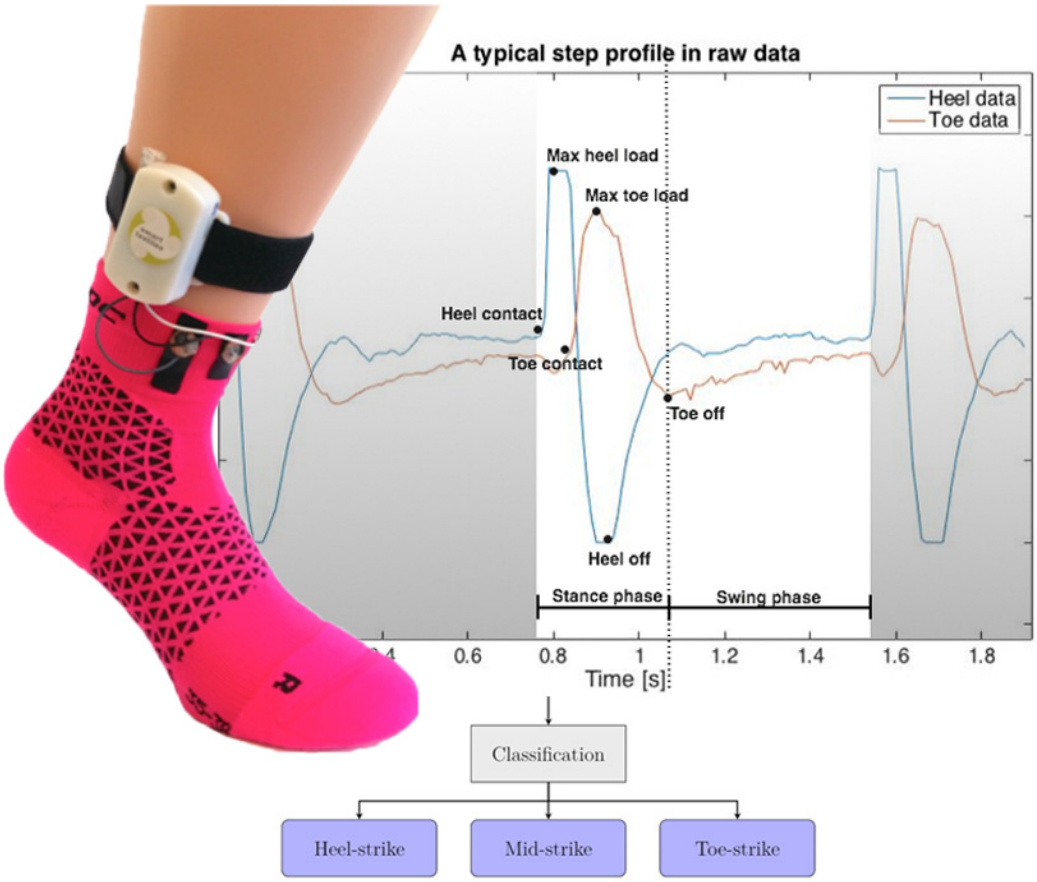
CHALMERS
UNIVERSITY OF TECHNOLOGY



Automated foot strike pattern recognition using a smart sock with textile piezoelectric sensors

Master's thesis in Biomedical Engineering

ANNA RAGNERIUS
FRIDA WIDELUND

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2016

Master's thesis 2016:EX032

Automated foot strike pattern recognition using a smart sock with textile piezoelectric sensors

ANNA RAGNERIUS
FRIDA WIDELUND

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Signals and Systems
Division of Signal processing and Biomedical engineering
Chalmers University of Technology
Gothenburg, Sweden 2016

Automated foot strike pattern recognition using a smart sock with textile piezoelectric sensors

ANNA RAGNERIUS

FRIDA WIDELUND

Supervisor: Leif Sandsjö, University of Borås/MedTech West
Examiner: Stefan Candefjord, Chalmers University of Technology/MedTech West

Master's Thesis 2016:EX032
Department of Signals and Systems
Division of Signal processing and Biomedical engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2016

Automated foot strike pattern recognition using a smart sock with textile piezoelectric sensors

FRIDA WIDELUND
ANNA RAGNERIUS
Department of Signals and Systems
Chalmers University of Technology

Abstract

Information about a runner's foot strike pattern is interesting as the foot strike is not only believed to impact the runner's performance but also the risk of sustaining running-related injuries. In this thesis a software system for recognition of foot strike patterns has been developed. The system makes use of signals from a sock instrumented with textile piezoelectric sensors in heel and toe. The purpose of the thesis was to develop software for automatic classification of runners as either heel-, mid- or toe-strike and provide information about foot strike patterns based on signals from the instrumented socks.

Data was collected on a treadmill while following a protocol. The protocol included walking and running at different speeds with the three strike types; heel-, mid- and toe-strike, resulting in a database with sequences tagged accordingly. Five subjects collected in total 186 minutes of walking/running data. A pattern recognition method composed of preprocessing, segmentation and classification with a supervised neural network was developed.

The resulting system succeeds to classify foot strike patterns correctly up to 97 %. This shows that it is possible to use the piezoelectric textile sensor to classify a runner's foot strike pattern, though the system has some limitations. The changing properties of the instrumented sock during use disables logging sessions throughout a full protocol. Therefore, it would be necessary to improve the sock, and possibly the hardware, before conducting software test on a larger test group and continuing the research in other areas of use than running.

Keywords: piezoelectric fibre, signal processing, segmentation, classification, gait, running, smart sock.

Acknowledgements

Firstly, we would like to thank our supervisor Leif Sandsjö for much appreciated advice and for steering us in the right direction. We would also like to thank Karin Rundqvist for her patience when helping us with the prototypes, Nils-Krister Persson for creative ideas and Erik Nilsson for providing us with material. Last but not least, we would like to thank our ruthless runners; Daniel Pihlquist, Johan Jonsson, Lisa Nilsson, Ulrika Fredborg and Marcel Nilsson.

Frida Widelund and Anna Ragnerius, Gothenburg, June 2016

Contents

1

Introduction

A person's gait type can be categorised by the foot strike pattern. A common way to do this is by assigning one of the three labels heel-, mid- or toe-strike, according to the runners landing. When it comes to distance runners about 80% are categorised as heel-strikers while most of the other 20% are mid-strikers, and very few are categorised as toe-strikers (Novacheck, 1998). When speed is increased, runners generally tend to go from heel- or mid-strike towards a clearer toe-strike (Mann and Hagy, 1980). The foot-strike type is not only believed to impact the runners speed but also the risk of sustaining running-related injuries (Kulmala et al., 2013).

1.1 Background

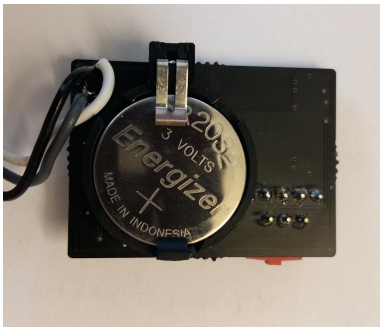
A system for logging and monitoring foot strike has been developed by *Smart Textiles* at *The University of Borås* in collaboration with *Voss ingenjörsfirma AB* and *Swerea IVF*. The work was initiated as a study to investigate the possibility to develop a portable sensor platform for foot strike patterns (Rundqvist et al., 2014b).

The developed system consists of a sock instrumented with two textile piezoelectric sensors, a blue-tooth unit and an Android application (Sandsjö et al., 2014). One of the sensors is placed under the heel and the other is placed under the toe. The blue-tooth unit is placed on the ankle, as can be seen in Figure 1.1. It performs signal conditioning and converts the signals from analog to digital. The digital signals are sent to the smartphone with a sampling frequency of 100 Hz (Sandsjö et al., 2014).

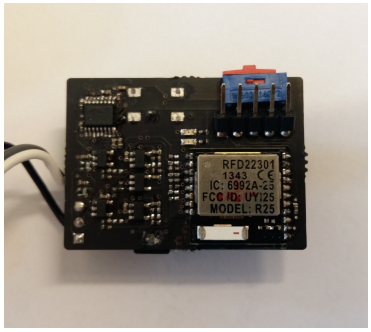
The piezoelectric sensor has a lower limit frequency of a few Hertz and therefore it is not possible to monitor the pressure while standing still. Thus, the amplitude of the signal will rise only when there is a change in pressure (Nilsson et al., 2013).

Figure 1.1: The current solution; the white box contains the blue-tooth unit and the phone shows the Android application with the time graph of the raw signals.

The blue-tooth unit is built up by a microcontroller that sends the signals via blue-tooth, an 8-bits AD-converter giving amplitude values between 0 and 255 (unitless) and a 3V battery. The microcontroller is turned on by a switch button. Figure 1.2a and 1.2b shows the placement of the battery and the microcontroller inside the unit.



(a) Battery



(b) Microcontroller and AD-converter

Figure 1.2: Images of the blue-tooth unit.

The Android application enables monitoring and logging of the signals. In the application, the raw signals are displayed in a time graph. The amplitude of the signals can be interpreted as the change in pressure. A screenshot can be seen in Figure 1.3 where blue is the heel sensor and yellow is the toe sensor. The application

has three buttons; 'SRV On' is used to start the logging, 'Toggle' to change visible graphs and 'Unit' to specify bluetooth unit name.

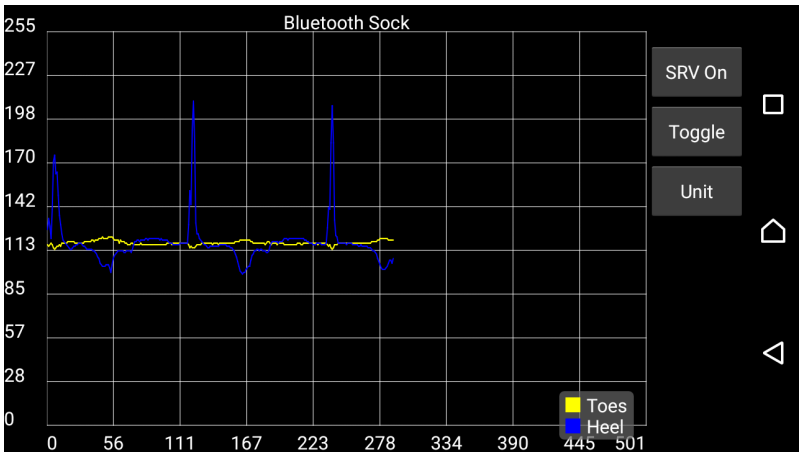


Figure 1.3: A screenshot of the existing Android application. In this logging moment the heel sensor (blue) worked well but the sensitivity in the toe sensor (yellow) was very low.

Though the now existing hardware is enough to log signals from the heel and toe, the solution needs further work to be able to give gait information based on the signals.

1.2 Purpose and Objectives

The purpose of this study is to investigate the feasibility of extracting gait information from signals recorded by a smartphone from a newly developed piezoelectric textile sensor integrated in the heel and toe of a smart textile sock. The aim is to develop a software for automated extraction of gait information and classification of foot strike pattern.

The objectives are:

- Compute the number of steps taken and cadence.
- Compute stance and swing time.
- Classify foot strike patterns as either heel-, mid-, or toe-strike.
- Look at the possibility to implement the software in real-time.

1.3 Delimitations

No development of the hardware will be performed. The focus will lie within software development.

Different algorithms will not be evaluated based on run-time or power efficiency.

1.4 Previous Work

The melt spun piezoelectric polyvinylidene fluoride (PVDF) fibres used in this project was developed and manufactured by Swerea IVF in 2011, as described in (Lund et al., 2012). The piezoelectric fibre has similar properties as conventional textile fibres, which enables integration in fabrics as a sensor without compromising the comfort. How to make use of the piezoelectric fibre as a textile based sensor was examined by Rundqvist beginning in 2013. The study concerned the possibility to weave with the PVDF fibre to achieve a textile based sensor for use in smart textiles (Rundqvist, 2013, Rundqvist et al., 2014a). Smart textile refers to a textile that has the ability to sense and respond to changes in its environment and in a textile based sensor, the textile itself is the sensor (Rundqvist et al., 2014a).

In her study Rundqvist tested three different weaving processes; the PVDF fiber in warp direction combined with different types of conductive yarn as outer electrode in weft direction. The study concluded that it is fully possible to weave with PVDF fibre and conductive yarn to create a fully piezoelectric textile sensor, as the samples from all three weaving processes showed a piezoelectric effect when strain was applied.

K. Rundqvist et al. continued the research by integrating the PVDF fibre into a regular sock with the aim to develop a system for logging foot strike data. The study showed that the areas where output is desired can be defined by the placement of the outer electrode. By placing the outer electrode under heel and toe the time difference between initial ground contact for the heel and toe could be measured (Rundqvist et al., 2014b).

2

Theory

In this chapter the underlying theories to the concepts used in this project are presented. First gait related theory and piezoelectric effect is explained to give an understanding of what is measured and how. In the end of the chapter a pattern recognition system is explained together with methods to evaluate such a system.

2.1 Gait Biomechanics

The human gait cycle can be divided into two main phases; the stance phase when the foot is in contact with the ground and the swing phase when the foot is in the air (Novacheck, 1998). In the work by Novacheck the gait cycle is defined to begin when one foot hits the ground (the stance phase begins) and end when the swing phase is over, which is when the same foot touches ground again.

The timing of the two phases is what differentiates walking from running. When walking, the stance phase of left and right foot are overlapping, such that both feet have contact with ground simultaneously and the stance time is above 50 % of the gait cycle. When running, left and right swing phase are overlapping such that there is a part of the cycle where no feet is in contact with the ground which means that the stance time is below 50 % of the gait cycle. (Novacheck, 1998, Mann and Hagy, 1980)

Multiple studies on gait associated parameters has been performed (Cavanagh and Lafortune, 1980, Novacheck, 1998, Mann et al., 2015). It has been shown that the step length, cadence and pace all increase as the speed of gait increases. Cadence is the number of full cycles taken within a minute, by the pair of feet, and pace is a ratio of the number of minutes it takes to cover a kilometer.

The cycle time when walking very slowly can be up to 1.5 s while the cycle time when sprinting can be as short as 0.55 s (Mann and Hagy, 1980). The stance time of the total cycle is also decreasing with speed, from around 60 % when walking to 30 % when running, and for elite sprinters it can be as short as about 20 % of the

gait cycle (Novacheck, 1998, Mann and Hagy, 1980).

The cadence can vary from 80 steps per minute when walking slowly up to 220 steps per minute when sprinting (Sundquist, 1997). The cadence can be related to pace by also analyzing step length. By increasing either step length or cadence the pace is increased.

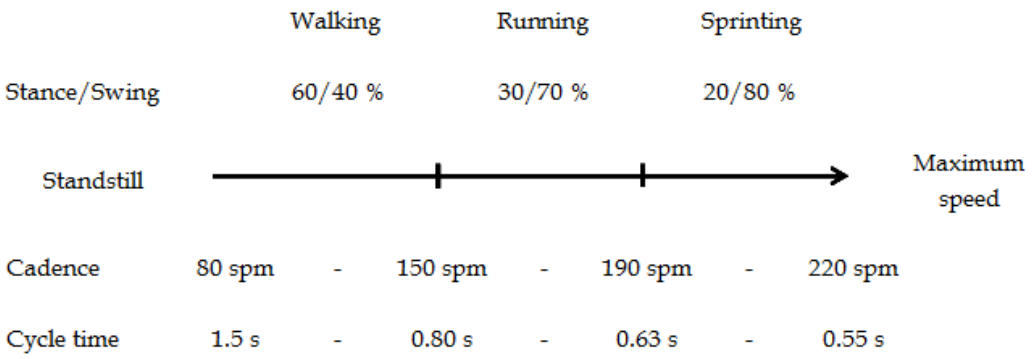


Figure 2.1: Overview of running associated parameters from walking to running.

Multiplying the cycle times in Figure 2.1 with the percentage belonging to the stance period results in a maximum stance time of 0.9 s when walking and a minimum stance time of 0.11 s when sprinting.

When discussing different types of running it is not only speed related measurements that are interesting. Ground reaction forces and the center of pressure at landing are also often used.

The foot strike pattern can be classified based on the location of the center of pressure relative to the length of the foot at initial ground contact (Cavanagh and Lafortune, 1980, Giandolini et al., 2014). An illustration of this model, called strike index, is shown in Figure 2.2. The three different types of runners are heel-, mid- and toe-strikers. In a heel-strike (HS) the heel lands first followed by a roll over the fore-foot. Mid strikers (MS) generally lands on the outside of the foot and after the initial contact both rear- and fore-foot is in contact with the ground. In a toe-strike (TS) the fore-foot lands first while the heel might land later or not at all (Daoud et al., 2012). Even though it is not necessarily so, many runners tend to shift from HS to MS or TS when the speed is increased (Keller et al., 1996).

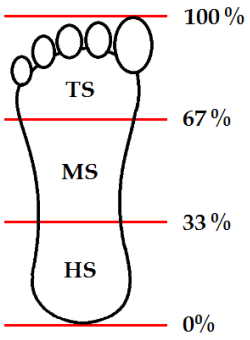


Figure 2.2: Strike index zones for classification of foot strike patterns as heel-, mid- and toe-strike (HS, MS and TS).

2.2 Gait Analysis

The possibilities to quantify the biometrics mentioned in the previous section have been studied by many researchers. Various measurement systems based on different types of data such as acceleration, knee-and wrist angles, angular velocity and ground reaction forces has been presented. The systems span from identifying gait phases to classifying foot strike patterns.

M. Saito et al. (Saito et al., 2011) developed a system with seven pressure sensors monitored in an insole. The system could monitor the plantar pressure during up to 20 hours. The resulting pressure measurements showed a clear stance- and swing phase. Though, the gait events could only be obtained by visual inspection and no automatic solution was presented.

Preece et al. (Preece et al., 2011) uses a knitted resistive strain sensor placed on the ankle to extract the salient features of ankle joint motion and thereby automatically identify gait events. Even though the signal varied between individuals, gait events could accurately be predicted. The study was only performed on walking subjects and the possibilities to use it in running is not discussed at all. Also the algorithm is not suited for usage in real time and there are according to the authors a number of individual-specific thresholds to be set.

The encouraging results in measuring gait events with textile based sensors inspired Tirosh et al. (Tirosh et al., 2013) to also develop a textile sensor sock but with the capability of long-term data capture. Although this device addresses the long-term recording, the analysis is not automated and can not be used in real-time.

Non-automatic classification of foot strike pattern has been successfully performed based on center of pressure measured by a force platform (Cavanagh and Lafortune, 1980, Dickinson et al., 1985), accelerometer (Giandolini et al., 2014) and camera

(Larson et al., 2011). Though these are mainly developed in studies performed to evaluate possible technologies but few of those are on the market today.

One solution that exists on the market is the Sensoria smart sock (Duffy, 2015, Sensoria, n.d.) Sensoria smart sock is infused with textile pressure sensors and tracks foot landing technique in real-time. The Sensoria sock is similar to the prototype used in this work but with the difference that it uses textile pressure sensors instead of piezoelectric fibres.

2.3 Piezoelectric Effect

A material with piezoelectric effect will give rise to a redistribution of the electrical charge when the material is exposed to mechanical press or pull. The word *piezo* derives from a Greek word that means press tight or squeeze (Kutz, 2015). The piezoelectric effect can arise in materials that have an electrical response to a mechanical force.

2.3.1 Piezeoelectric Fibre

The piezoelectric properties of Polyvinylidene fluoride (PVDF) were discovered already in 1969 (Kawai, 1969). The thread developed by Swerea-IVF consists of 24 piezoelectric meltspun PVDF fibres. Each fibre consists of a conducting core of carbon black and high density polyethylene, and a shell of PVDF (Rundqvist et al., 2014b).

PVDF is a polymorphic material which means that the crystallines in the polymer chain can be arranged in different forms. Depending on the crystalline form, the material is said to be in either α-,β-, γ- or δ-phase (Nilsson et al., 2013, Lund et al., 2012). The natural phase (i.e the most energetically beneficial) after melting is the α-phase, while the phase that has the highest net polarity is the β-phase (Nilsson et al., 2013). It is possible to transform the α-phase to β-phase by stretching (Lund et al., 2012).

When stretched into β-phase the backbone of the PVDF chains gets oriented along the stretching direction, though, the material is still not piezoelectric as the dipoles are randomly oriented in the plane perpendicular to the chains (Nilsson et al., 2013).

In order to get piezoelectric output voltage the fibre must have electrodes on opposing surfaces, which means in the core and on the outside of the fibre. The outer electrode should cover as much of the PVDF as possible (Nilsson et al., 2013). When both the electrodes are applied, the fibre can be poled. The process of poling means to align the dipoles in the plane perpendicular to the fibre, so that the nega-

tive charge of the dipoles point towards the inner electrode and the positive charge points towards the outer electrode (Lund et al., 2012).

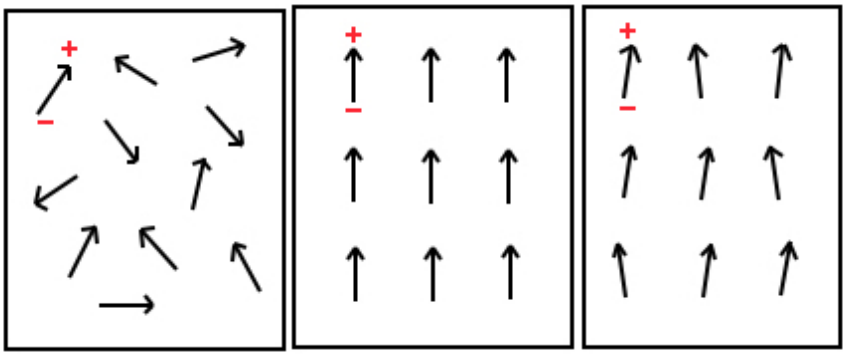


Figure 2.3: The dipole alignments in the plane perpendicular to the stretching direction before, during and after the poling process.

2.4 Pattern Recognition

There are many possible ways to setup a pattern recognition system, all depending on application. Though generally the process includes some or all of the following steps; data acquisition, preprocessing, segmentation, feature extraction, classification and post-processing (Jain et al., 2000).

2.4.1 Data Acquisition

Data acquisition is the process of measuring a physical phenomena, sampling the signal and converting the samples into numeric values ready for processing by a computer. Depending on the phenomena the signal can need more or less processing but the general stages are; sensing, signal conditioning and analog to digital conversion (Emilio Di Paolo, 2013).

2.4.2 Preprocessing

Data preprocessing comprises any type of processing performed on raw data in preparation for another processing procedure. The preprocessing concept includes a wide range of methods and tools and common elements are noise removal, baseline correction and data normalization.

2.4.3 Segmentation

There are generally three types of segmentation techniques, sliding windows, activity-based segmentation and event-based segmentation (Banos et al., 2014).

When using the first method the signal is divided into fixed-size windows. The windows can either be overlapping, non-overlapping or with a gap. This is the most common segmentation technique used in activity recognition and a wide range of window sizes has been used in previous studies. In the second method the data is partitioned based on changes in activity, which could be for example from standing still to running. The last method is based on the occurrence of specific events, which are then used to define successive segments. This method allows for non-evenly spaced segments of varying size. The method is potentially useful in gait recognition where heel-strike or toe-off might be used as the trigger event (Fida et al., 2015).

Each window, or segment, is evaluated separately and therefore the window size also determines the classification rate. A shorter window size allows for faster recognition but is more computationally demanding.

2.4.4 Classification

Classification can be described as determining to which category, in a set of categories, a new observation belongs to. An artificial neural network (ANN) is a classifier designed to mimic the human brain and its ability to solve complex perceptual problems (Jain et al., 1996). The cells in the brain processing information are the neurons. A neuron receives signals through its dendrites, processes the information in the cell body and then forwards the signal to other cells thorough an axon branching out to multiple synapses (Haykin, 2009).

In ANN the input to a neuron is multiplied with a weight simulating that some dendrites are of higher importance than others. Each neuron also contains an activation function representing the activation of biological neurons. The output of the activation function is forwarded as input to the neuron. In Figure 2.4 which shows an example network the weights are marked with w and the activation functions with f(x,w).

The neurons in ANNs are generally organized in layers which are of three different types, a single input layer, one or more hidden layers and a single output layer. Even though multiple hidden layers can be beneficial in some applications a single layer is sufficient for most input-output relationships (Bishop, 1995). The layers have different number of neurons dependent on how the data is formatted and how many output classes there are. The network in Figure 2.4 has an input layer with N neurons, one hidden layer with K neurons and an output layer with M neurons.

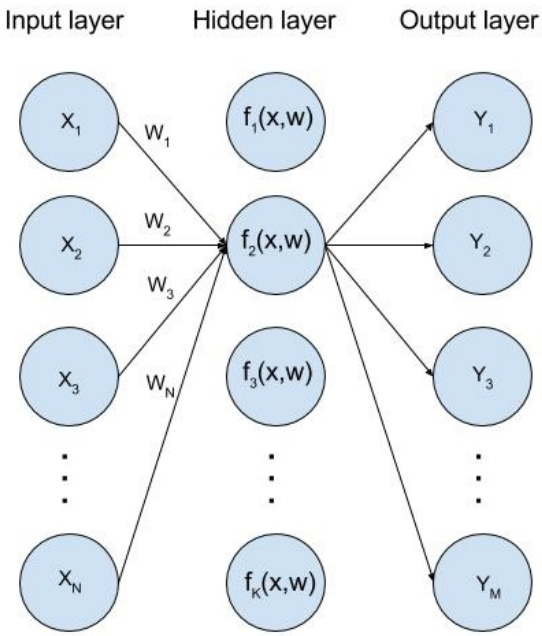


Figure 2.4: An illustration of a neural network with N input neurons, K neurons in the hidden layer and M output neurons.

Deciding the number of neurons in the hidden layer is an important design-issue with no clear answer. Too few neurons will result in underfitting while too many neurons can result in overfitting. Underfitting is when the the model cannot capture differences in the data and overfitting is when the model captures and adapts to noise in the data. Though, there are a lot of rule-of-thumb methods and many of them suggests that the number of neurons in the hidden layer should be between the number of inputs and the number of outputs (Heaton, 2008).

There exists an extensive amount of different neural networks and the appropriate design depends on the problem at hand. The different models and types have different advantages depending on the application. For pattern recognition, as is the case in this work, feed-forward networks have shown to be successful (Bishop, 1995). In a feed-forward network the signals only travels from input to output which means that the output of a neuron in a layer does not affect any neuron in the same layer.

In order to work the network has to be learned, i.e. the weights has to be tuned. How this is done divides neural networks into two subclasses; unsupervised learning models and supervised learning models (Zaknich, 1998). In unsupervised learning the input-samples are self-organized into groups according to their similarity and can be labeled in retrospect (Zaknich, 1998). In supervised learning known input-output pairs are presented to the network which iteratively self-adjusts until a predefined criterion is met.

The available input-output pairs are divided into three subsets; a training set, a validation set and a test set (Hagan et al., 1996). The training set is used to

adjust the weights in the network. The training can be done either in batch or incrementally. In batch means that all training data is used at once, and then the weights are updated, while incrementally means that the weights are updated after each presentation of an input-output pair to the network (Hagan et al., 1996).

After each iteration of training the validation set is used verify that the error is still decreasing. By verifying that the most recent adjustments also improves the performance on the validation set the risk of overfitting to noise in the training data is minimized (Bishop, 1995). Each iteration of training and validation is called an epoch. When the predefined criterion is met, the training is completed and the test set is used to confirm the actual performance of the network.

A simple algorithm that is widely used to train neural networks is standard back propagation (BP) (Le Cun et al., 1988). Although, BP has shown to have low performance on large-scaled problems which often is the case when training a neural network (Møller, 1993). For practical applications, the basic BP algorithm is often too slow (Hagan et al., 1996). For these reasons, variations of back propagation has been developed, such as numerical optimizations methods (Hagan et al., 1996). Examples of such methods are the standard conjugate gradient algorithm with line search (CGL), the one-step Broyden-Fletcher-Goldfarb-Shanno memoriless quasi-Newton algorithm (BFGS) and the scaled conjugate gradient method. These optimization methods are well suited to be used as learning algorithms, as they can handle large scaled problems (Møller, 1993). BFGS and SCG are methods developed from CGL (Møller, 1993).

Matlab provides a set of different neural networks within the *neural network toolbox*. Among these networks are pattern recognition network, feed forward network, cascade forward network, function fitting network and learning vector quantization (LVQ) network.

2.5 Performance Measures

A classifiers performance can be measured by the classification accuracy, i.e. the number of correct predictions divided by the total number of predictions.

$$\text{accuracy} = \frac{\text{true positives} + \text{true negatives}}{\text{all predictions}} \tag{2.1}$$

However, accuracy does not reveal information about the classifiers performance on a specific class, and therefore sensitivity and specificity can be used to get a more detailed view of the performance.

2.5.1 Sensitivity

The sensitivity of an algorithm, also called the hit rate, is the ratio between the number of true positives and the sum of true positives and the false negatives, as in 2.2 (Upton and Cook, 2014a).

$$\text{sensitivity} = \frac{\text{true positives}}{\text{true positives} + \text{false negative}} \tag{2.2}$$

2.5.2 Specificity

The specificity of an algorithm is the ratio between the number of true negatives and the sum of true negatives and the false positives, as in 2.3 (Upton and Cook, 2014a). In most cases, high sensitivity means lower specificity and vice versa.

$$\text{specificity} = \frac{\text{true negatives}}{\text{true negatives} + \text{false positives}} \tag{2.3}$$

2.5.3 Confusion Matrix

A confusion matrix is a type of table designed to visualize the performance of classification algorithms. Table 2.1 shows an example of a confusion matrix where the rows corresponds to the actual class and the columns shows how the algorithm has classified the instances. In this example, the algorithm can readily detect instances of class 3 but has some difficulties separating class 1 and class 2 instances. The correctly classified samples are centered in the diagonal of the matrix, making it easy to find the errors.

Table 2.1: An example of a confusion matrix with 3 classes.

		Predicted Class		
		Class 1	Class 2	Class 3
Actual Class	Class 1	10	2	0
	Class 2	3	12	0
	Class 3	0	2	15

2.5.4 Receiver Operating Characteristic (ROC) curve

A ROC curve shows the accuracy of the system in terms of false-positive rate against the true-positive rate (Upton and Cook, 2014a). An algorithm with 100% accuracy has both high sensitivity and high specificity. Figure 2.5 shows an example of a ROC-curve. The y-axis shows the sensitivity and the x-axis shows (100 - specificity). This gives a characteristic with the curve close to the upper left corner if the accuracy is high.

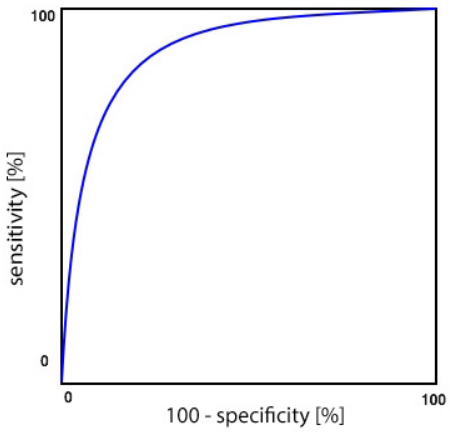


Figure 2.5: An example of a ROC-curve.

3

Method

In order to classify steps as either heel, mid or toe strike four questions were identified as critical to answer. First, does the data contain gait activity? Secondly, if it contains gait activity, where does a step start and where does it end? What part of a step differs between the three strike types? And last, how can a step be classified? A flow chart over a pattern recognition system was developed that considers these questions, the flowchart is presented in Figure 3.1 each step in the flow chart is further presented below.

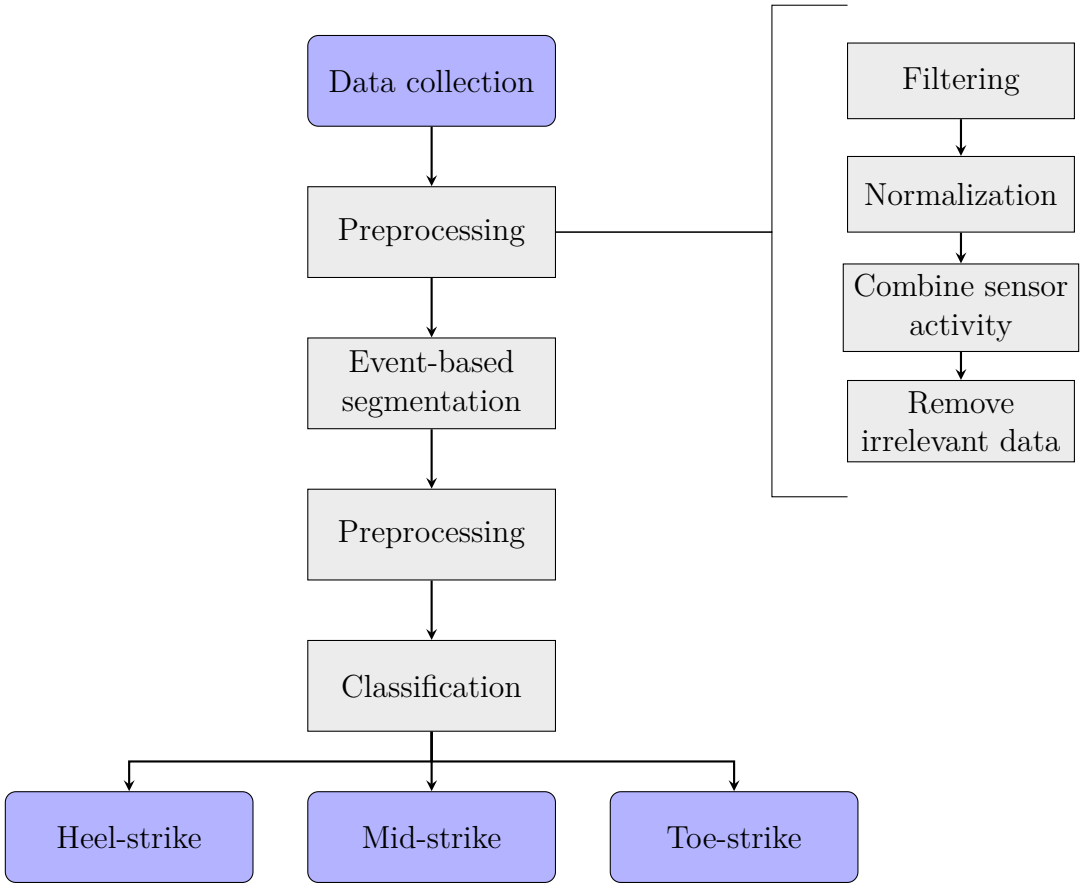


Figure 3.1: Flowchart over the developed pattern recognition system.

3.1 Data Acquisition

In order to collect data a data acquisition protocol was designed, socks were produced and a test group was formed. An occasion where one person uses one sock while following the protocol is a session.

3.1.1 Data Acquisition Protocol

Data was collected on a treadmill while following the protocol shown in Table 3.1. The protocol includes 13 one minute long intervals with speeds from slow walk to fast running. Interval 3-6 has natural pace as speed, meaning that the test person should pick a comfortable running speed that can be kept for several minutes.

Interval 4-12 are controlled intervals, with specified foot strike pattern, while interval 1-3 and 13 should be with the test persons natural step. As can be seen in Table 3.1 the three specified foot-strike patterns are heel-, mid- and toe-strike. The runner should imitate the strike-types as described in Section 2.1.

The intent of the protocol design was to get data sets with similar characteristics to be able to compare data from different logging sessions, socks and runners. For this reason, all intervals were tagged with the corresponding interval number and a class, 1, 2 or 3 corresponding to heel-, mid- or toe-strike, or 4 if neither one.

Table 3.1: Protocol used for data collection.

	Interval nbr												
	1	2	3	4	5	6	7	8	9	10	11	12	13
Speed (km/h)	4	6	natural pace	natural pace	natural pace	natural pace	14	14	14	10.5	10.5	10.5	max pace
Foot strike	natural	natural	natural	heel	toe	mid	heel	toe	mid	heel	toe	mid	natural

3.1.2 Socks

In order to be able to collect data 11 socks were produced. All socks were not produced at the same occasion but rather according as already produced socks stopped working. All socks were instrumented with one sensor in heel and one sensor in toe (positioned as in Figure 3.2), but the amount of fiber and the position of the sensing parts differs slightly between socks.

Figure 3.2: The blue circle marks the position of the heel sensor and the red circle marks the position of the toe sensor. The colors of the circles are the same ones used in plots throughout the report.

Figure 3.3 shows the socks used to collect data. Socks 1-7 are made of synthetic materials, and sock 6 and 7 are so called compression socks. Socks 8-11 are made of mostly cotton. Socks of different shape and material were used to be able to evaluate the influence of the sock characteristics on the sensor signals.

Figure 3.3: The eleven smart socks used for data collection.

3.1.3 Subjects

The characteristics of each person in the test group are presented in Table 3.2.

Table 3.2: An overview of the test group.

Runner Id	Sex	Height [cm]	weight [kg]	Age	Shoe size
A	female	171	60	23	39
B	female	178	61	26	39
C	male	189	95	27	45
D	female	174	65	24	39
E	male	171	66	24	42

3.1.4 Sessions

The resulting footstep database contains 186 minutes of walking/running data, collected by the 5 persons presented in Table 3.2, using the 11 different socks shown in Figure 3.3. Table 3.3 presents which sock, which runner and what intervals that were included in all logging sessions used to build the footstep database.

Table 3.3: A table showing which runner (A-E) performed each logging session and which intervals (1-13) that were included.

Session	Sock	Runner	Included intervals
1	2	B	1 2 3 4 5 6 7
2	8	B	1 2 3 4 5
3	2	B	1 2 3 4 5 6 7 8 9
4	6	B	1 2 3 4 5 6
5	5	B	1 2 3 4 5 6 7 8 9 10 11 12
6	6	B	10 11 12
7	5	B	7 8 9 10 11 12
8	9	A	1 2 3 4 5 6
9	2	A	1 2 3 4 5 6 7 8 9 10 11 12
10	10	A	1 2 3 4 5
11	8	A	1 2 3 4 5 6
12	2	A	1 2 3 4 5 6 7 8 9 10 11 12 13
13	6	A	1 2 3 4 5 6
14	3	A	1 2 3 4 5 6 7 8 9 10
15	7	A	1 2 3 4 5 6 7 8 9 10 11 12
16	7	A	7 8 9 10 11
17	5	A	10 11 12
18	9	D	1 2 3 4 5 6 7 8 9
19	3	E	1 2
20	7	E	1 2 3 4 5 6 7 8 9 10 11 12
21	1	B	4 5 6 7 8 10 11 12
22	1	A	4 5 6 7 8 9 10 11 12
23	1	A	4 5 6 10 11 12
24	11	C	1 2 3
25	11	C	1 2 3
26	8	C	4 5 6
27	10	C	4 5 6 7
28	10	C	1 2 3

3.2 Segmentation Preprocessing

The raw data from the heel- and toe-sensors were preprocessed in order to prepare the data for segmentation. The steps identified as necessary before segmenting the data into separate steps were filtering, amplitude normalization, combining sensor activity, and removal of irrelevant data sequences.

Filtering was necessary in order to remove signal content outside the frequency range of human gait, and for this a Butterworth band-pass filter was used to eliminate the high- and low-frequency noise. The resulting filter was designed with corner

frequencies 0.6 and 18 Hz. The lower limit was chosen based on an estimation of the lowest possible step frequency which is around 0.7 Hz. The higher limit was based on previous studies suggesting the highest frequency in foot motion data to be somewhere in the interval 15-18 Hz (Hamill et al., 1994). The filter was applied in forward and backward directions to prevent phase drifting.

Amplitude normalization is a well known method to make sensor signals of different amplitude comparable. Both the signal amplitudes were normalized according to Equation 3.1, where y is the raw signal, y' is the normalized signal and k is a scaling factor. y_{max} is the maximum value in the signal sequence that is analyzed.

$$y' = k\frac{y}{y_{max}} \tag{3.1}$$

The scale factor, k, was set to 10 to increase the absolute differences between the highest and the lowest peaks.

Detection of the start of a step should be independent on foot-strike pattern and which sensor that is active first. Therefore a method for detecting if any of the two sensors are active was developed, by combining the sensors.

Not all data collected during a logging session will contain footstep activity. Non-activity data is collected when the test person is wearing the sock but not running or walking, for instance when standing still, moving the foot in the air or jumping. It is desirable to remove such non-activity data to decrease the amount of data that is processed even though it is not actually walking or running.

3.3 Segmentation

In the segmentation only the stance phase was isolated as the ratio between stance and swing phase varies with speed and therefore the profiles of the entire step would not be comparable.

To isolate each stance phase event-based segmentation was used. The method was suitable as all stance-phases have a distinct start- and end-event. Also, the duration of the stance phase differs, disabling the use of a fixed window size.

The stance phase starts when the swing phase ends, which means that each stance is preceded by a time period where the signal amplitude theoretically should be zero. Independent on foot strike pattern, the stance phase always ends when the toes leaves ground. As toe-off results in a negative pressure difference this information can be used to identify the end of a stance phase.

3.4 Classification Preprocessing

The isolated stance-phases were preprocessed in order to prepare the segments for classification. As the duration of a stance phase is varying with cadence and speed, the segmentation will result in segments with varying length. This means that a heel-strike segment at low speed will not be similar to a heel-strike segment in high speed, and that heel-strike in low speed may be more similar to a toe-strike at low speed. The segments are therefor not suitable to use as feature space. To make the segments of a certain strike-type as similar as possible, and to differentiate segments of different strike-types, the segments were scaled in time.

To scale the segments in time, linear interpolation was used. Linear interpolation is a curve fitting method used to add new data points within an existing set of known data points. Equation 3.2 shows the formula used for linear interpolation.

$$y = y_0 \, (1 - \frac{x - x_0}{x_1 - x_0}) + y_1 \, (\frac{x - x_0}{x_1 - x_0}) \tag{3.2}$$

By choosing a point x, the corresponding value, y, is calculated as a weighted average. This means that if the point is closer to x_0 than x_1, then y_0 has more influence than y_1 and vice versa.

The longest possible stance time is 0.9 s which equals 90 samples with the sampling frequency 100 Hz. For linear interpolation, the desired number of samples was chosen not to be fewer than the longest possible stance phase.

3.5 Classification

A supervised feed-forward network was used to classify each segment as either heel-, mid-, or toe-strike. In order to prepare the data for classification the data from each isolated step was stored in a matrix, resulting in a feature space with heel, toe and summed data for all steps in the database. A matching target matrix containing either 1, 2 or 3 for each step depending on foot strike pattern (heel-, mid- or toe-strike) was also constructed. The input data (feature space and target matrix) was divided into training, validation and test sets with the distribution 70 % for training, 15 % for validation and 15% for tests. Further, the network was trained with a function that is based on the scaled conjugate gradient method. This method was used as training function as it has shown to be faster than other optimizations methods, such as BP, CGL and BFGS (Møller, 1993). *Matlabs neural network toolbox* was used to realize the supervised feed-forward network for classification of the different strike-types.

3.6 Network Design

For the classification both two and three classes were used respectively. In order to find what set of features to use and how many neurons to have in the hidden layer a series of test with different network setups was performed. The setup for each test is shown in Table 3.4 and the number of neurons was based on the rule of thumb mentioned in section 2.4.4. The result of each network was measured using both cross-entropy, mean-squared error and mean error performance measures. The sensitivity and the specificity was calculated, as well as the time it took to train each network and classify a random segment.

Table 3.4: Setup in tests of different networks. 'Heel and toe' means heel and toe data separate while 'Summed' refers to the summed heel and toe data described in Section 4.2.1

Feature space	Data points	Number of neurons									
Heel	100	10	20	30	40	50	60	70	80	90	100
Toe	100	10	20	30	40	50	60	70	80	90	100
Heel and toe	200	20	40	60	80	100	120	140	160	180	200
Summed	100	10	20	30	40	50	60	70	80	90	100
Heel, toe and summed	300	30	60	90	120	150	180	210	240	270	300

The networks in the neural network toolbox offers performance functions based on both cross-entropy and mean squared error (MSE). The cross-entropy performance function calculates the sum of the cross-entropy for each output-target pair according to Equation 3.3 based on the formula for cross-entropy (Hinton et al., 2012)

$$CE(\mathbf{T}, \mathbf{Y}) = \frac{1}{m}\frac{1}{n}\sum_{i=1}^{m}\sum_{i=1}^{n}\Big(- \mathbf{T} * log(\mathbf{Y})\Big) \tag{3.3}$$

where $\mathbf{T}$ is the target matrix, $\mathbf{Y}$ is the output matrix, m is the number of output classes and n is the number of output-target pairs.

The second performance function calculates the mean squared error (MSE) according to Equation 3.4 based on the general formula for MSE (Upton and Cook, 2014b)

$$MSE(\mathbf{T}, \mathbf{Y}) = \frac{1}{m}\frac{1}{n}\sum_{i=1}^{m}\sum_{i=1}^{n}\Big((\mathbf{T} - \mathbf{Y})^2\Big) \tag{3.4}$$

with same $\mathbf{T}$, $\mathbf{Y}$, m and n as in Equation 3.3.

3.6.1 Visualisation of the Performance

To visualize the performance of the resulting pattern recognition system a confusion matrix and a ROC curve were used.

4

Results

In this chapter the developed methodology for recognition of strike type patterns will be presented, including data acquisition, preprocessing, segmentation of stance phases, feature extraction and classification. Apart from the methodology, the results from the extraction of gait information and the classification performance will also be reported.

4.1 Data Acquisition

Only 1 of the 11 prototype socks were robust enough to collect data through the whole protocol. The endurance of the remaining 10 socks were varying, but commonly, the amplitude started to decrease at some point in the protocol, and decreased slowly until no response at all could be noticed. Most of the socks recovered within a couple of hours and could be used again but eventually all the socks stopped working. Figure 4.1 shows an example where both sensors shows this type of behaviour. From this session, only the steps from interval 1-3 was tagged and saved in the total footstep database.

Data collected by runner A during one logging session following the protocol is presented in Figure 4.2. As this data set is the only one containing data for all 13 interval this particular data set will from now on be used to illustrate the different stages of the recognition system.

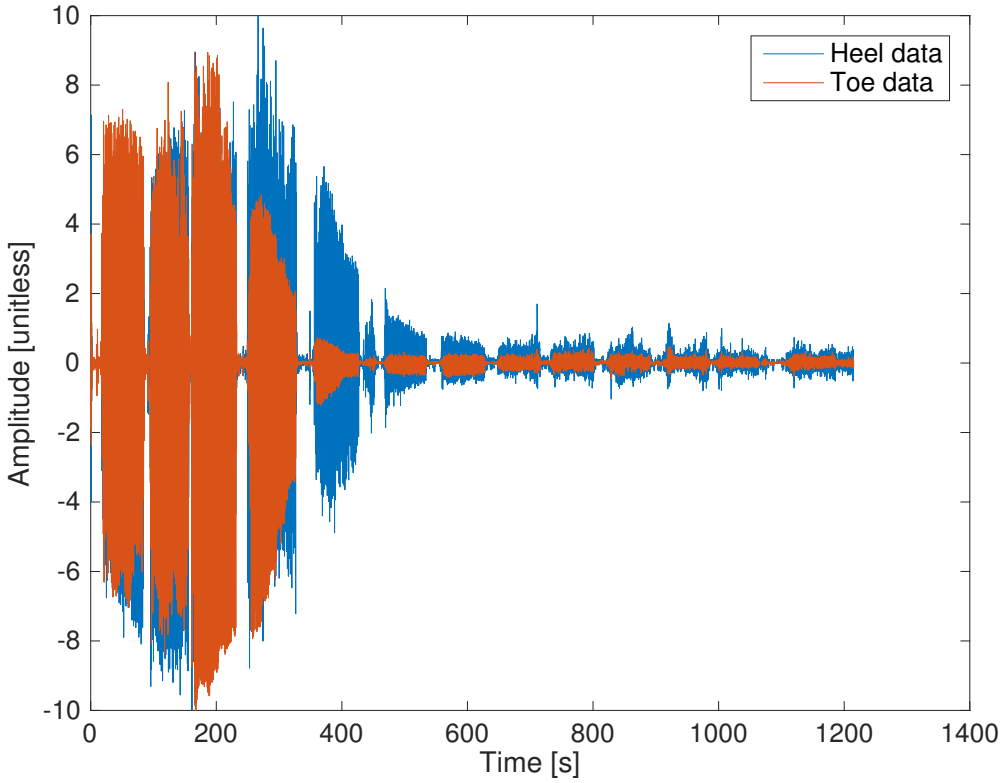


Figure 4.1: An example where the responses from both sensors decreases. The amplitude of the toe sensor (red in plot) starts to decrease after about 200 s and the amplitude of the heel sensor (blue in plot) starts to decrease after about 300 s.

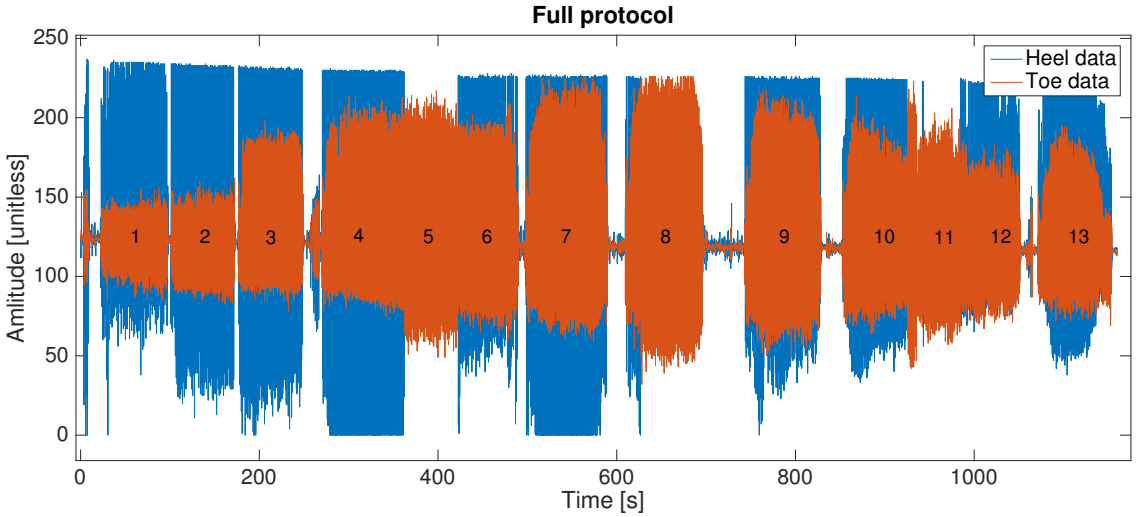


Figure 4.2: Raw data from a full protocol collected by runner A using sock number 2 . The numbers (1-13) shows which interval the data corresponds to. Heel data is blue and toe data is red.

The varying endurance made the amount of data collected with the different socks unequal. The final distribution of collected intervals (Interval nbr) over socks (Sock nbr) is presented in Table 4.1. The table shows that sock number 2 is the only sock that has given data corresponding to all 13 interval and that sock number 7 has been used most (41 intervals). In total 186 intervals were collected giving approximately 186 minutes of walking/running data.

Table 4.1: Number of intervals collected of each kind of interval number and sock.

Sock nbr \ Interval nbr	1	2	3	4	5	6	7	8	9	10	11	12	13	Sum
1	-	-	-	3	3	3	2	2	1	3	3	3	-	23
2	4	4	4	4	4	4	4	3	3	2	2	2	1	41
3	2	2	1	1	1	1	1	1	1	1	-	-	-	12
4	-	-	-	-	-	-	-	-	-	-	-	-	-	0
5	1	1	1	1	1	1	2	2	2	3	3	3	-	21
6	2	3	2	2	2	2	-	-	-	1	1	1	-	16
7	2	2	2	2	2	2	3	3	3	3	3	2	-	29
8	2	2	2	3	3	2	-	-	-	-	-	-	-	14
9	2	2	2	2	2	2	1	1	1	-	-	-	-	15
10	-	-	-	1	1	1	1	-	-	-	-	-	-	4
11	3	3	3	1	1	-	-	-	-	-	-	-	-	11
Sum	18	19	17	20	20	18	14	12	11	13	12	11	1	186

4.1.1 Data Analysis

A time plot of the raw data reveals that both the heel and the toe signal contains information that can be related to the events of a foot strike. The most significant events are marked with a black dot in Figure 4.3; initial heel and toe contact when heel or toe hits ground, maximum heel and toe contact as well as heel- and toe off when heel and toe leaves ground. The order of heel contact and toe contact is dependent on what foot strike pattern the runner has.

The step in 4.3 is a clear heel-strike, as both initial heel contact and maximum heel load occurs before toe contact and maximum toe load.

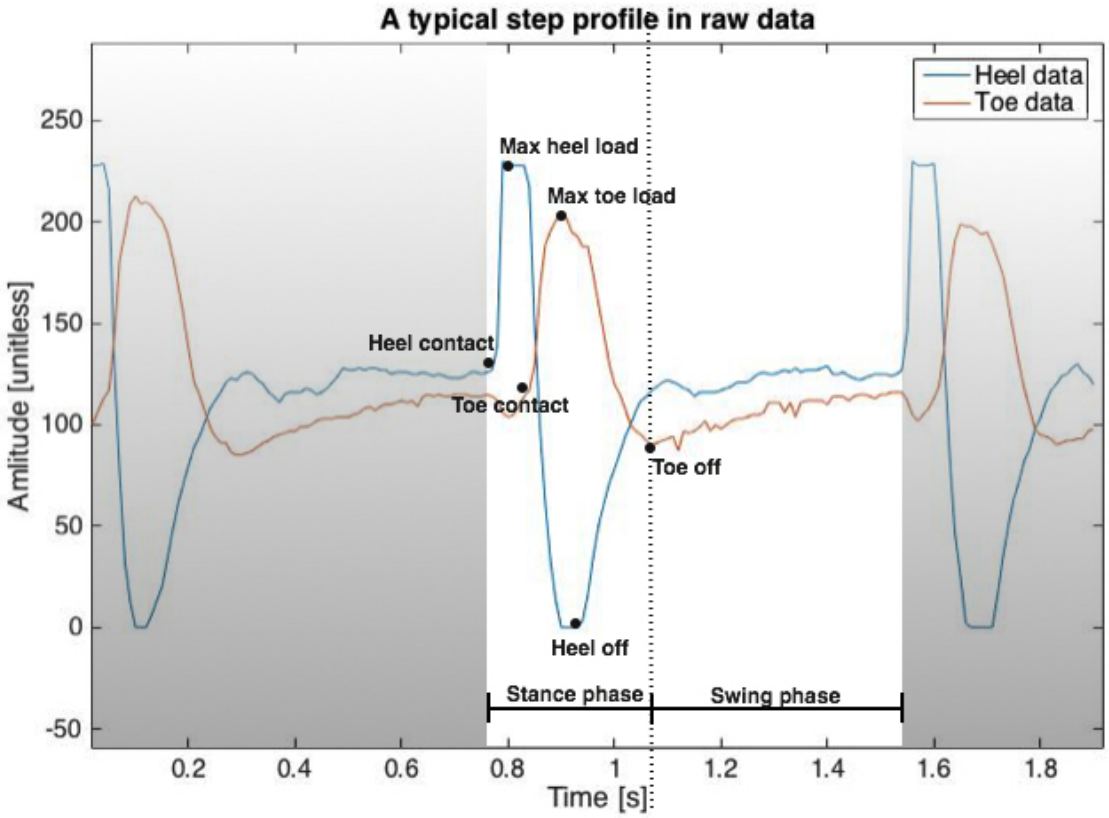


Figure 4.3: A single heel-strike profile with key events in the heel and toe data marked with a black dot.

The analysis also showed differences in the data depending on which sock is used, which person is wearing a certain sock and which logging session a certain sock is used by one person. Figure 4.4-4.6 indicate three situations with notable differences.

Figure 4.4 shows data collected by runner A with two different socks. The running speed is 10.5 km/h and from top to bottom is heel-, toe- and mid-strike steps. To the left is data from sock number 2 and to the right data from sock number 7. Sock number 7 is a compression sock while sock 2 is less tight on the foot. The figure shows that both heel and toe amplitude is slightly lower for sock 7. Sock 2 contains more noise in the toe signal than sock 7 toe signal, while sock 7 contains more noise in the heel signal than sock 2 heel signal.

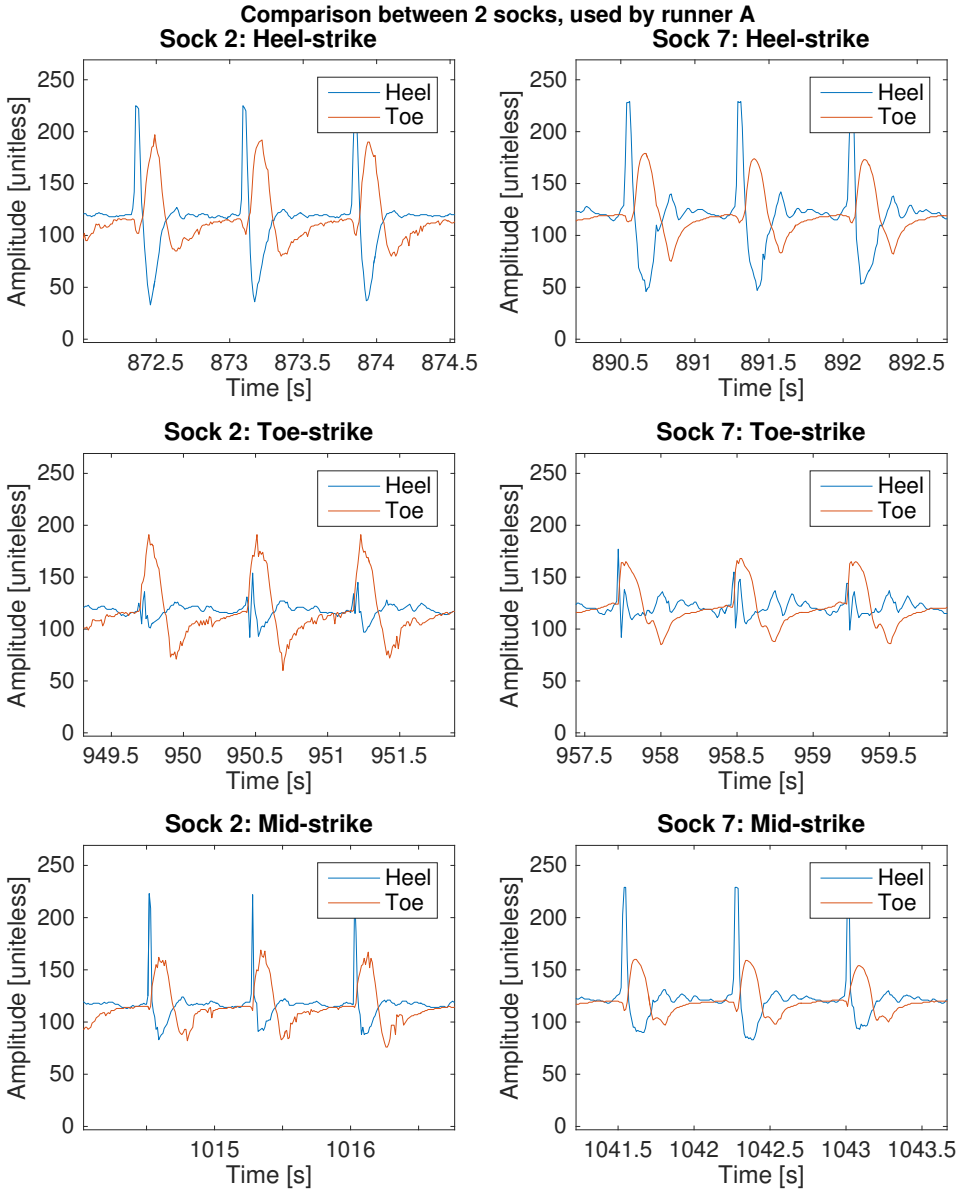


Figure 4.4: Comparison of data from two socks (2,7) used by the same runner (A).

Figure 4.5 shows data collected by runners A and B using the same sock. The running speed is 10.5 km/h and from top to bottom is heel, toe and mid-strike. Runner A and runner B has the same length, weight and shoe size. Even so the difference in amplitude is notable in both heel and toe sensors.

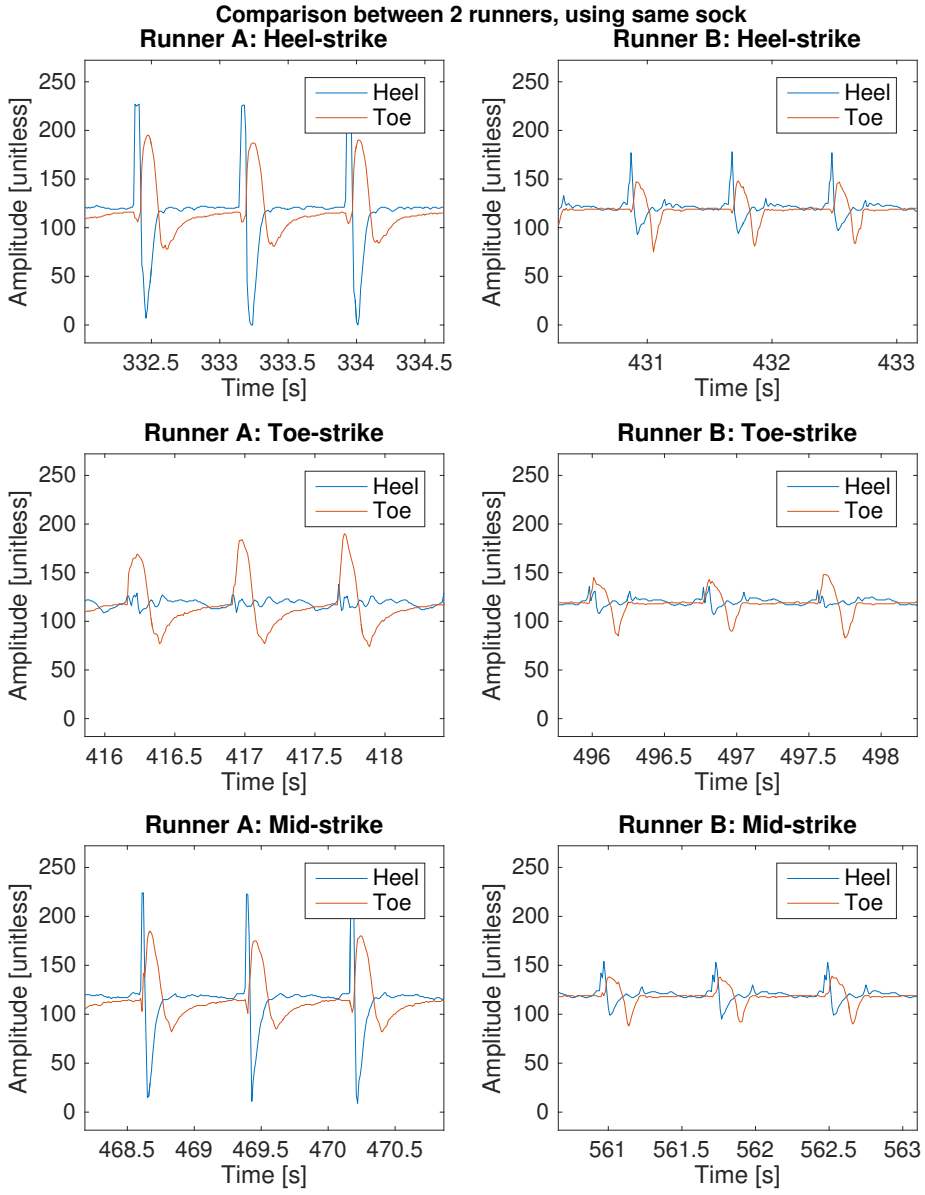


Figure 4.5: Comparison of data from the same sock (1) used by runner A and B.

Figure 4.6 shows data collected by runner B, with the same sock but from different logging sessions. The pace is 14 km/h and from top to bottom is heel-, toe- and mid-strike. The data in the right plots are collected one week later than the plots to the left. The differences are quite large even though the test setup is the same for both sessions. The signals from session 7 is less noisy, indicating that the sensitivity has decreased.

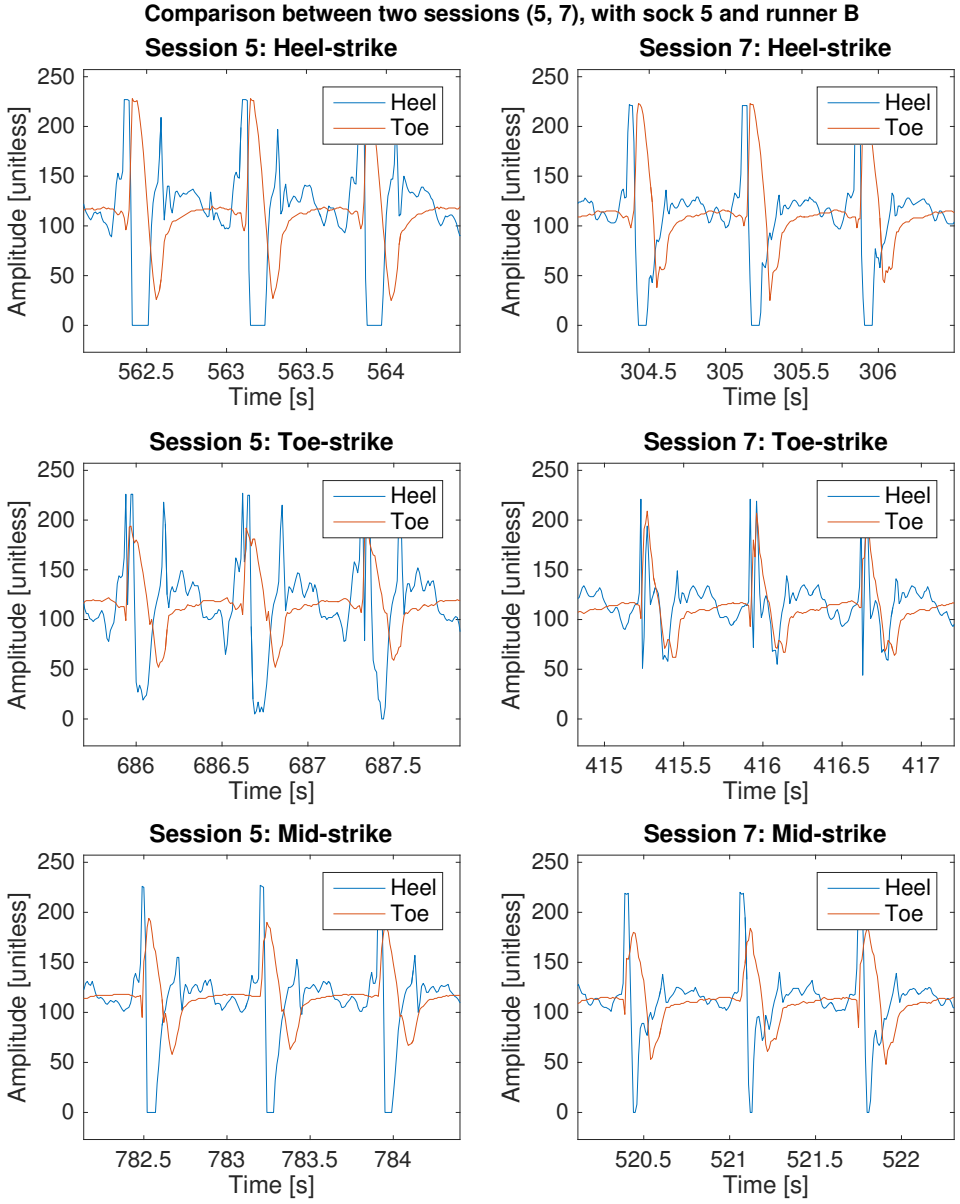


Figure 4.6: Comparison of data from the same sock (5) and runner (B), from two different logging sessions.

4.2 Preprocessing

The signal preprocessing included filtering, amplitude normalization, a summation and squaring of the signals and removal of irrelevant data. The two first parts were solved using well-known methods described in Section 3.2 while the last two parts are methods developed in this project and will be described more in detail in this Section.

Figure 4.7 shows the data after filtering and amplitude normalization. The baseline is straight and the data is centered around zero.

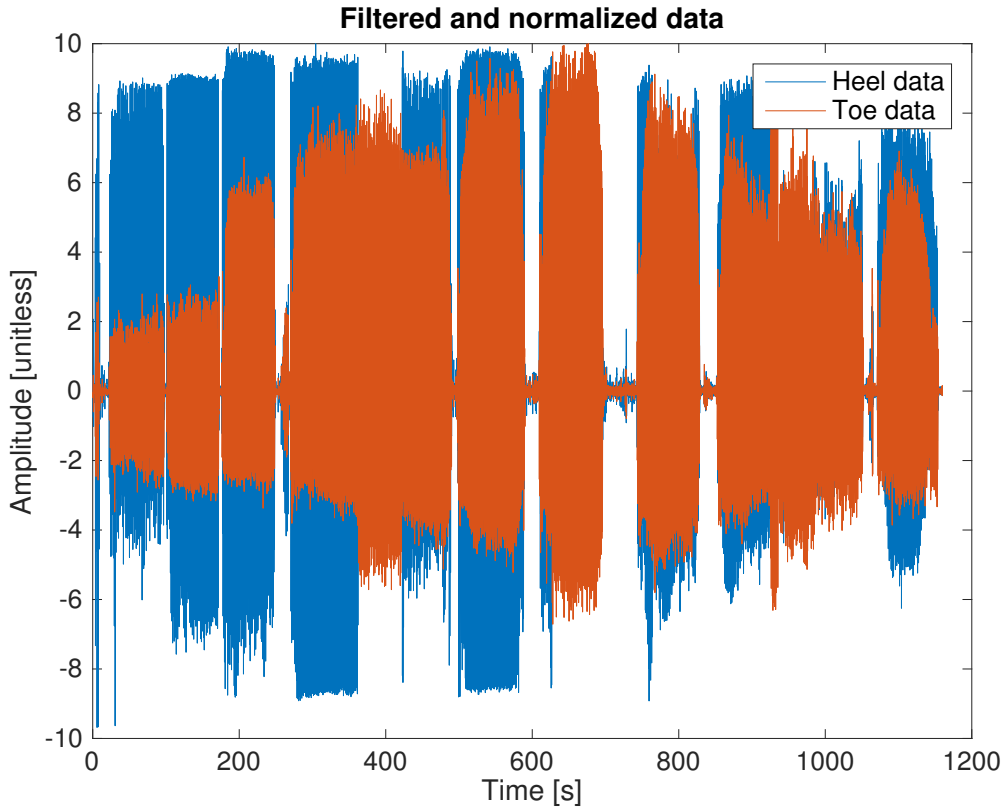


Figure 4.7: Same data sequence as in Figure 4.2 but after filtering and normalization. The baseline is straight and data is in the interval [-10, 10]. Blue line is heel data and red line is toe data.

4.2.1 Sum and Square Signal from Heel and Toe

To find when any of the two sensors were active the absolute values of the filtered and normalised heel and toe data were summed into one signal which was then squared. The square was used to enhance peaks and suppress the noise at zero-level which

made stance and swing phases even more distinct. Figure 4.8 shows the signal after this step, and Figure 4.9 shows different steps when zooming in, this is to show that the differences between steps are distinct also after the summation of the signals.

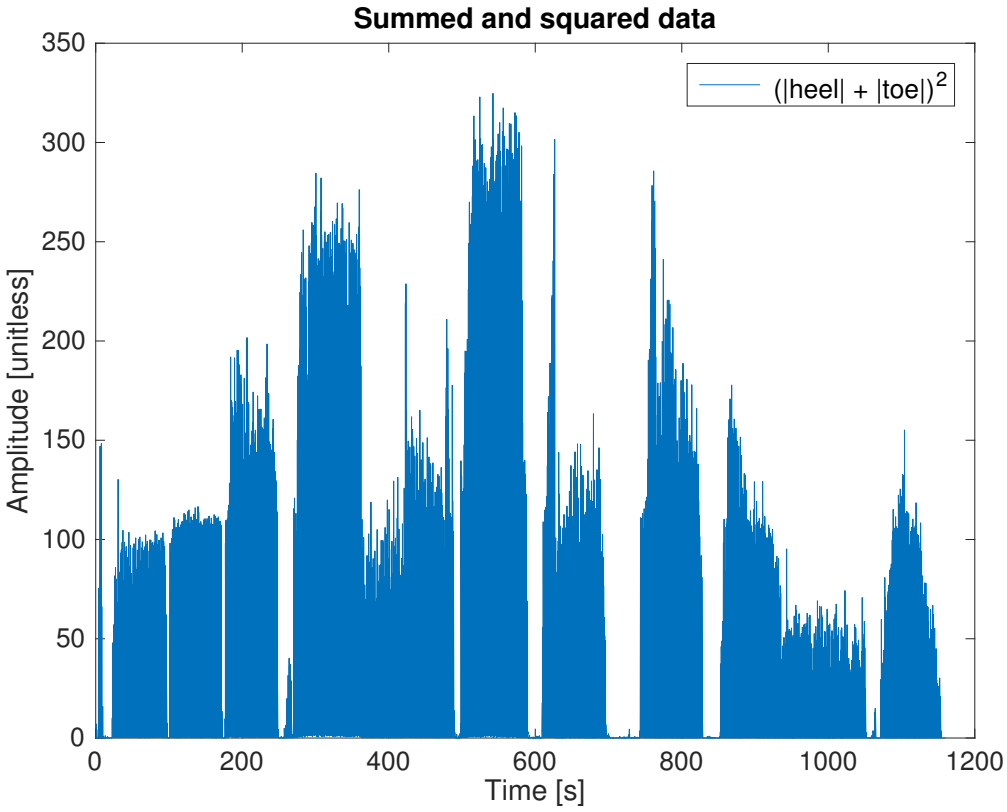


Figure 4.8: Summation and squaring of the absolute value of the data showed in Figure 4.7.

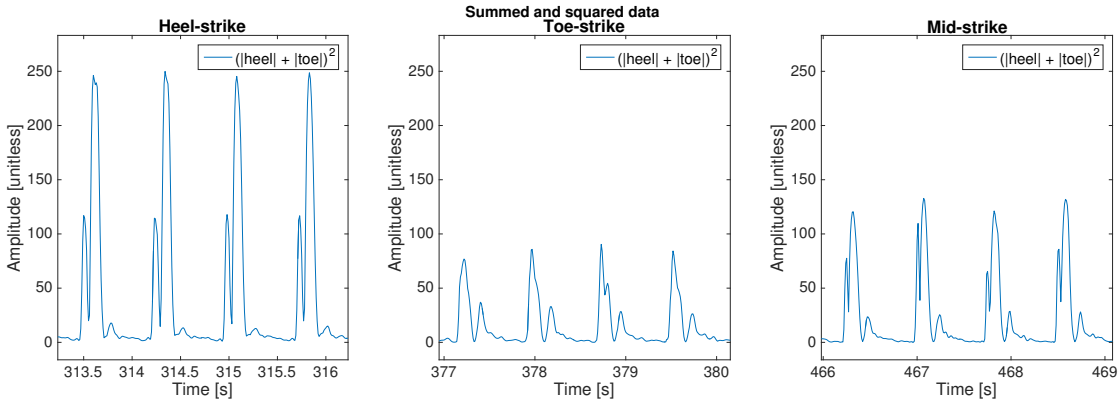


Figure 4.9: Zoom in on the summed and squared data in Figure 4.8. From left to right are parts of the data with heel-, toe- and mid-strikes respectively.

4.2.2 Removal of Non-Activity Data

The identification of non-activity data was achieved by analyzing the periodicity in consecutive data windows. The window size was set to 3 seconds with an overlap of 1 second. The periodicity is depending on the step frequency and it was found by autocorrelating the signal with the Matlab function `xcorr`. The autocorrelation of a signal shows how many cycles the signal is built up by and how long the duration of each cycle is. A pure sine-wave only has one cycle, but in the case of running or walking data, the signals contains multiple cycles with different duration. The sequences containing gait activity will have periodicity within the duration interval presented in Table 2.1. The period times during walking or running varies between 0.55 s and 1.5 s. After removing all data that does not contain periodicity within the interval, only the data containing foot step activity remained.

The left plot in Figure 4.10 shows the data before removal and the right plot shows the data after the removal. Most parts of the data where the runner is having a pause is removed. One of the pauses was not removed due to that the periodicity of the noise was within the interval of foot step activity. In this data set about 160 seconds of data was removed.

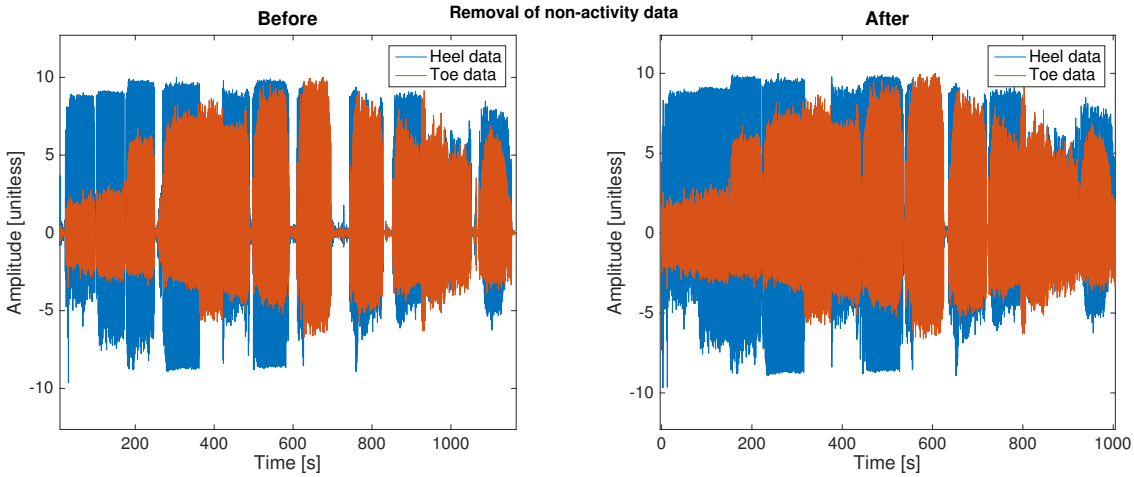


Figure 4.10: Left plot shows data all logged data and right plot shows data after the removal of non-activity parts. Blue lines are heel data and red lines are toe data.

4.3 Segmentation of Stance Phases

The segmentation of stance phases showed to be an extensive task and rendered a solution with four parts; peak detection, detecting start of the stance phase, detecting the end of the stance phase and extracting segments. In this section each part of the solution will be presented in detail.

4.3.1 Peak Detection

Peak detection is used to identify each separate step since all steps have at least one peak. The Matlab function `findpeaks` is used to locate all peaks. This function detect all positive local maximas which gives multiple peaks per stance phase. By using a minimum allowed distance between peaks, peaks belonging to the same step is sorted out and only the most prominent peak per step is saved. To be sensitive to variations in speed, the shortest allowed distance is based on the periodicity calculated in the activity-based segmentation. The periodicity was calculated with a resolution of 1 second, meaning that the minimum allowed distance is updated every second to be adjusted after the current speed.

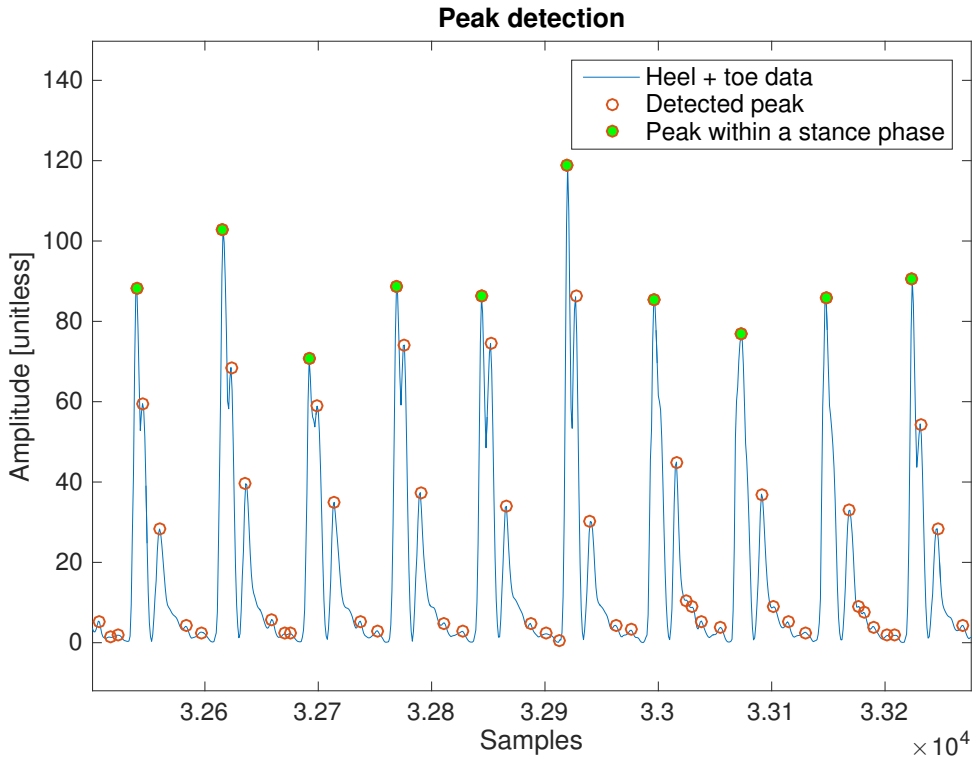


Figure 4.11: Figure showing the detected peaks and the sorted step-peaks, with each step-peak marked green. The blue line is the summed and squared signal (described in Section 4.2.1).

4.3.2 Detecting the Start of the Stance Phase

The shift from swing to stance could be identified as when the derivative of the signal changes. As the sensors ideally should give no response during the swing phase, both the amplitude and the derivative of the signal should be zero. The derivative should then start to increase as the stance phase begins.

By moving backwards from each detected step-peak until the amplitude and derivative was under a certain limit for at least two consecutive samples the start of a stance could be identified. To make the function robust and adaptive towards differences in the data sets, the limit was based on the signals standard deviation which showed best practice.

Since either heel or toe can hit the ground first, the slope detection function was applied to the summed and squared signals. The detected start points in a small part of the protocol can be seen in Figure 4.12.

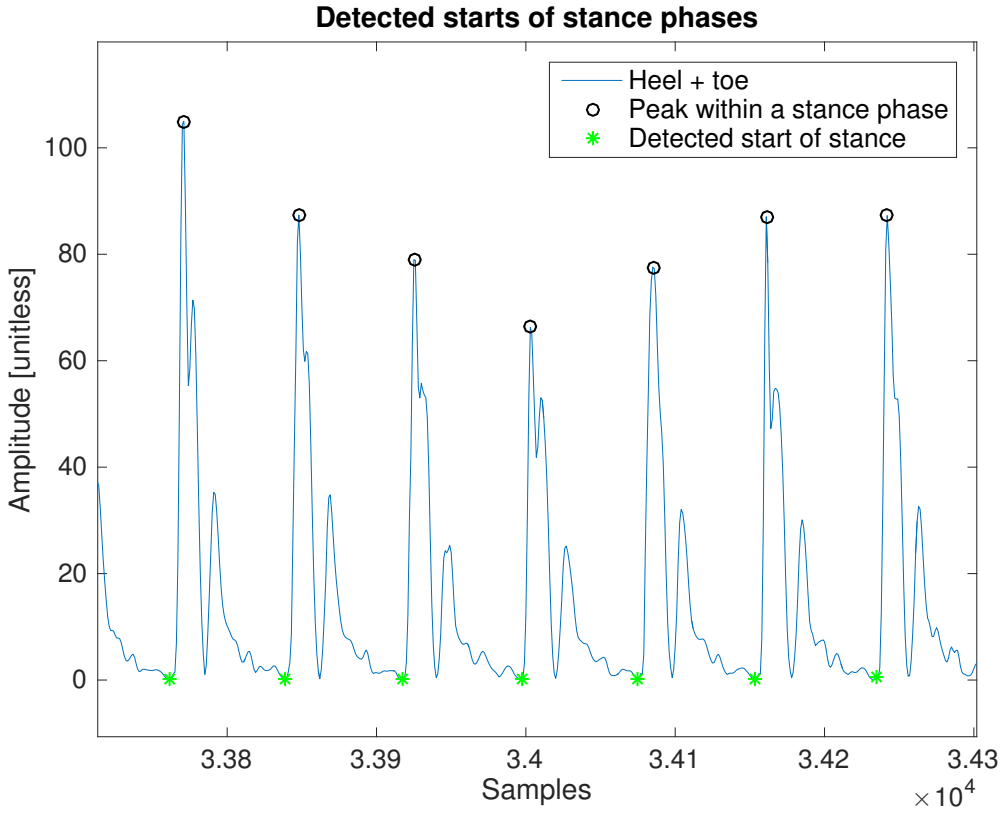


Figure 4.12: Figure showing the step-peaks (marked as black) and the detected start points for the stance phases (marked as green). The blue line is the summed and squared signal (described in Section 4.2.1).

4.3.3 Detecting the End of the Stance Phase

The toe is the part of the foot that leaves ground last when running or walking, independent of the foot strike pattern. Toe-off results in a negative pressure difference and can therefore be identified as a negative peak in the toe data. After the summation of the signals, this point is not always distinct, which is why the toe data is used to identify the end points. As the filtered data is centered around zero, the toe data was inverted by multiplying with (-1) and fed into the Matlab function `findpeaks` to locate negative peaks.

To ensure that the identified peak matches the start of stance a requirement that the ends should be within 0.11 to 0.9 s after the start of stance was added. Figure 4.13 shows a part of the protocol with located end points together with matching start points.

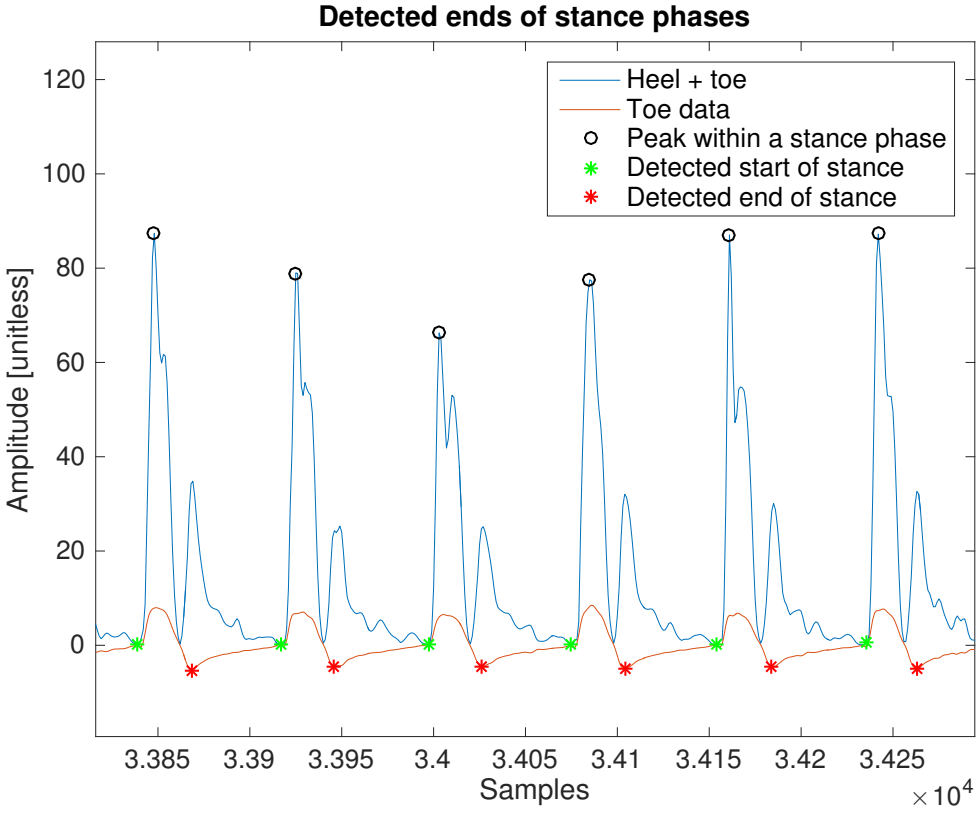


Figure 4.13: Figure showing the detected end of each stance phase, marked as red. The green marks are the start of the stance phases. The blue line is the summed and squared signal (described in Section 4.2.1).

4.3.4 Isolate All Segments

When the start and end of all stance phases are detected, the indices are used to extract the corresponding segments from the heel and toe-data. For each start of stance index, all higher end of stance indices are sorted and the smallest is picked as a match. If the difference between the start and the matching end index is larger than 11 and smaller than 90 a segment with the data between the two indices is saved as a segment. In the full protocol (presented in Figure 4.2) 1093 steps were isolated, which can be seen in Figure 4.14. No clear profile can be seen as all segments have different length.

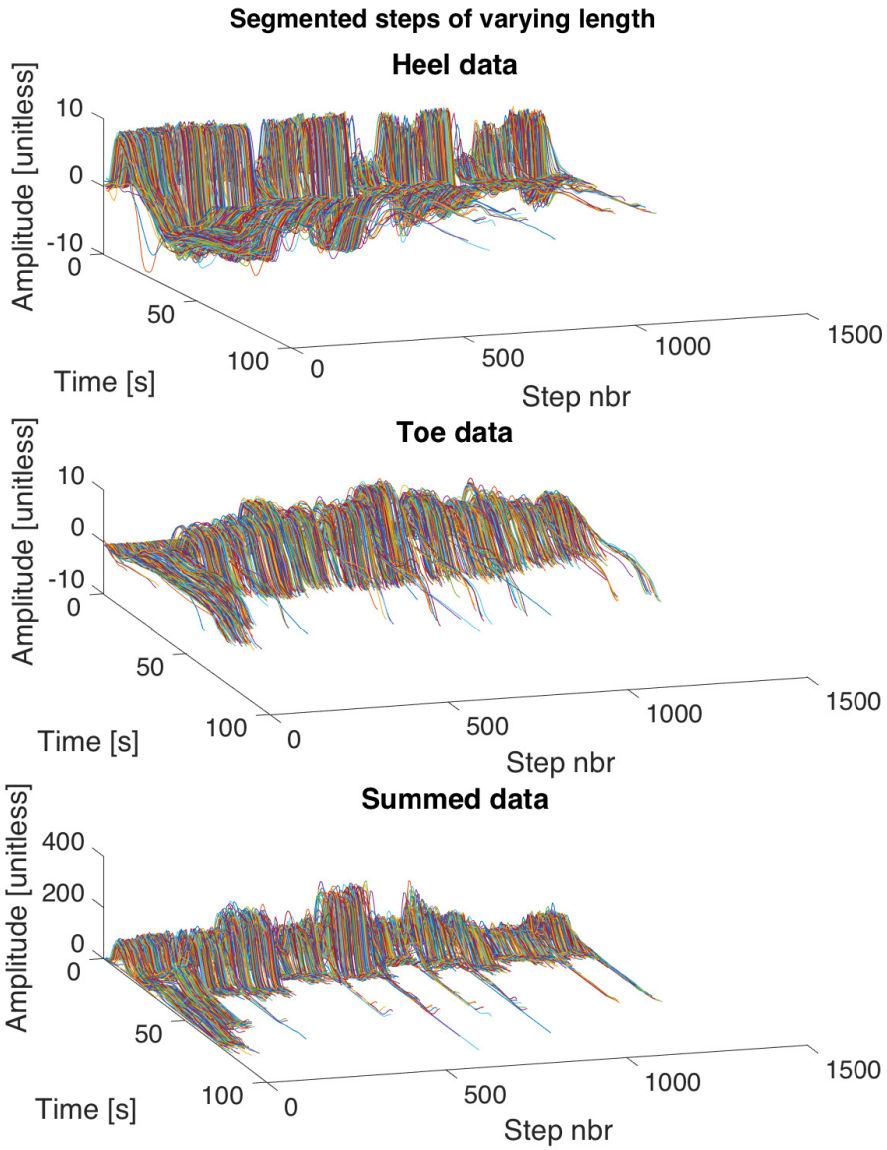


Figure 4.14: The extracted segments with varying length. From top to bottom is heel data, toe data and summed data for all 13 intervals.

4.4 Gait Analysis

The information about the extracted segments does also give other interesting running-related parameters. The amount of segments extracted indicates the number of steps taken during the logging session. The length of each segment indicates the duration of the stance phase. Combining the amount of segments and the length of the segments gives the mean cadence while running or walking during the logging session. These values were calculated for each interval in the full protocol (presented in Figure 4.2) and are shown in Table 4.2. In accordance with the theory (Section 2.1) the cadence increases while the stance time decreases with speed.

Table 4.2: Calculated running parameters for each interval in the data set presented in Figure 4.2

Interval Nbr	Duration [s]	Nbr Steps	Mean Cadence [steps/min]	Mean Stance Time [s]	Standard Deviation of Stance Times [s]	Min Stance Time [s]	Max Stance Time [s]	Mean Step Cycle Time [s]
1	61.40	29	56.68	0.75	0.02	0.71	0.79	1.14
2	63.51	27	51.02	0.61	0.02	0.58	0.67	0.95
3	60.31	79	157.21	0.34	0.02	0.29	0.38	0.75
4	86.64	115	159.35	0.36	0.03	0.28	0.42	0.74
5	60.11	72	143.76	0.33	0.06	0.28	0.50	0.77
6	61.96	78	151.21	0.29	0.02	0.25	0.35	0.76
7	67.11	96	171.68	0.29	0.02	0.24	0.33	0.69
8	59.31	86	174.03	0.26	0.03	0.22	0.40	0.68
9	69.11	99	171.92	0.27	0.03	0.23	0.33	0.68
10	65.21	85	156.44	0.32	0.02	0.28	0.36	0.75
11	59.81	80	160.54	0.27	0.01	0.24	0.31	0.73
12	64.01	85	159.38	0.28	0.02	0.24	0.34	0.73
13	61.01	91	179.02	0.26	0.02	0.21	0.32	0.65

4.5 Evaluation of Segmentation Method

The principal approach for evaluation of the segmentation has been by visual inspection of the identified stance start and end indices in the data. The main findings is that the method works well for most intervals and most often also in the transition between subsequent intervals. The only real drawback has been when walking in 4 km/h where the segmentation does not work as well for all socks. Examples of the different situations can be seen in Appendix B.

To further evaluate how well the segmentation performs the number of detected steps and missed steps in each logging session was estimated. The time distance between each segment was calculated by finding the time difference between the starting points of two subsequent segments. A step was considered as missed if the time difference was bigger than the longest possible step time, which is 1.5 s.

All segments were considered as subsequent, independent of the signal content in between the starts of two segments. This assumption means that if two segments are separated by a pause or transition steps, the time difference will result in an additional missed step. This also means that if there are more than one missed step in between the segments, these missed step will only be counted as one.

For the reasons stated above, this method of estimating the hit rate does not give the absolute number of missed steps but rather an indication of the performance. Table 4.3 shows the outcome of the evaluation method for each logging session. Session 28 has the lowest hit rate with only 87 % while session 6 has the highest hit rate with 100 % detected steps. All sessions with hit rate below 95 % includes interval number one (see Table 3.3) and all sessions where interval one is excluded has a hit rate above 98.4 % which aligns with the poor results observed for the segmentation of 4 km/h intervals.

Table 4.3: A table showing the hit rate of the segmentation algorithm for each logging session.

Session	Sock	Number of detected steps	Number of missed steps	Number of intervals	Hit rate [%]
1	2	510	7	7	98.7
2	8	313	8	5	97.5
3	2	638	12	9	98.2
4	6	400	10	6	97.6
5	5	810	20	12	97.6
6	6	223	0	3	100
7	5	447	7	6	98.5
8	9	429	6	6	98.6
9	2	971	11	12	98.9
10	10	363	4	5	98.9
11	8	397	6	6	98.5
12	2	1093	13	13	98.6
13	6	341	20	6	94.5
14	3	632	27	10	95.5
15	7	880	33	12	96.4
16	7	406	6	5	98.5
17	5	242	1	3	99.6
18	9	612	30	9	95.3
19	3	120	3	2	97.6
20	7	864	50	12	94.5
21	1	734	10	9	98.7
22	1	816	7	10	99.2
23	1	900	13	7	98.6
24	11	149	2	3	98.7
25	11	188	2	3	94.0
26	8	225	1	3	99.6
27	10	290	3	4	99.0
28	10	128	19	3	87.1

4.6 Classification Preprocessing

The segmented sequences of unequal length were scaled in time by linear interpolation and the result of time scaling the full protocoll (Figure4.2) can be seen in Figure 4.15. The profiles are now comparable and the segments can be used as feature space for classification.

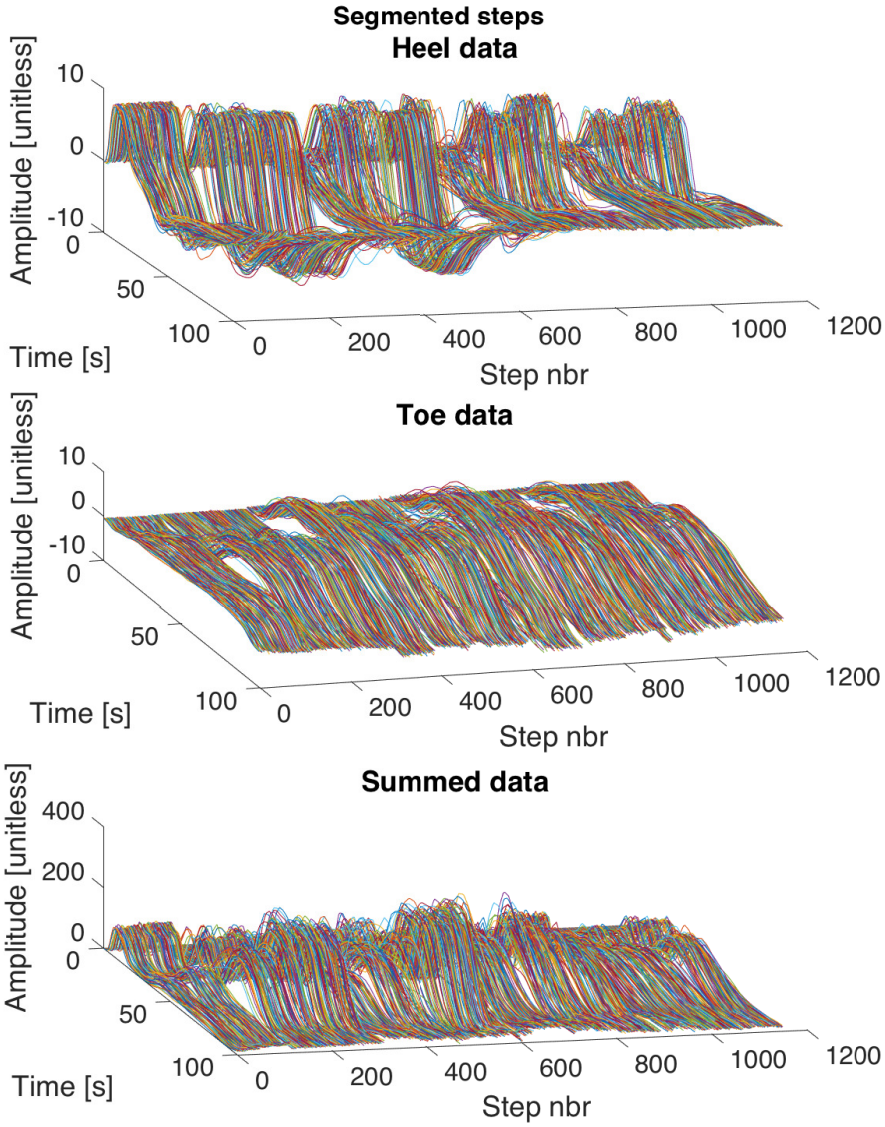


Figure 4.15: The segmented steps after the time scaling. From top to bottom is heel data, toe data and summed data for all 13 intervals.

4.7 Classification of Foot-Strike Pattern

In this subsection the results from training two different networks are presented. The first network has three classes while the second network has two classes. After removing all intervals not tagged as either heel-, mid- or toe-strike (see Table 3.1) from the step database described in Section 4.1 a total of 11856 steps remained after segmentation. Among these were 5158 heel-strikes, 3084 mid-strikes and 3369 toe-strikes. In the second network only heel- and toe-strikes were used and therefore all intervals tagged as mid-strike was also removed resulting in a total of 8464 steps.

With the aim to classify heel- toe- and mid-strike patterns, the neural network resulting in best performance takes 300 data points as features in the input layer, has one hidden layer with 140 neurons and gives 3 types of outputs in the output layer. The 300 input points is built up by one time scaled segment, with 100 points from heel data, 100 points from toe data and 100 points from the summed data.

The resulting confusion matrix for this neural network is showed in Figure 4.16.The plot shows that the network classifies the three classes with a hit rate of 97 % on the test data set. It also shows that when testing, the mid-strikes are falsely classified in 5.3 % of the testing mid-strikes, while toe-strike only fails with 2.8 % and heel-strike fails with 1.9 %.

Figure 4.16: The confusion matrix for the resulting network. One confusion matrix per data set divided by the network. The "all confusion matrix" is a combination of the other three confusion matrices.

The sensitivity and specificity are presented in Table 4.4. The corresponding ROC-curve is showed in Figure 4.17, where high sensitivity and high specificity are seen as the curves are close to the upper left corner.

Class	Foot-strike	Sensitivity	Specificity
1	Heel-strike	0.9886	0.9921
2	Mid-strike	0.9815	0.9903
3	Toe-strike	0.9877	0.9970

Table 4.4: Sensitivity and specificity for the resulting network.

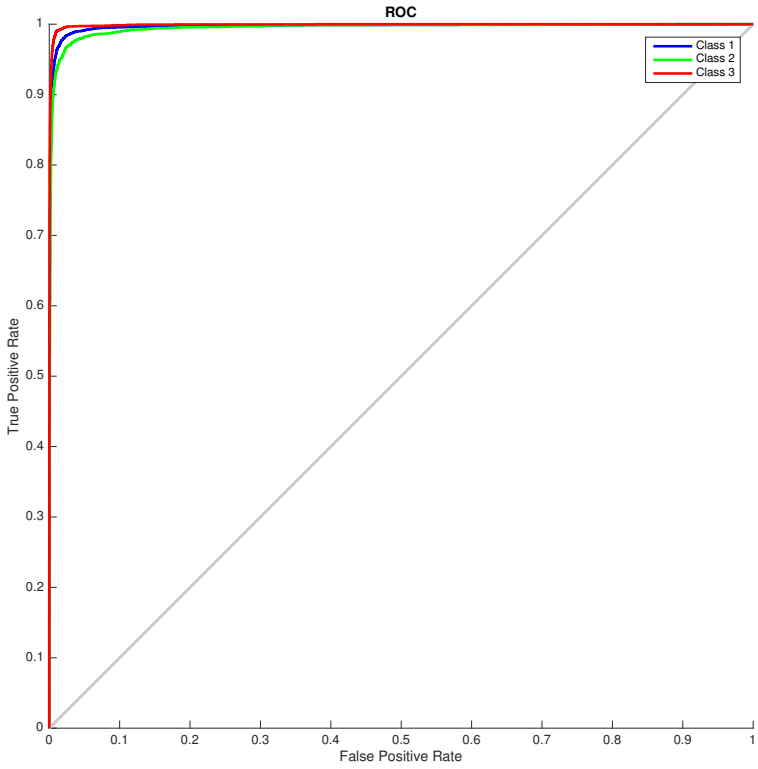


Figure 4.17: The ROC curve for the resulting network using three outputs.

The results of the evaluation of other network settings can be found in Appendix A.

When only using data tagged as heel- or toe-strike, the best performing network has other characteristics. It also takes 300 data points as features in the input layer, but it has 60 neurons in the hidden layer and gives 2 outputs per input.

The corresponding confusion matrix is shown in Figure 4.18. The plot shows that the network classifies the two classes with a hit rate of 98.7 % on the testing data. It can also be seen that the toe-strike classification fails with 0.8 % of the testing toe-strikes and the heel-strike classification fails with 1.6 %.

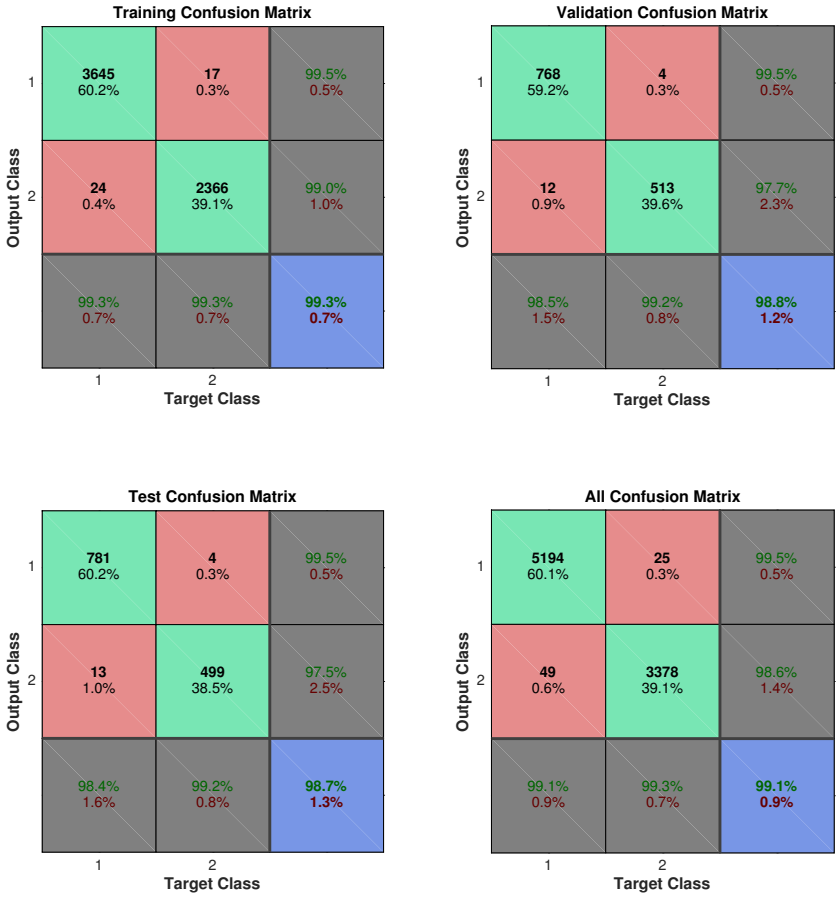


Figure 4.18: The confusion matrix for the network using only heel and toe as target classes.

The sensitivity and specificity are presented in Table 4.5. The corresponding ROC-curve is showed in Figure 4.19, where high sensitivity and high specificity is seen as the curves are close to the upper left corner.

Table 4.5: Sensitivity and specificity for the network where 2 classes (heel and toe) were used.

Class	Foot-strike	Sensitivity	Specificity
1	Heel-strike	0.9952	0.9857
2	Toe-strike	0.9857	0.9952

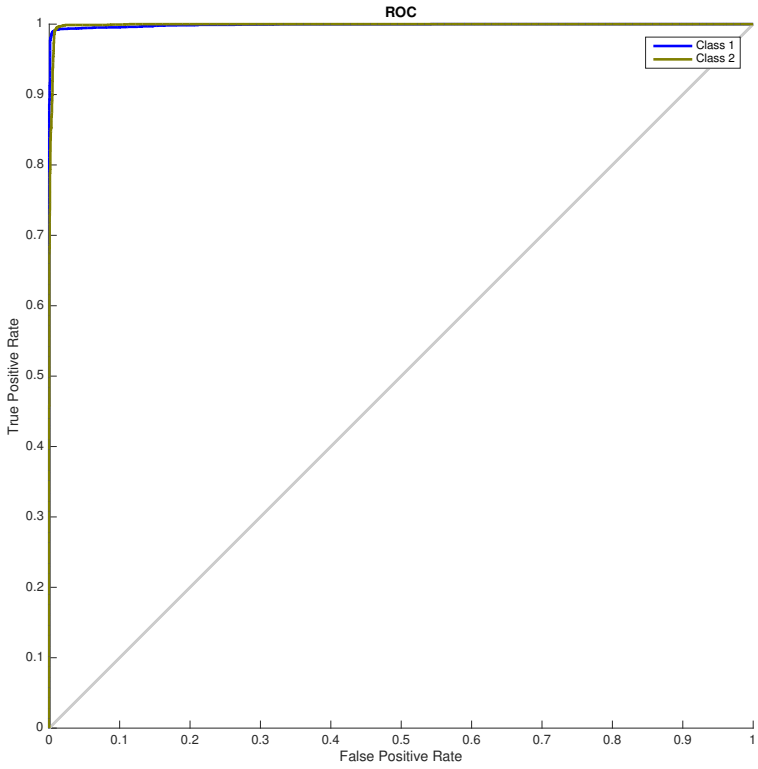


Figure 4.19: The ROC curve for the network using only heel and toe as target classes.

5

Discussion

In this study we have shown that with the use of a smart sock instrumented with textile piezoelectric sensors, foot-strike patterns can be correctly classified with a hit rate of up to 98 % and that other gait information such as cadence, stance time and cycle time can be provided. This consolidates the conclusions by Sandsjö et al (2014) that the sock can be used to provide information about foot-strike timing.

Though the results are very promising, as of today the signals from the textile piezoelectric sensors can only be used in a very generalized way. Small variations in the signals can not be connected to the person using the sock but rather to the sock itself. If a sock was to be used for rehabilitation in connection with a surgery for example, one would not be able to know if differences were dependent on the patients progress or a variation in the fitting of the sock. Also limiting the possible areas of use is the problem with socks being different from each other and the somewhat unpredictable function. If a sock stops working one can not simply replace it by another one without the risk of getting slightly different signals. If wanting to further develop the software to extract other gait information this might have an impact.

5.1 Data Acquisition

As presented in Section 4.1.1 the data analysis showed that all socks are individual giving data with different characteristics depending on the amount of piezoelectric fibre and the outcome of the poling process. In addition to this, each specific sock also gives different type of data depending on how well the sock is fitted under the foot and on the duration of the logging session. The collected database used for development and tests is not wide enough to conclude that the system can handle differences that may appear if the system is tested on a larger and less homogeneous group.

In all data sets where the sensor amplitude decreases before the logging session ends shows the same pattern; the toe sensor has lower endurance than the heel

sensor. A theory for this behaviour is that the toe sensor is further away from the connecting buttons. This increases the uncertainty since the fibre and conductive thread needs to be longer, which in turn means that charges has to travel a longer distance. The 24 filaments in the fibre thread are delicate and the longer thread is being used, the higher is the risk of breakage. Another theory for the decreasing amplitude behaviour is the occurrence of sweat. As the logging session proceeds the sock gets sweatier and sweatier and the sweat works as an electrolyte which enables the charges to travel in other ways than through the conductive thread.

5.2 Segmentation

The overall aim of the work was to classify foot-strike patterns, and to do so, finding step profiles of high quality has been a focal point. Though much work has been put into optimizing the segmentation, the evaluation of this method is somewhat imperfect. Some sessions have pauses between each interval while some sessions have no pauses at all. Since pauses result in an additional missed step, this way of counting missed steps will be unfair. To have a more fair evaluation method, which intervals as well as number of pauses should be considered as additional parameters. As described in Section 4.3.4 Sock 6 gave a 100 % hit rate in one of the logging sessions. This was possible as the protocol does not contain any pauses between the intervals. Also, this protocol only contains intervals with a speed of 10.5 km/h.

As stated in Section 4.3.4 the segmentation hit rate of 4 km/h intervals is often lower than for intervals with higher speed. This might be due to the higher sensitivity in the piezoelectric fibres in the beginning of a logging session, which also increases the noise. This may have been resolved by using a randomized protocol each logging session but that was not tested. The poor hit rate could also be due to the fact that the stance time is a bigger portion of the total step when walking, and due to overshoots in the signals the transition between stance and swing might be harder to recognize. Another aspect is that the impact between the foot and ground is lower in slower paces making the step profile less characteristic.

However, the varying result in the segmentation of 4 km/h intervals does not impact the classification result. Intervals tagged as natural foot strike are not included in the training database.

5.3 Classification

Since the interpretation of the different strike-types, and the ability to realize them when walking or running, can vary from person to person it is hard to know if foot-strikes tagged as heel-, mid- or toe strike actually is of the intended strike-type.

The strike-type commented as the hardest to forge was the mid-strike and that was also the strike-type for which all tested networks had the lowest sensitivity. This indicates that all mid-strikes being misclassified by the network might not actually be misclassified, some are likely wrongly tagged (or realized) and are in fact heel-strikes.

The result of the classifier was much dependent on which data was used to train the network. Even though using a protocol for data collection was hoped to give an equal amount of intervals of each strike type, issues with the sensors durability resulted in this uneven database, presented in Section 4.7. Classification has not been tested on a subset of the database containing an equal distribution of strike types and it is possible that this issue may have had an impact on the classification result.

Hypothetically, if the system would become a commercial product, should the network already be trained or should each customer collect data to train with? Best performance was achieved with the largest feature set, with data from all test persons and from all socks. Though, that was when the network was tested on data from multiple persons. A network only used by a single person might benefit from only training on data from this person. In Matlab, the training time was relatively low (less than a minute) while the possibility to perform such a network training on a phone has not been investigated. Possibly, if the processing time and computational requirements are too large, the application might require the network to be trained beforehand, and then remains the question of what training data to use. Potentially a pressure plate could be used as a reference method to collect data and build a database of controlled and appropriate steps.

5.4 Future Work

The difficulties with collecting data during any longer period of time has shown that in order to be able to conduct any further research within signal processing the robustness of the hard ware would need to be improved. To handle the problem with decreasing amplitude, changing to an adaptive AD-converter could be a starting point as there seems to still be information in the sensor signals, though with much lower amplitude.

Another field open for exploration is incorporating other sensors. Since accelerometer data is often used for activity classification (Banos et al., 2014), such information could be used in the preprocessing stage to classify when the person is actually running or walking.

Moreover, the data windows classified as non-activity data is cut out and the windows with activity data are added together. When adding the remaining data, if unfortunate, joining two windows may result in an unreal step profile. In this sit-

uation the segmentation algorithm will generate a segment of this assembled step. This aspect has not been dealt with and there might be a better way to handle this step in the system. The solution of incorporating accelerometer data might decrease the risk of creating these false step profiles, if the method is more accurate than using the periodicity-based solution.

Another possible improvement when combining accelerometer data with the sock data could be when detecting the start and end of each step or stance phase. The result might be more precise if the data from the three sensors are fused.

Last but not least, only one type of feed forward network has been tested in this study, and even though the results are promising, it could be interesting to test other types of classification methods.

6

Conclusion

The textile piezoelectric sensors can be used in a smart sock to classify heel-, toe- and mid-strike patterns. The developed pattern recognition system is composed of preprocessing, a comprehensive segmentation, and feature extraction prior to classification with a supervised neural network.

The system has a hit rate of up to 97 % when all three strike-types are used as targets. Even though the system had the lowest classification rate for mid-strikes, the hit rate was just slightly higher (98.7 %) when classifying only heel- and toe-strikes.

The pattern recognition system can estimate the number of steps taken as well as the cadence and the stance time. The system should be possible to implement as a real-time application as no data sequences longer than three seconds are needed in any part of the system.

The results are very promising, as of today the signals from the textile piezoelectric sensors can be used for classification, but larger tests would be needed to validate the results. It would be necessary to improve the sock, and possibly the hardware, before conducting software tests on a larger test group and continuing the research on other areas of use than running.

Bibliography

Banos, O., Galvez, J.-M., Damas, M., Pomares, H. and Rojas, I. (2014), 'Window size impact in human activity recognition', *Sensors* **14**(4), 6474–6499.

Bishop, C. (1995), *Neural networks for pattern recognition*, Oxford university press.

Cavanagh, P. and Lafortune, M. (1980), 'Ground reaction forces in distance running', *Journal of biomechanics* **13**(5), 397–406.

Daoud, A., Geissler, G., Wang, F., Saretsky, J., Daoud, Y. and Lieberman, D. (2012), 'Foot strike and injury rates in endurance runners: a retrospective study', *Medicine and Science in Sports and Exercise* **44**(7), 1325–34.

Dickinson, J., Cook, S. and Leinhardt, T. (1985), 'The measurement of shock waves following heel strike while running', *Journal of biomechanics* **18**(6), 415–422.

Duffy, J. (2015), 'Sensoria fitness smart socks bundle'. Available at: `http://www.pcmag.com/article2/0,2817,2486432,00.asp` [Accessed 09-03-2016].

Emilio Di Paolo, M. (2013), *Data Acquisition Systems*, Springer.

Fida, B., Bernabucci, I., Bibbo, D., Conforto, S. and Schmid, M. (2015), 'Pre-processing effect on the accuracy of event-based activity segmentation and classification through inertial sensors', *Sensors* **15**(9), 23095–23109.

Giandolini, M., Poupard, T., Gimenez, P., Horvais, N., Millet, G., Morin, J.-B. and Samozino, P. (2014), 'A simple field method to identify foot strike pattern during running', *Journal of biomechanics* **47**(7), 1588–1593.

Hagan, M. T., Demuth, H. B., Beale, M. H. and De Jesús, O. (1996), *Neural network design*, Vol. 20, PWS publishing company Boston.

Hamill, J., Milliron, M. and Healy, J. (1994), 'Stability and rear-foot motion testing: a proposed standard', *Proceedings of the Eigth Biennial Conference of the Canadian Society for Biomechanics* pp. 12–14.

Haykin, S. (2009), *Neural networks and learning machines*, Pearson Education Upper Saddle River.

Heaton, J. (2008), *Introduction to neural networks with Java*, Heaton Research, Inc.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N. et al. (2012), 'Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups', *Signal Processing Magazine, IEEE* **29**(6), 84.

Jain, A., Duin, R. and Mao, J. (2000), 'Statistical pattern recognition: A review', *IEEE Transactions On Pattern Analysis and Machine Intelligence* **22**(1), 4–37.

Jain, A., Mao, J. and Mohiuddin, K. (1996), 'Artificial neural networks: A tutorial', *IEEE Computer* **29**(3), 31–44.

Kawai, H. (1969), 'The piezoelectricity of poly (vinylidene fluoride)', *Japanese Journal of Applied Physics* **8**(7), 975.

Keller, T., Weisberger, A., Ray, J., Hasan, S., Shiavi, R. and Spengler, D. (1996), 'Relationship between vertical ground reaction force and speed during walking, slow jogging, and running', *Clinical Biomechanics* **11**(5), 253–259.

Kulmala, J.-P., Avela, J., Pasanen, K. and Parkkari, J. (2013), 'Forefoot strikers exhibit lower running-induced knee loading than rearfoot strikers', *Medicine and Science in Sports and Exercise* **45**(12), 2306–2313.

Kutz, M. (2015), *Mechanical Engineers' Handbook, Materials and Engineering Mechanics*, John Wiley & Sons.

Larson, P., Higgins, E., Kaminski, J., Decker, T., Preble, J., Lyons, D., McIntyre, K. and Normile, A. (2011), 'Foot strike patterns of recreational and sub-elite runners in a long-distance road race', *Journal of Sports Sciences* **29**(15), 1665–1673.

Le Cun, Y., Touresky, D., Hinton, G. and Sejnowski, T. (1988), A theoretical framework for back-propagation, *in* 'The Connectionist Models Summer School', Vol. 1, pp. 21–28.

Lund, A., Jonasson, C., Johansson, C., Haagensen, D. and Hagström, B. (2012), 'Piezoelectric polymeric bicomponent fibers produced by melt spinning', *Journal of Applied Polymer Science* **126**(2), 490–500.

Mann, R. and Hagy, J. (1980), 'Biomechanics of walking, running, and sprinting', *The American journal of sports medicine* **8**(5), 345–350.

Mann, R., Malisoux, L., Nührenbörger, C., Urhausen, A., Meijer, K. and Theisen, D. (2015), 'Association of previous injury and speed with running style and stride-to-stride fluctuations', *Scandinavian journal of medicine & science in sports* **25**(6), e638–e645.

Møller, M. F. (1993), 'A scaled conjugate gradient algorithm for fast supervised learning', *Neural networks* **6**(4), 525–533.

Nilsson, E., Lund, A., Jonasson, C., Johansson, C. and Hagström, B. (2013), 'Poling and characterization of piezoelectric polymer fibers for use in textile sensors', *Sensors and Actuators A: Physical* **201**, 477–486.

Novacheck, T. (1998), 'The biomechanics of running', *Gait & posture* **7**(1), 77–95.

Preece, S., Kenney, L., Major, M., Dias, T., Lay, E. and Fernandes, B. (2011), 'Automatic identification of gait events using an instrumented sock', *Journal of neuroengineering and rehabilitation* **8**(1), 1.

Rundqvist, K. (2013), Piezoelectric behaviour of woven constructions based on poly (vinylidene fluoride) bicomponent fibres, Master's thesis, University of Borås/Swedish School of Textiles.

Rundqvist, K., Nilsson, E., Lund, A., Sandsjö, L. and Hagström, B. (2014a), 'Piezoelectric textile fibres in woven constructions'. Ambience 1410i3m, Tampere, Finland, 7-9 september 2014.

Rundqvist, K., Sandsjö, L., Lund, A., Persson, N.-K., Nilsson, E. and Hagstöm, B. (2014b), 'Registrering av fotnedsättning baserat på piezoelektriska fibrer'. Medicinteknikdagarna 2014, 14-16 oktober, Göteborg, Sverige.

Saito, M., Nakajima, K., Takano, C., Ohta, Y., Sugimoto, C., Ezoe, R., Sasaki, K., Hosaka, H., Ifukube, T., Ino, S. et al. (2011), 'An in-shoe device to measure plantar pressure during daily human activity', *Medical engineering & physics* **33**(5), 638–645.

Sandsjö, L., Rundqvist, K., Lund, A. and Persson, N.-K. (2014), 'Monitoring of forefoot-rearfoot running using a piezoelectric sock'. Materials for Tomorrow/Sports Technology , 4-6 November, 2014, Gothenburg, Sweden.

Sensoria (n.d.). Available at: `http://www.sensoriafitness.com/` [Accessed 09-03-2016].

Sundquist, J. (1997), 'What is the ideal pace for walking running and cycling'. **URL:** *http://www.ontherunevents.com/ns0060.htm*

Tirosh, O., Begg, R., Passmore, E. and Knopp-Steinberg, N. (2013), Wearable textile sensor sock for gait analysis, *in* 'Sensing Technology (ICST), 2013 Seventh International Conference on', IEEE, pp. 618–622.

Upton, G. and Cook, I. (2014a), 'two-by-two table'. Available at: `http://www.oxfordreference.com/view/10.1093/acref/9780199541454.001.0001/acref-9780199541454-e-1679?rskey=XoP0I7&result=1` [Accessed 25-05-2016].

Upton, G. and Cook, I. (2014b), 'Mae'. Available at: `http://www.oxfordreference.com/view/10.1093/acref/9780199541454.001.0001/acref-9780199541454-e-2062?rskey=SnnBaF&result=4` [Accessed 20-06-2016].

Zaknich, A. (1998), 'Introduction to the modified probabilistic neural network for general signal processing applications', *IEEE Transactions on Signal Processing* **46**(7), 1980–1990.

List of Figures

List of Tables

A

Results from testing different number of neurons with different feature spaces

A.1 Heel data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
10	0.070	0.119	0.150	8.47	0.0014
20	0.055	0.098	0.115	4.31	0.0012
30	0.050	0.087	0.100	7.18	0.0013
40	0.035	0.064	0.070	10.98	0.0012
50	0.051	0.089	0.103	8.36	0.0010
60	0.046	0.082	0.093	9.62	0.0010
70	0.038	0.069	0.077	13.10	0.0011
80	0.035	0.063	0.068	14.48	0.0009
90	0.039	0.070	0.080	14.83	0.0009
100	0.059	0.101	0.125	10.91	0.0009

Table A.1: Part A: Performance measures for different number of neurons in the hidden layer, using only heel data as training data

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
10	0.864	0.744	0.929	0.879	0.909	0.978
20	0.891	0.817	0.937	0.913	0.925	0.985
30	0.910	0.835	0.945	0.920	0.937	0.988
40	0.934	0.888	0.964	0.950	0.955	0.988
50	0.907	0.832	0.942	0.921	0.935	0.984
60	0.907	0.859	0.949	0.933	0.936	0.988
70	0.933	0.876	0.952	0.943	0.950	0.988
80	0.937	0.887	0.968	0.947	0.957	0.990
90	0.925	0.878	0.951	0.944	0.946	0.987
100	0.880	0.801	0.931	0.901	0.918	0.987

Table A.1: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, using only heel data as training data

A.2 Toe data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
10	0.070	0.125	0.145	9.00	0.0016
20	0.061	0.111	0.123	3.77	0.0013
30	0.047	0.085	0.094	7.27	0.0010
40	0.042	0.076	0.084	9.27	0.0010
50	0.042	0.078	0.082	9.47	0.0010
60	0.040	0.075	0.079	10.84	0.0011
70	0.049	0.091	0.095	9.90	0.0010
80	0.050	0.091	0.100	9.40	0.0010
90	0.046	0.084	0.091	12.68	0.0009
100	0.045	0.083	0.090	14.73	0.0011

Table A.2: Part A: Performance measures for different number of neurons in the hidden layer, using only toe data as training data

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
10	0.885	0.779	0.877	0.927	0.908	0.948
20	0.900	0.817	0.895	0.942	0.919	0.956
30	0.933	0.853	0.914	0.955	0.937	0.969
40	0.943	0.863	0.926	0.961	0.946	0.969
50	0.943	0.873	0.920	0.959	0.946	0.973
60	0.941	0.874	0.933	0.960	0.948	0.974
70	0.927	0.848	0.923	0.952	0.937	0.968
80	0.927	0.839	0.915	0.950	0.937	0.964
90	0.932	0.857	0.920	0.953	0.941	0.969
100	0.935	0.859	0.918	0.950	0.942	0.971

Table A.2: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, using only toe data as training data

A.3 Heel and toe data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
20	0.033	0.065	0.062	6.65	0.0019
40	0.018	0.038	0.035	7.86	0.0019
60	0.017	0.037	0.034	10.31	0.0013
80	0.017	0.036	0.033	10.25	0.0011
100	0.017	0.035	0.031	11.05	0.0011
120	0.017	0.034	0.031	14.23	0.0011
140	0.014	0.028	0.026	15.50	0.0011
160	0.018	0.036	0.035	16.93	0.0012
180	0.013	0.027	0.024	23.24	0.0012
200	0.020	0.041	0.040	21.61	0.0015

Table A.3: Part A: Performance measures for different number of neurons in the hidden layer, using heel and toe data as training data

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
20	0.940	0.907	0.963	0.957	0.960	0.988
40	0.968	0.948	0.976	0.976	0.977	0.993
60	0.965	0.951	0.982	0.980	0.976	0.993
80	0.970	0.949	0.981	0.977	0.980	0.993
100	0.975	0.947	0.981	0.977	0.982	0.993
120	0.971	0.948	0.985	0.978	0.981	0.994
140	0.978	0.959	0.983	0.982	0.984	0.994
160	0.967	0.947	0.979	0.976	0.976	0.994
180	0.982	0.964	0.979	0.984	0.984	0.995
200	0.966	0.936	0.974	0.974	0.974	0.991

Table A.3: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, using heel and toe data as training data

A.4 Heel, toe and summed data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
30	0.021	0.042	0.038	13.34	0.0017
60	0.014	0.028	0.026	16.96	0.0013
90	0.012	0.025	0.023	23.55	0.0013
120	0.011	0.024	0.020	33.96	0.0012
150	0.011	0.022	0.019	36.52	0.0012
180	0.041	0.082	0.071	23.26	0.0012
210	0.010	0.022	0.019	54.88	0.0012
240	0.014	0.027	0.026	60.64	0.0012
270	0.014	0.029	0.026	65.37	0.0015
300	0.031	0.062	0.052	38.50	0.0013

Table A.4: Part A: Performance measures for different number of neurons in the hidden layer, using heel, toe and summed data as training data

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
30	0.968	0.949	0.965	0.974	0.975	0.992
60	0.977	0.962	0.982	0.982	0.984	0.995
90	0.982	0.965	0.982	0.984	0.985	0.995
120	0.982	0.971	0.985	0.987	0.986	0.996
150	0.983	0.976	0.982	0.989	0.986	0.996
180	0.941	0.889	0.948	0.950	0.958	0.982
210	0.985	0.971	0.985	0.988	0.988	0.996
240	0.977	0.964	0.978	0.983	0.981	0.996
270	0.978	0.961	0.979	0.9823	0.982	0.995
300	0.956	0.929	0.954	0.960	0.967	0.993

Table A.4: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, using heel, toe and summed data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
50	0.017	0.035	0.031	52.27	0.0013
60	0.012	0.026	0.022	49.50	0.0010
70	0.016	0.033	0.030	48.52	0.0012
80	0.016	0.032	0.030	50.94	0.0010
90	0.019	0.039	0.034	43.76	0.0010
100	0.018	0.038	0.035	44.62	0.0011
110	0.012	0.023	0.022	71.95	0.0010
120	0.014	0.029	0.026	66.81	0.0010
130	0.012	0.026	0.023	65.16	0.0010
140	0.008	0.016	0.014	90.23	0.0014
150	0.010	0.021	0.017	99.33	0.0017

Table A.5: Part A: Performance measures for different number of neurons in the hidden layer, using heel, toe and summed data as training data. Smaller spacing between the number of neurons were used to increase the resolution of the test.

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
50	0.970	0.961	0.975	0.981	0.977	0.995
60	0.982	0.964	0.985	0.984	0.987	0.996
70	0.975	0.958	0.975	0.980	0.981	0.994
80	0.974	0.953	0.980	0.979	0.981	0.994
90	0.970	0.948	0.977	0.976	0.978	0.994
100	0.972	0.951	0.968	0.977	0.975	0.995
110	0.980	0.971	0.982	0.986	0.985	0.996
120	0.976	0.965	0.980	0.983	0.983	0.995
130	0.981	0.967	0.980	0.985	0.985	0.995
140	0.989	0.982	0.988	0.992	0.990	0.997
150	0.985	0.974	0.988	0.989	0.988	0.996

Table A.5: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, using heel, toe and summed data as training data. Smaller spacing between the number of neurons were used to increase the resolution of the test.

A.5 Summed data as training data

Nbr neurons	MSE	H	Error	Train. time [s]	Class. time [s]
10	0.112	0.193	0.234	8.15	0.0015
20	0.075	0.133	0.157	9.18	0.0015
30	0.076	0.137	0.163	12.64	0.0013
40	0.087	0.155	0.182	9.22	0.0010
50	0.082	0.146	0.175	10.24	0.0010
60	0.064	0.115	0.134	23.57	0.0011
70	0.069	0.126	0.140	16.91	0.0011
80	0.086	0.151	0.181	12.34	0.0010
90	0.100	0.174	0.209	10.88	0.0011
100	0.096	0.168	0.205	9.64	0.0010

Table A.6: Part A: Performance measures for different number of neurons in the hidden layer, using summed data as training data

	Sensitivity			Specificity		
Nbr neurons	Heel	Mid	Toe	Heel	Mid	Toe
10	0.827	0.755	0.703	0.843	0.853	0.958
20	0.891	0.764	0.849	0.879	0.914	0.966
30	0.887	0.800	0.799	0.898	0.891	0.968
40	0.877	0.798	0.756	0.886	0.879	0.964
50	0.886	0.800	0.768	0.889	0.883	0.968
60	0.909	0.826	0.838	0.920	0.911	0.969
70	0.908	0.846	0.805	0.923	0.900	0.970
80	0.883	0.810	0.750	0.879	0.884	0.968
90	0.843	0.776	0.737	0.863	0.864	0.963
100	0.854	0.779	0.733	0.864	0.871	0.959

Table A.6: Part B: Sensitivity and specificity for different number of neurons in the hidden layer, summed data as training data

B

Descriptive examples of the segmentation

B.1 4 kph (interval 1) with sock 9

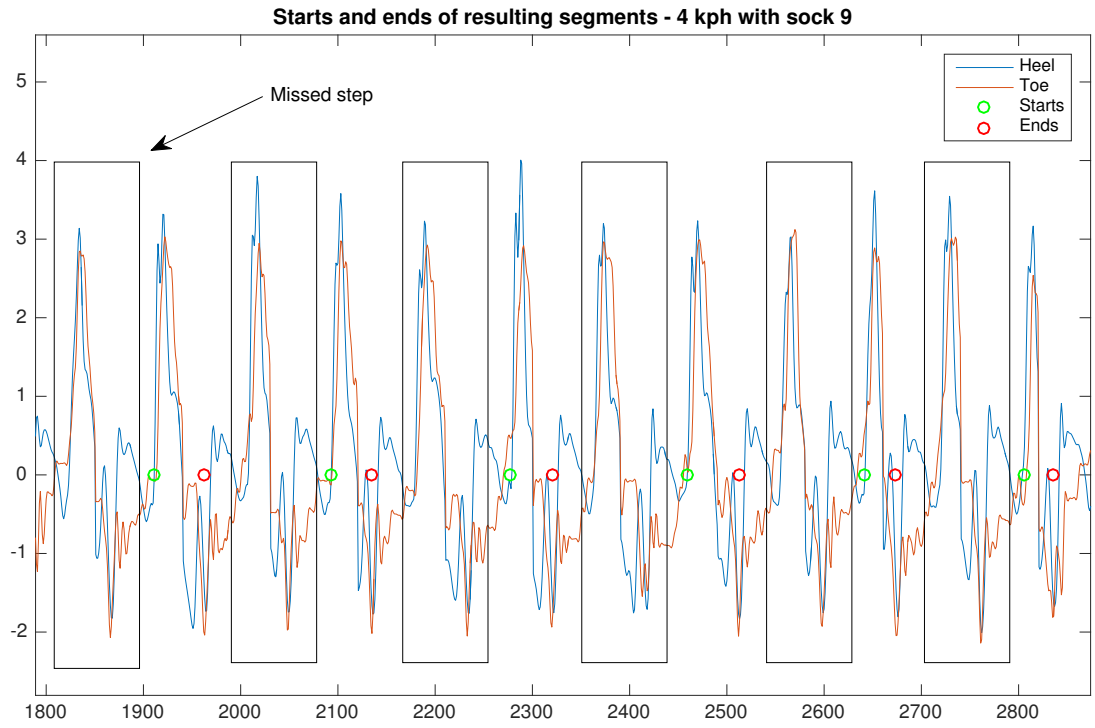


Figure B.1: All black boxes shows missed steps. The segmentation fails in every other step in 4 kph using sock 9.

B.2 4 kph (interval 1) with sock 2

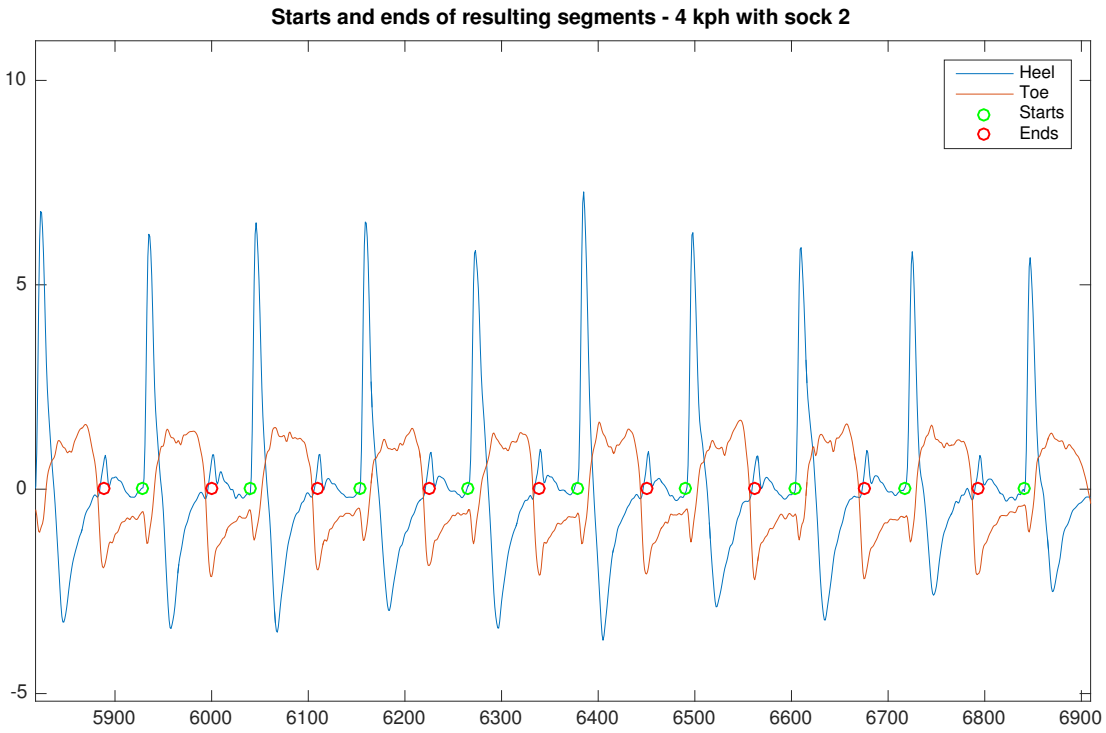


Figure B.2: The segmentation succeeds to find all steps even in 4 kph, using sock 2.

B.3 14 kph (interval 7) with sock 2

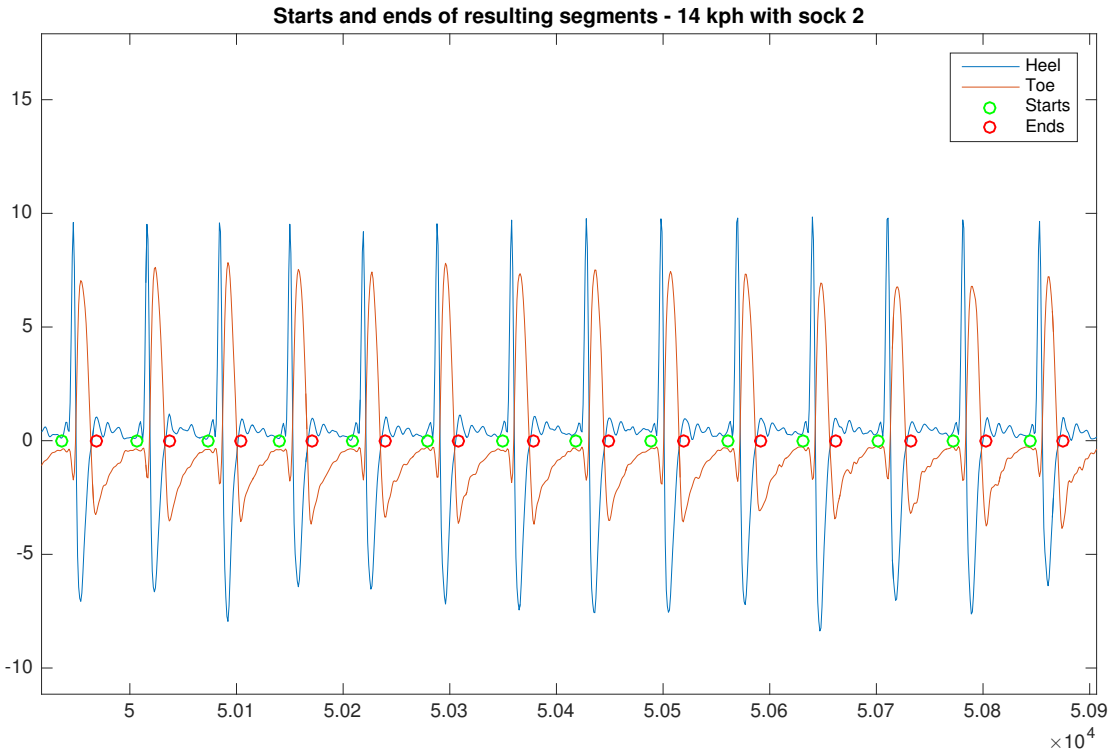


Figure B.3: The segmentation succeeds to find all steps in 14 kph, using sock 2.