POLITECNICO DI TORINO

Master of Science in Biomedical Engineering

Master's Degree Thesis

An Optimisation Algorithm for enhancing precision in stride segmentation using Multi-Dimensional Subsequence Dynamic Time Warping on sensor data

Supervisors:
Prof. Marco Knaflitz
Dr. Felipe Gomez
Dr. Pierre Cherelle

Candidate:
Andrea Crobu

July 2021

An Optimisation Algorithm for enhancing precision in stride segmentation using Multi-Dimensional Subsequence Dynamic Time Warping on sensor data

Politecnico di Torino

Master's Degree in Biomedical Engineering

Department of Mechanical and Aerospace Engineering

Axiles Bionics

Vrije Universiteit Brussel partner

Brussels, Belgium

Supervisors:

Prof. Marco Knaflitz

Politecnico di Torino

Torino, Italia

Dr. Felipe Gomez

Dr. Pierre Cherelle

Axiles Bionics - Vrije Universiteit Brussel

Brussels, Belgium

Candidate:

Andrea Crobu

Abstract

Gait analysis is the systemic study of human locomotion and plays an important role in detecting patterns in such activity.
Traditionally, gait analysis is carried out using video recording techniques, wherein the recording is reviewed in slow motion to permit an accurate assessment of the gait cycle fulfilled by a skilled clinician [1]; other optical techniques, such as Optoelectronic Systems, are also largely used in gait analysis to evaluate the motor performance of healthy subjects and patients [2]. The limitations of optical methods were partially overcome with the use of wearable Inertial Measurement Units (IMUs). Due to their very low consumption, these sensors can be battery powered and are promising tools for long-term ambulatory monitoring outside clinical facilities or laboratories; moreover, considering their low cost, inertial sensors have become particularly popular in the gait analysis research and development field.
Gait segmentation, and particularly stride segmentation, answers the specific clinical needs for analysing human gait. Stride segmentation is the procedure of dividing the gait into strides, where a stride begins with one 'heel strike' (i.e when the heel makes contact with the ground) and ends with the heel strike of the following step of the same foot. The ability to automatically and robustly segment individual strides from gait sequences derived using inertial sensors during different gait activities is crucial for the estimation of gait parameters and for the creation of a reliable gait dataset, without requiring the manual segmentation of recordings. When considering stride segmentation done with Inertial Measurement Sensors (i.e IMUs), multiple techniques and algorithms have been proposed: several algorithms for the analysis of individual strides from accelerometer-derived data are based on peak detection methods; other studies have used clearly defined signal characteristics like zero crossings in gyroscope and accelerometer data to determine gait events [3][4][5]. Machine Learning methods such as Hidden Markov Models (HMMs) [6][7][8][9], Support Vector Machines [10][11] and Decision Trees [12] have also been successfully used. In addition to these methods, template-based methods such as Dynamic Time-Warping algorithms have also shown promising performance due to their ability to identify multiple strides in a sequence, even though they might differ in length and amplitude.
In this work, a new algorithm for segmenting strides given raw data built upon the Multi-dimensional Subsequence Dynamic Time-Warping (msDTW) Algorithm proposed by *Barth et al.* [13] is presented, extending and extensively explaining the functionality and the procedures that need to be followed in order to implement it. With the goal of enhancing the performance of the msDTW, an Optimisation Procedure which improves the precision of the stride segmentation and reduces the computational time of execution of

the segmentation is proposed. The dataset used in this work is provided by *Luo et al.* [14], from which data from two IMU sensors (MTw Awinda,58 Xsens, Enschede, Netherlands) placed on the right shank and on the right thigh are considered, representing subjects *(1)* walking on a planar surface, *(2)* walking on a positive tilt, *(3)* walking on a negative tilt, *(4)* ascending stairs and *(5)* descending stairs. All these recorded data are associated with the category of *Free Walking*, i.e. uncontrolled environment. The performance of a Peak Detection Algorithm [15], the msDTW algorithm proposed by *Barth et al.* and the proposed msDTW Optimised Algorithm is compared in terms of *Accuracy*, *Recall*, *Precision* and *F1-Score*; moreover, a comparison of the msDTW Optimised method with *Barth et al.*'s msDTW is provided in terms of time of execution.
With the use of the proposed msDTW Optimised method it was possible to identify the best *Sensor Set* for each activity considered and achieve a *Precision* of 99.05% for the activity *Stair Ascent Walking*, an *Accuracy* of 98.26% for the activity *Downhill Walking* and a reduction of the computation time of up to 37.97%, when compared to *Barth et al.*'s msDTW. Finally, a demonstration that the proposed algorithm is a robust and reliable alternative method for the construction of a gait dataset which requires no human involvement is provided. Thanks to its high *Accuracy* and *Precision*, one can argue that the proposed method is suitable for clinically relevant applications and could be adapted to different gait activities and scenarios.
The proposed algorithm was concretely used for the automatic annotation of gait signals collected using inertial sensors during experimental data gathering at the partner company *Axiles Bionics*, resulting in the creation of a reliable and consistent gait dataset that is currently being used for the creation of Artificial Intelligence models capable of learning gait patterns from data.

The work presented in this thesis has been reviewed and approved by Dr. Felipe Gomez and Dr. Pierre Cherelle, respectively CTO and CEO of the partner company *Axiles Bionics*. Moreover, the main content of this work has been submitted to the journal *Sensors* (Manuscript ID: sensors-1276510) as an academic paper, thanks to fundings received by Innoviris - Brussels Region and the Flemish Government (AI Research Program) and reviewed by professors Efthymiadis K., De Pauw K., Steckelmacher D., Vanderborght B. and Nowé A.

Contents

List of Figures

List of Tables

Acronyms

IMU Inertial Measurement Unit

DTW Dynamic Time-Warping

msDTW Multi-dimensional Subsequence Dynamic Time-Warping

CTO Chief Technology Officer

CEO Chief Executive Officer

AI Artificial Intelligence

ROM Range Of Motion

HS Heel Strike

FF Foot Flat

HO Heel Off

TO Toe Off

NWS Non-Wearable Sensor

WS Wearable Sensor

IP Image Processing

FS Floor Sensor

LRS Laser Range Scanner

TOF time-of-flight

GRF Ground Force Reaction

CNN Convolutional Neural Network

SVM Support Vector Machine

HMM Hidden Markov Model

PSD Power Spectral Density

WSS Wide-Sense Stationary

IIR Infinite Impulse Response

GUI Graphical User Interface

ACF Accumulated Cost Function

TP True Positive

FP False Positive

TN True Negative

FN False Negative

Chapter 1

Fundamentals of human gait and gait analysis

1.1 Biomechanics of the ankle

The ankle complex is comprised of the lower leg and foot and forms the kinetic linkage that allows the lower limb to interact with the ground and allows dorsiflexion and plantar flexion of the foot, as well as some degree of pronation and supination. The ankle complex also acts as a shock absorber as the heel strikes the ground during the first phases of gait.
The ankle complex is made up of the twenty-six individual bones of the foot, together with the long bones of the lower limb, giving a total of thirty-one joints: this allows the ankle complex to play a fundamental role in many tasks, such as weight bearing and locomotion, and is responsible for complex movement functions such as gait adaption.
Among these joints, the ankle joint, also known as the talocrural joint (Figure 1.1[1]), forms the junction between the distal tibia, the fibula of the lower leg, and the talus. Thanks to its geometry, the ankle joint allows movement such as dorsiflexion and plantaflexion of the foot and contributes to the stability of the ankle complex. Moreover, it limits its freedom of rotation.

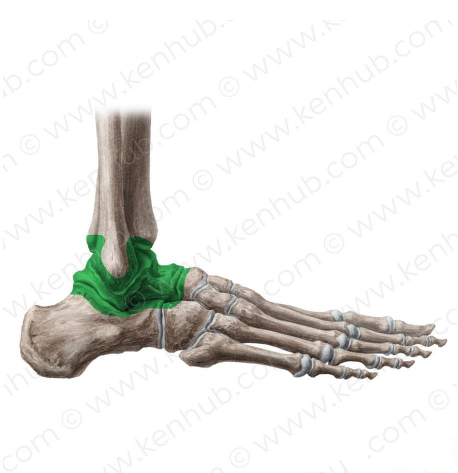


Figure 1.1: Talocrural joint.

[1] https://www.kenhub.com/en/library/anatomy/transverse-tarsal-joint

The subtalar joint, also known as talocalcaneal joint (Figure 1.2 [2]), consists of the anterior and posterior articulations between the talus and calcaneus. Most inversion and eversion of the foot is provided by this articulation.

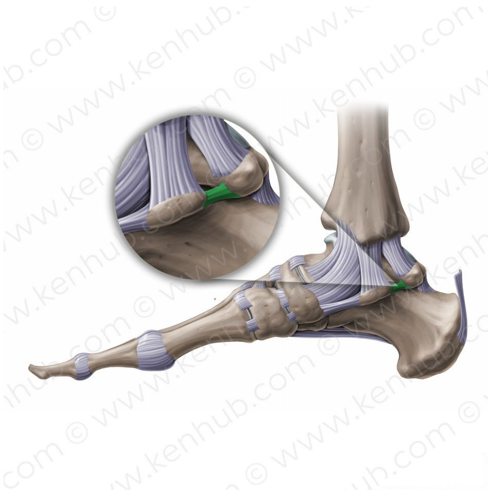


Figure 1.2: Talocalcaneal joint.

Given the articulated structure of the ankle complex, several movements are permitted (Figure 1.3) over the planes of motion (Figure 1.4):

- Plantarflexion and Dorsiflexion, occurring in the sagittal plane
- Abduction (lateral rotation) and Adduction (medial rotation), occurring in the transverse plane
- Inversion and Eversion, occurring in the frontal plane

Combinations of these motions create three-dimensional motions called Pronation and Supination (Figure 1.3): both terms define the position of the plantar surface of the foot (sole). During supination, a combination of plantarflexion, inversion, and adduction causes the sole to face medially. In pronation, dorsiflexion, eversion, and abduction act to position the sole facing laterally [16].

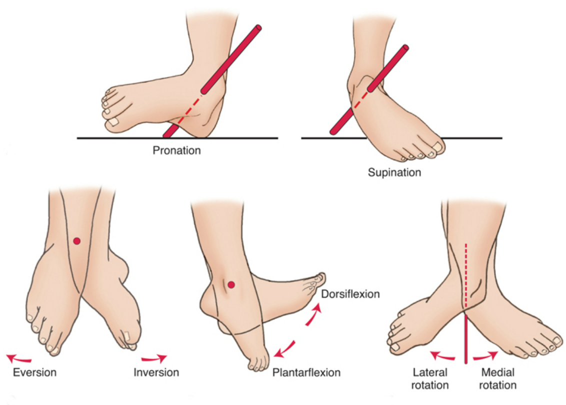


Figure 1.3: Movements of the ankle.

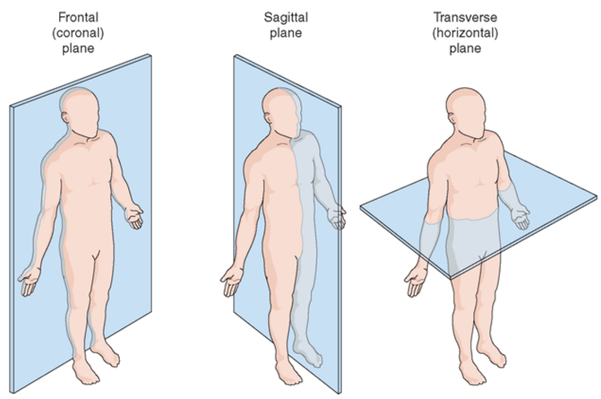


Figure 1.4: Planes of motion.

[2] https://www.kenhub.com/en/library/anatomy/transverse-tarsal-joint

More specifically, the rotation in the saggital plane occurs along the axis passing through the lateral melleoli and the medial malleoli (dotted line, Figure 1.5); the coronal plane axis of rotation occurs around the intersecting point between the malleoli and the long axis of the tibia in the frontal plane (Figure 1.5).

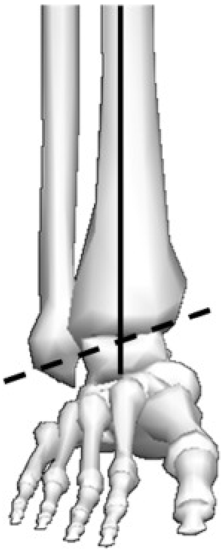


Figure 1.5: Sagittal and frontal plane axis of rotation of the ankle joint complex.

The transverse plane axis of rotation occurs around the long axis of the tibia intersecting the midline of the foot (Figure 1.6)[16]. Studies [17] have shown that this axis may vary as motion changes.

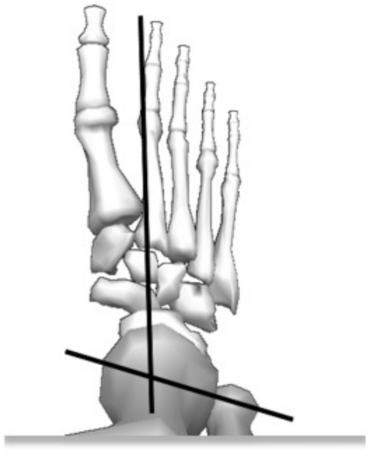


Figure 1.6: Sagittal and frontal plane axis of rotation of the ankle joint complex.

The range of motion (ROM) of the ankle complex has been shown to vary significantly between individuals due to geographical and cultural differences in daily activities [18]. On average, in the saggital plane the range of dorsiflexion is between 10 and 20 degrees, while the normal range of plantarflexion is between 40 and 55 degrees. Considering the frontal plane, the avarage range of motion for inversion is around 23 degrees and for eversion is around 12 degrees.
To support all these movements, the ankle complex is made up of twelve extrinsic muscles contained within four compartments (Figure 1.7). The anterior compartment produces

dorsiflexion, inversion and eversion of the foot. The lateral compartment produces plantarflexion and eversion of the foot. The posterior compartment contributes to plantarflexion of the foot. The deep posterior compartment produces plantarflexion and inversion of the foot.

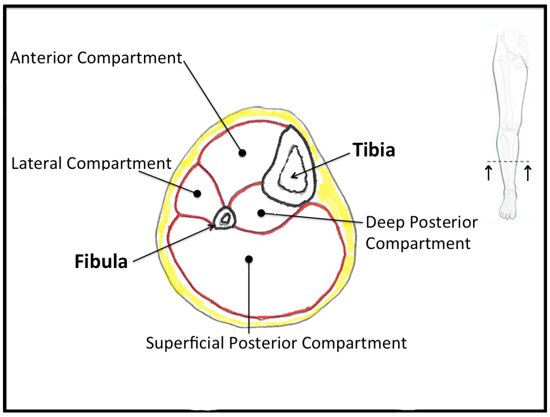


Figure 1.7: Muscles compartments.

1.2 Importance of gait analysis

Gait reveals key information about a person's quality of life. Accurate and reliable knowledge of gait characteristics at a given time and — even more importantly — monitoring and evaluating them over time would enable early diagnosis of diseases and their complications and would help to identify the most appropriate treatment.
Due to the complex structure of the human musculoskeletal system, gait analysis becomes the preliminary requirement in understanding the complex dynamics of its locomotion strategy. Research on gait analysis promises new horizons for clinical and pathological gait diagnosis, such as monitoring sports and athletic performances, observation of training and rehabilitation exercises, and designing anthropomorphic gaits, exoskeletal systems, and prosthetic limbs [19].
Technological advances have given rise to devices and techniques which allow an objective evaluation of different gait parameters, resulting in more efficient measurement and providing specialists with a large amount of reliable information on patients' gaits [20].
Abnormal gait is the main cause for many physical problems, such as back pain, joint pain in the lower limbs, muscle strain, etc. [21], so the development of robust and trustworthy methods for analysing gait are fundamental. On the other hand, gait abnormalities can occur for a variety of reasons; such as a biomechanical problem, injury, stroke, fracture, neurological disorder, and so on [22]; and thus require tools capable of supporting clinicians in the diagnosis and treatment process.
In the particular case of amputees, the prosthetic ankle has a reduced range of movement compared to the anatomical ankle. This results in prolonged heel strike and weight bearing through the heel before flat foot contact, with delayed forefoot loading. Due to the reduced ankle movement of the prosthesis, the range of extension at the hip is reduced to approximately half of that of the opposite limb. The stance time on the non-prosthetic side is also increased compared to the prosthetic side [23]. Therefore, there are deviations which an amputee will adopt to compensate for the prosthesis, leading to muscle weakness or tightening, lack of balance and fear. Such deviations create an altered gait pattern and it is important that they are recognised, as rehabilitation of the gait will need to

encompass corrections of these deviations [24][25][26].
Through the determination of the gait cycle and gait phases, and by validating gait events, it is possible to obtain a series of parameters related to gait, and thereby permit the classification and the analysis of different gait characteristics.

1.3 Segmentation of gait cycle

Gait can be defined as a series of rhythmic, alternating movements of the lower limbs which results in the forward progression of the body [1]. A gait cycle starts with a so-called heel strike (HS), the moment in which the heel of one leg (hereafter referred to as the reference leg) touches the ground, and it ends at the next heel strike of the same leg. A single gait cycle is denominated as a 'stride'.
The gait activity is a cyclical repetition of strides, each composed of two main phases:

- Stance phase: the phase in which both feet are in contact with the ground. It begins with a heel strike and ends when the toe of the reference leg loses contact with the ground. This latter event is referred to as toe off (TO). This phase makes up approximately 62% of the gait cycle, with an average duration of approximately 0.59 to 0.67 seconds.
- Swing phase: the single-support phase of the gait cycle. This phase begins in correspondence with the toe off of the reference leg and terminates with a new heel strike. During this phase all body weight is borne by a single leg, while the reference leg oscillates over the ground towards a forward position. This phase takes approximately 38% of the gait cycle with an average duration of approximately 0.38 to 0.42 seconds.

Figure 1.8 summarises the phases thus far described and introduces a further division of a stride.

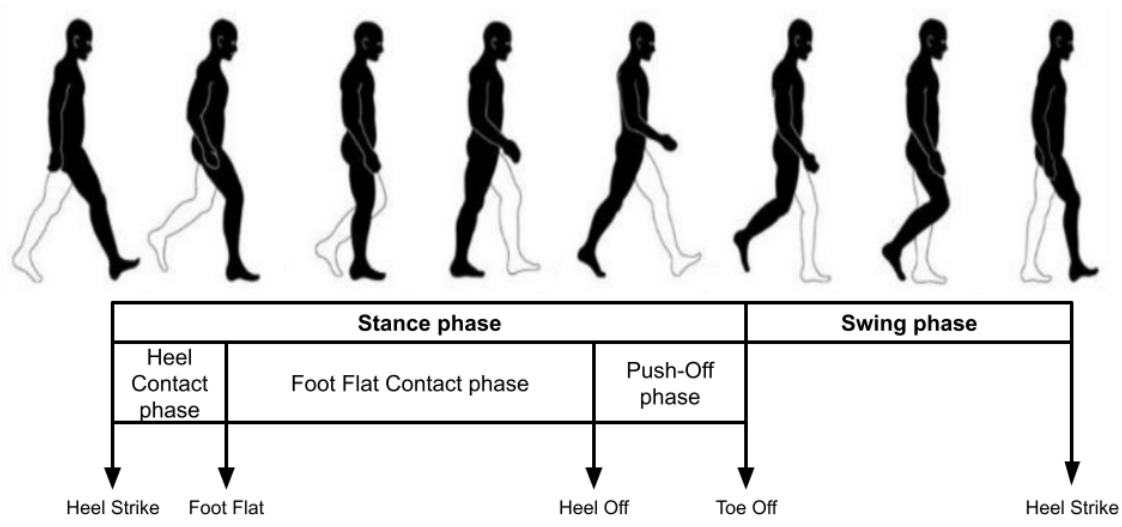


Figure 1.8: Division of gait cycle. A stride can be divided into two main phases: stance phase and swing phase. Furthermore, the stance phase can be additionally divided into: 'heel contact' phase, 'foot flat contact' phase and 'push-off' phase. The limits of these phases are defined by four events identified as heel strike (HS), foot flat (FF), heel off (HO) and toe off (TO).

In figure 1.8 one can to observe the further division of the stance phase:

- Heel contact phase: the phase in which the reference leg's heel rolls over the ground but the toe has yet to make contact with the surface. This phase begins with a heel strike and ends when the foot is completely in contact with the ground, i.e. the foot flat event.
- Foot flat contact phase: the phase in which the foot is completely in contact with the ground and the upper body loads body weight on to the reference leg, while the other leg is swinging forwards. This phase begins when the foot is completely in contact with the ground (foot flat) and ends when the heel of the reference leg leaves the surface.
- Push-off phase: the phase in which the reference leg prepares for the swing phase. It begins as soon as the heel of the reference leg starts to move away from ground (heel off event) but the toe remains in contact with the surface and ends when the toe also loses contact with the ground (toe off).

Various techniques have been employed with the purpose of studying human gait through stride segmentation. Among them, devices used to study the human gait can be classified according to two different approaches: those based on non-wearable sensors (NWS) and those based on wearable sensors (WS). The former require control of the environment in which the sensors are located and only allows the capture of gait data while the subject walks in predefined scenarios. In contrast, wearable sensor systems allow data analysis outside of laboratory settings and permit the capture of information concerning human gait during a person's everyday activities[20]. NWS systems can be further divided into two subgroups: (1) those based on image processing (IP); and (2) those based on floor sensors (FS). IP systems capture data on a subject's gait through one or more optic sensors and take objective measurements of the different parameters through digital image processing. Analogue or digital cameras are the mostly commonly used devices. Other types of optic sensors such as laser range scanners (LRS), infrared sensors and time-of-flight (ToF) cameras are also used. Optical techniques may or may not employ the use of markers (marker-based and marker-less methods, respectively)[20].

Figure 1.9: Example of optical system. Optoelectronic motion capture systems based on markers. This allows the precise study of a subject's motion using markers placed in strategic spots and the recording of this motion through a multi-camera system. These devices are often used in combination with force platforms and/or inertial sensors.

On the other hand, FS systems are based on sensors located along the floor on so-called force platforms where gait information is measured through pressure sensors and ground reaction force sensors (GRF) which measure the force exerted by the subject's feet on the floor when they walk.

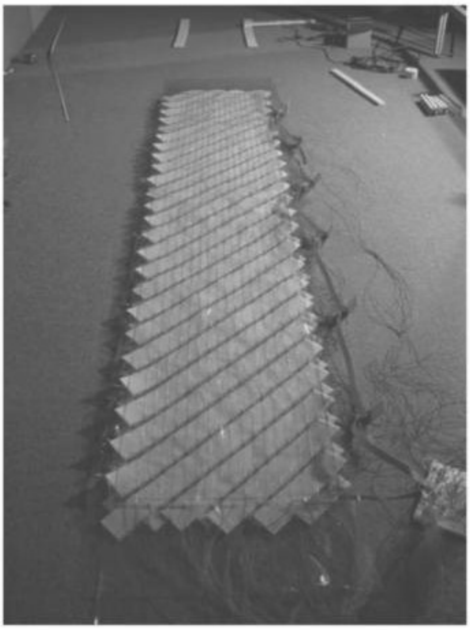


Figure 1.10: Example of a force platform. Measuring instruments that perform ground reaction force measurements and can be used to quantify pressure patterns under the subject's foot.

WS systems use sensors located on several parts of the body, such as the feet, knees, thighs, or waist. Different types of sensors are used to capture the various signals that characterise human gait. These include accelerometers, gyroscopic sensors, magnetometers, force sensors, extensometers, goniometers, active markers, and electromyography [20].
In this work, Inertial Measurement Units (IMUs) are used.

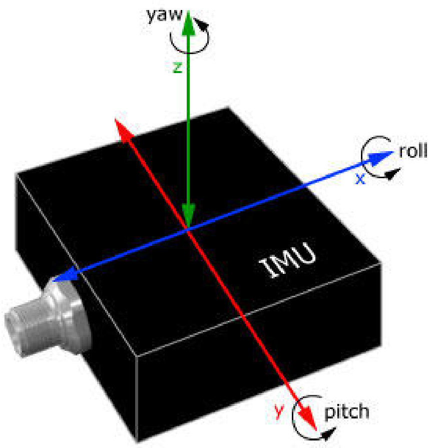


Figure 1.11: Example of a wearable sensor. Capable of sensing accelerations, angular speeds and orientation in the earth's magnetic field, IMUs are widely used in gait analysis for studying subjects' kinematics.

1.3.1 Stride segmentation

Gait analysis is the systemic study of human locomotion and plays an important role in detecting patterns within such activity. Stride segmentation is the procedure of dividing the gait into strides, where one stride is defined between one heel strike and the heel strike of the following step, as previously explained. The ability to automatically and robustly segment individual strides from gait sequences derived from inertial sensors while performing different gait activities is crucial for the estimation of gait parameters and for the creation of a reliable gait dataset, without requiring the manual segmentation of recordings.
In the past few years, different techniques have been proposed to achieve stride segmentation. In the following section, an overview of these techniques is presented, with a focus on the detection of heel strikes in IMUs' recorded signals. Among them, zero-crossing techniques based on the analysis of gyroscope signals located on the shank have been proposed by several authors [27][28][29], as well as techniques based on the detection of peaks in acceleration and/or gyroscope signals [15][30][31]: these techniques are simple but have shown respectable performances in terms of detection of heel strikes. However, they perform relatively poorly in uncontrolled environments, such as during free walking and have shown poor generalisation capabilities. The majority of these algorithms are structured as a set of rules applied to the values of the IMU signals taken into consideration (accelerations and angular speeds) for the detection of the Initial Contact (i.e. heel strike): an example is given by *Salarian et al.* [15] and *Gouwanda et al.* [30], who in their work have implemented an algorithm for the detection of heel strikes based on the research of minimum values in gyroscope signals. Salarian's work has been used as a baseline in this work and his algorithm is presented in section 3.1.
The limitations of the techniques presented thus far are partially overcame by techniques which use template recognition, such as Dynamic Time Warping [13][32]: these algorithms perform better in terms of generalisation and correct detection in different real-life environments. These techniques use information from a pre-constructed example of a stride to search for that pattern in the recorded signals. In this work the algorithm proposed by *Barth et al.* [13] is comprehensively analysed (3.2) and extended through an optimisation algorithm (3.4). In addition to this, many Wavelet Transform based algorithms [33][34] have also been proposed for stride segmentation.
In recent years, Artificial Intelligence methods such as Machine Learning and Deep Learning techniques have become increasingly popular. Among those methods, *Zhao et al.* [35] proposed a gait recognition algorithm based on a seven-layer convolutional neural network (CNN), *Gurchiek et al.* [36] work aimed to detect asymmetric gait patterns in patients recovering from anterior cruciate ligament reconstruction with the use of a Support Vector Machine (SVM). In addition, *Mannini et al.* [6][7] and *Panahandeh et al.* [37] proposed an alternative solution based on a Hidden Markov Model (HMM), with the advantage of having the possibility of evaluating the uncertainty of the prediction, but it is limited by the assumption that IMU data is gaussian distributed.

In this work, a Dynamic Time Warping algorithm is developed starting from the method proposed by *Barth et al.* [13] with the aim of performing precise stride segmentation on the data collected by IMU sensors, placed on points of interest such as the right shank and right thigh, with the final goal of building a robust and reliable method for the automatic construction of a gait dataset of segmented signals.

Chapter 2

Materials and methods

2.1 Subject protocol

This work makes use of the public dataset[1] published in Nature by *Luo et al.* [14]. This dataset contains data recorded while participants performed several walking trials over different surfaces wearing IMU sensors.

Data was gathered analysing the gaits of 30 healthy subjects (15 male and 15 female) with no pre-existing pathology while *(1)* walking on a planar surface, *(2)* walking on a positive and *(3)* negative tilt, *(4)* ascending stairs and *(5)* descending stairs.
Table 2.1 shows the anthropometric characteristics of the participants, measured at the time of the test.

	Mean ± STD
Age [years]	23.5 ±4.2
Height [cm]	169.3 ±21.5
Body mass [kg]	70.9 ±13.9

Table 2.1: Anthropometric characteristics of the subjects.

No restrictions on the shoes were imposed in order to enlarge the range of applications.

[1] https://www.nature.com/articles/s41597-020-0563-y

In this work, the following activities are considered:

- *Walking*: Walking on an horizontal, paved surface
- *Stair Ascent*: Ascending concrete stairs
- *Stair Descent*: Descending concrete stairs
- *Uphill Walking*: Walking on a upward sloping concrete surface
- *Downhill Walking*: Walking on a downward sloping concrete surface

Figure 2.1: Measurement sites for walking and stair trials.

Figure 2.1 displays the surfaces on which the activities were performed: participants were instructed to walk at their normal pace and to let their arms swing naturally. Surfaces were presented in a randomised order and adequate rest was provided to prevent fatigue between trials. Participants walked six times on each surface.

2.2 Measurement protocol

Participants performed the activities while wearing six IMU sensors (MTw Awinda, Xsens, Enschede, Netherlands) secured to the body using the bands provided by the manufacturer such that they were:

- Centred on the wrist on the dorsal forearm
- Centred on both the anterior thighs
- Centred 5 cm above the bony processes of both ankles
- Posterior at level of L5/S1 joint

Following the most common set-ups used for gait analysis, in this work the IMUs placed on the right shank and on the right thigh were considered (Figure 2.2). This permitted the comparison of the obtained results with the literature.

Figure 2.2: Sensor placement setup. In orange: the IMU sensors positioned on the right shank and on the right thigh of one participant.

The frequency of sampling was 100Hz for all sensors.

In Figure 2.3 it is possible to see the orientation of the reference axis.

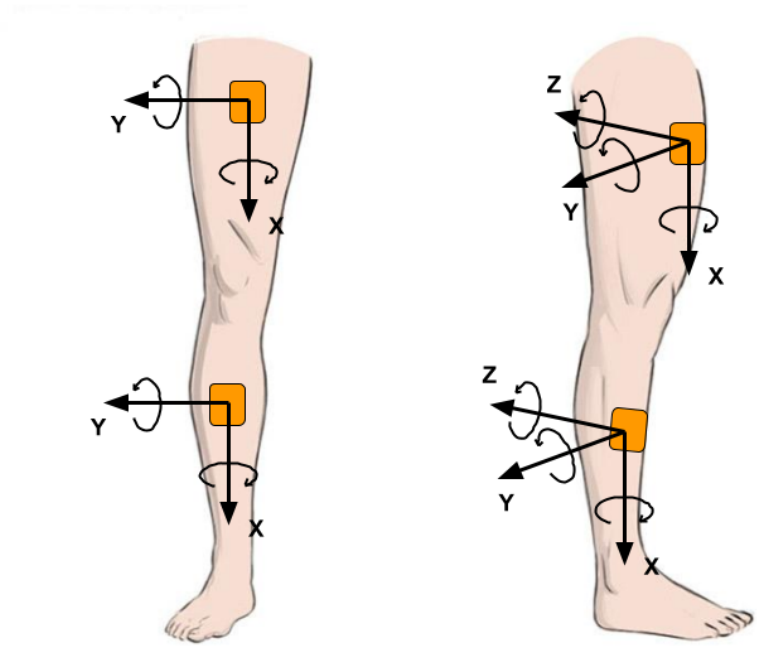


Figure 2.3: Orientations of the axis of the IMUs. The axis X follows the upper-lower direction, the axis Y follows the mediolateral direction and the axis Z follows the anterior-posterior direction.

Using the IMU sensors, accelerations along the three axes (a_x, a_y, a_z) and angular velocities around the three axes (ω_x, ω_y, ω_z) were registered. In addition to these data, combinations of the signals were considered in order to compute the magnitude of the vectors acceleration and angular velocity (2.1).

$$\begin{aligned} magnitude_{acc} &= \sqrt{acc_x^2 + acc_y^2 + acc_z^2} \\ magnitude_{\omega} &= \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2} \end{aligned} \tag{2.1}$$

Accelerations without the gravity component were also considered.

Table 2.2 reports all the signals used in this work.

Signal	Axis	Signal Identification Number
shank acceleration	x	0
shank acceleration	y	1
shank acceleration	z	2
shank acceleration (without gravity)	x	3
shank acceleration (without gravity)	y	4
shank acceleration (without gravity)	z	5
shank acceleration vector	xyz	6
shank acceleration vector (without gravity)	xyz	7
thigh acceleration	x	8
thigh acceleration	y	9
thigh acceleration	z	10
thigh acceleration (without gravity)	x	11
thigh acceleration (without gravity)	y	12
thigh acceleration (without gravity)	z	13
thigh acceleration vector	xyz	14
thigh acceleration vector (without gravity)	xyz	15
shank angular velocity	x	16
shank angular velocity	y	17
shank angular velocity	z	18
shank angular velocity vector	xyz	19
thigh angular velocity	x	20
thigh angular velocity	y	21
thigh angular velocity	z	22
thigh angular velocity vector	xyz	23

Table 2.2: Signals collected.

2.3 Preprocessing of data

Initially, a Power Spectral Density (PSD) estimation was performed so as to better understand the spectral components of the signals: a Welch PSD estimation was used for this purpose. Signals are considered Wide-Sense Stationary processes (WSS) inside windows of 2 seconds and therefore a theoretical resolution in terms of frequency of 500mHz is achieved. A Hamming window of a length of 50 samples was used. The overlap set for the Welch estimation was 50%. The number of points on which representing the PSD was chosen equal to 1024 in order to have an apparent resolution of around 100mHz. The obtained PSD is shown in Figure 2.4.

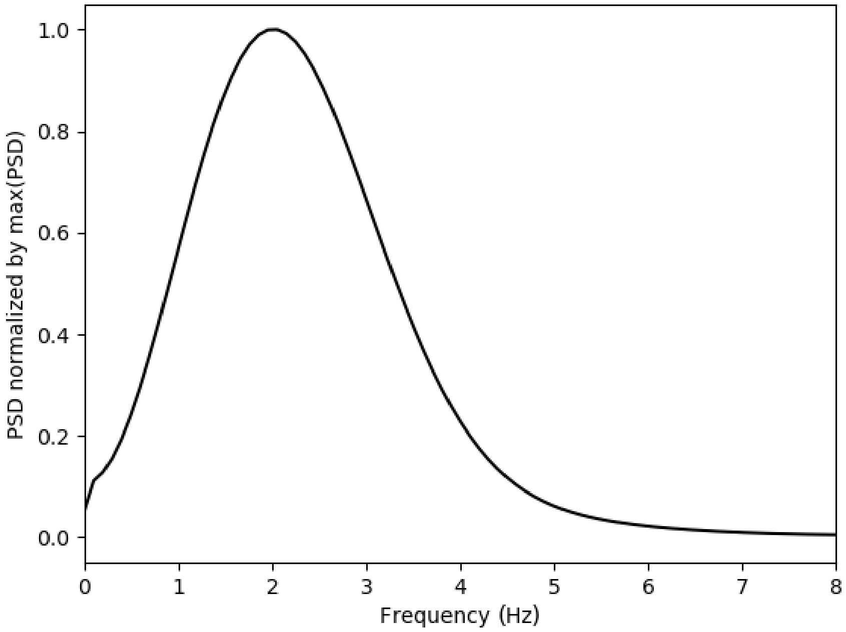


Figure 2.4: Power Spectral Density of the shank gyroscope signal (axis Y).

The results shown here refer to the signal of the gyroscope positioned on the shank (axis Y), but relatively similar results were obtained for all other signals.

Considering the obtained results, an IIR low-pass filter was implemented to filter data. The cut-off frequency chosen was equal to 7Hz and the filter order equal to 10. The response of the filter can be observed in Figure 2.5 and Figure 2.6.

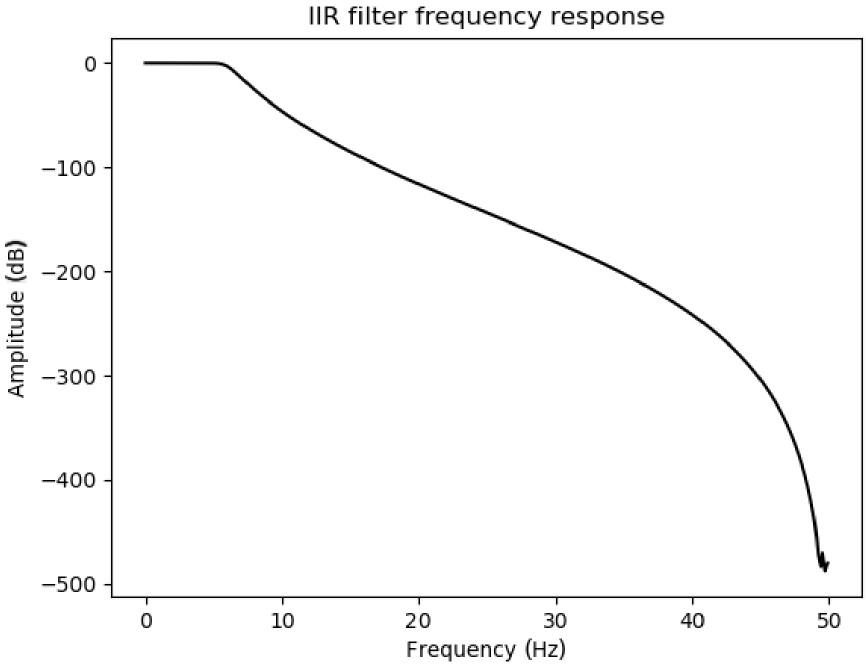


Figure 2.5: Magnitude of the filter: the magnitude of the filter at the cut-off frequency of 7Hz is -3dB, while at 8Hz the magnitude is -16dB.

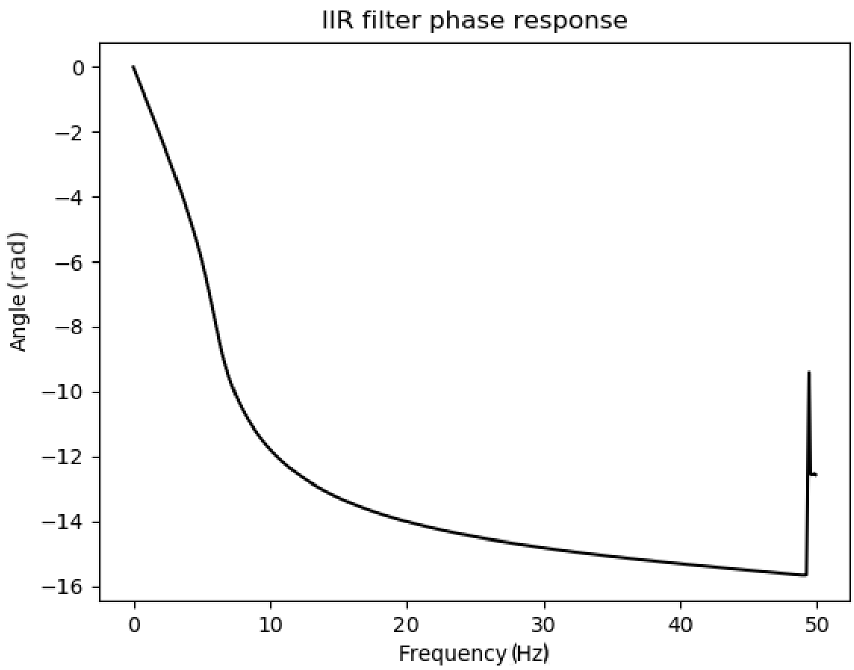


Figure 2.6: Phase response of the filter.

The filter was then applied forward and backward to the signals, so as to avoid distortion stemming from the non-linear phase of the IIR filter.

2.4 Manual labelling of data

In order to compare and evaluate the results of the segmentation algorithms, data were manually labelled by three users using a dedicated Graphical User Interface (GUI) developed for the purpose. The recordings were labelled following the results of several studies related to Gait Segmentation using inertial sensors positioned on the shank or in its proximity [1][15][38][39][40][41][42]. More specifically, the heel strike is believed to be located in correspondence with a minimum in the gyroscope signal, as will be detailed later in section 3.1.
The annotations made by the three users were compared to obtain more accurate and reliable labelling: only the annotations selected by at least two of the three users were considered. Moreover, differences among the annotations were always observed to be lower then $\pm 10ms$: each valid annotation was calculated as the average of the annotations of each individual user.

Chapter 3

Segmentation algorithms

The first algorithm implemented was a Peak Detection-based algorithm (3.1) and serves as a baseline. The second algorithm was based on the technique of the Dynamic Time Warping (3.2) and was implemented in such a way that more then one signal is used for computing the segmentation (3.3). In the third algorithm, a method on top of DTW is proposed to enhance performance, reduce the computational cost and boost the speed of execution (3.4).

In this section data relating to the activity *Walking* are used as examples in order to explain the segmentation procedures, but the same operations were executed on the signals recorded for each activity under consideration.

3.1 Peak Detection Algorithm

The Peak Detection Algorithm implemented for this work is the one proposed by *Salarian et al.* [15], adapted to the data used.
The pipeline of the algorithm is reported in Figure 3.1.

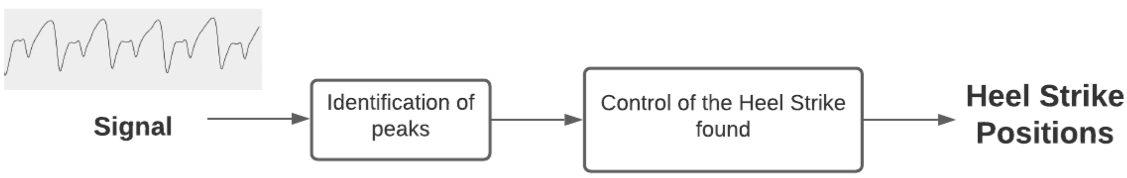


Figure 3.1: Pipeline of the Peak Detection Algorithm.

The first step of the algorithm is the identification of peaks in the signal (shank angular velocity in the medio-later direction) with height higher than 100 $°/s$ and an inter-distance higher than $500ms$ (Figure 3.2).

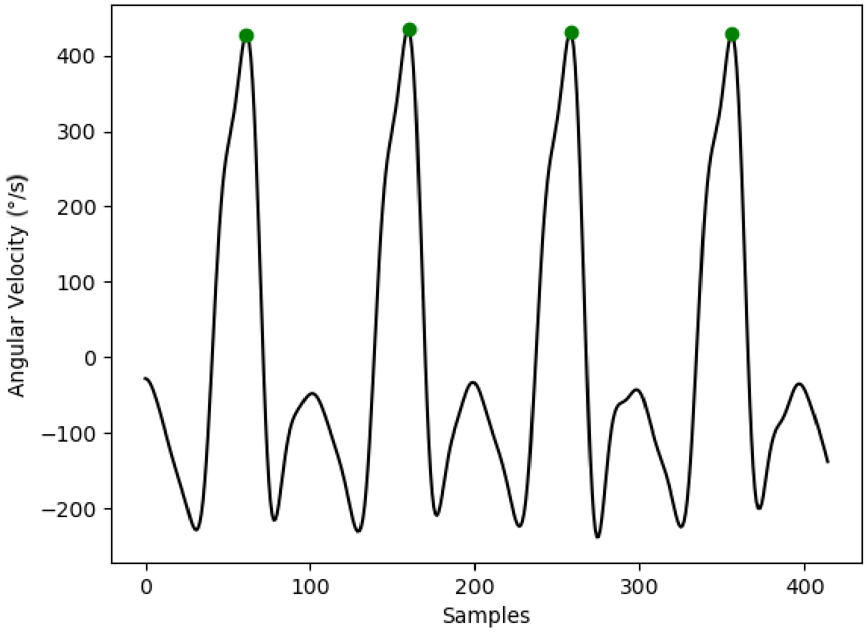


Figure 3.2: Identification of peaks in the shank angular velocity (axis Y) signal. In green: the identified peaks; these peaks correspond to the middle swing phase peaks.

The second step of the algorithm is to identify the heel strike: after each identified peak (green points in Figure 3.2) the first local minimum in the signal within a range of $100ms$ after the identified peak is believed to be the heel strike (red points in Figure 3.3).

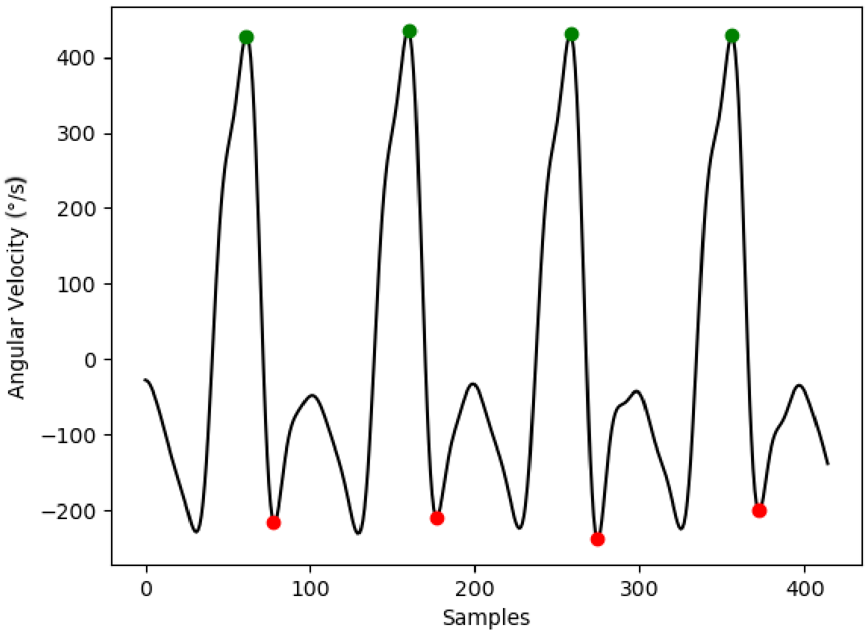


Figure 3.3: Identification of heel strikes in the shank angular velocity (axis Y) signal. In red: the identified heel strikes.

Among the identified heel strikes, only those which presented a negative value in the shank angular velocity (axis Y) signal were accepted [40][43].

3.2 Dynamic Time Warping Algorithm

Dynamic Time Warping (DTW) [13] is a matching algorithm used for computing the similarity between two time series. In the case of gait analysis, and in particular in the case of stride segmentation, this algorithm is widely used to research a given time sequence and identify the stride (hereafter called *Template*) inside a *Target Signal* (Figure 3.4).

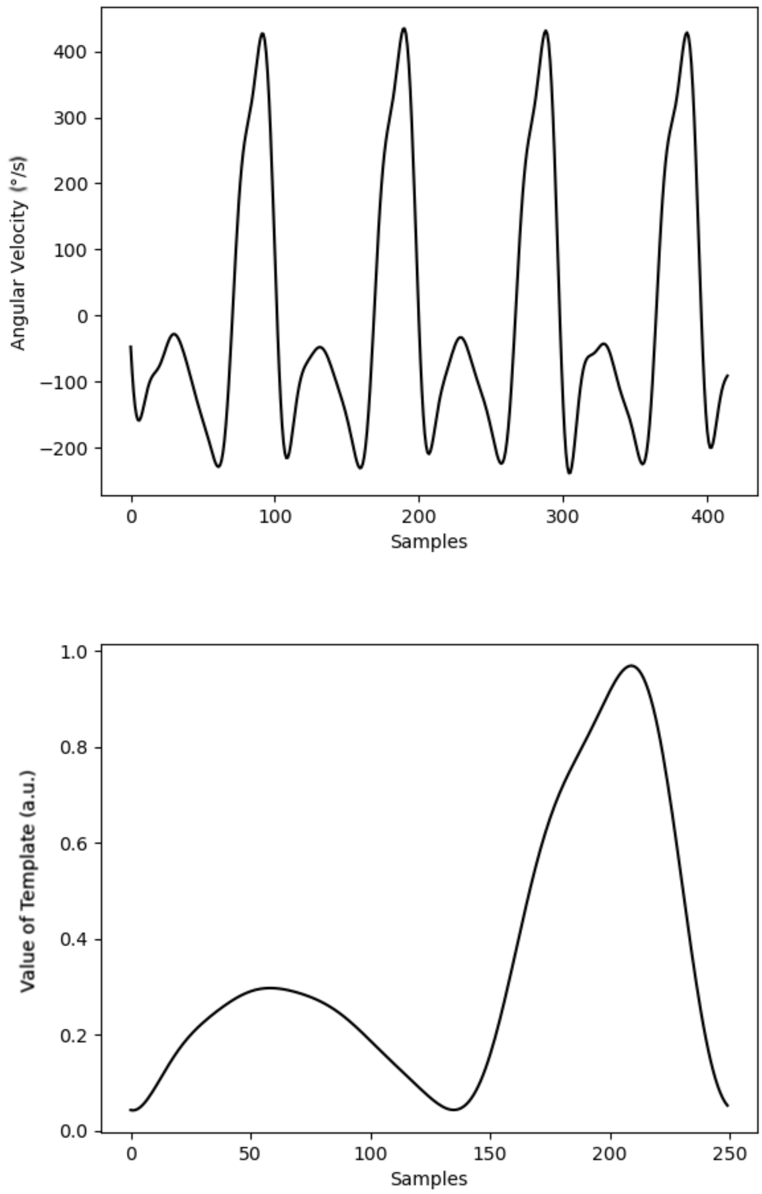


Figure 3.4: Example of a *Target Signal* (top) and its *Template* (bottom) for the activity *Walking*. The signal considered is the shank angular velocity (axis Y). The *Template* is normalised between 0 and 1 and has a length of 250 samples.

DTW is a distance-based algorithm, therefore it allows the identification of patterns with different lengths and also matches signals non-linearly so that sub-parts of the template are stretched or shortened for an optimal fit by warping the *Template* upon the *Target Signal* [13].
The *Template* is defined as a time-dependent sequence $T = \{t_0, .., t_{M-1}\}$ where each sample t_m with $m \in \{0, M-1\}$ consists of data from the inertial sensors.
The *Target Signal* is defined as a time-dependent sequence $S = \{s_0, .., s_N - 1\}$ where each sample s_n with $n \in \{0, N-1\}$ consists of data from the inertial sensors, from which strides are segmented.
A detailed explanation of the *Template* generation is given in *subsection 3.2.1*.

The pipeline of the DTW algorithm is shown in Figure 3.5.

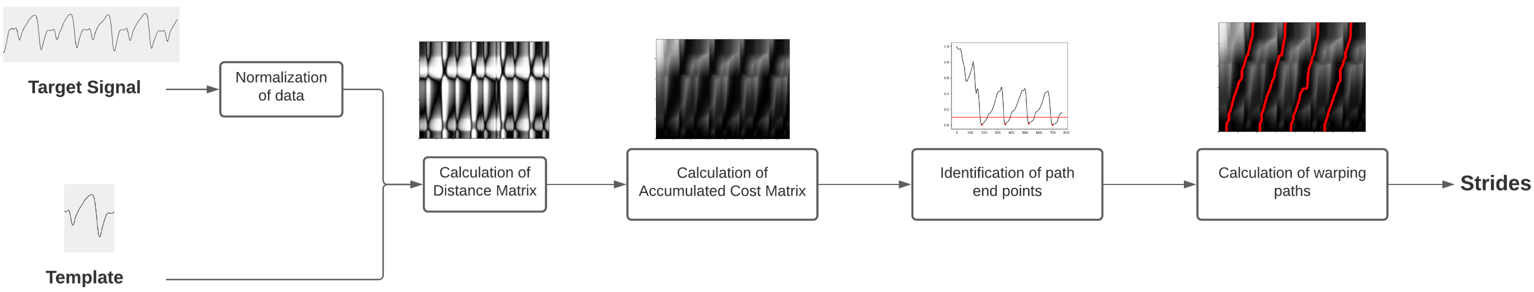


Figure 3.5: Pipeline of the DTW algorithm. The first step of the algorithm is the normalisation of data; the algorithm then computes the matrix of distance D, followed by the computation of the matrix of costs C. The algorithm then identifies the end-points of each stride using a threshold on the Accumulated Cost Function, derived from C. Through a back-tracking procedure on the matrix C, heel strikes are identified in the signal.

As can be seen in Figure 3.5, the first step of the algorithm is the normalisation of data: in this work a *min-max scaling* technique was used (3.1).

$$s_n = \frac{s_n - min(S)}{max(S) - min(S)}, \ n \in \{0, N-1\} \tag{3.1}$$

The algorithm then calculates the pairwise distance between each sample of the *Template* and each sample of the *Target Signal*: in this work the Euclidean distance was used (3.2).

$$D(m, n) = \sqrt{(t_m - s_n)^2} \ \forall \ m \in \{0, \ldots, M-1\}, n \in \{0, \ldots, N-1\} \tag{3.2}$$

The result of this operation is the matrix of distance (D) (Figure 3.6).

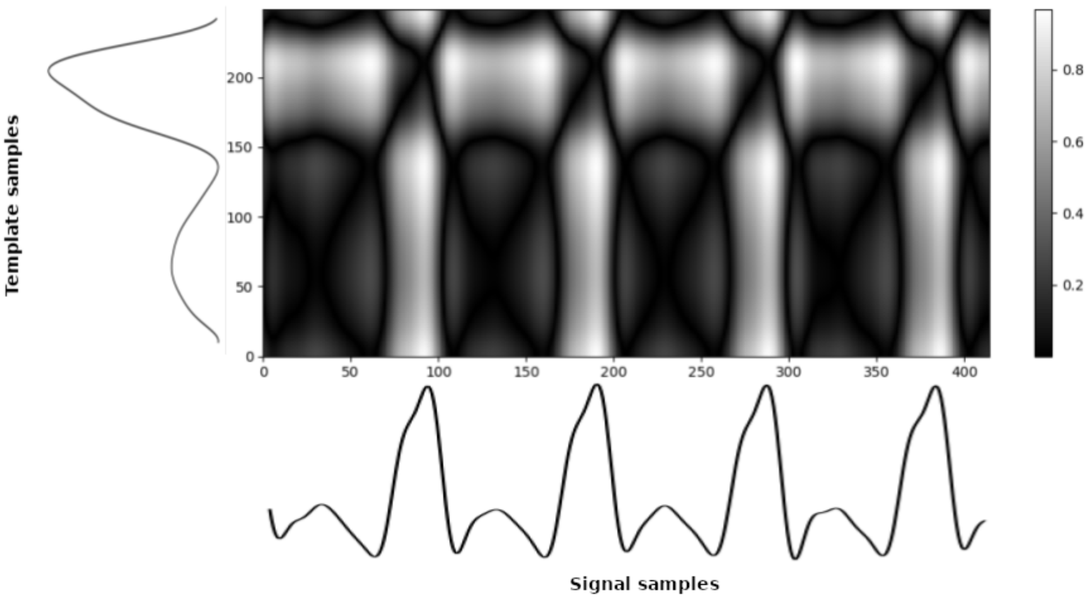


Figure 3.6: Example of a matrix of distance. A darker colour in the image means a small distance between one sample in the *Target Signal* and one other sample in the *Template*. The matrix D shown in the figure is the one obtained by calculating the pairwise distance of each sample of the shank angular velocity (axis Y) signal, and its *Template*.

Assuming that a small distance between the *Template* and the *Target Signal* is an index of a good match in that point, it is possible to identify diagonal paths (hereafter referred as *Warping Paths*) along the image that are associated to a good match between the *Template* and the *Target Signal*. With the purpose of emphasizing those *Warping Paths*, a *min-max scaling* normalisation over rows and columns of D was performed: in other words, the difference between low-level pixels and high-level pixels is highlighted, i.e the contrast of the image is enhanced (Figure 3.7).

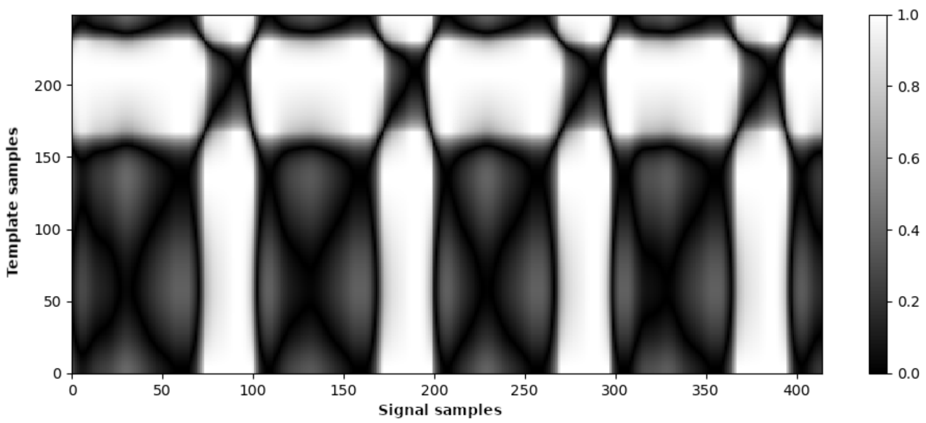


Figure 3.7: Example of a matrix D normalised. The matrix D shown is obtained by taking the pairwise distance of each sample of the shank angular velocity (axis Y) signal, and its *Template* and then normalising the image. A dark colour represents a small value of distance between a certain point of the *Target Signal* and another point of the *Template*; on the other hand, a lighter colour indicates a larger distance.

In order to better identify those *Warping Paths*, a new matrix with the same size as D, defined as accumulated cost matrix C, was computed: C represents not only the distance between a *Template* and a *Target Signal*, but also the accumulated costs of warping the *Template* to parts of a *Target Signal* [13].

The first row of C is equal to the first row of D:

$$C(0, n) = D(0, n) \ \forall \ n \in \{0, \dots, N-1\} \tag{3.3}$$

The first column of C is defined as:

$$C(m,0) = \sum_{i=0}^{m} D(i,0) \ \forall \ m \in \{0, \dots, M-1\} \tag{3.4}$$

All the other elements of C are defined as:

$$\begin{aligned} C(m, n) = min\{C(m-1, n-1), C(m-1, n), C(m, n-1)\} + D(m, n) \\ \forall \ m \in \{1, \dots, M-1\}, n \in \{0, \dots, N-1\} \end{aligned} \tag{3.5}$$

For the same reason as before, a *min-max scaling* normalisation over rows and columns of C was performed.
An example of an accumulated cost matrix C is shown in Figure 3.8.

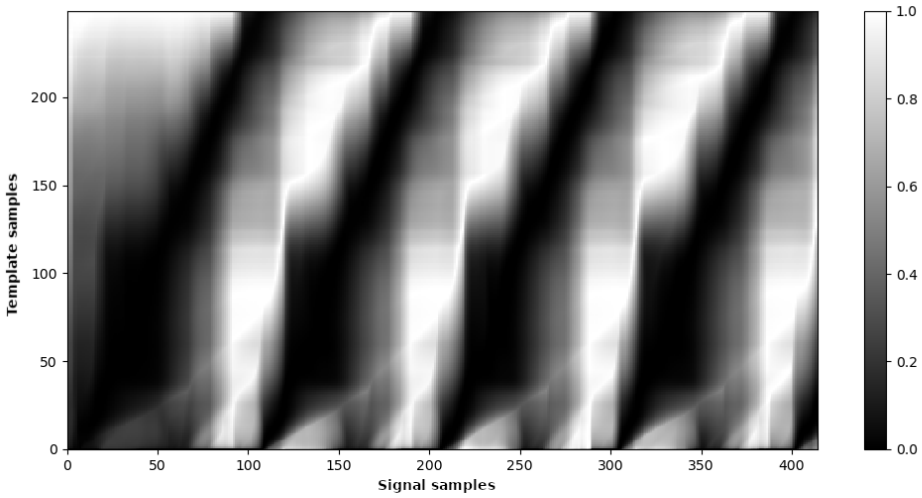


Figure 3.8: Example of an accumulated cost matrix C. The matrix C shown in figure is obtained from matrix D of the shank angular velocity (axis Y) signal. It is possible to notice the effects that the operation has on diagonal paths, which are now highlighted.

As a result of this operation, the top row of C represents the cost accumulated during the warping procedure and is thus called the Accumulated Cost Function (ACF) (Figure 3.9)(3.6).

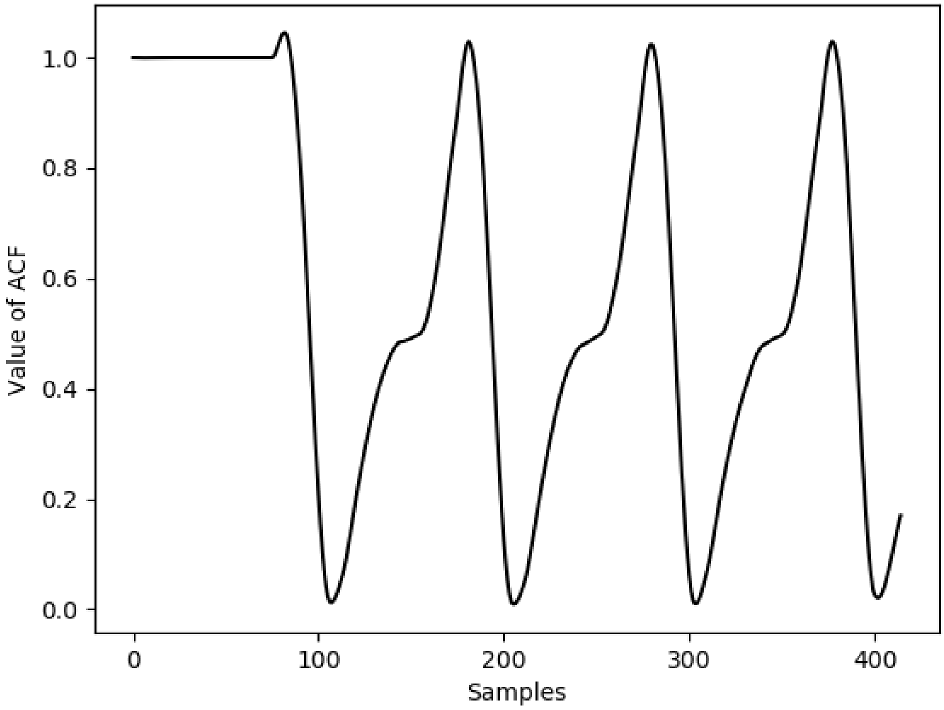


Figure 3.9: Example of an Accumulated Cost Function. The ACF shown is the one referred to in the top row of matrix C obtained for the shank angular velocity (axis Y) signal.

Troughs in the ACF identify the end points of the warping paths. The last row of C also identifies the stride end. The criteria used to identify the end points (p_{end}) is the threshold τ (Figure 3.10) (3.7).

$$ACF = C(M-1, n) \; \forall \; n \in \{0, \ldots, N-1\} \tag{3.6}$$

$$p_{end} = \{p_{end_i} \in \{0, \ldots, N-1\} \mid ACF[p_{end_i}] \leq \tau\}, \; i \in \{0, \ldots, K\} \tag{3.7}$$

Where K is the number of identified strides.

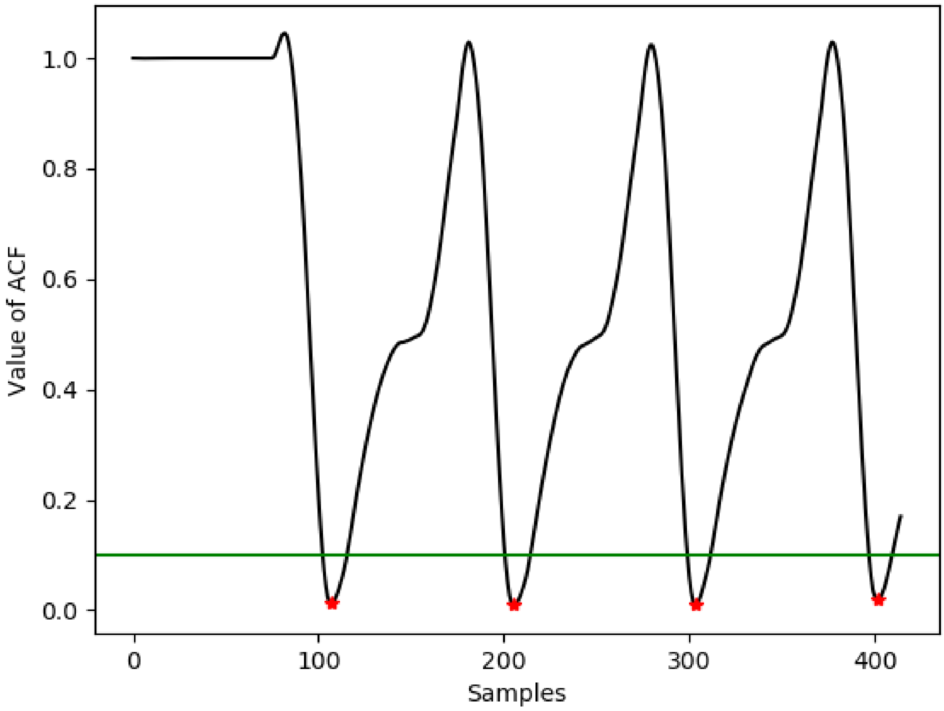


Figure 3.10: Example of an Accumulated Cost Function. The ACF shown is the one referred to in the top row of matrix C obtained for the shank angular velocity (axis Y) signal. In green: the threshold τ. In red: the end points p_{end} identified.

Starting from each p_{end_i} it is possible to obtain the starting point of the path by backtracking on matrix C. A path P_i is defined as follows:

$$\begin{aligned} P_i = \{p_j \in C \mid p_j = min\{C(m-1,n), C(m-1,n-1), C(m,n-1)\}\} \\ with\ p_j = (m,n), m \in \{0,\dots,M-1\}, n \in \{0,\dots,N-1\}\ for\ j = \{0,\dots,L-1\} \end{aligned} \tag{3.8}$$

Where p_j indicates each point (m,n) composing a path P_i and L indicates the length of the path P_i in samples. Given its nature, the DTW algorithm allows the identification of paths with different lengths L. As observed in 3.8, each warping path is defined between a p_{end_i} and a p_{start_i}, where p_{start_i} identifies the point p_j where the stride starts. This means that each path is forced to start in the first row of C and end in the last row of C, as well as to follow a diagonal path.

Figure 3.11 shows an example of this warping procedure.

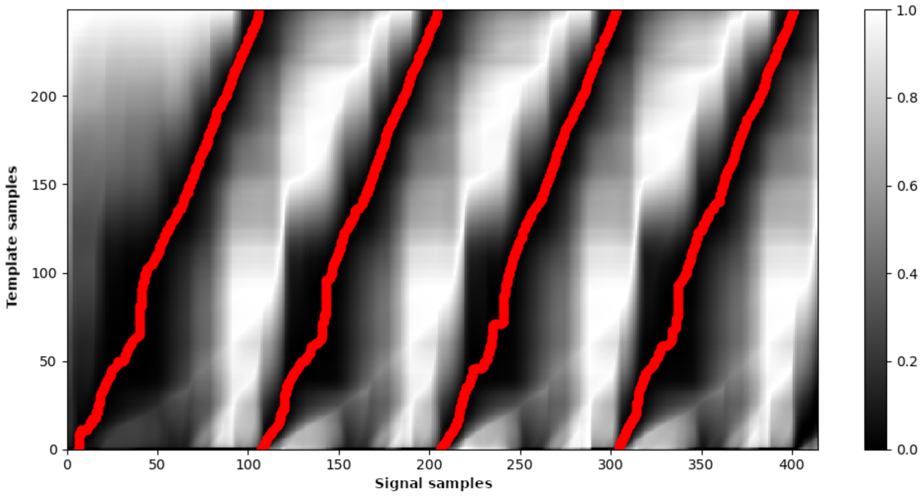


Figure 3.11: Example of an accumulated cost matrix C. The matrix C shown is obtained from matrix D of the shank angular velocity (axis Y) signal. In red: the *Warping Paths* identified using the DTW algorithm.

Ultimately, the point obtained at the end of the warping procedure is identified as the starting point (p_{start_i}) of each stride, i.e. the position of the heel strikes.

3.2.1 Creation of templates

The *Templates* for computing the DTW algorithm were created by calculating the mean of 360 manually labelled strides randomly selected from among those present in the recorded signals. The labelled strides were resampled to 250 samples.
Labelling was carried out following the knowledge acquired from the literature using a dedicated tool developed for the purpose [38][1][39][40][41][42][15].
Figure 3.12 shows the template obtained for the shank angular velocity (axis Y) signal during the activity *Walking* and the variability associated to each sample.

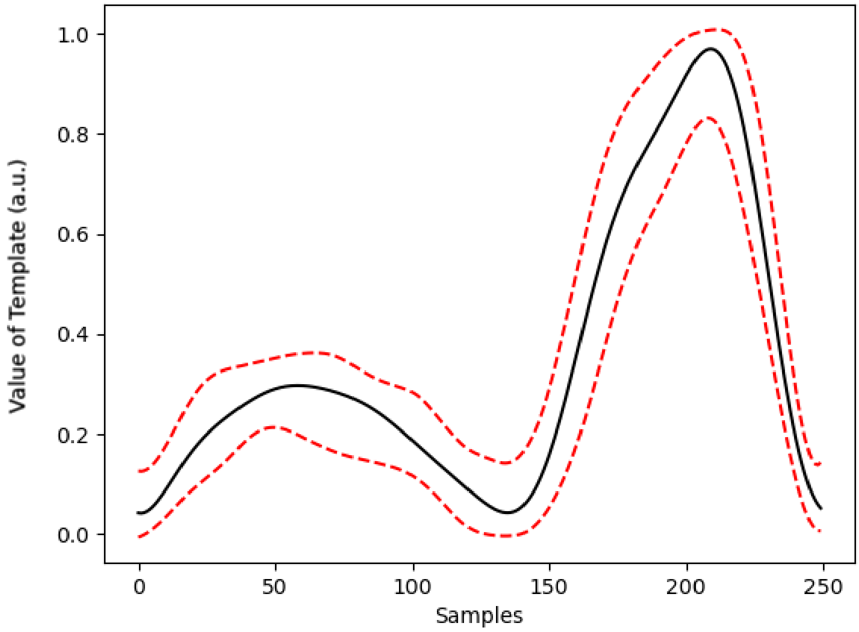
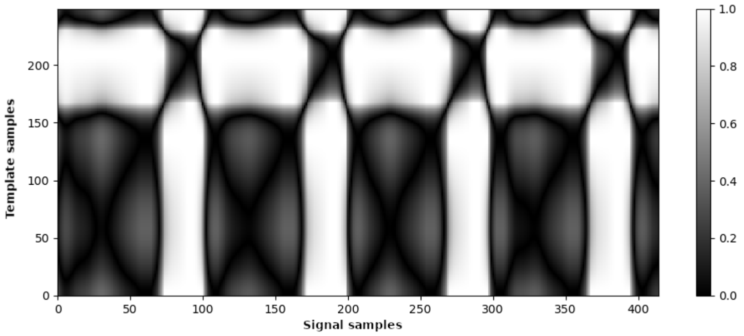


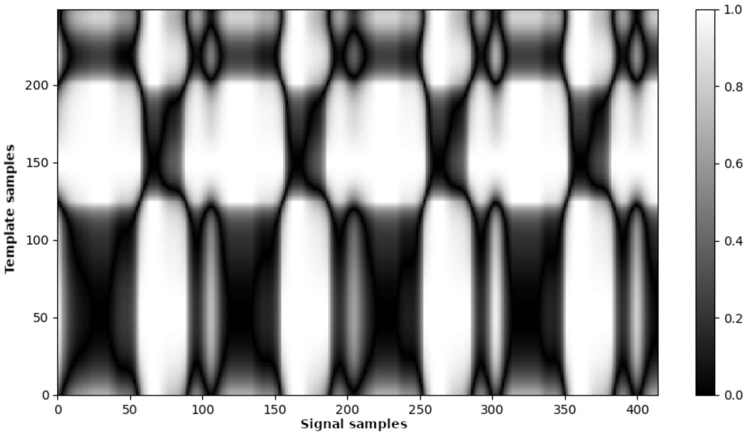
Figure 3.12: Example of a *Template* and its variance. The *Template* shown is the one corresponding to the shank angular velocity (axis Y) signal. The red dotted line represents the limits (maximum and minimum values) for each sample of the *Template*: as can be seen there is a higher variability in the beginning of the *Template*, when compared to the end part. This phenomenon was observed on several occasions while creating the *Template*. Moreover, the same behaviour was noted for all the activities.

3.3 Multi-Dimensional Subsequence Dynamic Time Warping Algorithm

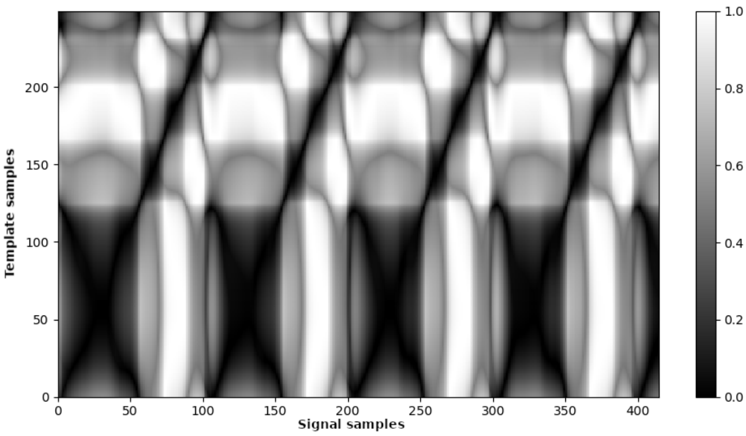
The main limitation of the DTW algorithm is its inability to take into consideration more than one *Target Signal* at a time. For this reason, a version of the algorithm which simultaneously considers more than one *Target Signal* was implemented: this version of the algorithm is called the Multi-Dimensional Subsequence Time Warping Algorithm (msDTW).
In this work, the msDTW implemented takes a cue from the algorithm proposed by *Barth et al.* [13], and extends it through a detailed explanation of the functionality of and the motivation behind each procedure.
This version of the DTW algorithm computes a matrix of distance D for each of the *Target Signals* that will be used and then computes the element-by-element sum of the D matrices to combine them. In fact, by summing the matrices it is possible to emphasise the difference between areas where the two matrices *'agree'* (in both matrices there are low/high values) and areas where they *'disagree'* (in one matrix there are low/high values and in the other the there are high/low values). Therefore, this procedure allows greater selectivity when analysing the diagonals (*Warping Paths*) inside the matrix.
An example of the procedure is shown in Figure 3.13.



(a) Matrix D of shank angular velocity (axis Y) signal, normalised.



(b) Matrix D of thigh angular velocity (axis Y) signal, normalised.



(c) Sum of the two matrices.

Figure 3.13: Example of summing of matrices D. The summing operation does not affect areas where the two matrices *'agree'* (for example on the diagonal paths and in some white areas) but penalises areas where they do not agree (such as certain areas outside the diagonal paths and in other areas of the matrices). As a consequence, in figure *(c)* the *Warping Paths* are more evident than in figure *(a)* and *(b)*. When computing the matrix C this translates to more accurate mapping of the *Warping Paths* and, in turn, to a more precise Accumulated Cost Function (3.9, 3.6) which will result in more accurate identification of the heel strikes.

The result obtained from summing the two matrices D will differ depending on the matrices concerned. For example, in Figure 3.14 one observes the matrix D resulting from the summing of matrix D of the shank gyroscope signal (axis Y) and matrix D of the shank accelerometer signal (axis Y).

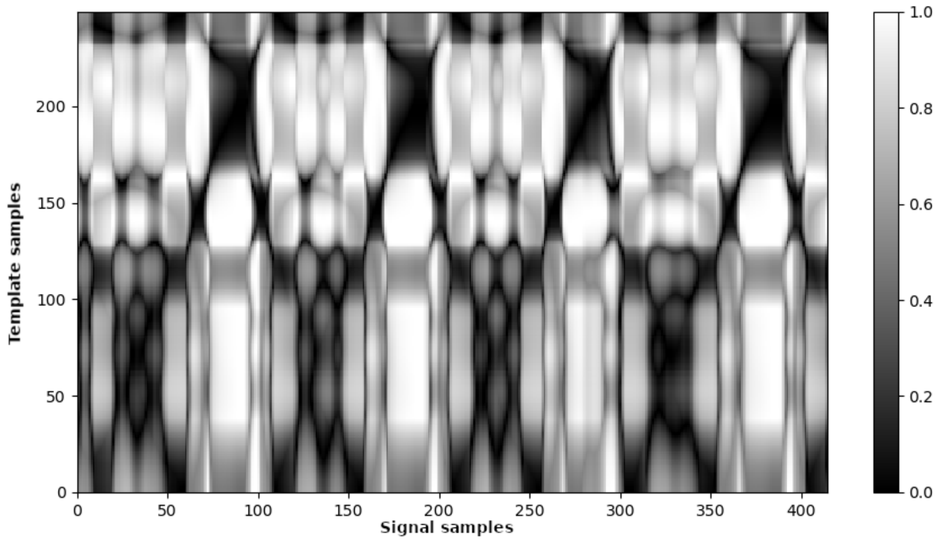


Figure 3.14: Result of the summing of matrix D of the shank angular velocity (axis Y) signal and matrix D of the shank acceleration signal (axis Y).

Figure 3.14 shows how the result of summing up two matrices D can differ depending on the matrices concerned. In this case, the matrix D obtained by summing is noisier and contains less accurate information about the location of the *Warping Paths*. Since the algorithm is based on the correct identification of *Warping Paths* for the localisation of the strides, the performance of the algorithm in terms of finding the correct position of the heel strikes will decrease compared to the case presented in Figure 3.13 (c) .
The summing of matrices D can be extended to more than two matrices so as to improve the performance of the msDTW algorithm yet further.

It was therefore necessary to better understand which *Target Signals* were more informative and which were less so. '*Informative*' here shall mean that a *Target Signal* improves the performance of the algorithm when considered in the sum of matrices.
In order to resolve this question, an Optimisation Algorithm was implemented (section 3.4).

3.4 Optimisation Algorithm

In order to optimise the msDTW Algorithm, some considerations were made:

1. A small distance in matrix D means that the pair of points involved in that distance represents a good match. That particular pair of points can, however, represent either a correct match (the point of the *Template* correctly corresponds to the point of the *Target Signal*) or an incorrect match (the match has a small distance but in reality the point of the *Template* shouldn't correspond to the point of the *Target Signal*).

2. The perfect and correct match between *Template* and *Target Signal* would be over a diagonal path that goes from the bottom of the image to the top (from the beginning of the *Template* to the end, and vice versa). Therefore, if a matrix D contains correct information, this information will be over diagonal paths.

Given these considerations, only regions presenting small distances along diagonal paths should be considered.

The goal of the Optimisation Algorithm is to recognise which *Target Signals* can actually improve the performance of the msDTW Algorithm and which ones would instead hamper performance.

Considering the complexity of the problem, it was necessary to make two assumptions so as to avoid having to test all possible combinations of *Target Signals*:

1. It is possible to divide the *Target Signals* (and thus the corresponding matrices D) in two groups: *informative* and *not informative*. As already specified, a signal is considered *informative* if its presence in the sum of matrices D improves the performance of the msDTW Algorithm.
   This differentiation is made possible by considering all possible two-by-two combinations of the matrices D, which can then be divided into two groups: combinations where the relevant matrices D *agree* with each other and combinations where the relevant matrices D *disagree*. As already specified, two matrices D *agree* on a certain region if, in that particular region, both matrices give low/high values. On the other hand, two matrices D *disagree* if in one matrix, in a particular region, there are low/high values and in the other matrix, in the same region, there are high/low values.
   The combinations belonging to the *disagree* group are not considered, as it would be impossible to tell which of the two matrices D involved contains the correct information regarding the position of the paths.
   It was then necessary to investigate which of the combinations belonging to the *agree* group (and thus the matrices D relating to those combinations) are informative and which are not. It is reasonable to expect that two matrices D may 'agree' on incorrect information, i.e. they agree on the wrong position of the paths.

2. If a *Target Signal* is present in most of the combinations belonging to the *agree* group, that *Target Signal* is *informative*. Therefore, all the *Target Signals* agreeing with the informative one are also informative.

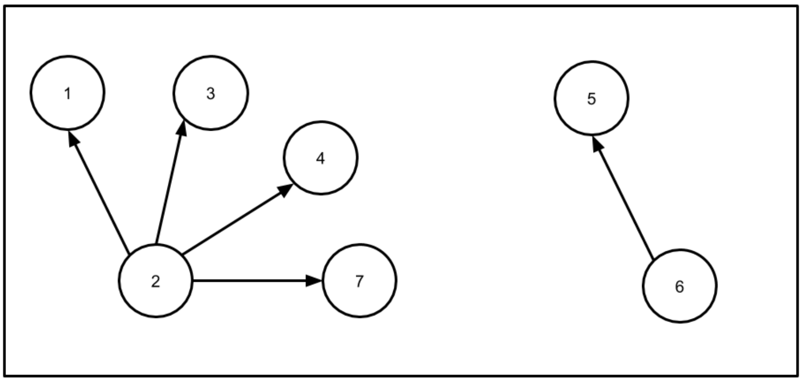


Figure 3.15: As an explanation of Assumption 2, let us consider the illustrated situation. *Target Signal* 2 is the one that agrees most with the other *Target Signals*, thus it is *informative*. Since *Target Signal* 1, 3, 4 and 7 agree with *Target Signal* 2, they are also considered informative. *Target Signal* 5 and *Target Signal* 6 agree with each other but are considered *not informative*, since there is a higher probability that they are agreeing on something incorrect, i.e noise.

For the considerations made, the comparison between the matrices D was focused on the diagonal paths contained in the matrices.
To begin, the images (the matrices) are binarised in order to isolate the diagonal paths present in the images. This procedure is executed with the help of the manual annotations. An example of binarisation can be seen in Figure 3.16

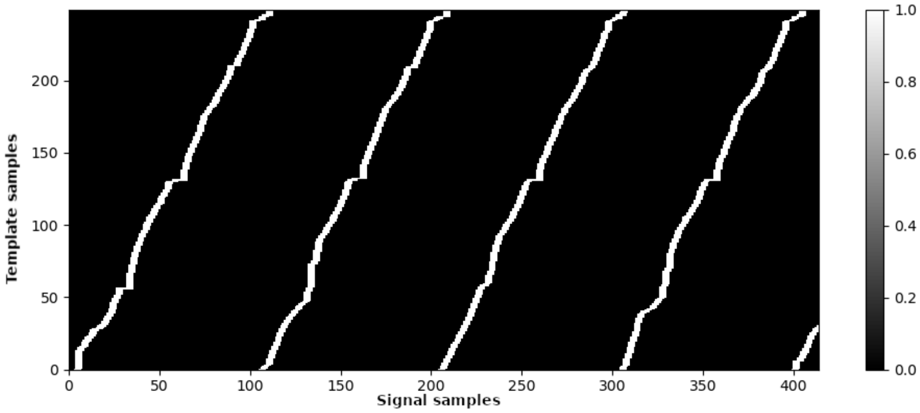
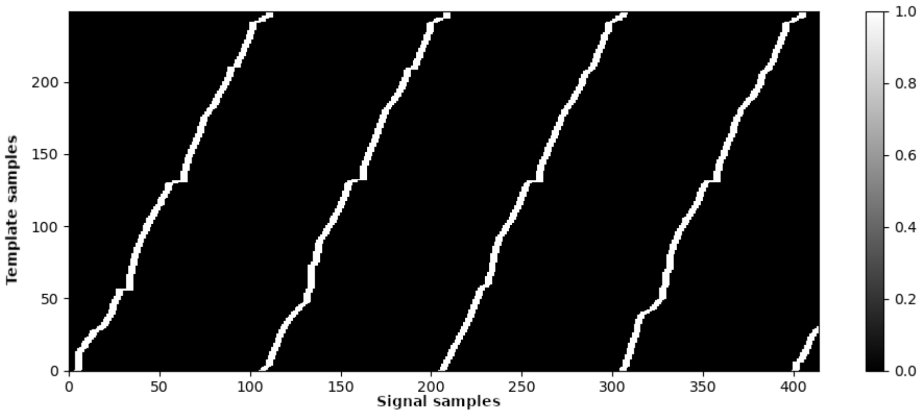
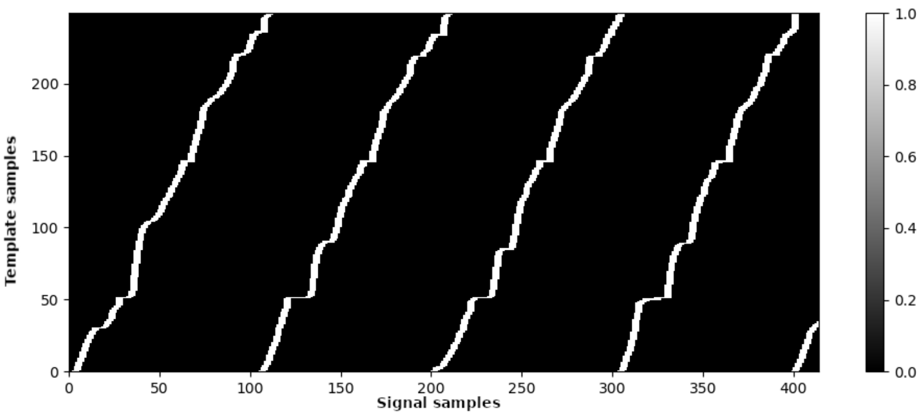


Figure 3.16: Example of binarisation. Binarised matrix obtained from the shank angular velocity (axis Y) signal.

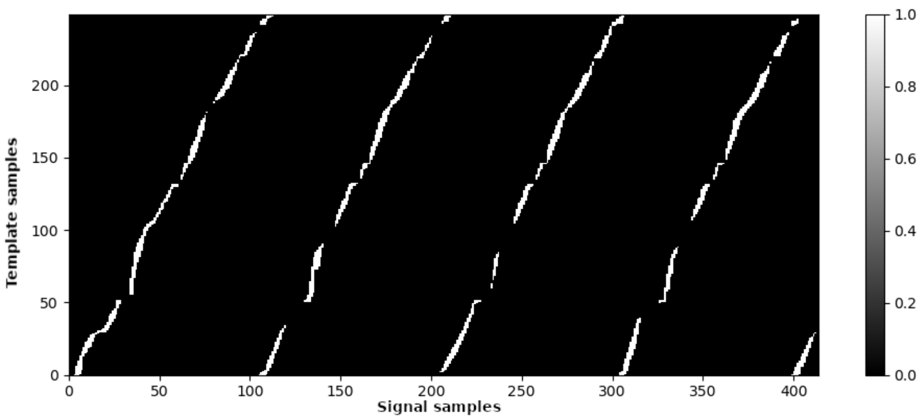
After the binarisation of all matrices, they are compared. For a given combination, the relevant matrices are multiplied element by element; this operation is equivalent to a logical-AND between the matrices. As expected, if two matrices agree (now only on a diagonal path) the resulting matrix preserves most of its regions. Otherwise, if two matrices disagree on most of the positions of the diagonal paths, the resulted matrix will lose most of its regions having logical-1 values.



(a) Binarised matrix obtained from the shank angular velocity (axis Y) signal.



(b) Binarised matrix obtained from the thigh angular velocity (axis Y) signal.

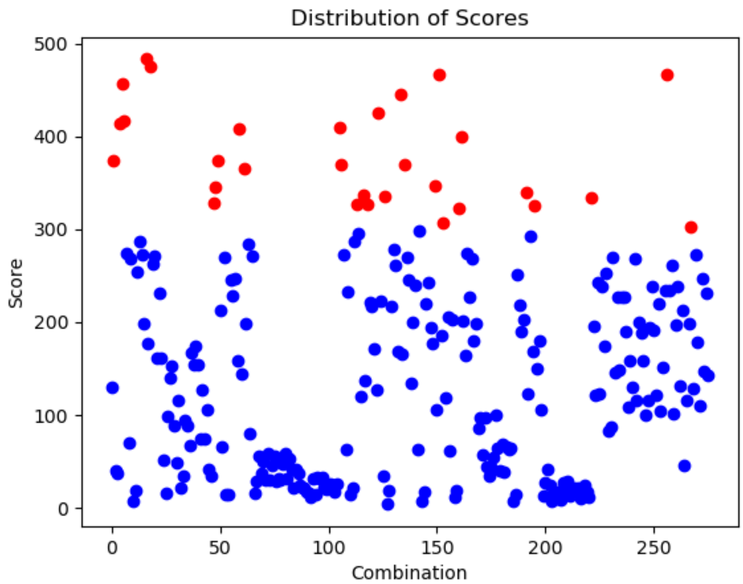


(c) Result of the product of the two matrices.

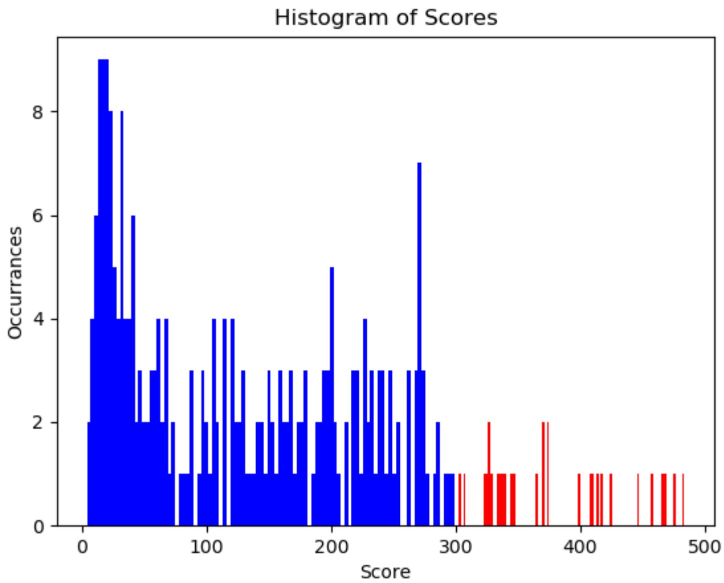
Figure 3.17: Example of the product of Binarised matrices.

For each combination, the sum of each element of the resulting matrix is computed; the number obtained is referred to as the *Score of the Combination*.
Combinations composed by matrices that agree in most regions will give a high *Score*, while combinations composed by matrices that disagree on most regions will give a low *Score* as result.

It is then possible to divide the combinations into the groups *Agree* and *Disagree* by considering all the *Scores* obtained, as well as a threshold set by taking into account the distribution and the histogram of the *Scores* (Figure 3.18) associated to each combination of *Target Signals*.



(a)



(b)

Figure 3.18: Distribution (a) and Histogram (b) of the *Scores*. On top: the distribution of the *Scores* along all possible two-by-two combinations of *Target Signals*. Marked in red: the combinations giving a *Score* higher than a prefixed threshold. In blue: discarded combinations. The histogram of the *Scores* is shown below. The threshold is adapted for each individual trial, though behaviour similar to that shown in the figure is observed for all recordings.

After dividing the combinations between the two groups, only combinations belonging to the *Agree* group are considered. Among those, the occurrences of each *Target Signal* are computed. Following the assumptions made, the signal presenting the highest number of occurrences in the combinations is considered *Informative*, as are all other signals that agreed with it. The set of those signals is denominated *Sensors Set*.

In order to obtain an unbiased result, the operation thus explained is performed on 70% of the recordings collected for a specific activity. As a final result, a *Sensors Set* for each recording considered is obtained. For the same reason, only the *Target Signals* present in at least a certain percentage of *Sensors Sets* are considered to be informative *Target Signals* (Figure 3.19).

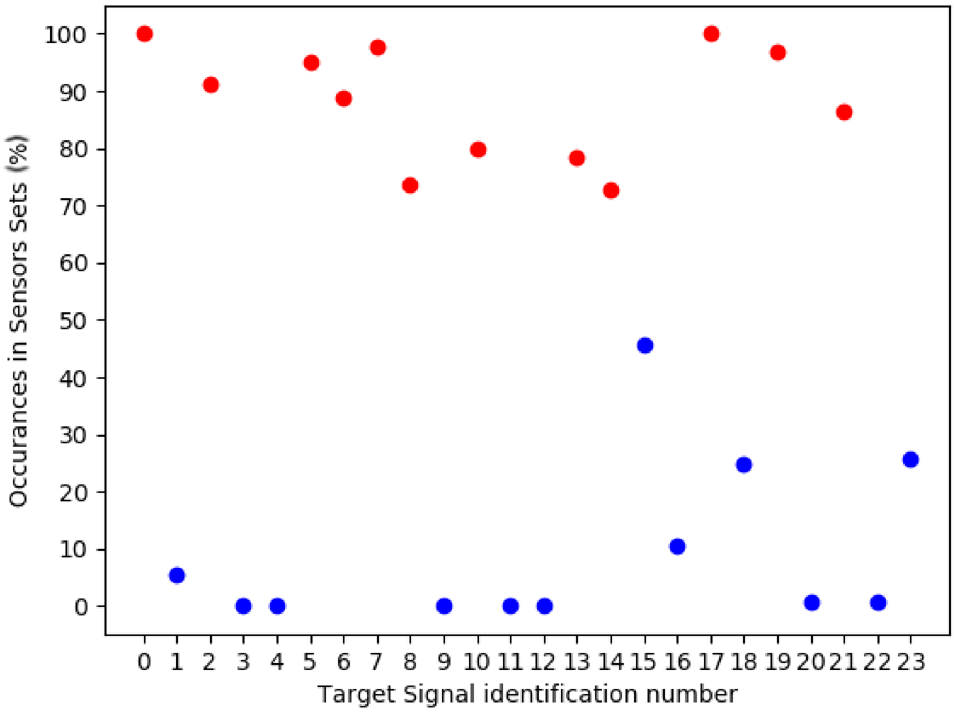


Figure 3.19: Percentage occurrence of each *Target Signal* in the *Sensors Sets*. In red: the *Target Signals* present in at least 70% of the *Sensors Sets* computed; these *Target Signals* make up the final *Sensors Set*. The signal identification number's legend can be found in *Section 2.1*.

The whole optimisation procedure was executed 5 times to consider the stochastic component of the algorithm, though the final *Sensors Set* did not change.
The operations thus illustrated all refer to the activity *Walking*, but the same optimisation procedure was executed for each activity, adapting parameters such as thresholds for each case.

Chapter 4

Results

4.1 Results of the Optimisation Procedure

The Optimisation algorithm was tested in the five activities: *Walking*, *Uphill Walking*, *Downhill Walking*, *Stair Ascent Walking* and *Stair Descent Walking*. Table 4.1 shows the results of the Optimisation.

Signal	**Axis**	**Walking**	**Uphill Walking**	**Downhill Walking**	**Stair Ascent**	**Stair Descent**
shank acceleration	x	X	X	X	X	X
shank acceleration	y					
shank acceleration	z	X	X	X	X	X
shank acceleration (without gravity)	x		X			
shank acceleration (without gravity)	y					X
shank acceleration (without gravity)	z	X	X	X	X	X
shank acceleration vector	xyz	X	X	X	X	X
shank acceleration vector (without gravity)	xyz	X	X	X	X	X
thigh acceleration	x	X	X	X	X	X
thigh acceleration	y					
thigh acceleration	z	X	X	X	X	X
thigh acceleration (without gravity)	x		X		X	
thigh acceleration (without gravity)	y					X
thigh acceleration (without gravity)	z	X	X	X	X	X
thigh acceleration vector	xyz	X			X	X
thigh acceleration vector (without gravity)	xyz				X	X
shank angular velocity	x					
shank angular velocity	y	X	X	X	X	X
shank angular velocity	z					
shank angular velocity vector	xyz	X	X	X	X	X
thigh angular velocity	x					
thigh angular velocity	y	X	X	X	X	X
thigh angular velocity	z					
thigh angular velocity vector	xyz				X	

Table 4.1: Results of the Optimisation Procedure.

4.2 Result of the segmentation

The three algorithms (Peak Detection (3.1), msDTW considering all the *Target Signals* (3.3) and msDTW Optimised (3.4)) were ultimately used for the segmentation of the strides. The results and methods of evaluation of this operation are shown in this section.

4.2.1 Methods for Error Measurement

Performance was evaluated using four different indicators calculated from the confusion matrix obtained after the segmentation. The confusion matrix was constructed following several considerations:

- Strides identified in the same position as the manual annotation were considered *True Positive (TP)*
- Strides wrongly undetected by the algorithms were considered *False Negative (FN)*
- Strides wrongly detected by the algorithms were considered *False Positive (FP)*
- Strides correctly undetected by the algorithms were considered *True Negative*

The four indicators used to evaluate the performance are:

1. *Accuracy*. This indicates the correctness of the segmentation; the percentage of correctly detected strides considering the whole matrix.

$$\mathit{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

2. *Recall*. This indicates how well the strides are identified, considering how many of them where correctly detected or undetected.

$$\mathit{Recall} = \frac{TP}{TP + FN} \tag{4.2}$$

3. *Precision*. This indicates how well the algorithms avoid detecting wrong strides; the higher the number of wrongly detected strides, the lower the Precision.

$$\mathit{Precision} = 1 - \frac{FP}{TP + FP} \tag{4.3}$$

4. *F1-Score*. This is the harmonic mean of *Precision* and *Recall* and takes into account missing strides and wrongly detected strides equally.

$$\mathit{F1\text{-}Score} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{4.4}$$

In this work, the parameter *Precision* is considered the most important. The desired result of the optimisation procedure is to enhance the performance of the msDTW algorithm, so as to improve the detection of heel strikes in terms of reducing the number of *False Positives*. This factor becomes important when the ultimate intention of the segmentation procedure is to automatically generate a dataset that will be used for future works focused on Artificial Intelligence models. The remaining indicators were used for a more complete comparison.

4.2.2 Evaluation of performance

Considering the activity *Walking*, the following performances were obtained for the three algorithms:

performance for the activity ***Walking***				
Algorithm	Accuracy	Recall	Precision	F1-Score
PeakDetection	92.02%	**99.94%**	92.08%	95.84%
msDTW (all *TS*)	95.02%	99.61%	95.38%	97.44%
msDTW Optimised	**95.87%**	99.77%	**96.05%**	**97.87%**

Table 4.2: Performances obtained using the three algorithms to annotate the data related to the activity *Walking*.

.

Considering the activity *Uphill Walking*, the following performances were obtained for the three algorithms:

performance for the activity ***Uphill Walking***				
Algorithm	Accuracy	Recall	Precision	F1-Score
PeakDetection	96.11%	99.97%	96.14%	98.01%
msDTW (all *TS*)	96.86%	**100%**	96.86%	98.40%
msDTW Optimised	**96.98%**	**100%**	**96.97%**	**98.46%**

Table 4.3: Performances obtained using the three algorithms to annotate the data related to the activity *Uphill Walking*.

Considering the activity *Downhill Walking*, the following performances were obtained for the three algorithms:

performance for the activity ***Downhill Walking***				
Algorithm	Accuracy	Recall	Precision	F1-Score
PeakDetection	96.08%	**100%**	96.08%	98.00%
msDTW (all *TS*)	97.66%	99.85%	97.81%	98.82%
msDTW Optimised	**98.26%**	99.78%	**98.44%**	**99.11%**

Table 4.4: Performances obtained using the three algorithms to annotate the data related to the activity *Downhill Walking*.

Considering the activity *Stair Ascent Walking*, the following performances were obtained for the three algorithms:

performance for the activity *Stair Ascent Walking*				
Algorithm	Accuracy	Recall	Precision	F1-Score
PeakDetection	82.22%	95.92%	85.20%	90.24%
msDTW (all *TS*)	**96.21%**	**97.87%**	98.27%	**98.07%**
msDTW Optimised	95.48%	96.32%	**99.05%**	97.67%

Table 4.5: Performances obtained using the three algorithms to annotate the data related to the activity *Stair Ascent Walking*.

Considering the activity *Stair Descent Walking*, the following performances were obtained for the three algorithms:

performance for the activity *Stair Descent Walking*				
Algorithm	Accuracy	Recall	Precision	F1-Score
PeakDetection	85.89%	**98.67%**	86.89%	92.42%
msDTW (all *TS*)	**91.78%**	92.94%	98.65%	**95.71%**
msDTW Optimised	91.13%	92.12%	**98.82%**	95.35%

Table 4.6: Performances obtained using the three algorithms to annotate the data related to the activity *Stair Descent Walking*.

Another important parameter that was taken into consideration for comparing the algorithms is the time of execution. The Peak Detection Algorithm has not been considered in this comparison since its architecture is not comparable with either of the two versions of the msDTW Algorithm.
The time of execution of the segmentation was considered to be as important as the *Precision*. It was possible to calculate decreases in percentage of the time of execution of the msDTW algorithm when comparing the version that takes into account all available *Target Signals* and the Optimised version. Table 4.7 shows the results of this evaluation.

Decreases in % of time of execution for msDTW and msDTW Optimised	
Activity	Decreases [%]
Walking	37.01%
Uphill Walking	36.72%
Downhill Walking	37.97%
Stair Ascent Walking	35.76%
Stair Descent Walking	36.02%

Table 4.7: Decreases in the percentage of time of executing the msDTW algorithm, comparing the two versions proposed. The decreases consider the reduction of time in percentage terms between the original version of the msDTW and the msDTW Optimised.

Chapter 5

Discussion

5.1 Optimisation Procedure

As displayed in Table 4.1, the Optimisation Procedure leads to some interesting observations. Some signals, the shank acceleration (axis Y) for example, are never selected as *informative Target Signals*, while signals such as the shank angular velocity (axis Y) are always selected in all activities. Moreover, certain signals are selected in some activities but not in others.
These results are perfectly aligned with expectations. From a biomechanical point of view, most motion of the articulations while performing a gait activity occurs in some particular planes of the space; several studies suggest that the greatest part of the motion is placed in the sagittal and coronal planes, while the range of motion in the transverse plane is far less significant.
Analysing the data and the results obtained from the Optimisation Procedure, one can confirm that signals describing the movement of the IMU sensor in certain directions, such as the mediolateral displacements, are not as relevant as displacements along the vertical or anterior-posterior direction. The same behaviour was observed for the rotations. Several studies note that the angular displacements of the joints of the lower limb are principally located in the sagittal plane. This is confirmed by the results as the angular velocity, and thus the angular rotation, in the transverse plane (around axis X) and in the coronal plane (around axis Z) are never selected as *informative Target Signal*, while, on the other hand, the angular velocity around the axis Y (i.e. in the sagittal plane) is always selected, both in the shank and in the thigh data.
Moreover, one observes that removing the gravity component from the acceleration signals (axis X) in both the shank and the thigh IMU implies the non-selection of those signals for most activities, while the same signals with the gravity component included are always selected. On the other hand, the acceleration signals (axis Z) in the shank and in the thigh IMU are always selected, either with and without the gravity component, for all activities.

5.2 Performance of the algorithms

Considering that the purpose of this work was to increase the performance of the msDTW Algorithm in terms of reducing the number of *False Positives* and decreasing the time of computation, results shown in Table 4.2, Table 4.3, Table 4.4, Table 4.5, Table 4.6, and

Table 4.7 confirm that this goal has been achieved.
More specifically, when comparing the msDTW performances with those of msDTW Optimised, for all the activities one observes increases in the indicator *Precision* ranging from +0.11% up to +0.78%.
Moreover, for the activity *Walking*, *Uphill Walking*, and *Downhill Walking*, the msDTW Optimised gave better performances both in terms of *Accuracy* and *F1-Score*. Considering the *Accuracy*, increases between +0.85% and +1.49% were observed, while, for the *F1-Score*, increases ranging from +0.06% up to +0.43% were obtained. Still considering these three activities, the parameter *Recall* showed different results depending on the activity, but in each case the performance remained higher than 99.61%.
A slightly different behaviour was noted for the activities *Stair Ascent Walking* and *Stair Descent Walking*. In these two cases, marginal decreases in *Accuracy* can be observed, always in the face of an increase in *Precision*. This response from the algorithm can be explained from a biomechanical point of view. In fact, when analysing the variance of the manually labelled strides in the activity related to the *Stair Ascent/Descent*, a much higher stride variability was noted when compared to the activities related to Walking (both on a flat surface and on a slope). These results led us to the conclusion that the biomechanical movements taking place during the climbing/descending of stairs differ greatly according to the person performing the activity.
On the other hand, as previously mentioned, an increase in *Precision* can be appreciated. This led us to conclude that the msDTW Optimised is less subject to the detection of *False Positive* strides and is thus better from this perspective compared to the non-optimised version. In order to improve the *Accuracy* in these two activities, more than one *Template* should be created, to allow the algorithm to better adapt to the different movements that a person carries out while approaching a Stair activity. Considering the time of execution of the msDTW algorithms, it was noted that the Optimised version showed good improvements in terms of speed and reduction of unnecessary computational cost. This is the consequence of the non-computation of the matrices D of those signals not considered in the *Sensor Set* of a specific activity. This enhancement can be seen in Table 4.7, quantified as a decrease in the time of execution of the algorithm between 35.76% and 37.97%.
Furthermore, when comparing the results obtained with the msDTW Optimised and the Peak Detection algorithm, an increase in *Accuracy*, *Precision*, and *F1-Score* was observed for all activities. More specifically, with regards to *Accuracy*, increases ranging from +0.87% up to +13.26% can be noted. As expected, considering the high variability in the activities involving stairs, the *Accuracy* is subjected to a substantial improvement. In fact, since the Peak Detection algorithm is based on simple rules that consider only the amplitude of peaks in the shank angular velocity (axis Y), the errors made by this algorithm are much higher in the case of *Stairs Activities* than in the case of *Walking Activities* (both on a flat surface and on a slope), where a much higher regularity can be observed. On the other hand, the nature of the msDTW algorithm allows highly effective differentiation of correct strides from the noisy parts of the recordings. In terms of *F1-Score* the same results were obtained; increases in this parameter were quantified between a minimum of +0,45%, up to a maximum of +7.43%. Considering the *Precision*, for the same reasons already mentioned, an enhancement of +0.83% up to +13.85% was observed. This confirms the capability of the msDTW Optimised algorithm to be even more selective in terms of the detection of correct strides, leading to more reliable segmentation overall. In particular, the greatest improvement in *Precision* is observed for the activity *Stair Ascent Walking*, where the best performance is also obtained.

Chapter 6

Conclusions

The ability to automatically and robustly segment individual strides from gait sequences derived from wearable inertial sensors while performing different gait activities is crucial for the estimation of gait parameters and for the creation of a reliable gait dataset, without requiring the manual segmentation of recordings.
In this work, an Optimisation Procedure for the msDTW Algorithm is presented. The proposed algorithm aims to outperform the other segmentation algorithms in terms of precision and/or computational speed. Firstly, a Peak Detection method is implemented as a baseline and allows comparison of the obtained performance. Secondly, the msDTW Algorithm proposed in [13] is extended and extensively explained in terms of functionality and procedures. Finally, the proposed msDTW Optimised method built upon msDTW is presented. Performance is evaluated in terms of *Accuracy*, *Recall* and *F1-Score* with particular attention given to the *Precision* and the computational cost. The obtained results lead us to affirm that the proposed method of msDTW Optimised is a robust and reliable solution for stride segmentation, capable of identifying the majority of the strides recorded during a gait activity with a higher precision compared to msDTW and Peak Detection, and with a better performance-computational cost ratio.

Bibliography

[1] T. Gujarathi and K. Bhole, "Gait analysis using imu sensor," *10th ICCCNT 2019*, 2019.

[2] R. DiMarco, S. Rossi, F. Patanè, and P. Cappa, "Technical quality assessment of an optoelectronicsystem for movement analysis," *IMEKO*, 2015.

[3] F. Casamassima, A. Ferrari, B. Milosevic, P. Ginis, E. Farella, and L. Rocchi, "A wearable system forgait training in subjects with parkinson's disease," *Sensors*, 2014.

[4] J. R. Rebula, L. V. Ojeda, P. G. Adamczyk, and A. D. Kuo, "Measurement of foot placement and itsvariability with inertial sensors.," *Gait Posture*, 2013.

[5] N. T. N. Trung, Y. Makihara, H. Nagahara, Y. Mukaigawa, and Y. Yagi, "Inertial-sensor-based walking action recognition using robust step detection and inter-class relationships," *21st International Conference on Pattern Recognition (ICPR)*, 2012.

[6] A. Mannini and A. M. Sabatini, "A hidden markov model-based technique for gait segmentation usinga foot-mounted gyroscope," *33th Annual International Conference of the IEEEEngineering in Medicine and Biology Society (EMBC)*, 2011.

[7] A. Mannini, D. Trojaniello, U. Della Croce, and Sabatini, "Hidden markov model-based strategy for gaitsegmentation using inertial sensors: application to elderly, hemiparetic patients and huntington's diseasepatients," *37th Annual International Conference of the IEEE Engineering in Medicineand Biology Society (EMBC)*, 2015.

[8] G. Panahandeh, N. Mohammadiha, A. Leijon, and P. Handel, "Continuous hidden markov model for pedestrian activity classification and gait analysis," *IEEE Trans. Instrum. Meas.*, 2013.

[9] T. Pfau, M. Ferrari, K. Parsons, and A. Wilson, "A hidden markov model-based stride segmentation technique applied to equine inertial sensor trunk movement data," *J. Biomech*, 2008.

[10] R. Begg, M. Palaniswami, and B. Owen, "Support vector machines for automated gait classification," *IEEE transactions on bio-medical engineering*, 2005.

[11] D. Das and S. Saharia, "Human gait analysis and recognition using support vector machines," *Computer Science an Information Technology*, 2014.

[12] F. Sakr, F. Bellotti, R. Berta, and A. De Gloria, "Machine learning on mainstream microcontrollers," *Sensors*, 2020.

[13] J. Barth, C. Oberndorfer, C. Pasluosta, and S. Schülein, "Stride segmentation during free walk movements using multi-dimensional subsequence dynamic time warping on inertial sensor data," *Sensors*, 2015.

[14] Y. Luo, S. M. Coppola, P. C. Dixon, S. Li, J. T. Dennerlein, and B. Hu, "A database of human gait performance on irregular and uneven surfaces collected by wearable sensors," *Nature*, 2020.

[15] A. Salarian, H. Russmann, F. J. G. Vingerhoets, C. Dehollain, Y. Blanc, P. R.

Burkhard, and K. Aminian, "Gait assessment in parkinson's disease: toward an ambulatory system for long-term monitoring," *IEEE transactions on biomedical engineering,*, 2004.

[16] L. Brockett Claire and J. Chapman Graham, "Biomechanics of the ankle," *Orthopaedics and trauma*, vol. 30, no. 3, pp. 232–238, 2016.

[17] C. Barnett and J. Napier, "The axis of rotation at the ankle joint in man.its influence upon the form of the talus and the mobility of thefibula," *J Anat*, 1952.

[18] S. K. Grimston, B. M. Nigg, D. A. Hanley, and J. R. Engsberg, "Differences inankle joint complex range of motion as a function of age," *FootAnkle Int*, 1993.

[19] M. Akhtaruzzaman, A. A. Shafie, and M. Raisuddin Khan, "Gait analysis: systems, technologies, and importance," *World Scientific*, 2016.

[20] A. Muro de la Herran, B. Garcia-Zapirain, and A. Mendez-Zorrilla, "Gait analysis methods: an overview of wearable and non-wearable systems, highlighting clinical applications," *Sensors*, 2014.

[21] G. Taga, "A model of the neuro-musculo-skeletal system for human locomotion," *Biol Cybern*, 1995.

[22] S. Chowdhury and N. Kumar, "Estimation of forces and moments of lower limb joints from kinematics data and inertial properties of the body by using inverse dynamics technique," *J Rehabil Robot*, 2013.

[23] D. Smith, J. W. Michael, and J. H. Bowker, "American academy of orthopaedic surgeons. atlas of amputations and limb deficiencies," *Surgical, prosthetic and rehabilitation Principles*, 2011.

[24] A. Esquenazi, "ait analysis in lower-limb amputation and prosthetic rehabilitation," *Physical Medicine and Rehabilitation Clinics*, 2014.

[25] A. K. Silverman, N. P. Fey, A. Portillo, J. G. Walden, G. Bosker, and R. R. Neptune, "Compensatory mechanisms in below-knee amputee gait in response to increasing steady-state walking speeds," *Gait posture*, 2008.

[26] D. P. Murphy, "Fundamentals of amputation care and prosthetics," *Demos Medical Publishing*, 2013.

[27] R. Wu, J. Wu, and W. Xiao, "Gait detection using a signle accelerometer," *IEEE 15th International Conference on Control and Automation (ICCA)*, 2015.

[28] M. Grimmer, J. Zeiss, F. Weigand, G. Zhao, S. Lamm, M. Steil, and A. Heller, "Lower limb joint biomechanics-based identification of gait transitions in between level walking and stair ambulation," *Plos One*, 2020.

[29] Y. Yao, L. Pan, W. Feng, X. Xu, X. Liang, and X. Xin, "A robust step detection and stride length estimation for pedestrian dead reckoning using a smartphone," *IEEE SENSORS JOURNAL*, 2020.

[30] D. Gouwanda and A. A. Gopalai, "A robust real-time gait event detection using wireless gyroscope and its application on normal and altered gaits," *Medical Engineering and Physics*, 2014.

[31] L. Markus, K. Warner, B. Joao P, N. Chuong, B. Cornelius, and S. Leonhardt, "Peak detection algorithm for gait segmentation in long-term monitoring for stride time estimation using inertial measurement sensors," *IEEE EMBS International Conference on Information Technology Applications in Biomedicine (ITAB)*, 2019.

[32] B. Nikolaos V., P. Konstantinos N., and D. Hatzinakos, "Gait recognition using dynamic time warping," *IEEE 6th Workshop on Multimedia Signal Processing*, 2016.

[33] S. Khandelwal and N. Wickstrom, "Identification of gait events using expert knowledge and continuous wavelet transform analysis," *7th International Conference on Bio-inspired Systems and Signal Processing*, 2016.

[34] B. Mohamed, B. Olivier, D. Vincent, S. Cédric, D. Marie, G. Gaëtan, and G. Jacques, “Segmentation of gait cycles using foot-mounted 3d accelerometers,” *International Conference on 3D Imaging (IC3D)*, 2015.

[35] Z. Yongjia and Z. Suiping, “Wearable device-based gait recognition using angle embedded gait dynamic images and a convolutional neural network,” *Sensors*, 2017.

[36] R. D. Gurchiek, R. H. Choquette, and B. D. Beynnon, “Remote gait analysis using wearable sensors detects asymmetric gait patterns in patients recovering from acl reconstruction,” *International Workshop on Wearable and Implantable Body Sensor Networks (BSN)*, 2016.

[37] G. Panahandeh, N. Mohammadiha, A. Leijon, and P. Handel, “Continuous hidden markov model for pedestrian activity classification and gait analysis,” *. IEEE Trans. Instrum. Meas.*, 2013.

[38] J.-T. Z. Shuozhi Yang a, C. Alison, Novak, B. Brouwer, and Q. Li, “Estimation of spatio-temporal parameters for post-stroke hemiparetic gait usinginertial sensors,” *GaitPosture*, 2013.

[39] H. Ghassemi, J. Hannink, F. Martindale, H. Gabner, M. Muller, H. Klucken, and B. Eskofier, “Segmentation of gait sequences in sensor-based movement analysis: a comparison of methods in parkinson’s disease,” *Sensors*, 2017.

[40] K. Aminian, B. Najafi, and L. Robert, “Spatio-temporal parameters of gait measured by an ambulatorysystem using miniature gyroscopes,” *Journal of Biomechanics*, 2002.

[41] B. Coley, B. Najafi, A. Paraschiv-Ionescu, and K. Aminian, “Stair climbing detection during daily physical activityusing a miniature gyroscope,” *GaitPosture*, 2005.

[42] A. M. Sabatini, G. Ligorio, and A. Mannini, “Fourier-based integrationof quasi-periodic gait accelerations for drift-free displacement estimation using inertial sensors,” *BioMedical Engineering*, 2015.

[43] K. Aminian, C. Trevisan, B. Najafi, H. Dejnabadi, C. Frigo, E. Pavan, A. Telonio, F. Cerati, E. C. Marinoni, P. Robert, and P.-F. Leyvraz, “Evaluation of an ambulatory system for gait analysis in hip osteoarthritis and after total hip replacement,” *Gait Posture*, 2003.

[44] J. Barth, C. Oberndorfer, P. Kugler, D. Schuldhaus, and J. Winkler, “Subsequence dynamic time warping as a method for robust step segmentation using gyroscopesignals of daily life activities,” *In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2013.

[45] D. M. Powers, “Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation,” *International Journal of Machine Learning Technology*, 2011.

[46] G. A. Ten Holt, M. J. T. Reinders, and E. A. Hendriks, “Multi-dimensional dynamic time warping forgesture recognition,” *In Proceedings of the Thirteenth Annual Conference of the Advanced Schoolfor Computing and Imaging*, 2007.

[47] L. Prjiot, C. Schwartz, J. Watrin, A. Mendes, J. Croisier, B. Forthomme, and V. Denoel, “Algorithm for extracting initial and terminal contact timings duringtreadmill running using inertial sensors,” *Biosignals*, 2020.

[48] R. Alvarez, E. Pulido, and D. A. Sierra, “Climbing/descending stairs detection using inertial sensors and implementing pca and a svm classifier,” *Claib 2016*, 2016.

[49] M. Schmid, F. Riganti-Fulginei, I. Bernabucci, A. Laudani, D. Bibbo, R. Muscillo, A. Salvini, and S. Conforto, “Svm versus map on accelerometer data to distinguish among locomotor activities executed at different speeds,” *Hindawi Publishing*, 2013.

[50] L. Prijot, C. Schwartz, J. Watrin, A. Mendes, J.-L. Croisier, B. Forthomme, V. Denoël, O. Brüls, and M. Boutaayamou, “Algorithm for extracting initial and terminal

contact timings duringtreadmill running using inertial sensors," *Biosignals*, 2020.

[51] A. Psarras, D. Mertyri, and P. Tsaklis, "Biomechanical analysis of ankle during the stance phaseof gait on various surfaces: a literature review," *Human Movement*, 2016.

[52] A. Leardini, J. J. O'Connor, and S. Giannini, "Biomechanics of the natural, arthritic, andreplaced human ankle joint," *Journal of foot and ankle research*, 2014.

[53] L. Metsavaht and G. Leporace, "Current trends in the biokinetic analysis of thefoot and ankle," *Journal of the FootAnkle*, 2020.

[54] N. de Boer, H. Abdi, M. Fielding, and S. Nahavandi, "Segmentation and feature extraction of human gait motion," *DesTech Conference*, 2017.

[55] V. Rosso, L. Gastaldi, V. Agostini, R. Takeda, and S. Tadano, "Gait measurements in the transverse plane using awearable system," *IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2017.

[56] S. Winiarski, J. Pietraszewska, and B. Pietraszewski, "Three-dimensional human gait pattern: reference data for young, active women walking with low, preferred, and high speeds," *Hindawi*, 2019.

[57] N. Roth, A. Küderle, M. Ullrich, T. Gladow, F. Marxreiter, J. Klucken, B. Eskofier, and F. Kluge, "Hidden markov model based stride segmentation on unsupervised free-living gait data in parkinson's disease patients," *Research square*, 2021.

[58] H. T. T. Vu, F. Gomez, P. Cherelle, D. Lefeber, A. Nowé, and B. Vanderborght, "Ed-fnn: A new deep learning algorithm to detect percentage of the gait cycle for powered prostheses.," *Sensors*, 2018.

[59] H. T. T. Vu, D. Dong, H. L. Cao, T. Verstraten, D. Lefeber, B. Vanderborght, and J. Geeroms, "A review of gait phase detection algorithms for lower limb prostheses.," *Sensors*, 2020.