs Time.
 - **Scatter plot**: Sample i vs Sample j .
 - * **Heat Map**: matrix of correlation among repetitions of different subjects.
 - * Mean and standard deviations of the correlations.
- **Correlations Intra-subject**
 - **Heat Map**: matrix of the mean of correlations of all subjects and all M_k .
 - Mean and standard deviations of all the correlations for all the subjects.
- **Best Intra-subject correlations**
 - For each M_k , a table with the highest correlations of each subject and the number of the repetitions for which these correlations were calculated.
- **Correlations Inter-Subject**
 - **Heat Map**: matrix of the mean of correlations for each subject and M_k . Means calculated from the intra-subject correlations.
- **Subjects Resemblance**
 - **Heat Map** for each M_k : matrix with the means of the four correlations values that were calculated between pairs of subjects.
 - **Global Heat Map**: matrix with the means of the values of the *Heat Maps* for each M_k . It gives an idea of how similar two subjects perform the same motion.

6.1.3. Main Results of the Resemblance Analysis

This section gives insights of the most relevant results from the resemblance analysis. First, we will take a look at the results of the *Intra-Subject* analysis with the target of discovering which are the subjects with more variability in the motion's performance. To do so, twelve different motions have been considered and the global correlation results for each subject have been calculated (mean of M_x , M_y and M_z). The results obtained will be used in Section 6.3.6 to identify if low correlations in certain subjects affect on the motion classification accuracies for these subjects.

Table 6.1 and Table 6.2 show the means of M_x , M_y and M_z of the global correlation for all the motions. The motions *Stand Up*, *Sit Down* and *Lift Object* have the highest correlations which might mean that these motions are easier to repeat similarly than the others. On the other hand *Walking Forward* and *Turn Left* gave the lowest values, so these were not repeated similarly by the subjects. In Table 6.3 we can find an overview without differentiating motions, but split by the moment vector's components. Additionally, the last column shows the mean of these values. In rough lines, subjects ID1717 and ID1719 are the ones that gave the highest correlation values. That means that these subjects carried out a motion more similarly each time they repeated it. In the opposite case, we find subjects like ID1718 and ID1722, which gave the lowest global correlation values. As it will be checked in Section 6.3.6, we will see if these correlation results affect the performance of the HHMs for the motion classification.

Concerning the analysis of correlations among subjects, it was pursued to look into two different questions. The first one was to analyse which are the motions that are usually more similarly performed by the subjects. With this in mind, the mean of correlations between subjects were calculated for each motion. Table 6.4 shows these results split by moment vector's components and the total mean in the last

Subject	Walking Forw.	Walking Back.	Turn Right	Turn Left	G. Upstairs	Stand Up
ID1717	77.37	80.31	94.10	75.69	95.93	87.39
ID1718	62.20	67.79	74.71	57.86	80.53	93.47
ID1719	67.53	87.62	89.47	66.12	93.73	96.68
ID1720	65.02	85.06	81.55	69.11	90.18	94.25
ID1722	67.48	48.14	77.93	67.18	93.11	94.89
ID1723	72.49	83.35	89.71	66.70	77.63	93.91
ID1724	54.09	56.09	89.42	54.83	91.90	93.76
ID1725	52.85	73.35	90.84	66.12	89.57	92.26
ID674	71.33	90.64	90.68	64.94	83.61	87.90
ID917	69.21	85.74	84.52	66.62	73.73	93.10
Mean	69.96	75.81	86.29	65.52	87.00	92.76

Table 6.1.: Mean of the global correlations ρ (mean of $\rho(M_x)$, $\rho(M_y)$ and $\rho(M_z)$) for each subject (%).

Subject	Sit Down	S. Step Right	S. Step Left	Lift Object	Drop Object	G. Downstairs
ID1717	95.41	81.97	79.68	89.65	88.01	91.21
ID1718	93.61	50.35	68.16	84.72	66.99	70.68
ID1719	97.31	71.43	76.31	97.99	94.60	87.26
ID1720	92.79	74.45	86.82	87.28	71.32	88.66
ID1722	92.12	73.67	80.37	83.05	71.34	69.22
ID1723	87.20	67.13	65.70	91.68	82.14	79.38
ID1724	92.48	84.48	80.11	96.36	88.29	92.17
ID1725	93.76	70.96	77.45	94.79	92.50	87.67
ID674	87.90	77.69	78.02	88.29	85.86	92.19
ID917	76.61	77.66	67.77	83.89	91.43	85.93
Mean	90.02	73.01	76.04	89.77	83.25	84.44

Table 6.2.: Mean of the global correlations ρ (mean of $\rho(M_x)$, $\rho(M_y)$ and $\rho(M_z)$) for each subject (%).

column. If we take a look at the results, the motions *Stand Up*, *Sit Down* and *Lift Object* are the ones with the highest resemblances among the subjects. The three motions have in common that they are carried out in a short period of time and made up by simple movements. It makes sense that *Stand Up* is the one with the highest correlation, since it is just a fixed position.

The second question set out was to search for pairs of subjects that gave high correlations and try to find the reason by taking a look on the physical characteristics of the subjects (Table 3.1). The Table 6.5 contains pairs of subjects which gave the highest correlations considering 6 different motions. Subjects ID674, ID1717 and ID1723 were the ones that appeared more frequently pairing up among them or with other subjects. Subjects ID674 and ID1723 have similar physical characteristics, the height is almost the same and the leg dimensions are quite similar. However, subject ID1717 is taller and the leg dimensions differ. As long as the subjects used for the recordings have similar physical characteristics (one of the requirements set for the recordings in Beil et al. (2018)) and no obvious similarities were spotted among the repetitions with highest correlations, we conclude that it is not possible to use the *Analysis of Inter-Subjects* to prove physical resemblances among subjects.

Subject	$\rho(M_x)$	$\rho(M_y)$	$\rho(M_z)$	Global ρ
ID1717	85.76	88.27	85.16	86.39
ID1718	62.38	75.95	79.44	72.59
ID1719	80.77	89.20	86.55	85.50
ID1720	72.43	83.96	90.30	82.33
ID1722	70.10	80.49	79.05	76.54
ID1723	68.89	88.36	82.00	79.75
ID1724	79.40	80.64	83.46	81.16
ID1725	77.75	83.48	84.30	81.84
ID674	80.46	86.25	83.05	83.25
ID917	74.84	81.78	82.43	79.69

Table 6.3.: Mean of correlations for all the motions of M_x , M_y and M_z and the global one (%).

Motion	$\rho(M_x)$	$\rho(M_y)$	$\rho(M_z)$	Global ρ
Walking Forward	48.62	71.84	77.41	65.96
Walking Backward	51.70	64.60	64.19	60.16
Turn Right	36.20	66.90	67.31	56.80
Turn Left	45.53	67.52	83.50	65.52
Going Upstairs	56.16	59.81	57.48	57.82
Stand Up	66.42	94.42	71.20	77.35
Sit Down	63.14	92.61	67.06	74.27
Side Step Right	51.40	50.18	65.28	55.62
Side Step Left	52.09	61.24	53.30	55.54
Lift Object	66.99	83.41	66.01	72.14
Drop Object	52.81	89.55	61.59	67.98
Going Downstairs	39.21	71.74	54.30	55.08
Mean	52.52	72.82	65.72	63.69

Table 6.4.: Mean of correlations for all the inter-subjects correlations of M_x , M_y and M_z and the global one (%).

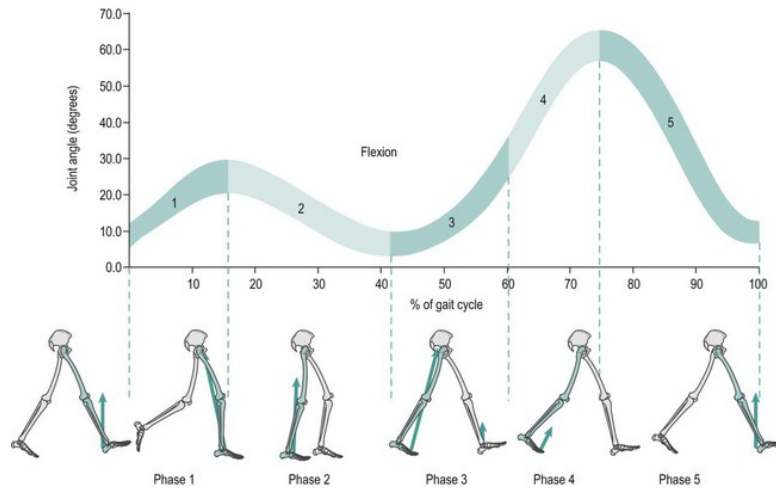
Motion	Pair 1	ρ	Pair 2	ρ	Pair 3	ρ
WF	ID1723, ID674	87.98	ID1717, ID1723	87.16	ID674, ID917	85.19
WB	ID1719, ID674	76.56	ID1723, ID674	73.73	ID1720, ID917	73.38
TR	ID1722, ID1723	81.55	ID1717, ID1725	76.45	ID1717, ID1722	75.74
TL	ID1717, ID1725	88.82	ID1717, ID1719	87.34	ID1717, ID674	81.55
GU	ID1717, ID1719	84.92	ID1717, ID1723	80.20	ID1719, ID1725	78.60
SU	ID1719, ID1724	94.18	ID1718, ID1724	92.43	ID674, ID917	91.06

Table 6.5.: Pairs of subjects that got the highest correlations in the *Inter-Subject analysis*. The motions considered are: "WF - Walking Forward", "WB - Walking Backward", "TR - Turn Right", "TL - Turn Left", "GU - Going Upstairs" and "SU - Stand Up". Correlation values ρ are in %.

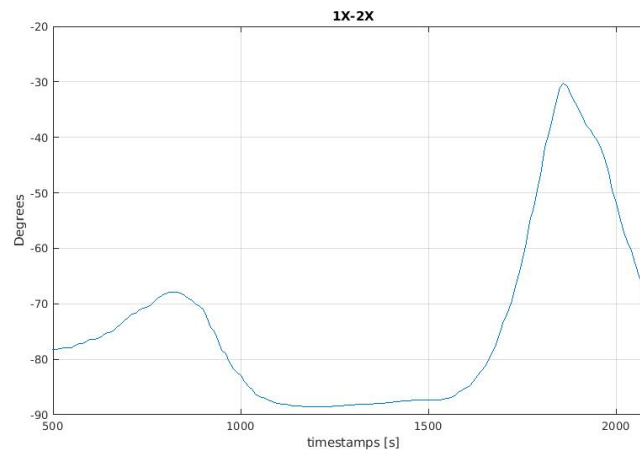
6.2. Joint Angles Analysis

In this part the theoretical joint angles of the knee and the ankle angles of the gait cycle are analysed and compared with the angles extracted from the recordings of the motion *Walking Forward* using the Expressions 4.12 and 4.13. The knee and ankle joint angles are relative angles, which provide an angular variation between the orientations of the thigh and the shank (joint angle of the knee) and the angular variation between the shank and the foot (joint angle of the ankle).

Figure 6.11(a) shows the theoretical joint angle of the knee during a gait cycle fragmented in gait phases. In Figure 6.11(b) the data of the calculated knee joint angle is plotted during also a gait cycle, but this time the X axis is temporal. The title of this plot *1X-2X*, means that the components about X-axis of the IMUs 1 and 2 were used to calculate this angle. Comparing both plots we can highlight that the knee joint angle variation during the whole gait cycle is approximately 70 degrees in both cases and the patterns are comparable.



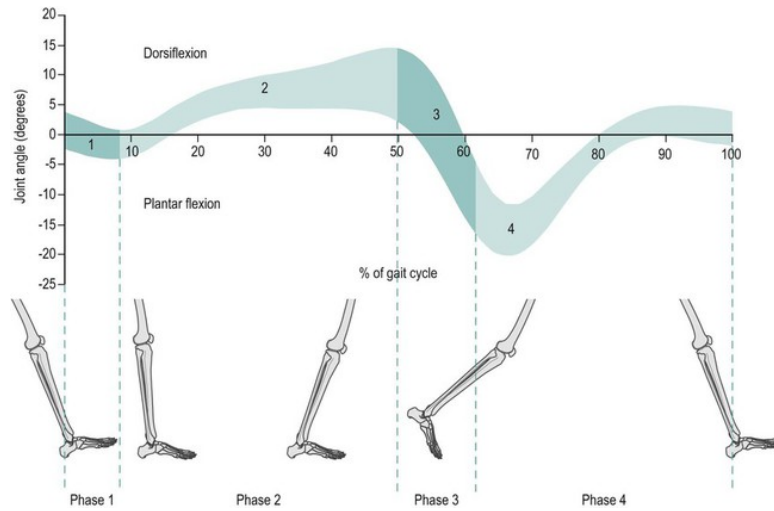
(a) Theoretical joint angle of the knee during a gait cycle. Picture from Musculoskeletal Key (2008).



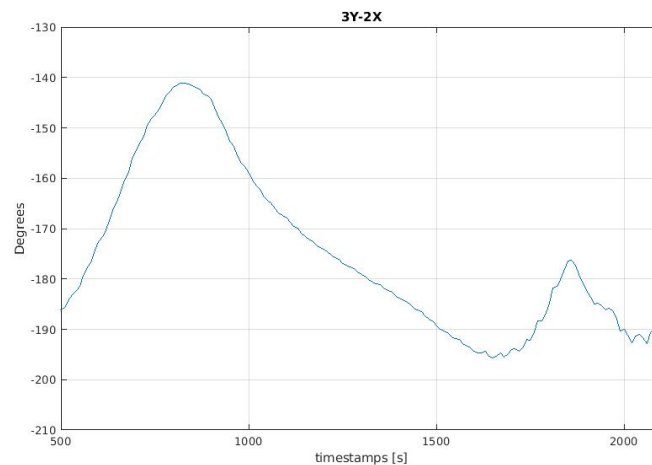
(b) Joint angle of the knee calculated from the recordings of *Walking Forwards*, subject ID1718 and repetition number 4.

Figure 6.11.: Comparison between the theoretical joint angle of the knee with the one calculated from the recordings.

Meanwhile in the Figure 6.12, we can see a comparison like the last one but this time between the theoretical joint angle of the ankle and the one calculated from the recordings. The title of the Figure 6.12(b) 3Y-2X stands also for the two IMUs use to calculate the joint angle. This time the Y direction of the IMU 3 and the X direction of the IMU 2 were the two values used. In this case, the angle variation is around 40 degrees and although this time both patterns are not as similar as in the previous comparison, the angle variation take place in the same intervals of the gait cycle. As mentioned in Section 4.2 the angles calculated are approximations to the knee and ankle joint angles of a human leg. Orientation, location of the IMUs and the fact that the lower limb exoskeleton can only perform rotations around one axis are some of the reasons why the real angles can not be extracted.



(a) Theoretical joint angle of the ankle during a gait cycle. Picture from Musculoskeletal Key (2008).



(b) Joint angle of the ankle calculated from the recordings of *Walking Forwards*, subject ID1718 and repetition number 4.

Figure 6.12.: Comparison between the theoretical joint angle of the ankle with the one calculated from the recordings.

6.3. Motion Classification

During this chapter, the results obtained during all the testing process are shown. This chapter targets to analyse the set of possible configurations of raw data with derived features in order to find a suitable setup to carry out the classification. Throughout the testing process, different factors have been taken into account, like the amount of sensors used or the computational time needed. Additionally, one of the sections explains that some corrupted raw data was found and how it was dealt with this problem.

6.3.1. Results with Raw Data

In this section the previous evaluations without using any derived feature will be shown in order to compare them in subsequent sections. These results have been extracted from Patzer, Isabel and Asfour, Tamim (2019), where it was pursued to obtain a high accuracy using the minimum amount of sensors. In Table 6.6 there are the results ordered in an increasing way from the amount of 2 sensors to the maximum set of 10 sensors. These evaluations were carried out with the 10 subjects of Table 3.1, the 14 motions introduced in Section 3.2 (the motion *Stand* is not included), a window size of 300ms and using a 5-fold cross validation.

These results show that the IMUs are the sensors that get better accuracy if they are used alone. When two types of sensors are used together, a tendency exists in which IMUs 1 and 3 combined with force sensors give the best performance. From the amount of using 5 sensors, the results improve a little reaching the best result an accuracy of 92,85% using 8 sensors. That means an increment just lower than the 2% when comparing it with the result of 91.16% obtained using five sensors.

Tests on all subjects, all motions, W300 and S14				
Num. Sensors	Num. Features	Force Sensor	IMU Sensor	Accuracy (%)
2	12	–	1,3	86.35
	9	7	3	85.34
	12	–	1,2	84.66
	9	5	3	84.60
3	15	5	1,3	89.53
	15	7	1,3	89.08
	15	6	1,3	89.04
	15	3	1,3	88.81
4	18	6,7	1,3	90.73
	18	3,6	1,3	90.69
	18	5,6	1,3	90.48
	18	3,5	1,3	90.44
5	24	3,5	1,2,3	91.16
6	27	1,3,5	1,2,3	92.20
7	30	1,5,6,7	1,2,3	92.61
8	33	1,2,5,6,7	1,2,3	92.85
9	36	1,2,3,5,6,7	1,2,3	92.77
10	39	1,2,3,4,5,6,7	1,2,3	92.40

Table 6.6.: Tests results on all motions, all subjects, window size of 300ms and HMMs with 14 states. More results with different sensor combinations can be found in Patzer, Isabel and Asfour, Tamim (2019).

6.3.2. Corrupted Data

This section explains some irregularities that were found in the raw data found in the data base. More specifically the corrupted data is found in the values corresponding to the force sensor 4 and some Euler angle values in the recordings of some motions. Before explaining these issues, the results done with original data are presented

Tests run with the original data

This section presents the first tests using the new derived features and the whole raw data that were carried out before noticing that some data was corrupted. The set of raw data used is the same as the one in Beil et al. (2018) and Patzer, Isabel and Asfour, Tamim (2019).

Table 6.7 shows evaluations using derived features alone or combined with raw data of tests run on all the subjects, window size of 300 ms and using HMMs with 14 states. For each configuration a test using the differences between feature vectors was run, as explained in Section 5.2.1. The Table 6.7 shows when this preprocessor was done when *Diff=Yes*. Meanwhile Table 6.8 shows the same setup configuration than in the previous table, but this time all the evaluations were done on single subjects. The results of the last column are the average accuracies of the accuracy from the evaluation of each subject.

From this first tests we want to highlight the accuracy obtained using just knee moments with more than a 34% evaluating on all the subjects and around 75% evaluating on single subjects. Although the results are lower than the ones of Table 6.6, only 3 features are used. Also we realised that the use of linear acceleration (configuration of *Moments* and *IMU(l)*), the performance of the classification drops off significantly when the differences among feature vectors are calculated.

All subjects, W300 and S14				
Features	Num. Features	Num. Sensors	Diff.	Accuracy(%)
J.angles	2	2	No	41.92
			Yes	29.43
Moments	3	7	No	34.60
			Yes	34.42
Moments, IMUs (l)	12	10	No	82.22
			Yes	67.48
Moments, IMUs (e)	12	10	No	82.69
			Yes	88.28
Moments, IMUs	21	10	No	87.60
			Yes	89.07
Moments, FSs	39	7	No	89.34
			Yes	92.27
Moments, IMUs, FSs	42	10	No	88.53
			Yes	92.02

Table 6.7.: Results and compute times for configurations using only raw data. Tests were run on all the subjects, window size of 300 ms and HMMs of 14 states.

Dismission of Force Sensor 4

During the Resemblance Analysis some anomalous values in the motion data were found corresponding to the Force Sensor 4 (FS4). In some cases components on X, Y and Z directions of this sensor take values inside the range of values 30 and 100 Newtons. The magnitude is much bigger as the one that this sensor usually takes, as it can be seen in the Table 5.4.

Single subjects, W300 and S14				
Features	Num. Features	Num. Sensors	Diff.	avg. Accuracy(%)
J.angles	2	2	No	76.70
			Yes	66.19
Moments	3	7	No	75.74
			Yes	75.27
Moments, IMUs (l)	12	10	No	97.59
			Yes	88.77
Moments, IMUs (e)	12	10	No	98.81
			Yes	95.66
Moments, IMUs	21	10	No	99.15
			Yes	95.85
Moments, FSs	39	7	No	99.80
			Yes	97.98
Moments, IMUs, FSs	42	10	No	99.82
			Yes	96.85

Table 6.8.: Results and compute times for configurations using only raw data. Tests were run on all the subjects, window size of 300 ms and HMMs of 14 states.

To show the relevance of this issue, Table 6.9 shows some tests carried on all the subjects with the three features of the moments. For the data setup of the tests in rows 2 and 4, the feature of the moments was calculated without using the components regarding the FS4. If we take a look at the two configurations presented, the classification results in the cases where FS4 was not used, the results improve about a 2%.

In Patzer, Isabel and Asfour, Tamim (2019), a dimensional reduction was carried out selecting the best sensor setup to do the motion classification using an N-Best Feature Subset Exploration. The last force sensor selected to include in the tests was just FS4, which it even makes the accuracy in the classification decrease. We know now the main cause of this bad performance. Table 6.6 shows the results of the aforementioned work where this fact can be observed. In order to face this problem it was decided to not count on this force sensor during the motion classification tests.

Tests on all subjects, all motions, W300 and S14			
Differences	Features	Number	Accuracy(%)
No	Moments	3	34.78
No	Moments without FS4	3	37.60
Yes	Moments	3	34.02
Yes	Moments without FS4	3	36.42

Table 6.9.: Tests results carried out on all the subjects that compare a configuration where the moments values were calculated using FS4 and an other where they were not used.

Dismission of some IMUs files

Another problem found on the data was some Euler angles values of the recordings which do not have any value assigned, just a Not a Number string ('NaN'). The Expressions 5.3, 5.4 and 5.5 are the ones that were used to calculate these values from the quaternion values recorded by the IMUs. There is the possibility that the dominator in this equations gives sometimes a number close to 0, which would explain these results. Due to a lack of time, it was decided just remove the files of IMU's data where this

problem appears. These data files are not too many, just for the subject 1722 and the motions *Sit Down* and *Stand Up*.

6.3.3. Comparison of Derived and Primary Features

The tests that appear in this section target to compare the new derived features in comparison with the raw data from where they were extracted. First each derived feature is evaluated alone and afterwards it is mixed with other derived features or with the primary features from where they were calculated (e.g combination of 'Moments' and 'Force Values' or combination of 'Joint Angles' and 'Euler angles').

In Table 6.10 can be seen some setups and the corresponding results. In the first four rows moments features are evaluated alone. Results show that the norm of the moment vector is the single feature that performs better. Only using the three components of the moments the results are around 37% of accuracy. If we compare the results of 'Forces' and 'Forces and Moments', the last case, in which a higher number of features is used, it even makes the results a bit worse. The main reason of this fact, might be an excess of data used (over-fitting in the training process) and additionally the moments are not adding new data to train the model. As it is said in Freeman (2014), redundant and correlated features increase classifier complexity without adding additional information.

Moments and Forces			
Differences	Features	Number	Accuracy(%)
No	Mx	1	17.64
No	My	1	16.88
No	Mz	1	16.31
No	M	1	20.55
No	Moments	3	37.60
Yes	Moments	3	36.42
No	Forces	18	81.78
Yes	Forces	18	82.32
No	Moments and Forces	21	81.43
Yes	Moments and Forces	21	81.31

Table 6.10.: Tests results carried out on all the subjects that compare different configurations of derived features (knee moments) and their primary features (forces from six force sensors). Window size of 300 ms and 14 states used for the HMMs.

In Table 6.11 we can see a similar comparison like the one previously done. Using the joint angles of the leg as a single feature, the results obtained are better than using one of the moment components as a single feature to train the HMMs. Joint angles together, being just two features, perform also better than when using the three features from the moment vector. Moreover, the same effect occurs when mixing together derived and primary features, since the results did not improved when adding the joint angles to the Euler angles. We conclude by saying that for future studies it will be better not to face the improvement of the motion classification by mixing derived features with features that were used to calculate them

6.3.4. Performance of Filtered Data

In this section the results obtained using the filter defined in Section 5.2.1 are presented. These test where only carried out for classifications where features of the knee moment vector are considered, since the filter was only designed for this kind of data.

As it can be seen in Table 6.12 and if we consider the configuration where the data differences were not calculated, the improvement on single subject tests is almost a 7% increment. Meanwhile in Table 6.13, for tests on all the subjects, the results improved in nearly a 3% using the configuration with filtered

Joint Angles and Euler Angles (IMUs)			
Differences	Features	Number	Accuracy(%)
No	Knee Angle	1	21.32
No	Ankle Angle	1	24.22
No	Joint Angles	2	41.92
Yes	Joint Angles	2	29.43
No	IMU (e)	9	81.22
Yes	IMU (e)	9	87.00
No	Joint Angles, IMU (e)	11	80.36
Yes	Joint Angles, IMU (e)	11	86.11

Table 6.11.: Tests results carried out on all the subjects that compare different configurations of derived features (joint angles) and their primary features (Euler angles from IMUs, *IMU(e)*). Window size of 300 ms and 14 states used for the HMMs.

data. In Section 6.3.5 it is possible to see also an improvement on the results for the test configurations where the data has been filtered. It is important to outline that the filter needs to be tuned for each type of feature, otherwise the results can get worse. This is the reason why when the filter is applied in moment data where the differences have been calculated, the accuracy decreases.

For future work it would be interesting to design filters tuned for each feature and see if the classification improves. At the same time we should take a look on the compute times, because this kind of preprocessor might increase the preprocessor time. Also it would be good to think about how to apply this approach during on-line execution.

Tests on single subjects, all motions, W300 and S14				
Differences	Filtered	Features	Number	avg Accuracy(%)
Yes	No	Moments	3	75.27
No	No	Moments	3	75.74
Yes	Yes	Moments	3	71.74
No	Yes	Moments	3	82.42

Table 6.12.: Tests results carried out on single subjects that compare different configurations with data filtered and not filtered. Window size of 300 ms and 14 states used for the HMMs.

Tests on all subjects, all motions, W300 and S14				
Differences	Filtered	Features	Number	Accuracy(%)
Yes	No	Moments	3	36.42
No	No	Moments	3	37.60
Yes	Yes	Moments	3	34.50
No	Yes	Moments	3	41.13

Table 6.13.: Tests results carried out on all the subjects that compare different configurations with data filtered and not filtered. Window size of 300 ms and 14 states used for the HMMs.

6.3.5. Comparison of Topologies

This chapter refers to Section 3.3.3 where different types of HMMs were introduced. In order to try the performance of other topologies and find a better representation of motions' models, some tests were

carried out changing the transition matrix. At the same time, models with lower number of states were tested (until now testing always with 14 states) to see if less complex HMMs would represent better the motions.

Figure 6.13 shows a multiplot where each subplot displays the accuracy result versus the compute time of the topologies *Fully Connected*, *Left-Right* and *Cyclic*. The tests were run using the moment feature and as always during this thesis, a window size of 300 ms. Each column of the plot shows results of tests run using the same number of states for the model representations (5, 10 and 14 states). The first column corresponds to tests where the differences were not calculated on the data. In the second column we can find the results using data where the differences between consecutive values were calculated. The first fact that stands out is the difference of computational time between the *Fully Connected* topology and the other two. For 5 states the computational time of the *Fully Connected* is approximately 3 times more, for 10 states almost 5 times more and for 14 states more or less 3 times more. At the same time we have to underline that although the accuracy results did not improve, they didn't get too much worse. For 5 states the accuracy drops in a range of 2-4%, for 10 states 4-5% and for 14 states 5-6%. Also it seems that accuracy results with the type of HMM *Left-Right* are usually a bit better than the results with *Cyclic* topology. In order to compare the results of the different configurations under the same conditions, all the tests were run in the same computer. The computer was an Intel Core i7-8700K CPU 3.70GHz and 11 cores were used for multiprocessing.

As these topologies allow a great reduction regarding compute times, it was considered to run tests with the best raw data configurations using 14 states and with the cyclic topology. In that way, we can have an idea of how is the performance using simpler models in terms of accuracy and compute time for the best configurations of previous works. Table 6.14 shows pairs of results, one setup corresponding with the best configurations found in Patzer, Isabel and Asfour, Tamim (2019) and the other setup with the same configuration except for the topology used (cyclic). We can see that the compute times for a cyclic topology are about one-fifth of the compute time used for the fully connected topology. With respect to the accuracies, it drops in the cyclic case between 1 and a 4%. The more number of sensors used, less significant is the difference in accuracy between both topologies.

Tests on all subjects, all motions, W300 and S14				
Num. Sensors	Features	HMM Topology	Accuracy(%)	Time[min]
2	IMU:1, 3	Fully connect	86.99	236.69
		Cyclic	82.95	58.55
3	FS:5 IMU:1, 3	Fully connect	89.71	244.98
		Cyclic	86.25	51.13
4	FS:6, 7 IMU:1, 3	Fully connect	90.76	250.83
		Cyclic	88.05	51.55
5	FS:3, 5 IMUs	Fully connect	91.17	253.27
		Cyclic	89.76	55.93
6	FS:1, 3, 5 IMUs	Fully connect	92.30	255.98
		Cyclic	90.12	55.91
7	FS:1, 5, 6, 7 IMUs	Fully connect	92.72	256.91
		Cyclic	91.08	55.24
8	FS:1, 2, 5, 6, 7 IMUs	Fully connect	92.98	256.78
		Cyclic	91.20	56.82
9	FS:1, 2, 3, 5, 6, 7 IMUs	Fully connect	92.77	289.12
		Cyclic	91.56	57.75

Table 6.14.: Tests results of tests carried out on all the subjects to compare classification performances for HMM of type full-connected and cyclic. Window size of 300 ms and 14 states used for the HMMs.

In conclusion, we could say that *Left-Right* and *Cyclic* topologies are a good option when the compute

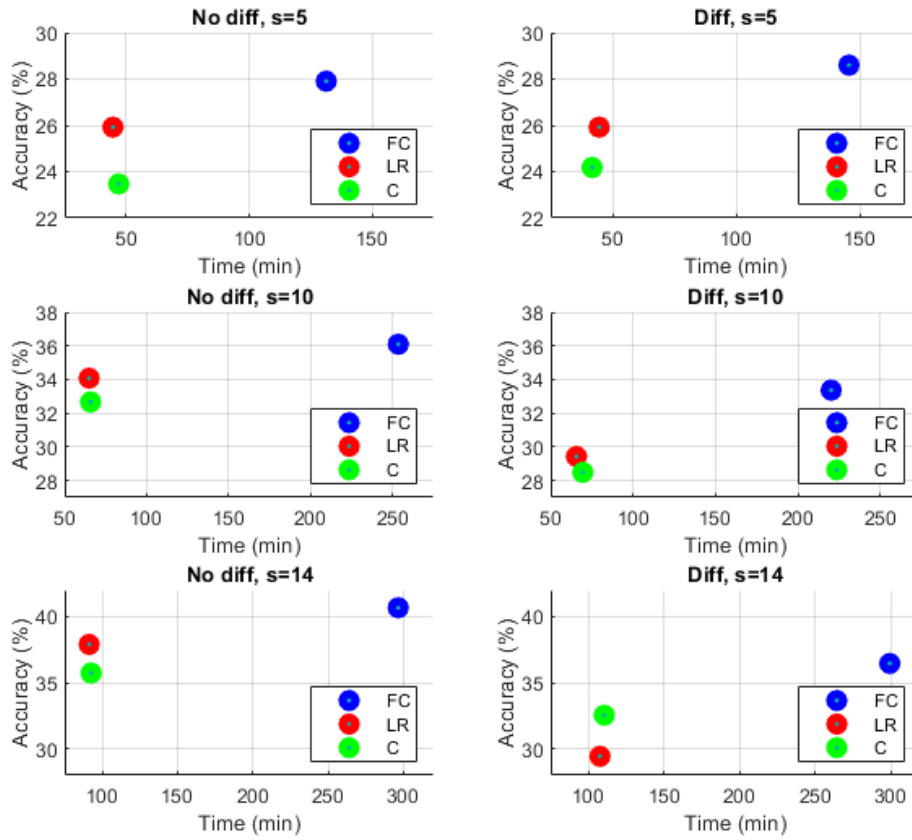


Figure 6.13.: Plot of results for different type of HMM and different number of states. The tests were carried out using the moment feature, on all the motions and window size of 300 ms. *FC*: Fully Connected (blue dots), *LR*: Left-Right (red dots) and *C*: Cyclic (green dots). The *S* on the top of each subplot refers to the number of states.

time is critical in the classification or is worth to reduce it to the detriment of accuracy. For future work it would be interesting to try other type of HMMs and number of states in order to find the better configuration an the most suitable trade-off between accuracy and compute time.

6.3.6. Effect of the Correlations

This section will connect the results and conclusions from the *Resemblance Analysis* (Section 6.1) to try to prove the hypothesis set in the Section 5.1: "One reason of having a high accuracy in the classification for a subject, in comparison with others, is because that subject performs a motion always in a really similar way". To prove it, the results of different tests using the knee moment vector were connected to the results of the resemblance analysis (Section 6.1.3). Tests carried out on all the subjects were considered from where the single subjects results were pulled out to perform an *Inter-Subject* analysis.

Table 6.15 shows the setup of each test and a letter to identify them. Table 6.16 shows the accuracy obtained for each subject in the tests designed with a letter in the aforementioned table. Finally, Figure 6.14 displays a plot of all the results to show in a qualitative way how are the global correlations connected (shown in Table 6.3) and the single subject accuracy results. On the last figure it can be observed that exists a proportional tendency between the accuracy obtained and the intra-correlation calculated for a subject. Only tests using the knee moment features were run for this study since the correlations were calculated using only these features. In future work it would be interesting to calculate correlations among accelerations or Euler angles and do a similar study. In that way this tendency might would be better identified.

To end this section we could simply say that data used for training with less variability (higher correlations) would improve the results of the tests of the motion classifier. However, working in the improvement of this classifier with this ideal data could be detrimental in real applications. In real live the variability of motions is much wider that the one found in the data base of motion recordings. Therefore, motion data that present this variability and cause models not so well defined, help us to prepare the classifier in real situations and to develop a robuster way to identify motions.

All subjects, W300 and S14					
Differences	Filtered	Features	Number	avg Accuracy(%)	Test Name
Yes	Yes	Moments	3	32.49	A
No	Yes	Moments	3	37.42	B
Yes	No	Moments and IMU (e)	12	88.28	C
Yes	No	Moments and IMU (l)	12	67.48	D
No	No	Moments and IMU (e)	12	82.69	E
No	No	Moments and IMU (l)	12	88.22	F
Yes	No	Moments and IMU	21	89.07	G
No	No	Moments and IMU	21	87.60	H

Table 6.15.: Different test setups that were run to do an study between the accuracy and correlation values among subjects.

6.3.7. Best Evaluations with Derived Features

In this last section of this chapter, the results of the most significant configurations tested are shown and discussed. Table 6.17 shows the results and the compute times of the best configuration sets that were found in Patzer, Isabel and Asfour, Tamim (2019) where a dimensional reduction was pursued. The configurations are ordered from less to more number of features and sensors. In all the cases the differences of the feature vectors were calculated, a window size of 300 ms was used, tests were run on all the subjects and a 5-fold cross validation was used to test the performance. Meanwhile in Table 6.18 we can see evaluations with the same configuration than before, but using derived features or a combination

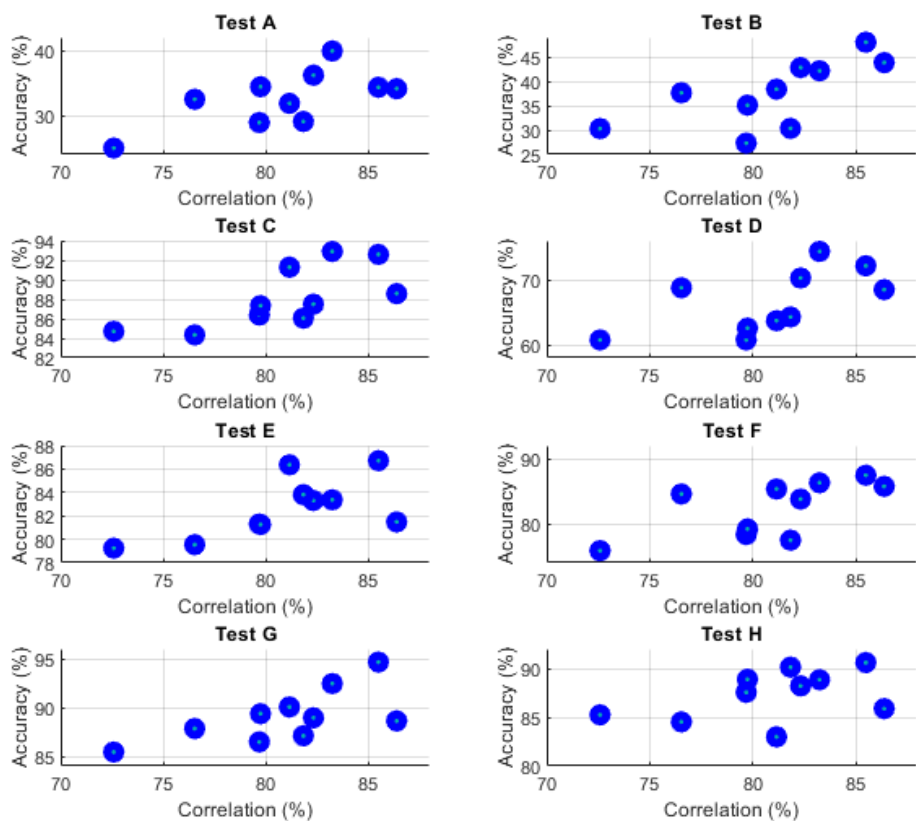


Figure 6.14.: Plots of the results *Accuracy(%)* vs *Correlation(%)* of the different tests shown in Table 6.15 and Table 6.16.

Subject	A	B	C	D	E	F	G	H	mean
ID1717	34.10	37.64	88.58	68.47	81.48	85.73	88.62	85.88	73.49
ID1718	24.98	30.29	84.71	60.74	79.24	75.84	85.43	85.24	68.96
ID1719	34.32	48.00	92.59	72.13	81.26	87.44	94.65	90.59	77.69
ID1720	36.18	42.80	87.49	70.27	83.31	83.77	88.94	88.21	75.05
ID1722	32.48	43.81	84.34	68.76	83.36	84.56	87.84	84.52	73.61
ID1723	34.41	35.10	87.34	62.55	86.71	79.19	89.36	88.89	73.99
ID1724	31.86	38.37	91.28	63.69	79.54	85.33	90.05	82.99	72.65
ID1725	29.57	30.34	86.03	64.27	83.81	77.46	87.10	90.13	71.89
ID674	39.89	42.14	92.89	74.34	83.37	86.28	92.45	88.84	76.78
ID917	28.89	27.32	86.38	69.74	81.29	78.33	86.47	87.55	70.48

Table 6.16.: Subject single results of the accuracy (%) for each test shown in Table 6.15. On the first row appear the letters that connect these results with the table were the setups of the tests are written on.

of primary features (raw data) and derived features. In this table the results are ordered from less to more number of features and for each configuration two tests were carried out: one where the differences of the feature vectors were calculated (*Diff=Yes*) and another one where they weren't (*Diff=No*). The *Moments* features refers to the three components of the knee moment vector. Regarding the compute times, in both tables and all cases the same type of computer run the tests, an Intel Core i7-8700K CPU 3.70GHz and 11 cores were used for multiprocessing.

Considering the features of the moment vector, we notice that the feature of the norm of it is really representative of the forces. Firstly, as we could seen in Table 6.10, the norm of the moment vector was the feature that got the best performance comparing with the use of other single features extracted from the forces. And now in Table 6.18, if we check the setup using IMI and IMUs, we got a good accuracy around 89% considering that we are representing the 18 force features from the forces in just one feature. Even when we compare the setup using the three components of the moment vector and the IMUs (*Moments and IMUs*) with the aforementioned configuration (*IMI and IMUs*), even though with the first configuration two more features are used, the results are almost the same but the compute time increases in the case with more features, as expected. To sum up, we can state that the norm of the moments vector is a good representative of all the forces and a good option in order to carry out a dimensional reduction in the data set.

About the features regarding the joint angles, combining them with forces (*J.angles and Forces*) the accuracy achieved is 88.27% in the case where the differences were calculated. Considering that all the Euler angles (9 features) are presented as only two features, the results are notable. If we compare the results of only using forces (setup using *Forces* from the Table 6.10) with the addition of also these two angles, the increment is around a 6%. If we take into account that with only forces we already obtained around 82%, the increment is significant using only two more features.

Using together the two types of derived features (*J.angles, Moments*) the accuracy achieved is 78.43% in the case where the differences were calculated. Although it is a 10% lower than the first configuration of the Table 6.17 in which kinematic and dynamic variables are used (*FS5 and IMU:1,3*), in the first case we are using one third of features. Moreover, the compute time is reduced a 13%. In an other setup where both derived features are used together with linear accelerations (*J.angles, Moments and IMUs(l)*), the results achieved are around 89% in the case when no differences were calculated. This result is comparable with the one where a similar amount of features is used (*FS5 and IMU:1,3*) of the Table 6.17.

With regard to the data preprocessing, it was spotted that when the differences were calculated the performance of the motion algorithm is usually better in almost all the cases. Only when the linear accelerations are used, the accuracy on the classification obtained is higher if the differences were not calculated. This can be seen in Table 6.18 and setup configuration of (*J.angles and IMUs(l)*), (*Moments*

and $IMUs(l)$ and ($J.angles$, $Moments$ and $IMUs(l)$).

Finally, regarding the compute times, we can see that the more features are used the higher is the time needed. However, it seems that from the amount of approximately 20 features the addition of new features increase the compute time does not increased a lot.

Primary Features - All subjects, W300 and S14					
Features	Num. Feat.	Num. Sens.	Diff.	Accuracy(%)	Time[min]
IMU:1, 2	12	2	Yes	86.99	236.69
FS5 IMU:1, 3	15	3	Yes	89.71	244.98
FS:6, 7 IMU:1, 3	18	4	Yes	90.76	250.83
FS:3, 5 IMUs	24	5	Yes	91.17	266.78
FS:1, 2, 3, 5, 6, 7 IMUs	36	9	Yes	92.77	303.34

Table 6.17.: Results and compute times for configurations using only raw data. Tests were run on all the subjects, window size of 300 ms and HMMs of 14 states.

Primary and Derived Features - All subjects, W300 and S14					
Features	Num. Feat.	Num. Sens.	Diff.	Accuracy(%)	Time[min]
J.angles, Moments	5	9	No	70.98	213.83
			Yes	78.43	211.47
J.angles, IMUs (1)	11	3	No	84.94	227.98
			Yes	76.61	216.15
Moments, IMUs (1)	12	9	No	83.90	233.53
			Yes	69.54	235.26
J.angles, Moments, IMUs (1)	14	9	No	88.93	241.75
			Yes	83.61	239.29
M , IMUs	19	9	No	88.88	244.79
			Yes	89.39	247.17
J.angles, Forces	20	9	No	84.18	250.55
			Yes	88.27	253.28
Moments, IMUs	21	9	No	88.62	261.49
			Yes	89.89	259.40
J.angles, Forces, Moments	23	9	No	83.96	256.78
			Yes	88.07	259.11
J.angles, Forces, IMU (1)	29	9	No	88.87	256.00
			Yes	89.31	255.49

Table 6.18.: Results of different configurations using raw data and derived features. The *Momments* feature refers to the knee moments. Tests run on all the subjects, window size of 300 ms and HMMs of 14 states.

7. Conclusion and Outlook

In this thesis the use of derived features extracted from data recorded by a passive exoskeleton was analysed in the context of motion classification. The initial set of raw data consisted of 3D force vectors recorded by force sensors and 3D linear accelerations and Euler angles recorded by three IMUs. The data set consists of recordings of 10 subjects performing 14 different daily motions, each one repeated 10 times. It was used as the training and testing set for the motion classifier.

In order to improve the motion classification by using more meaningful features, derived features were extracted from the raw data aforementioned. The usage of these new features can additionally reduce the amount of training data required. A first set of new features corresponding to the knee moments were calculated. These features give a dynamic description of the motion and they are extracted from the force vectors recorded. The other set of derived features are the joint angles of the knee and ankle, which give a kinematic meaning of a motion and are calculated using the Euler angles.

Before starting the search of the best setup for the classification system, the new derived features of the knee moment vector were analysed in order to study the consistency and spot patterns among data from the same subject and from different subjects. A resemblance analysis was carried out in which correlation between recordings of data were calculated to quantify the similarities among them. Firstly, only recordings of the same motion were considered. An analysis, named *Intra-Subject Analysis*, was focused on the study of single subjects. The repetitions performed by the subjects ID1717 and ID1719 gave the highest correlations meaning that they performed motions each time in a more similar way than the other subjects. This fact was later connected with the motion classification to see which influence has in the results of the classification. A second analysis named *Inter-Subject Analysis* had as a target spot the motions that are usually more similarly performed by different subjects. The motions *Stand Up*, *Sit Down* and *Lift Object* obtained the highest correlation indicating that they are easier to repeat more similarly than the others. Meanwhile, motions like *Going Downstairs* or *Walking Backward* obtained the lowest correlations, probably because they are more complex and take more time to perform. For future work a resemblance analysis using the joint angles could be considered to study also this new derived features and also in that way the motions could be analysed kinematically. About the study of the joint angles feature, these new features were compared with the theoretical joint angles of the knee and ankle. The knee joint angle calculated bears resemblance with the theoretical one and the angular variation is almost the same. Regarding the ankle joint angle, although the similarities in the pattern bears less resemblance, the angular variation is comparable.

Once the new derived features were analysed, the classification system based on Hidden Markov Models was evaluated with them. Considering the results in Beil et al. (2018) and in Patzer, Isabel and Asfour, Tamim (2019) a sliding window approach using a window size of 300ms concatenated each 10ms was used as the data to train the HMMs. Each HMM was formed by 14 states fully connected and Gaussian distributions were used to represent the observations. A 5-fold cross validation was used to test the performance of the classifier. Different sets of derived features were tested alone and in combination with the raw data, looking for the setup that achieves the best performance. Not only the features used were changed, but also other characteristics of the classification system, like the data preprocessor or the structure of the Hidden Markov Models (HMMs) used to represent each motion.

As a first step, a comparison of the performance of the classification using derived features and primary features was carried out. As a single feature, the norm of the moment vector was the dynamic feature that achieved the best classification with a 20.55% of accuracy. Using the whole moment vector of only three features, an accuracy of 37.60 % was obtained. Considering the joint angles, the ankle angle got a better result of 24.22% than the knee angle with a 21.32%. As a combination, the two kinematic derived features achieved a 41.92% while the knee moment vector's accuracy was beyond 38%. These results showed that joint angles perform better alone than the moment vector. Besides, when combining

a derived features with the features from where they were calculated, the classification results do not improve or even get worse in comparison with a setup where only the primary features are used. For future studies this kind of combination is not recommendable since features are correlated and become redundant, which increases the complexity of the models without adding new information.

To aim an improvement of the motion classification results, a filtering of the knee moment vectors was considered. A digital low-pass filter was tuned in a way that singularities and noise were mitigated. Results of evaluations on single subjects using data filtered achieved a 7% increment with regard to evaluations using data not filtered. For the case of testing on all the subjects the increment was around the 3%. Only tests with the knee moment feature were run, since the filter was not designed for the other features. This preprocessor could be done on all the features since a notable improvement might be obtained. A specific filter should be tuned for each kind of data in order to get the best performance. The preprocessor compute time should be checked, if an on-line application of the classification system is pursued.

The idea of improving the classification method by creating better models for the motions was also studied. Hidden Markov Models with different number of states and topologies were evaluated looking for a better performance of the classification method. Until now a fully-connected topology and 14 states were used to train the models. As each motion is formed by short movements that have always a continuity on time, topologies that represent a successive set of actions were considered. Therefore, evaluations were carried out using Left-Right and Cyclic topologies. The results shown a huge reduction of the compute time of the no fully connected topologies of about one third part of the time needed when in both configuration 14 states were used. The classification accuracy results using these new topologies dropped in a range of 2-4%. The complexity reduction of the HMMs and consequently the compute times that *Left-Right* and *Cyclic* topologies achieve, make these topologies a really suitable option if the compute time is critical in the classification. For future work it would be interesting to try other types of HMMs combined with different number of states in order to find the best configuration. Additionally, an specific HMM structure for each motion could be studied. As it was proved in the resemblance analysis, some motions are more complex than others causing more variability among the motion data. With that idea in mind, complex motions could be represented by HMMs with more states and more complex topologies, and the other way around could be applied for the simplest motions.

Different setups using derived features and raw data were tested looking for the best motion classification results. About these evaluations we would like to highlight the performance of the norm of the moment vector. This feature is a good representative of the force vectors if we consider that an accuracy of 89% was obtained when combined with linear accelerations and Euler angles. Regarding the joint angles combined with forces, the accuracy obtained was about the 88.2%, a good result if we consider that the kinematic information of the motion is represented by only 2 features. If the usage of derived features with raw data is compared with the best results of the configurations setups from Patzer, Isabel and Asfour, Tamim (2019) (e.g. 90.73% of accuracy using 18 features and 4 sensors), not better accuracy results are obtained. However, some results of setups using derived features, are comparable with the best setups where only raw data was used (e.g. 89.39% of accuracy using $|M|$ and the IMUs). Although the best classification results were not improved, the usage of derived features turned out to be a good representation of their primary data. Continuing in this direction of improvement, new derived features could be extracted and tested like the ankle moment vector, which was calculated but not evaluated in this thesis because of a lack of time.

Table A.1 in the Appendix shows a proposal of derived features that could be used for the motion classification. All the features on that table, except for the last one, should be calculated considering a data window. All the features considered until now are time dependent and for each timestamp a set of these features is obtained. If features calculated from the data of a set of values is considered the current approach of the classification system should be changed. An option could be the use of a single HMM with a number of states equal to the motions considered. In that case the classified data would be sequences of data formed by different motions.

Another promising direction of improvement would be to carry out a dimensional reduction with all the features now available, as it was done in Patzer, Isabel and Asfour, Tamim (2019), using the algorithm *N-Best Feature Subset Exploration* from Mandery et al. (2016).

A. Appendix

A.1. Derived Features

Suggestions of derived features for motion classification			
Name	Data used	Dim.	Reference
Maximum/Minimum	IMUs + FSs	39	Feldhorst and Hompel (2013)
Arithmetic mean	IMUs + FSs	39	Chen et al. (2016b)
Standard deviation	IMUs + FSs	39	Chen et al. (2016b)
Variance	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Signal Magnitude Area	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Mean Absolute Deviation	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Interquartile range IQR	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Zero-accel crossings freq	IMUs	9	Lin and Kulic (2014)
Accel peaks freq	IMUs	9	Lin and Kulic (2014)
Zero-angle crossings freq	IMUs	9	
Angle peaks freq	IMUs	9	
Energie, Fourier transf	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Wavelet transform	IMUs + FSs	39	Óscar D. Lara and Labrador (2013)
Inertias of the leg	IMUs	1 or 3	

Table A.1.: Set of derived features used for motion classification. Except for the last one, all the features should be applied considering the data of a time window.

Description of the features suggested

Joint angles Using IMUs accelerations, compute the angle of the knee and ankle articulations. Only relative angles could be calculated.

Maximum/Minimum Get the maximum/minimum value for each set of data values contained in a window (each window would be reduced into 39 values if all the raw data is used).

Arithmetic mean For each set of data values contained in a window, compute the Arithmetic mean value (each window would be reduced into 39 values if all the raw data is used).

Standard deviation For each set of data values contained in a window, compute the Standard deviation value (each window would be reduced into 39 values if all the raw data is used). First the arithmetic mean should be calculated.

$$\sigma = \sqrt{\frac{\sum_{k=1}^n (x_k - \mu)^2}{n}}$$

Variance For each set of data values contained in a window, compute the Standard deviation value. Each window would be reduced into 39 values if all the raw data is used. First the arithmetic mean

should be calculated.

$$\sigma^2 = \frac{\sum_{k=1}^n (x_k - \mu)^2}{n}$$

Signal magnitude area (SMA) Statistical measure of the magnitude of a varying quantity. For each set of data values contained in a window, compute the SMA. Each window would be reduced into 39 values if all the raw data is used.

$$x_{sma} = \sum_{i=1}^n x_i$$

Mean Absolute Deviation Average of the absolute deviations or the positive difference of the given data and the mean value of it. Each window would be reduced into 39 values if all the raw data is used.

$$x_{sma} = \frac{1}{n} \cdot \sum_{i=1}^n |x_i - \mu|$$

Interquartile range IQR Measure of statistical dispersion, being equal to the difference between 75th and 25th percentiles, or between upper and lower quartiles. Each window would be reduced into 39 values if all the raw data is used.

$$IQR = Q_3 - Q_1$$

Zero-acceleration crossings frequencies For each set of data from the linear acceleration (contained in a window), compute the frequency of zero-acceleration values. Relative times (timestamps) from the recorded data should be considered. Each window would be reduced into 9 values if all the IMUs data is used.

$$(x_t = 0 \parallel \text{sign}(x_{t-1}) \neq \text{sign}(x_t))$$

(e.g if the previous condition happens 10 times during a window of 300 milliseconds size, the *Zero-acceleration crossings frequency* of that window would be 30ms^{-1}).

Acceleration peaks frequencies For each set of data from the linear acceleration (contained in a window), compute the frequency of acceleration peaks. Relative times (timestamps) from the recorded data should be considered. Each window would be reduced into 9 values if all the IMUs data is used.

$$[(x_{t-1} > x_t) \& (x_{t-1} > x_t)] \parallel [(x_{t-1} > x_t) \& (x_{t-1} > x_t)]$$

Zero-angle crossings frequencies For each set of data from the Euler angles (contained in a window), compute the frequency of zero-angle values. Relative times (timestamps) from the recorded data should be considered. Each window would be reduced into 9 values if all the IMUs data is used.

$$(x_t = 0 \parallel \text{sign}(x_{t-1}) \neq \text{sign}(x_t))$$

Angle peaks frequencies For each set of data from the Euler angles (contained in a window), compute the frequency of angle peaks. Relative times (timestamps) from the recorded data should be considered. (each window would be reduced into 9 values if all the IMUs data is used).

$$[(x_{t-1} > x_t) \& (x_{t-1} > x_t)] \parallel [(x_{t-1} > x_t) \& (x_{t-1} > x_t)]$$

Energie, Fourier transformation Consideration of the frequency domain. Preprocessor may be necessary. The scale of the data should be change to detect low frequencies (logarithmic scale). The coefficients of the fourier transformation would be used as a new derived feature.

Inertias of the leg Resistance of the thigh, shank and foot to change in motion. It could be calculated for the knee and/or the ankle's axis of rotation. The angular velocity w could be approximated as the differences of consecutive accelerations divided by the sample time.

$$I_i = \frac{L_i}{w_i} \qquad L_i = r_i \times p_i$$

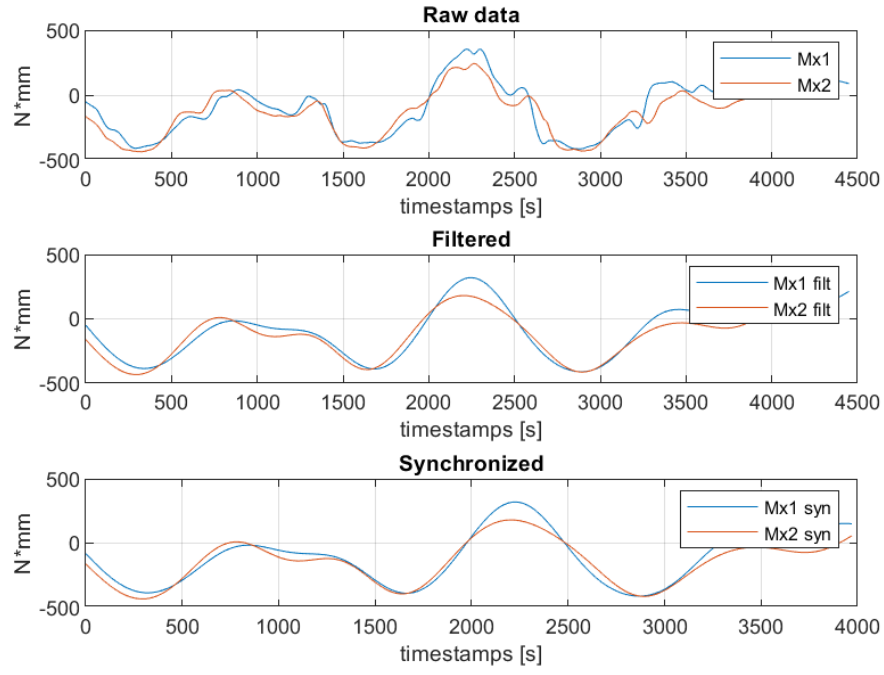
$$w_i = \frac{(a_t - a_{t-1})}{T} \cdot \frac{1}{r_i} \qquad p_i = m_i \cdot v_i$$

$$v_i = \frac{(a_t - a_{t-1})}{T}$$

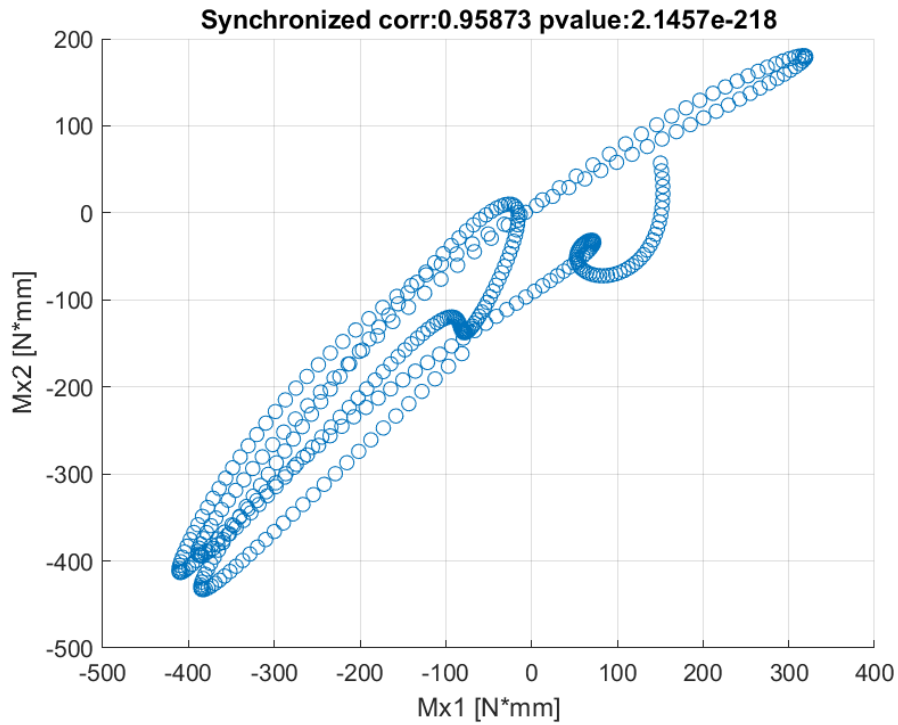
I_i : moment of inertia [$Kg \cdot m^2$], L_i : angular momentum [$Kg \cdot m^2 \cdot s^{-1}$], w_i : angular velocity [rad/s], p_i : momentum [$Kg \cdot m/s$], v_i : linear velocity [m/s], a_i : linear acceleration [m/s^2], T : sample time [s], r_i : perpendicular distance of the Center of Masses of i to the axis of rotation [m] and m_i : mass of the inertia's solid [Kg].

A.2. Correlation Analysis

In this section we can find more examples of correlation analyses of the knee moment features for different motions. For each example of correlation analysis, the corresponding time series plot and scatter plot are shown evaluating two repetitions of a specific subject. Always the M_x moment is displayed, which is the one corresponding to the axis of rotation of the knee joint. The motions that will be shown are: *Walking Backward*, *Going Downstairs Backward*, *Going Upstairs* and *Stand Up*.

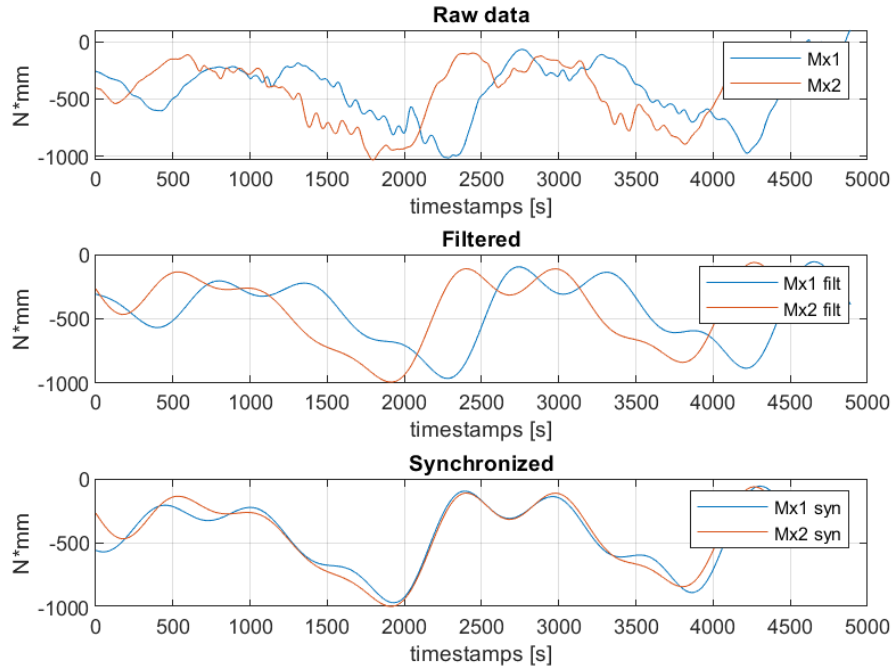


(a) Series plot

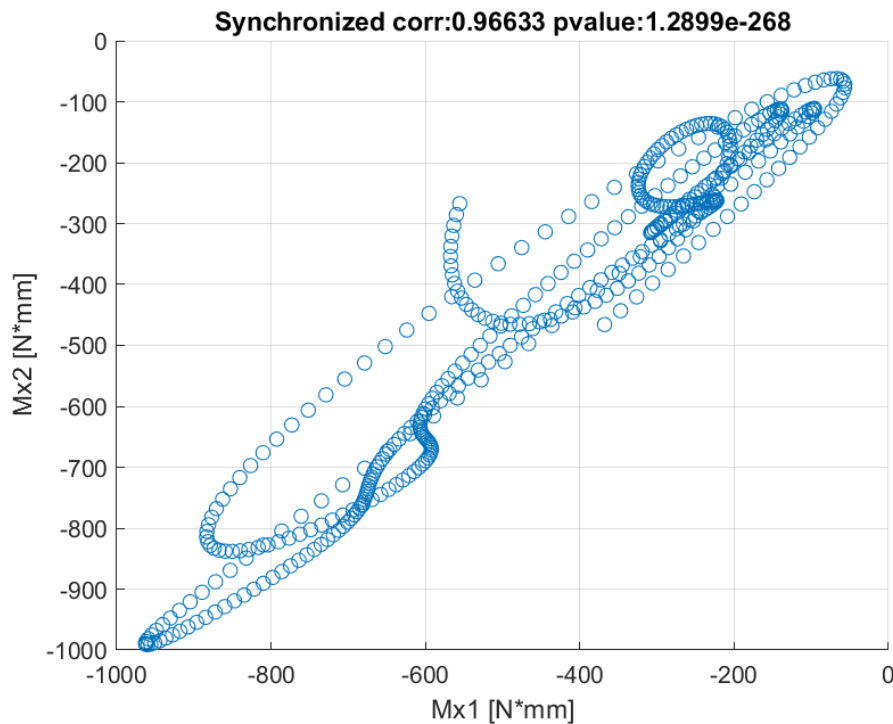


(b) Scatter plot. The correlation results are on the top

Figure A.1.: Samples of the subject 674, motion *Walking Backward*, knee moment M_x and repetitions 1 and 8.

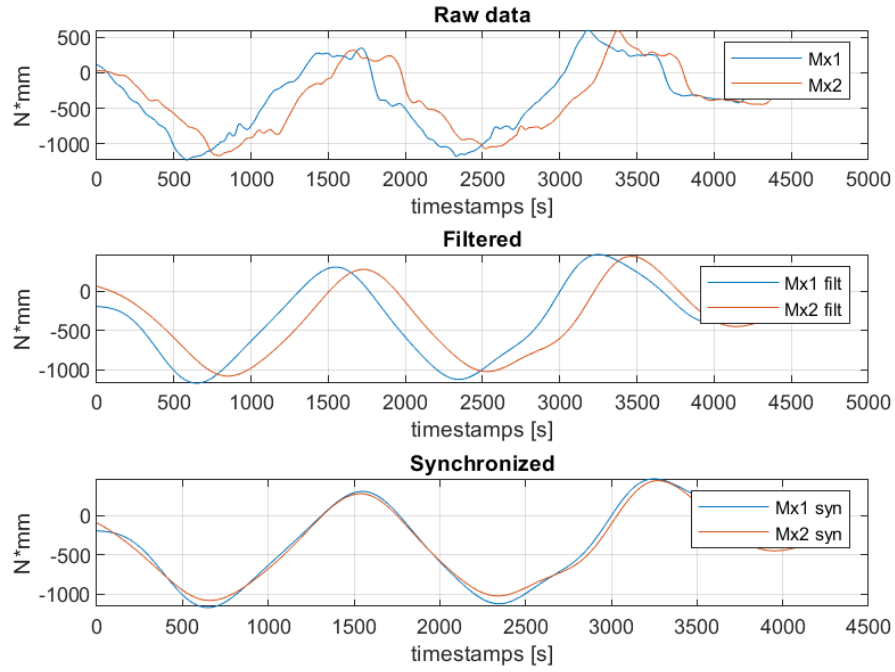


(a) Series plot

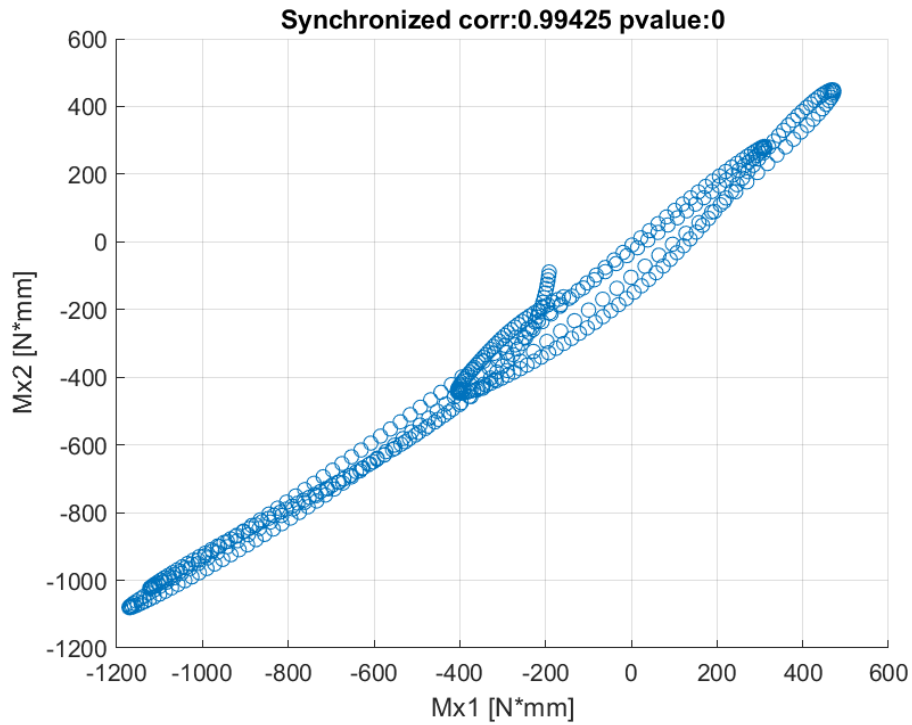


(b) Scatter plot. The correlation results are on the top

Figure A.2.: Samples of the subject 674, motion *Going Downstairs Backward*, knee moment M_x and repetitions 4 and 8.

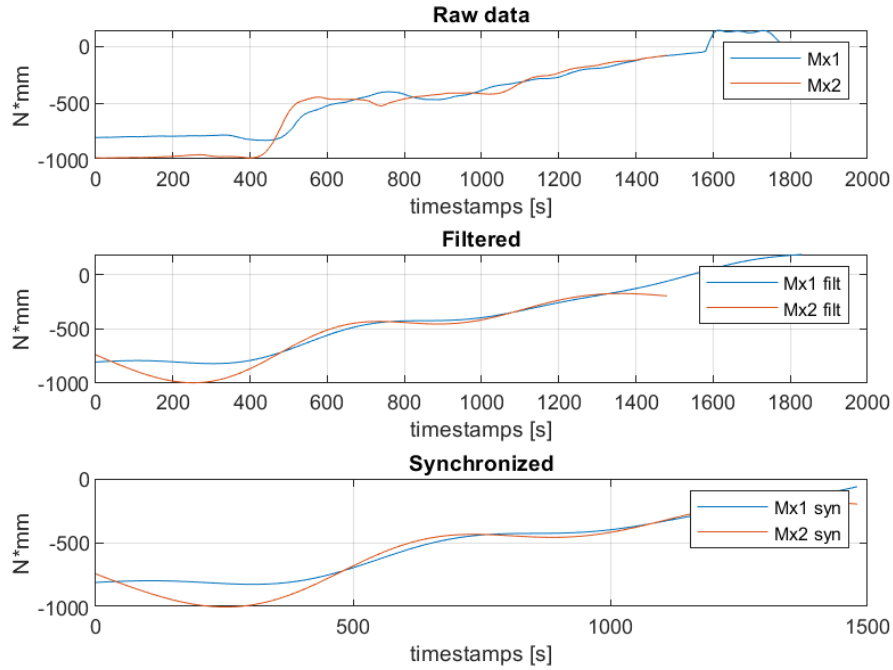


(a) Series plot

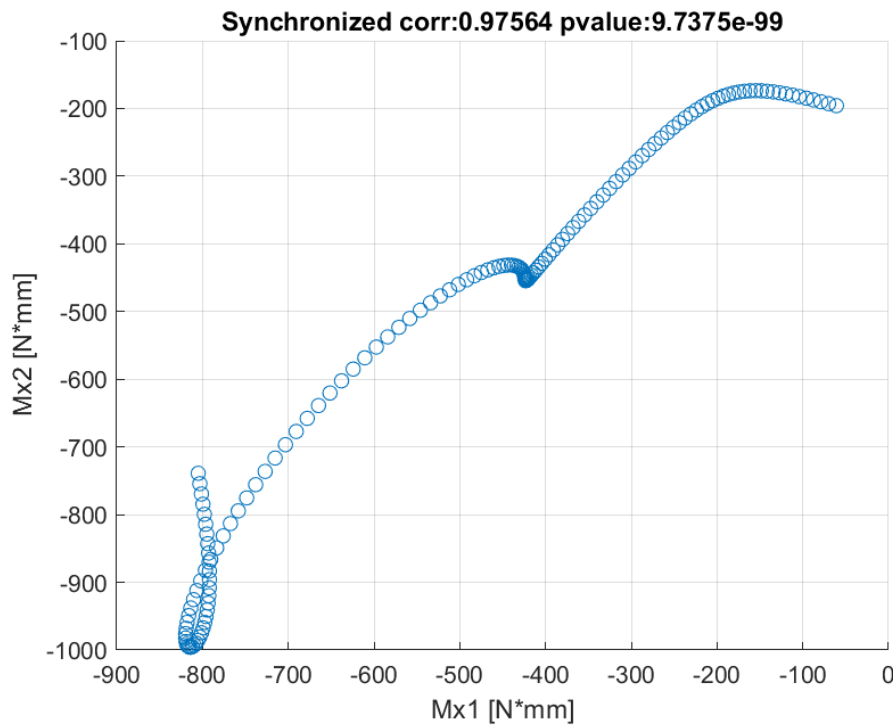


(b) Scatter plot. The correlation results are on the top

Figure A.3.: Samples of the subject 1717 , motion *Going Upstairs*, knee moment M_x and repetitions 1 and 2.



(a) Series plot



(b) Scatter plot. The correlation results are on the top

Figure A.4.: Samples of the subject 1718, motion *Stand Up*, knee moment M_x and repetitions 6 and 9.

Bibliography

- A. Akl, C. Feng, and S. Valaee. A novel accelerometer-based gesture recognition system. *IEEE Transactions on Signal Processing*, 59(12):6197–6205, 2011. ISSN 1053587X. doi: 10.1109/TSP.2011.2165707. 4
- M. D. S. Anjo, E. B. Pizzolato, and S. Feuerstack. A real-time system to recognize static gestures of Brazilian sign language (libras) alphabet using Kinect. *Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems - IHC '12*, 5138:259–268, 2012. URL <http://dl.acm.org/citation.cfm?id=2393574>. 6
- J. Beil, G. Perner, and T. Asfour. Design and control of the lower limb exoskeleton KIT-EXO-1. *IEEE International Conference on Rehabilitation Robotics*, 2015-September:119–124, 2015. ISSN 19457901. doi: 10.1109/ICORR.2015.7281186. 7, 8
- J. Beil, I. Ehrenberger, C. Scherer, C. Mandery, and T. Asfour. Human motion classification based on multi-modal sensor data for lower limb exoskeletons. pages 5431–5436, 2018. doi: 10.1109/iros.2018.8594110. 2, 7, 8, 9, 10, 14, 17, 30, 32, 33, 34, 44, 49, 59
- D. J. Bemdt. Insights into phases of liquid water from study of its unusual glass-forming properties. *KDD-94 Workshop on Knowledge Discovery in Databases*, 398:359–370, 1994. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.407.3857>. 6
- S. Bhatnagar and S. Agrawal. Hand Gesture Recognition for Indian Sign Language: A Review. *International Journal of Computer Trends and Technology*, 21(3):121–122, 2015. doi: 10.14445/22312803/ijctt-v21p122. 4
- Z. P. Bian, L. P. Chau, and N. Magnenat-Thalmann. Fall detection based on skeleton extraction. *Proceedings - VRCAI 2012: 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pages 91–94, 2012. doi: 10.1145/2407516.2407544. 7
- C. M. Bishop. *Pattern recognition and machine learning*. Information science and statistics Computer science. Springer, New York, NY, 2006. ISBN 0-387-31073-8; 978-0-387-31073-2. URL <http://swbplus.bsz-bw.de/bsz250316129cov.htm>; <http://swbplus.bsz-bw.de/bsz250316129vor.htm>; <http://swbplus.bsz-bw.de/bsz250316129kap.htm>; <http://swbplus.bsz-bw.de/bsz250316129inh.htm>. Hier auch später erschienene, unveränderte Nachdrucke. 5, 14
- I. Borg and S. G. Sireci. BOOK REVIEW Modern Multidimensional Scaling : Theory and Applications. pages 277–280, 1997. 5
- Bosch Sensortec GmbH. Data sheet bno055 rev. 1.4. Technical report, Bosch Sensortec GmbH, Juli 2014. 10
- R. A. Brookhuis, T. S. Lammerink, R. J. Wiegerink, M. J. De Boer, and M. C. Elwenspoek. 3D force sensor for biomechanical applications. *Sensors and Actuators, A: Physical*, 182:28–33, 2012. ISSN 09244247. doi: 10.1016/j.sna.2012.04.035. URL <http://dx.doi.org/10.1016/j.sna.2012.04.035>. 10
- B. Chen, H. Ma, L. Y. Qin, F. Gao, K. M. Chan, S. W. Law, L. Qin, and W. H. Liao. Recent developments and challenges of lower extremity exoskeletons. *Journal of Orthopaedic Translation*, 5:26–37, 2016a. ISSN 2214031X. doi: 10.1016/j.jot.2015.09.007. URL <http://dx.doi.org/10.1016/j.jot.2015.09.007>. 1

- B. Chen, C. H. Zhong, X. Zhao, H. Ma, X. Guan, X. Li, F. Y. Liang, J. C. Y. Cheng, L. Qin, S. W. Law, and W. H. Liao. A wearable exoskeleton suit for motion assistance to paralysed patients. *Journal of Orthopaedic Translation*, 11(March):7–18, 2017. ISSN 2214031X. doi: 10.1016/j.jot.2017.02.007. 4
- C. Chen, R. Jafari, and N. Kehtarnavaz. A survey of depth and inertial sensor fusion for human action recognition. *Multimedia Tools and Applications*, 2015. 5
- C. Chen, R. Jafari, and N. Kehtarnavaz. A real-time human action recognition system using depth and inertial sensor fusion. *IEEE Sensors Journal*, 16(3):773–781, 2016b. 2, 61
- R. A. Clark, Y. H. Pua, A. L. Bryant, and M. A. Hunt. Validity of the Microsoft Kinect for providing lateral trunk lean feedback during gait retraining. *Gait and Posture*, 38(4):1064–1066, 2013. ISSN 09666362. doi: 10.1016/j.gaitpost.2013.03.029. URL <http://dx.doi.org/10.1016/j.gaitpost.2013.03.029>. 6
- C. Dondrup, N. Bellotto, F. Jovan, and M. Hanheide. Real-time multisensor people tracking for human-robot spatial interaction. *Workshop on Machine Learning for Social Robotics at International Conference on Robotics and Automation (ICRA)*, 2015. URL <http://eprints.lincoln.ac.uk/17545/1/dondrup.pdf>. 4
- Drouin, Alexander and Laviolette, François. Introduction to machine learning, 2019. URL <https://aldro61.github.io/microbiome-summer-school-2017/>. 17
- Eric Adams. Power-multiplying exoskeletons are slimming down for use on the battlefield, 2018. URL <https://www.popsci.com/army-exoskeletons-lockheed-martin/>. 2
- S. Feldhorst and M. Hompel. Bewegungsklassifikation mithilfe mobiler Sensoren zur Analyse des Kommissionierprozesses. pages 1–5, 2013. doi: 10.2195/lj. 61
- P. Filipe and P. Henriques. Gesture recognition with microsoft kinect tools for socially assistive robotics scenarios. (May), 2017. 7
- C. Freeman. Feature selection and hierarchical classifier design with applications to human motion recognition. pages 1–297, 2014. 5, 6, 51
- D. Gehrig, H. Kuehne, A. Woerner, and T. Schultz. HMM-based human motion recognition with optical flow data. *9th IEEE-RAS International Conference on Humanoid Robots, HUMANOIDS09*, pages 425–430, 2009. doi: 10.1109/ICHR.2009.5379546. 6
- P. F. Guo, P. Bhattacharya, and N. Kharma. Automated synthesis of feature functions for pattern detection. *Canadian Conference on Electrical and Computer Engineering*, (2):1–4, 2010. ISSN 08407789. doi: 10.1109/CCECE.2010.5575224. 5
- P. Händel, B. Enstedt, and M. Ohlsson. Combating the effect of chassis squat in vehicle performance calculations by accelerometer measurements. *Measurement: Journal of the International Measurement Confederation*, 43(4):483–488, 2010. ISSN 02632241. doi: 10.1016/j.measurement.2009.12.019. URL <http://dx.doi.org/10.1016/j.measurement.2009.12.019>. 4
- Q. He, H. Li, B. Zhou, H. Wen, J. Li, B. Xiao, K. Zhang, W. C. Hodgson, and X. Yu. TA-2, a thrombin-like enzyme from the Chinese white-lipped green pitviper (*Trimeresurus albolabris*): Isolation, biochemical and biological characterization. *Blood Coagulation and Fibrinolysis*, 23(5):445–453, 2012. ISSN 09575235. doi: 10.1097/MBC.0b013e32835496b2. 4
- D. Holden, J. Saito, T. Komura, and J. Thomas. Learning motion manifolds with convolutional autoencoders. 313(July):504–507, 2006. 5
- Hybrid Assistive Limb. Medical robot suit hal 3, Aug. 2013. URL <https://www.cyberdyne.jp/english/products/HAL/index.html>. 1

- Hyundai. Hyundai chairless exoskeleton, Oct. 2018. URL <https://www.hyundai.news/uk/brand/hyundai-motor-group-ventures-further-into-new-robotics-industry/>. 2
- J. Jang, K. Kim, J. Lee, B. Lim, J.-K. Cho, and Y. Shim. Preliminary study of online gait recognizer for lower limb exoskeletons. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5818–5824. IEEE, 2017. 2, 4
- I. T. Jolliffe and J. Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374 (2065), 2016. ISSN 1364503X. doi: 10.1098/rsta.2015.0202. 5
- G. Khalili Moghaddam and C. R. Lowe. *Health and Wellness Measurement Approaches for Mobile Healthcare*, chapter Ex Vivo Biosignatures, pages 51–104. SpringerBriefs in Applied Sciences and Technology. Springer International Publishing, Cham, 2019. 5
- KIT H2T. Kit whole-body human motion database, Aug. 2014. URL <https://motion-database.humanoids.kit.edu/list/subjects/>. 7, 11
- S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition. *Bell System Technical Journal*, 62(4):1035–1074, 1983. ISSN 15387305. doi: 10.1002/j.1538-7305.1983.tb03114.x. 16
- J. F. S. Lin and D. Kulic. Online segmentation of human motion for automated rehabilitation exercise analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(1):168–180, 2014. ISSN 15344320. doi: 10.1109/TNSRE.2013.2259640. 61
- Lokomat. Highly intensive physiological gait rehabilitation for severely impaired neurological patients, 2019. URL <https://www.hocoma.com/solutions/lokomat/>. 1
- R. Lun and W. Zhao. *A Survey of Applications and Human Motion Recognition with Microsoft Kinect*, volume 29. 2015. ISBN 0218001415550. doi: 10.1142/S0218001415550083. 4, 5, 6
- R. C. Madeo, C. A. Lima, and S. M. Peres. Gesture unit segmentation using support vector machines: Segmenting gestures from rest positions. *Proceedings of the ACM Symposium on Applied Computing*, pages 46–52, 2013. doi: 10.1145/2480362.2480373. 6
- C. Mandery, M. Plappert, J. Borràs, and T. Asfour. Dimensionality reduction for whole-body human motion recognition. *2016 19th International Conference on Information Fusion (FUSION)*, pages 355–362, 2016. 7, 17, 19, 34, 60
- A. Mansur, Y. Makihara, and Y. Yagi. Inverse dynamics for action recognition. *IEEE Transactions on Cybernetics*, 43(4):1226–1236, 2013. ISSN 21682267. doi: 10.1109/TSMCB.2012.2226879. 20
- T. Matsubara and J. Morimoto. Bilinear modeling of EMG signals to extract user-independent features for multiuser myoelectric interface. *IEEE Transactions on Biomedical Engineering*, 60(8):2205–2213, 2013. ISSN 00189294. doi: 10.1109/TBME.2013.2250502. 5
- L. Miranda, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. M. Campos. Data through key poses learning and decision forests. 7
- L. Miranda, T. Vieira, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. Campos. Real-time gesture recognition from depth data through key poses learning and decision forests. *Brazilian Symposium of Computer Graphic and Image Processing*, pages 268–275, 2012. ISSN 15301834. doi: 10.1109/SIBGRAPI.2012.44. 7
- Musculoskeletal Key. Biomechanics, clinical gait analysis, Oct. 2008. URL <https://musculoskeletalkey.com/biomechanics-2/>. 25, 46, 47

- C. S. Myers, L. R. Rabiner, and A. E. Rosenberg. Performance trade-offs in dynamic time warping algorithms for isolated word recognition. 34, 1979. doi: 10.1121/1.2017729. 6
- S. Nomm and K. Buhhalko. *Monitoring of the human motor functions rehabilitation by neural networks based system with Kinect sensor*, volume 12. IFAC, 2013. ISBN 9783902823410. doi: 10.3182/20130811-5-US-2037.00062. URL <http://dx.doi.org/10.3182/20130811-5-US-2037.00062>. 6
- S. Ohannessian. Historical background. *Language in Zambia*, pages 271–291, 2017. doi: 10.4324/9781315106786-12. 17
- Optoforce Ltd. Omd-30-se-100n datasheet v2.2. Technical report, Optoforce Ltd., Dezember 2016. Optoforce Ltd. 10
- Óscar D. Lara and M. A. Labrador. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys & Tutorials*, 15(3), 2013. 61
- G. Panahandeh, N. Mohammadiha, A. Leijon, and P. Händel. Continuous hidden markov model for pedestrian activity classification and gait analysis. *IEEE Transactions on Instrumentation and Measurement*, 62(5), Mai 2013. 6, 14
- I. P. Pappas, M. R. Popovic, T. Keller, V. Dietz, and M. Morari. A reliable gait phase detection system. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 9(2):113–125, 2001. ISSN 15344320. doi: 10.1109/7333.928571. 4
- C. Park, M. C. Wang, and E. B. Mo. Probabilistic penalized principal component analysis. *Communications for Statistical Applications and Methods*, 24(2):143–154, 2017. ISSN 23834757. doi: 10.5351/CSAM.2017.24.2.143. 5
- Patzer, Isabel and Asfour, Tamim. Minimal Sensor Setup in Lower Limb Exoskeleton for Motion Classification based on Multi-Modal Sensor Data. pages 5431–5436, 2019. doi: 10.1109/iros.2019.8594110. 3, 8, 14, 17, 25, 33, 48, 49, 50, 53, 55, 59, 60
- V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi, and T. Sorsa. Computational auditory scene recognition. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2:1941–1944, 2002. ISSN 15206149. doi: 10.1109/ICASSP.2002.5745009. 6
- J. L. Pons. Wearable Robots: Biomechatronic Exoskeletons. *Wearable Robots: Biomechatronic Exoskeletons*, pages 1–338, 2008. doi: 10.1002/9780470987667. 1
- L. R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. (Februar 1989), 1989. 12, 15, 16, 17
- M. Saerens. A continuous-time dynamic formulation of Viterbi algorithm for one-Gaussian-per-state hidden Markov models. *Speech Communication*, 12(4):321–333, 1993. ISSN 01676393. doi: 10.1016/0167-6393(93)90081-U. 14
- C. M. Senanayake and S. M. Senanayake. Computational intelligent gait-phase detection system to identify pathological gait. *IEEE Transactions on Information Technology in Biomedicine*, 14(5):1173–1179, 2010. ISSN 10897771. doi: 10.1109/TITB.2010.2058813. 3, 4
- M. Stamp. A Revealing Introduction to Hidden Markov Models. *Introduction to Machine Learning with Applications in Information Security*, pages 7–35, 2018. doi: 10.1201/9781315213262-2. 12, 13, 15
- H. Stern, D. Frolova, and S. Berman. Hand gesture recognition for TV remote control using tree-based ensemble and LCS classifiers. *Proceedings of the 2010 International Conference on Image Processing, Computer Vision, and Pattern Recognition, IPCV 2010*, 2(June 2014):687–693, 2010. 4

- P. T. Szemes, H. Hashimoto, and P. Korondi. Pedestrian-behavior-based mobile agent control in intelligent space. *IEEE Transactions on Instrumentation and Measurement*, 54(6):2250–2257, 2005. ISSN 00189456. doi: 10.1109/TIM.2005.858824. 4
- J. Taborri, E. Scalona, S. Rossi, E. Palermo, F. Patanè, and P. Cappa. Real-time gait detection based on hidden markov model: Is it possible to avoid training procedure? In *IEEE International Symposium on Medical Measurements and Applications (MeMeA) Proceedings*, pages 141–145, Mai 2015. doi: 10.1109/MeMeA.2015.7145188. 4
- C. W. Tan and S. Park. Design of accelerometer-based inertial navigation systems. *IEEE Transactions on Instrumentation and Measurement*, 54(6):2520–2530, 2005. ISSN 00189456. doi: 10.1109/TIM.2005.858129. 4
- J. Tchorz, S. Wollermann, and H. Husstedt. Classification of Environmental Sounds for Future Hearing Aid Applications. pages 294–299, 2017. 6
- Ö. Terlemez, S. Ulbrich, C. Mandery, M. Do, N. Vahrenkamp, and T. Asfour. Master Motor Map (MMM) - Framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots. *IEEE-RAS International Conference on Humanoid Robots*, 2015-Febru(Mmm): 894–901, 2015. ISSN 21640580. doi: 10.1109/HUMANOIDS.2014.7041470. 7
- The three million suit. The three million suit. wyss institute wins darpa grant to further develop its soft exosuit, 2014. URL <https://news.harvard.edu/gazette/story/2014/09/the-3-million-suit/>. 2
- Toyota. Welwalk ww-1000, Aug. 2017. URL <https://global.toyota/en/detail/15989382>. 1
- N. Tsuruta and S. K. Aly. Self organizing map vs. spectral clustering on visual feature extraction for human interface. 2006. 5
- UniExo. Provider of rehabillitaion assistance with modular exoskeleton devices, 2019. URL <https://www.uniexo.com/>. 1
- C. J. Van Rijsbergen. Information Retrieval, 2nd edition. *Butterworths*, page Ch.7, 1979. URL <http://www.dcs.gla.ac.uk/Keith/Preface.html>. 18
- S. Viteckova, P. Kutilek, G. de Boisboissel, R. Krupicka, A. Galajdova, J. Kauler, L. Lhotska, and Z. Szabo. Empowering lower limbs exoskeletons: state-of-the-art. *Robotica*, 36(11):1743–1756, 2018. doi: 10.1017/S0263574718000693. 1, 2
- W. Wan, H. Liu, L. Wang, G. Shi, and W. J. Li. A hybrid HMM/SVM classifier for motion recognition using μ IMU data. *2007 IEEE International Conference on Robotics and Biomimetics, ROBIO*, pages 115–120, 2007. doi: 10.1109/ROBIO.2007.4522145. 4
- World Health Organization. Spinal cord injury, Aug. 2019. URL <https://www.who.int/news-room/fact-sheets/detail/spinal-cord-injury/>. 1
- D. Wu, F. Zhu, and L. Shao. One shot learning gesture recognition from rgb-d images. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 7–12, 2012. ISSN 21607508. doi: 10.1109/CVPRW.2012.6239179. 6
- C. C. Yang and Y. L. Hsu. A review of accelerometry-based wearable motion detectors for physical activity monitoring. *Sensors*, 10(8):7772–7788, 2010. ISSN 14248220. doi: 10.3390/s100807772. 4
- A. J. Young, T. A. Kuiken, and L. J. Hargrove. Analysis of using emg and mechanical sensors to enhance intent recognition in powered lower limb prostheses. *Journal of Neural Engineering*, 11(5), 2014. ISSN 17412552. doi: 10.1088/1741-2560/11/5/056021. 5

- M. Zhu. Where are linear feature extraction methods applicable? *Analysis*, 27(12):1934–1944, 2005. 5
- H. Zou, T. HASTIE, and R. TIBSHIRANI. Sparse Principal Component Analysis Component Analysis. 15(2):265–286, 2012. ISSN 1061-8600. doi: 10.1198/106186006X113430. 5