UNIVERSITY OF TWENTE.

Faculty of EEMCS
Department of Electrical Engineering
Biomedical Signals and Systems

Automatic indentification of inertial sensors on the human body segments

January 28, 2011

Report nr: BSS 11-05

Master thesis
Electrical Engineering

Author: D. Weenk

Committee:
dr. ir. B.J.F. van Beijnum
Prof. dr. ir. P.H. Veltink
dr. ir. H.J. Luinge
ir. C.T.M. Baten

Abstract

In the last few years, the combination of inertial sensors (accelerometers and gyroscopes) with magnetic sensors was proven to be a suitable ambulatory alternative to traditional human motion tracking systems based on optical position measurements. While accurate full 6 degrees of freedom information is available, these inertial sensor systems still have some drawbacks, e.g. each sensor has to be attached to a certain predefined body segment.

This thesis is part of the 'Fusion Project'. The goal of this project is to develop a 'Click-On-and-Play' ambulatory 3D human motion capture system, i.e. a set of (wireless) inertial sensors which can be placed on the human body at arbitrary positions, because they will be identified and localized automatically.

In this thesis the automatic identification (or classification) of the inertial sensors is investigated, i.e. the automatic identification of the body segment to which each inertial sensor is attached.

Walking data was recorded from ten healthy subjects using an Xsens MVN motion capture system with full body configuration (17 inertial sensors). Subjects were asked to walk for about 5-8 seconds at normal speed (about 5 km/h). After rotating the sensor data to the global frame and aligning the walking directions for all the subjects with the positive x-axis, features as variance, mean, and correlations between sensors were extracted from x, y and z-components and from magnitudes of the accelerations and angular velocities. As a classifier a decision tree based on the C4.5 algorithm was developed (with cross-validation) using Weka (Waikato Environment for Knowledge Analysis).

From 31 walking trials (527 sensors), 523 sensors were correctly identified (99.24 %). For left/right identification inter-axis correlation coefficients were used. The accelerations of sensors on the right side of the body showed higher correlations between the positive y-axis (pointing to the left) and the positive x-and/or z-axis (pointing to the front and/or up) than the accelerations of sensors on the left side of the body.

Contents

Chapter 1

Introduction

1.1 Capturing human motion

Motion capture (mocap) is a term used to describe the process of recording human movement and to map this movement to a biomechanical model. In most cases this model consists of several rigid bodies (representing the body segments) which are connected by joints.

Motion capture is used to measure and or to calculate the positions of the segments and the angles of the joints [13]. There are several ways to capture human motion, for example optical, mechanical, inertial or acoustic sensing. In this report the focus is on inertial sensing.

The analysis of human motion is important for several disciplines. It is used for example for rehabilitation, sports training, and entertainment [2, 12, 14].

1.2 Fusion project

More and more people get locomotor problems which lead to an increased demand for accurate human motion capture techniques in rehabilitation and physiotherapy. The current motion capture systems using inertial sensors are time consuming to use and require the user to have prior knowledge of the technical details of the system.

This master's project is part of the 'Fusion' project: different research groups and companies[1] collaborate to develop a 'Click-On-and-Play' ambulatory 3D human motion capture and feedback system, comprising a set of

[1]The companies and research groups involved in the Fusion project are: Roessingh Research and Development (RRD), Xsens Technologies B.V. (Xsens), University of Twente - Biomedical Signals and Systems (UT-BSS), University of Twente - Biomechanical Engineering (UT-BW), Technical University of Delft - Biomechatronics and Biorobotics (TUD), Technology Trial Centre / Groot Klimmendaal (TTC), and Sint Maartenskliniek Research (SMR).

wireless motion sensors which can be placed on different segments of the human body in an arbitrary order, without the need of any prior knowledge.

1.3 Project goal

The goal of this project is to develop a new method to automatically identify human body segments to which inertial sensors are attached during walking.

To achieve this goal, information obtained from inertial sensors during walking at 'normal' speed (about 5 km/h), will be analyzed. When the inertial sensors are identified for a certain set of measurements, the performance of this identification method for new measurements is investigated.

1.4 Outline of the report

In Chapter 2 several important background topics (from literature) that are necessary for achieving the project goal are explained. This information is needed in a later stadium when a method for the automatic identification of inertial sensors is developed. Chapter 3 describes a pilot study, that investigated the properties and possibilities of the inertial sensor data. A proof of concept, by means of a decision tree used for identifying the inertial sensors is presented and explained. After this pilot study, a trial study is performed from which the measurement set-up and the methods are described in Chapter 4. The results of the measurements are presented in Chapter 5 and discussed in Chapter 6. This report ends with conclusions and recommendations in Chapter 7.

Chapter 2

Background

In this Chapter several important topics that are necessary for achieving the project goals are explained. In Section 2.1 the principle of inertial sensors and the physical meaning of the sensor output is described.

The estimation of the sensor location, i.e. the identification of the segment to which the sensor is connected, is a so-called *classification problem*. More about classification and pattern recognition is described in Section 2.2.

In several search databases publications regarding the automatic identifying of inertial sensors on the human body, have been sought but yielded no results. A detailed view of the searched databases and the used keywords is presented in Appendix A.

2.1 Inertial sensors

2.1.1 Accelerometers and gyroscopes

Inertial sensing is based on change of position and orientation estimation, using inertial sensors (accelerometers and gyroscopes).

A 3D *accelerometer* consists of a mass in a box, suspended by springs. The distances between the mass and the box ($\boldsymbol{x}$) are measured at all sides, yielding the inertial forces ($\boldsymbol{F}$) acting on the mass (m), using *Hooke's law* ($\boldsymbol{F} = k\boldsymbol{x}$). This force can be divided by the mass, using *Newton's second law* ($\boldsymbol{F} = m\boldsymbol{a}$), to obtain the *acceleration* ($\boldsymbol{a}$).

Gyroscopes are used to measure *angular velocity*. If a vibrating mass is rotated with an angular velocity ($\boldsymbol{\omega}$) while it has a translational velocity ($\boldsymbol{v}$), a Coriolis force $\boldsymbol{F_C}$ will act on the mass ($\boldsymbol{F_C} = 2m\boldsymbol{\omega} \times \boldsymbol{v}$). This force causes a vibration orthogonal to the original vibration. From this secondary vibration, the angular velocity can be determined.

The angular velocity of the gyroscopes has to be integrated in order to obtain the change of orientation. To obtain the change of position, the acceleration from the accelerometer has to be integrated twice. The accelerometer measures the sum of sensor acceleration ($\boldsymbol{a}$) and gravitational acceleration ($\boldsymbol{g}$). This gravitational component can be removed when the orientation with respect to the global frame is known.

2.1.2 Three-dimensional space

In *three-dimensional space*, or R^3, an arbitrary but fixed point is specified and called the origin. Through this origin three mutually perpendicular lines are specified, the x-axis, the y-axis, and the z-axis. Each of these axes are real number lines, with the zero points at the origin. In this thesis these axes are oriented to form a so called *right-handed coordinate frame*, i.e. if the index finger of the right hand is pointed forward, the middle finger bent inward (at a right angle) and the thumb placed at a right angle to both, then these three fingers indicate the x-, y-, and z-axes of a right handed coordinate system. The thumb indicates the x-axis, the index finger the y-axis and the middle finger the z-axis [5].

Points in three-dimensional space are represented by triplets (x, y, z) of real numbers. The origin, for instance, has coordinates (0,0,0). In R^3 any given point $p = (x, y, z)$ can be represented as a vector $\boldsymbol{v}$ from the origin O to the point p.

Because an accelerometer measures the sum of sensor acceleration $\boldsymbol{a}^s$ and gravitational acceleration $\boldsymbol{g}^s$,

$$\boldsymbol{s}^s = \boldsymbol{a}^s - \boldsymbol{g}^s,$$

both in the sensor frame, it is difficult to compare the 3D-accelerations of the different inertial sensors throughout the body (because the relative orientation between the sensors is unknown). Therefore it is necessary to express the accelerations of all the inertial sensors in the same global coordinate system. In this first step of expressing the accelerations in the global coordinate frame, the accelerations are rotated in such a way, that the z-axis of the accelerations is pointing upwards. This allows us to subtract the gravitational component easily (from the z-component of the 3D-acceleration). The heading, i.e. the orientation of the sensors in the horizontal plane, remains unchanged during this procedure.

To change the 3D-accelerations from sensor coordinate frame ψ^s to global coordinate frame ψ^g, the orientation of the inertial sensor, with respect to the global coordinate frame, has to be estimated. This can be done by combining the initial orientation and integration of the angular velocity, measured by the gyroscopes. The following differential equation can be used to integrate the angular velocities to angles [14]:

$$\dot{\boldsymbol{R}} = \boldsymbol{R}_s^g \tilde{\boldsymbol{\omega}}_s^{s,g}.$$

In this equation, the 3D *rotation matrix* representing the change of coordinates between sensor frame ψ_s and global frame ψ_g is indicated as $\boldsymbol{R}_s^g$ and its time derivative as $\dot{\boldsymbol{R}}_s^g$. $\tilde{\boldsymbol{\omega}}_s^{s,g}$ is a skew-symmetric matrix consisting of the components of the angular velocity vector of frame ψ^s with respect to ψ^g, expressed in ψ^s:

$$\tilde{\boldsymbol{\omega}}_s^{s,g} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}.$$

So, for the 3D sensor acceleration in the global coordinate frame, the following equation holds:

$$\boldsymbol{a}^g(t) = \boldsymbol{R}_s^g(t)\boldsymbol{s}^s(t) + \boldsymbol{g}^g,$$

with $\boldsymbol{g}^g = (0, 0, -9.81)$.

2.1.3 Motion capture

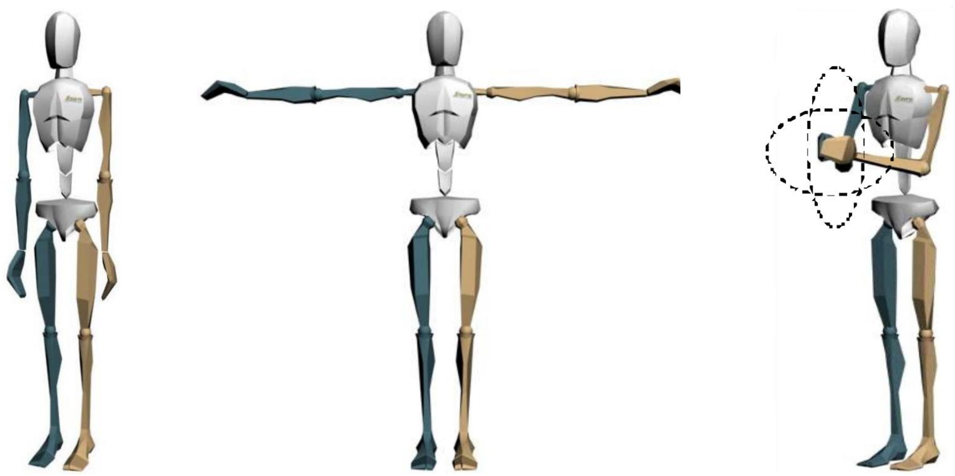
An example of a motion capture system using inertial sensors in combination with magnetic sensors is the Xsens MVN system [2, 13, 22]. As described in the previous Chapter, motion capture is a term used to describe the process of recording human movement and to translate this movement to a certain model. This model consists of several segments, rigid bodies, connected by joints. The Xsens MVN system uses a 23 segment biomechanical model for this. Not all these segments are measured directly with inertial sensors. Only 17 of these segments are measured directly, the other segments are calculated using the biomechanical model. See Appendix B for a detailed description of the biomechanical model.

In the current situation, the MVN system is not plug-and-play, i.e.

- All 17 sensors have a unique id, i.e. they have to be placed on a predefined body segment.
- When the sensors are attached to the body, the exact position on the segment is unknown.
- When the sensors are attached to the body, the exact orientation with respect to the segment is unknown.

Regarding these last two points, a *calibration* procedure has to be performed in order to determine the initial positions and orientations of the sensors with respect to the segments.

The basic calibration pose is the *neutral pose* (N-pose). It is similar to the anatomical pose, but with the thumbs pointed in forward direction instead of pointing laterally (Figure 2.1(a)). Another calibration pose is the *T-pose*, it is the same as the N-pose, but with the arms extended horizontally (the thumbs forward) (Figure 2.1(b)). If the knee orientations of these calibrations can not be determined correctly, the *squat calibration* can be performed. In this procedure one has to bend and straighten the knees (not to deep), starting from the n-pose, keeping the knees in the sagittal plane. For a higher accuracy of the upper body kinematics, a *hand touch calibration* can be performed (Figure 2.1(c)). During this calibration procedure the hand palms are placed together and the arms are moved slowly while the shoulders are kept steady [22].



(a) N-pose (b) T-pose (c) Hand touch calibration pose

Figure 2.1: Calibration poses in MVN Biomech. Calibration is needed in order to determine the initial orientations of the sensors with respect to the segments (from Xsens MVN BIOMECH User Manual [22]).

These segment calibrations and the fact that all the sensors have to be placed on the body on a specific place, are time consuming and therefore a future goal of the Fusion project is to develop an auto-calibration method.

2.1.4 Advantages and disadvantages of inertial sensors

Great advantages of motion capture systems based on inertial sensors are that there is no limited measurement volume and there are no line of sight problems. The costs are in most cases significantly lower than other motion capture systems were expensive camera's are required.

A disadvantage of inertial sensing is that, in the current situation, all sensors have a unique location ID, i.e. each sensor has to be attached to a certain, predefined body segment.

Also the fact that the relative positions and orientations of the sensors with respect to the body segments are unknown is a disadvantage. This can be resolved by calibrating the system, a procedure in which the positions and orientations of the sensors are linked to the positions and orientations of the body segments, under the assumption that the subject is standing in a predefined position.

Another disadvantage is the integration drift caused by noise, this can be minimized by sensor fusion algorithms. [12, 13, 14, 16].

2.2 Statistical signal classification and pattern recognition

Statistical signal classification is a process whereby a certain pattern or sampled signal is assigned to a certain predefined class [9]. It is sometimes referred to as *pattern recognition*, because the data can be divided into several classes with different patterns. A training procedure determines the decision boundaries between these classes.

A statistical signal classification system typically contains a feature extractor followed by a pattern classifier, as can be seen in Figure 2.2.

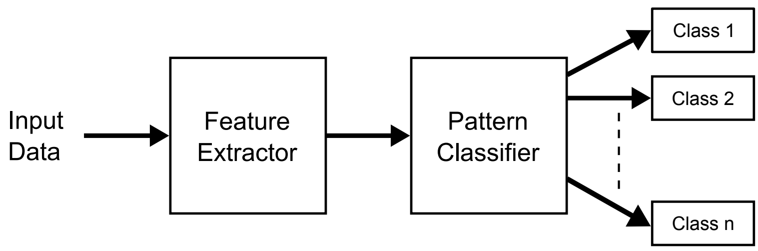


Figure 2.2: Block diagram of a typical statistical signal classification system.

2.2.1 Feature extraction

The purpose of feature extraction is to determine the characteristics of a data segment that accurately represents the original signal. These signal features, also referred to as a *feature set* or a *feature vector*, can then be used as input to classification algorithms. Features can be extracted from the signal in the time domain as well as from the frequency domain [3].

Time-domain features

Time domain features can be extracted from the input data directly. Examples of time-domain features are mean, variance, root mean square, or correlations between signals (in this case correlations between different body segments). In case of inertial sensors one could think of extracting time-domain features from acceleration or angular velocity signals.

Frequency-domain features

The focus of frequency-domain features is on the periodic structure of the signal. These periodic properties can be derived, for instance, from Fourier transforms. Examples of frequency-domain features are spectral energy and spectral entropy.

Dimensionality reduction

Using the extracted features directly as inputs for the classification and recognition methods might cause computational problems and cause the system to be less accurate. To avoid this, a *dimensionality reduction* method is used in which the dimensionality of the feature set is reduced by, for instance, making a selection of the most discriminative features or the features that have a higher contribution to the performance of the classifier. The dimension of the feature set can also be reduced by using feature transform techniques, i.e. try to map the high-dimensional feature space into a much lower dimension, yielding uncorrelated features that are a combination of the original features. This can be done for instance with *principal component analysis* (PCA). PCA is a linear transformation that transforms the features to a new coordinate system with the (uncorrelated) features sorted in descending order, corresponding to the variances of the extracted components [3, 6, 9].

2.2.2 Classification and recognition systems

The extracted feature set is then, after the dimensionality reduction, used as an input to the pattern classifier (see Figure 2.2). The pattern classifier contains classification and recognition methods. The two mostly used classification and recognition methods are *threshold-based techniques* and *pattern recognition techniques*:

- Threshold based classification systems can be used to distinguish signals with different intensities, for instance by using energy features [3]. Veltink et al. [17] used threshold based classification system for distinguishing between the static or dynamic nature of activities.

- Examples of pattern recognition classification systems are: decision tables, decision trees, nearest neighbor, Naïve Bayes, Markov Models, Hidden Markov Models, and Gaussian mixture models [3].

Extracting patterns from data is also referred to as *data mining*.

Data mining and machine learning The extraction of implicit, previously unknown, information from data is called data mining. One way of extracting this information is by designing and developing algorithms and let computers do the rest of the work. This way of extracting information from raw data is called *machine learning* [20].

There are many machine learning techniques, but an easy way of using (most of) them is by use of *Weka*[1]. Weka (Waikato Environment for Knowledge Analysis) is a comprehensive (free and open source) software resource, written in the Java language, developed at the University of Waikato, New Zealand [1]. It provides many popular learning schemes that can be used for practical data mining or for research.

Concepts, instances and attributes Before looking into machine learning methods into detail, some basic terms and in- and outputs are explained.

The input to the learner takes form of *concepts*, *instances*, and *attributes*. A *concept*, or concept description, is the thing to be learned. An *instance* is an individual example of the concept to be learned, it is the input to the machine learning scheme. The set of instances are the things that are to be classified. Each individual instance is characterized by its values on a fixed, predefined set of *features* or *attributes*. So each dataset is a matrix where the rows and columns represent the instances and attributes respectively. The attributes can be either *nominal* or *numeric*. Nominal (or categorical) features can take several prespecified values, while numeric features can be real or integer valued. Somewhere in between these two types are the *ordinal features*, which make it possible to rank the categories, so there is a notion of ordering, but there is no notion of distance between the values [20].

Different types of learning Basically four different types of learning appear in data mining applications:

- Classification learning
- Association learning
- Clustering
- Numeric prediction

In *classification learning*, a set of classified examples is presented from which a way of classifying is expected. *Association learning* means that association among features (the columns of the dataset) is sought, so this is not just a prediction of a certain class. In *clustering*, groups of instances (the rows of the dataset) that belong together are sought. While in *classification learning* the outcome to be predicted is a category, in *numeric prediction* the outcome is a numeric quantity [20].

Classification learning is also called *supervised learning* because the outcome (or the class) of each instance is made available to the machine learner.

[1] The Weka (pronounced to rhyme with Mecca) or woodhen is a flightless bird, found only on New Zealand.

In this thesis supervised learning will be used to identify the inertial sensors.

For these four types of learning several classifiers can be used, e.g. decision tables, decision trees,

Decision trees Decision trees are widely used, because they are simple to understand and interpret, they require little data preparation, and they are able to handle both numerical and nominal features. Another advantage is that decision trees perform well with large datasets in a relative short time [3, 20].

In Weka the J4.8 algorithm, which is an implementation of the *C4.5 algorithm*, can be used to create decision trees. The C4.5 algorithm builds decision trees from a set of training data, using the concept of *information entropy*. Information entropy (H) (in bits) is a measure of uncertainty and is defined as:

$$H = -\sum_{i=1}^{n} p(i) \log_2 p(i),$$

where $p(i)$ is the probability, estimated as the proportion of instances in the dataset. Information gain is the difference in entropy, before and after selecting one of the features for making a split.

At each node of the decision tree, the C4.5 algorithm chooses one feature of the dataset that splits the data most effectively, i.e. the feature with the highest information gain is chosen to make the split.

The main structure of the C4.5 algorithm is [4, 20]:

1. If all instances belong to the same class, then finish
2. Calculate the information gain for all features
3. Use the feature with the largest information gain to split the data
4. Return to step 1.

Training and testing A natural way to measure a classifier's performance is by means of *error rate*. If the classifier correctly predicts the class of an instance, it is counted as a success and if not, it is counted as an error.

What we are interested in is the performance of the classifier on new data, so not (only) the performance of the data used for training the classifier. This is why the classifier needs to be tested on a so-called *test set*, a dataset that is not used in the formation (training) of the classifier. There are several different techniques for predicting the performance of a classifier based on a limited dataset. One of these techniques is simply splitting the dataset in a test set and a *training set*. Another technique is the *cross-validation technique*, which is especially useful when the amount of data for training and testing is limited. In this method, the process of training and testing is repeated several times with different samples. In each iteration a certain proportion of the data is randomly selected for training, while the remainder is used for testing. The error rates are then averaged over the iterations. The standard way of predicting the error rate of a learning technique is the *10-fold cross-validation*, in which the data is divided randomly into 10 parts. Each part is then held out in turn and the learning scheme is trained on the remaining nine-tenths. Then the error rate is calculated on the holdout set. This is repeated 10 times and the error estimates are then averaged.

Instead of 10, any other number of folds can be used to get an estimate of the error, but 10 has become the standard.

Another point of discussion is *overfitting*. Overfitting occurs when a decision tree is too complex, while not being predictive for other data then the set used for training. This is usually occurring when a decision tree has too many branches, while in each branch only a few instances are classified. An overfitted tree performs very good on the training data, but will probably perform less on independent test data. This problem of overfitting can be resolved by a process called *pruning*, i.e. reducing the size of the decision tree [20].

2.2.3 Preprocessing

The input data to the feature extractor shown in Figure 2.2, is not directly the data from the inertial sensors but it is preprocessed data.

Preprocessing is necessary, for example, to remove the gravitational acceleration from the accelerometer data. This can be done, for instance, by using a high-pass filter (not ideal) or by calculating the acceleration in global coordinates and then subtracting the gravitational constant [3], see Section 2.1.2.

2.3 Activity monitoring

For an accurate estimation of the sensor location it might be important to have information about the activity performed by the subject, because signal features might differ while performing different activities.

Mannini and Sabatini described in [7] a way to classify human physical activity using on-body accelerometers. For this purpose, they used computational algorithms with classifiers based on Hidden Markov models.

Wassink et al. [18] monitored human activities using a trainable system, also based on Hidden Markov modeling. Data was collected using inertial sensors attached to the S4, T10 and C7 vertebrae. On a set of eight different human activities, including lifting a load, walking, standing and sitting, a score of up to 95.5±1.9% was obtained.

Veltink et al. also investigated the detection of static and dynamic activities of daily living in [17]. For example standing, sitting, lying, walking, ascending stairs, descending stairs, and cycling were distinguished using a small set of two or three uniaxial accelerometers mounted on the body.

Avci et al. surveyed the different approaches for activity recognition using inertial sensors in [3].

2.4 Conclusion

In this Chapter the background information, needed for developing a suitable algorithm for identifying the inertial sensors, was described. First the basics of inertial sensors is described, followed by the measurements in three-dimensional space. To calculate accelerations and angular velocities in the global frame, rotation matrices are needed. Because the relation between the inertial sensors and the human body segments is unknown, in the current situation a sensor-segment calibration procedure is needed.

Statistical signal classification can be used to divide signals into several classes. A typical signal classifier consists of a feature extractor, followed by a pattern classifier.

Weka can be used to classify signals automatically. The J48 is an implementation of the C4.5 decision tree algorithm and is a fast and easy way to classify data.

Because inertial sensor signal features might differ while performing different activities, monitoring these activities is needed.

Chapter 3

Pilot study

This Chapter describes a pilot study (a proof of concept) in which the identification of inertial sensors during walking is demonstrated. The assumption has been made that all 17 inertial sensors are attached correctly to the body, i.e. on the predefined positions as described in Appendix B, Table B.1. In this pilot study, only the magnitudes of the sensor signals are used so the relative orientations between the sensors is of no influence.

3.1 Measurement description

During an internship at Xsens Technologies B.V. [2], walking trials are recorded for three subjects using an MVN motion capture system with full body configuration [22]. The subjects were asked to walk over about four meters with normal velocity. The sensor accelerations and sensor angular velocities of these measurements are used for the development of the protocol for identifying the inertial sensors. The MVN system consisted of:

- 17 *MTx sensors* with an accelerometer range of 18 g, and a rate gyroscope range of 1200 deg/s.
- Two *Xbus Masters* (XM), delivering power to the MTx's and retrieving their data exactly synchronized.
- Two *Wireless Receivers* (WR-A), for handling the data traffic between the XMs and the PC. Each WR-A is connected to a USB port.

The sampling frequency (F_s) used for the measurements was 120 Hz. The data was saved in an MVN file format, converted to XML and loaded into MATLAB for further analysis.

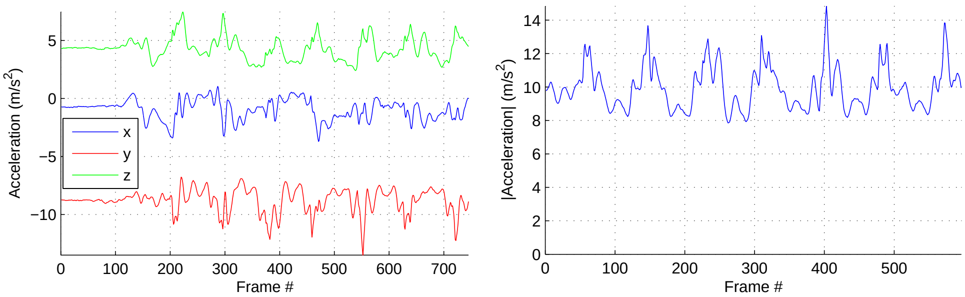
3.2 Preprocessing

For sensor identification, i.e. determining which sensor is attached to which body segment, several steps are required and analyzed.

As a start, the data is manually (visual inspection) shortened in order to proceed with the walking data only, see Figure 3.1(a). The remaining segment length is 500-600 samples ($\pm$4-5 s).

The next step is to calculate the magnitudes of the 3D acceleration and the 3D angular velocity. This is done because the relative orientation between the sensors is unknown, which may cause errors when x, y, or z components of different sensors are compared to each other. The calculation of the norm (or magnitude) is done in MATLAB by taking the square root of the sum of the x, y, and z components of the signal squared (see Figure 3.1(b) for an example of the magnitude of the sensor acceleration).

From this preprocessed data several features will be extracted in the next Section.



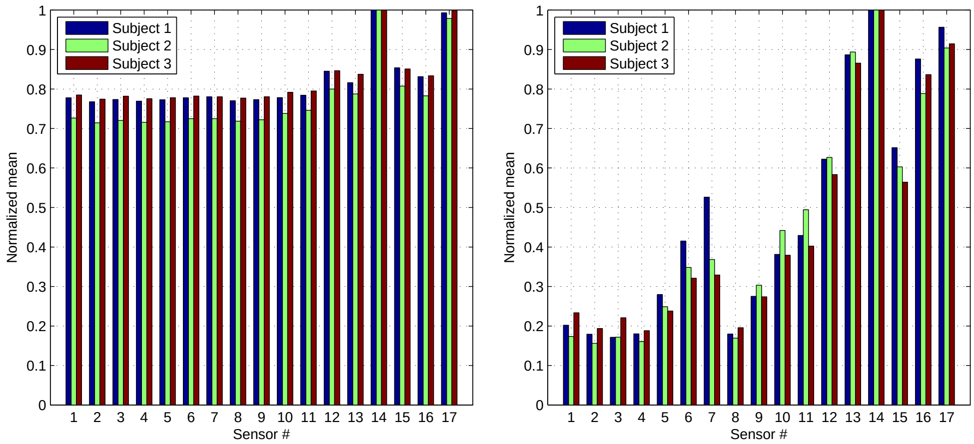
(a) x, y, and z components. (b) Norm of the walking data.

Figure 3.1: Acceleration measured on the left shoulder of subject 1 during normal walking. Two preprocessing steps are performed: The first 150 frames (F_s = 120 Hz, so the first 1.25 seconds) of the measurement, where the subject is standing still before he/she starts walking, are deleted (by visual inspection) from the original signal (left) and the norm of the x, y, and z components is calculated (right). The gravitational component is not removed in this pilot study.

3.3 Signal features

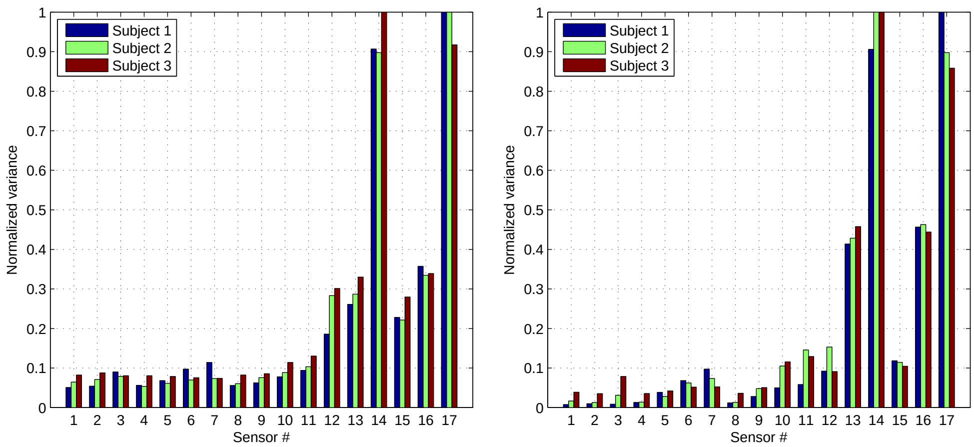
The first feature that is investigated is the mean of the preprocessed acceleration and angular velocity. The mean of all these signals is normalized for each subject (divided by the mean of the signal with the maximum mean of a subject) in order to get a better comparison between different subjects. For the three subjects, the result is shown in figure 3.2 for both the magnitudes of the accelerations and the angular velocities of the 17 sensors.

Next, the variance of the preprocessed signals is calculated and normalized in the same way (Figure 3.3).



(a) Normalized mean of acceleration (b) Normalized mean of angular velocity

Figure 3.2: Normalized mean of the magnitudes of the accelerations and angular velocities of the 17 sensors for the subjects 1, 2, and, 3, during walking at normal speed.



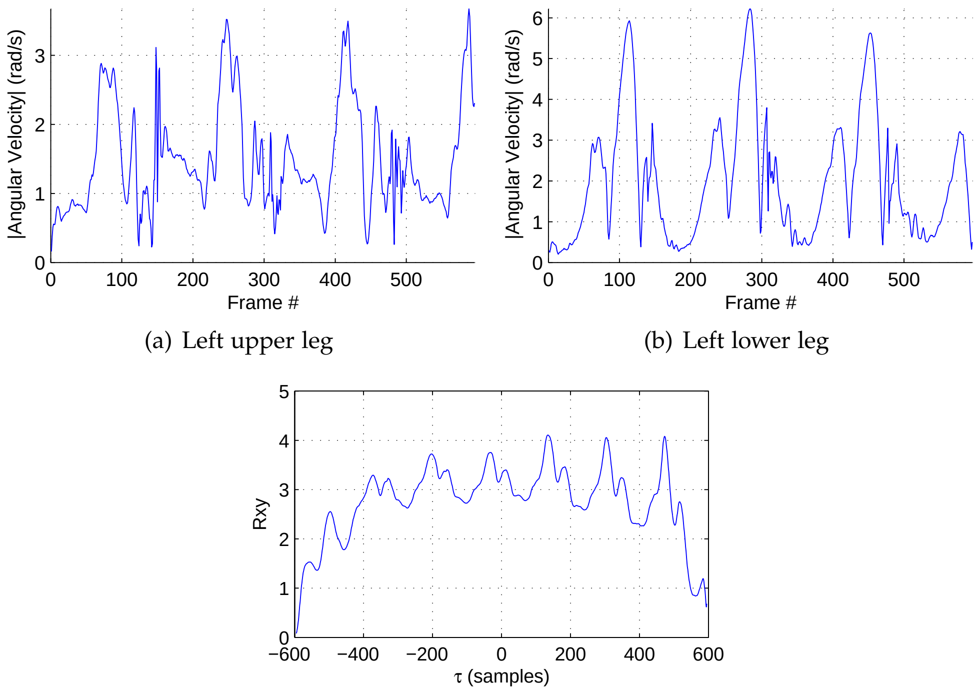
(a) Normalized variance of acceleration (b) Normalized variance of angular velocity

Figure 3.3: Normalized variance of the magnitudes of the accelerations and angular velocities of the 17 sensors for the subjects 1, 2, and, 3, during walking at normal speed.

Another feature that is extracted is the (unbiased estimate of the) cross-correlation function (R_{xy}), calculated in MATLAB by

$$R_{xy}(m) = \begin{cases} \frac{1}{N-|m|} \sum_{n=0}^{N-m-1} x(n+m)y(n) & m \geq 0 \\ R_{yx}(-m) & m < 0 \end{cases} \tag{3.1}$$

with N, the number of samples of the signals. An example of the cross-correlation between the angular velocities of the sensors on the left lower and upper leg is shown in Figure 3.4.



(a) Left upper leg

(b) Left lower leg

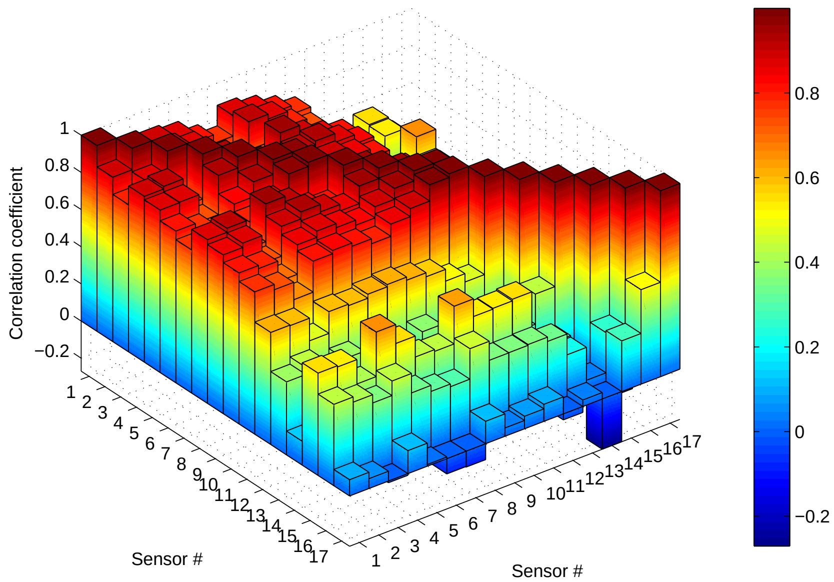
(c) Cross-correlation

Figure 3.4: Example of the cross-correlation (Rxy) (c) between the angular velocity of the sensors on the left upper (a) and lower (b) leg. From measurement of subject 1 during walking at normal speed.

Related to the cross-correlation, the correlation coefficients between signals (ρ) is used as a feature. These (linear) correlation coefficients are calculated using

$$\rho = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{3.2}$$

with σ_{xy} the covariance and σ_x, σ_y the standard deviations of the signals. The correlation coefficient is always between -1 and +1 and if it is equal to zero, the signals are uncorrelated [15]. An example of the correlation coefficients for subject 1 during walking at normal speed are shown in Figure 3.5(a) (3D view) and Figure 3.5(b) (top view). In Appendix C all the correlation coefficients for the three subjects are shown, for the angular velocity as well as for the acceleration signals (Figures C.1 and C.2).



(a) 3D view

Figure 3.5: Correlation coefficients of the magnitude of the sensor acceleration of subject 1, while walking at normal speed. (continued...).

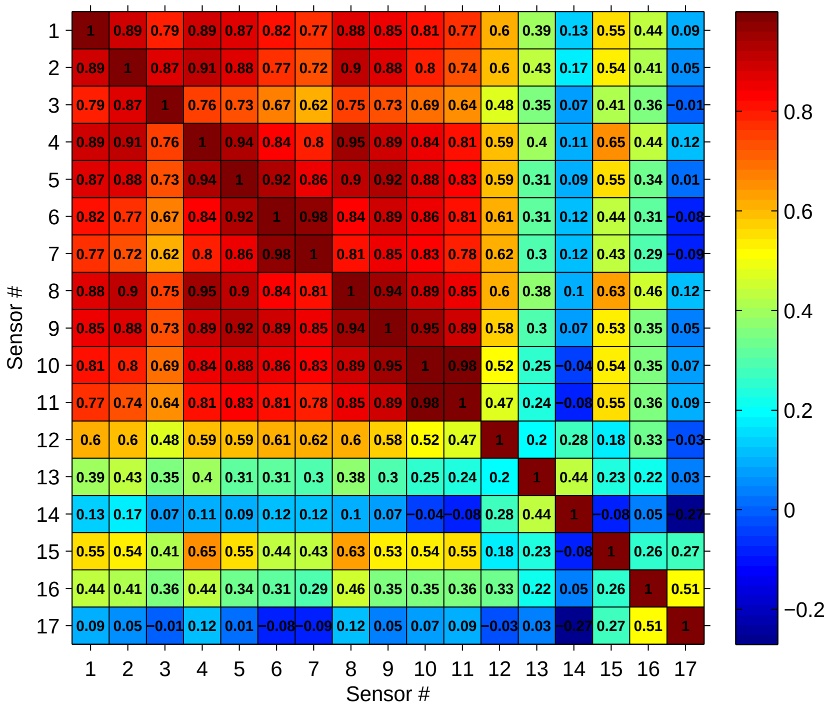
The power spectra of the signals (the Fourier transformation of the correlation functions) are also investigated, but no extra information for distinguishing different sensors can be obtained from this. Most of the signal power is located between 0-3 Hz.

3.4 Signal classifier

3.4.1 Threshold based signal classifier

After looking closely at the selected features, a suitable classifier is chosen. In this pilot study the features are representing different values, instead of containing patterns, so a threshold based classification system has to be used (see Section 2.2.2).

In this pilot study a classifier is created by trial and error (as a proof of concept), so the choice is made to use a decision tree, because they are relatively simple to understand and interpret (see Section 2.2.2).



(b) Top view, including the correlation coefficients

Figure 3.5: (...continued). Correlation coefficients of the magnitude of the sensor acceleration of subject 1, while walking at normal speed.

3.4.2 First version of the decision tree

The first decision tree that was produces was based on threshold values derived from the bar plots in Figures 3.2 and 3.3 and from the correlation coefficients in the Figures C.1 and C.2 in Appendix C.

This decision tree worked well for these three subjects, but after looking at the walking trials of eight other subjects[1], the classifier caused some errors in identifying the upper legs. This is due to fact that the normalized mean of the angular velocity was higher for the forearms and hands than for the upper legs in some walking trials (In Figure 3.2(b), it can be seen that for subject 1 the right hand (sensor 7) has a normalized mean of about 0.52, while the normalized mean of the right upper leg (sensor 12) is about 0.61. From this it can be concluded that working with threshold values for distinguishing between hands and upper legs might cause problems in other walking trials).

[1]Walking trials of eight other subjects wearing normal shoes were used for this. These trials, four trials per subject and about 1500 frames per trial (12.5 s), were measured by Pim Pellikaan during his Bachelor assignment - *What is the effect of wearing 'unstable' Masai Barefoot Technology (MBT) shoes on the walking motion, balance and posture of a person compared to conventional 'stable' shoes* [10]. A similar MVN motion capture system was used for these measurements.

3.4.3 Final version of the decision tree

Because these threshold values caused errors in some other walking trials, a new approach is chosen. A new (final) decision tree is created (by visual inspection of all the extracted features and by trial and error) and shown in Figure 3.6. This decision tree will be explained step by step in this subsection.

Because it is not possible to identify left and right (without 3D information in the global frame), but only to determine whether or not sensors are on the same lateral side of the body, the codes 01/02 or i/j are used in the decision tree to distinguish between the lateral sides. These codes stand for left/right (or vice versa). In the final decision tree, if one of the sensors (except a sensor on the pelvis, sternum or head) can be verified to be on the left or on the right side of the body, all the other sensors can be identified correctly.

The first step is to distinguish the feet from the sensors, which is done using the variance of the angular velocity. The two sensors with the largest variance of angular velocity are the feet.

Secondly, the 15 remaining sensors are split in two groups, one with eight sensors with the largest mean of the angular velocity (upper legs, lower legs, forearms and hands), and one group with the seven sensors with the smallest mean in angular velocity (pelvis, sternum, head, shoulders and upper arms).

From the group of eight sensors with the largest mean (mostly the sensors on the extremities, subjected to relatively high angular velocities), the forearms and hands are identified by means of the acceleration correlation coefficients (twice). Because the correlation coefficients between forearms and hands are large (>0.95) compared to correlation coefficients of other segments (see Figure 3.5(b)), these four sensors can be identified easily. To make a distinction between forearms and hands, the mean of the angular velocity can be used, because the hands have a larger angular velocity than the forearms (during walking).

What is left are the upper and lower legs, which can be separated by using the variance of the angular velocity. The lower legs have a larger variance than the upper legs. The lower legs are then distinguished in lateral sides(i and j) by calculating the maximum of the cross-correlation function of both lower legs with one of the upper legs, and the corresponding time lag. The time lag between ipsilateral upper and lower legs is smaller than the time lag between contralateral upper and lower legs.

Within the group of seven sensors, obtained after the second step, the upper arms are identified by using the mean of the angular velocity. The two sensors with the largest mean are the upper arms. After calculating the correlation coefficient between each of these sensors and forearm01, upper arm01 can be obtained because there is a higher correlation between an upper

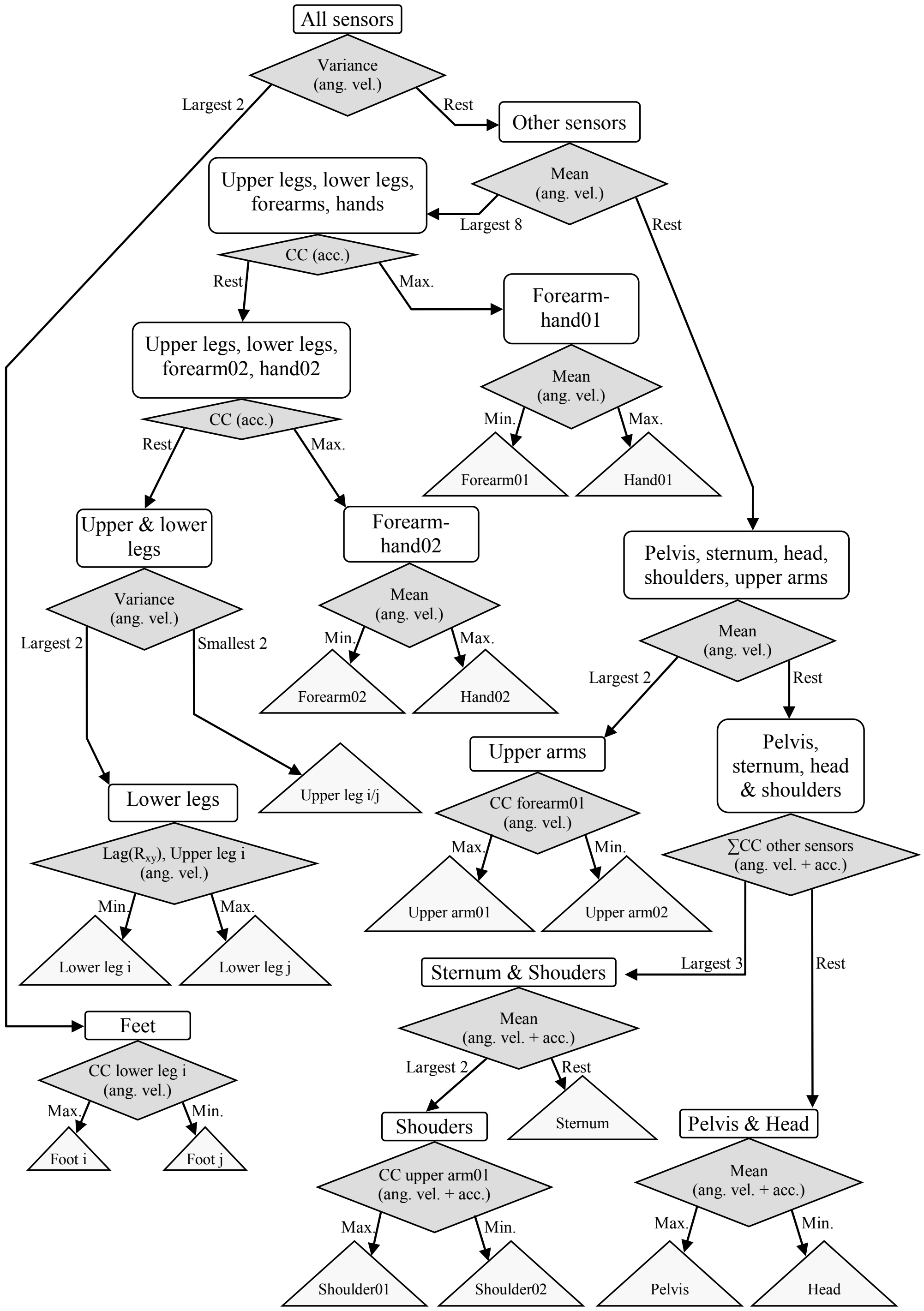


Figure 3.6: Decision tree used for identifying the sensors. By calculating the correlation coefficients (CC) between leg i/j and arm 01/02, it can be determined on which lateral side of the body the leg is on, side 01 or side 02 of the body. There is a larger correlation between a leg and the contralateral arm, than between a leg and the ipsilateral arm. Ang. vel. stands for angular velocity, acc. for acceleration of the inertial sensor(s).

arm and forearm on one lateral side of the body, than between contralateral upper- and forearms.

This leaves a group of five sensors (pelvis, sternum, head and shoulders) from which the sternum and shoulders are split, by calculating (for each sensor in this group of five sensors) the sum of the correlation coefficients with all other sensors (in this group of five sensors). The largest three values correspond with the sternum and the shoulder sensors. The sum of the correlation coefficients is calculated to get an impression of the correlation of a sensor with respect to all the other sensors and to preserve the distinction between positive and negative correlation coefficients. Negative correlation coefficients indicate an anti phase between segments, e.g. the correlation coefficient of the magnitude of the sensor acceleration between the left and right foot in Figure 3.5(b) is -0.27. Another way to get an indication of the correlation of a sensor with respect to the other sensors would be to take the sum of the absolute values, but then this information about signals in antiphase is neglected.

Next the mean of the angular velocity and the acceleration is calculated. The largest two sensors are the shoulder sensors, the smallest is the sensor on the sternum. To distinguish between left and right shoulder, the correlation with the upper arm is calculated.

Finally, the sensor on the pelvis can be identified by calculating the maximum of the mean of the angular velocity as well as the acceleration signals. The remaining sensor is the one on the head.

By calculating the correlation coefficients (CC) between leg i/j and arm 01/02, it can be determined whether the leg is on lateral side 01 or side 02 of the body. There is a larger correlation between contralateral legs and arms, than between ipsilateral legs and arms.

3.5 Results of the decision tree

The decision tree is now tested for eleven subjects. Although not all correlations, correlation coefficients and mean and variances can be shown here, the identification process worked correctly for eight subjects. For three subjects there were some problems identifying the sensors, the results are shown in Table 3.1. For subject 9 the only problems were to distinguish left from right shoulder and left from right upper arm. Subjects 8 and 11 also showed some sensors that were identified completely wrong.

Table 3.1: Identified sensor numbers for the three subjects that could not be identified correctly by the decision tree. The sensors that are not correctly identified are indicated with a *. If there was a problem distinguishing left from right this is indicated with **.

	Identified sensors		
Body segment	Subject 8	Subject 9	Subject 11
Pelvis	5*	1	1
Sternum	2	2	2
Head	3	3	3
Right shoulder	4	8**	4
Right upper arm	9*	9**	5
Right forearm	6	6	10**
Right hand	7	7	11**
Left shoulder	8	4**	8
Left upper arm	1*	5**	6*
Left forearm	10	10	9*
Left hand	11	11	7**
Right upper leg	15**	12	12
Right lower leg	16**	13	13
Right foot	17**	14	14
Left upper leg	12**	15	15
Left lower leg	13**	16	16
Left foot	14**	17	17

3.6 Discussion

So far, the decision tree seems to work well, because from the eleven subjects only three subjects caused problems. These problems are due to the fact that these three subject did not walk as expected. One of these subjects had one arm hanging still, while the other two subjects showed an arm movement less than average. This caused problem in identifying the sensors correctly, especially the ones on the arms.

The first classifier that was developed, caused problem in identifying the sensors correctly because thresholds were used. Since the subjects all walk at a (slightly) different speed, mean and variance values change from all sensor, but especially of the sensors on the arms. This was also the case after normalizing these features. Instead of using threshold values, in the final version of the decision three, the n largest values of the sensor features are used for the decision making process. With this new decision tree the sensors on all the subjects that walked “normally”, i.e. with normal arm movement, were identified correctly.

3.7 Conclusions and recommendations

In this pilot study, a decision tree that can be used for identifying inertial sensors on the human body is presented. The decision tree is created by trial and error (a proof of concept), but was able to identify the sensors correctly, however considering the following constraints:

- All 17 inertial sensors are present and attached correctly to the body according to Table B.1 in Appendix B.
- The subject is walking "normally", i.e. with a normal, symmetric gait pattern, at normal speed, and with normal arm movement.
- The features used by the classifier are extracted from the complete walking trial, i.e. no segmentation is applied and the length of the trials might differ between subjects.
- The eleven subjects used for this pilot study are considered training data, the decision tree has to be tested with additional subjects (test data) to get a real impression of its accuracy.
- Left and right identification is not possible based on only magnitudes of inertial sensor data, only contra-/ipsilateral identification is possible.

Instead of creating a tree using trial and error, it is recommended to look into automated classifier algorithms. This is done in the next chapters of this thesis. The results are compared with this pilot study in Chapter 6.

Chapter 4

Measurement set-up and methods

In this Chapter the measurement set-up and methods are described. The measurement set-up, used to identify the inertial sensors is described in Section 4.1. Section 4.2 describes the methods to analyze the inertial sensor data and create a classifier with use of Weka.

4.1 Measurement set-up

Measurements were obtained partly from my internship at Xsens Technologies B.V. and partly from the Bachelor project of Pim Pellikaan [10]. In both cases walking trials were recorded using an Xsens MVN system (Xsens Technologies B.V.) [2, 13, 21, 22]. Three walking trials were recorded from three subjects wearing an MVN suit (measurements from internship), while 28 other walking trials were recorded from seven other subjects wearing an MVN system with (Velcro) straps (measurements from Pellikaan).

In both cases a full body configuration was used, i.e. 17 inertial sensors are placed on 17 different body segments as indicated in Figure 4.1 and listed in Table 4.1.

Both MVN systems consist of:

- 17 *MTx sensors* with an accelerometer range of 18 g, and a rate gyroscope range of 1200 deg/s.
- Two *Xbus Masters* (XM), delivering power to the MTx's and retrieving their data exactly synchronized.
- Two *Wireless Receivers* (WR-A), for handling the data traffic between the XMs and the PC. Each WR-A is connected to a USB port.

The sampling frequency (F_s) used for the measurements was 120 Hz. The data was saved in an MVN file format, converted to XML and loaded into

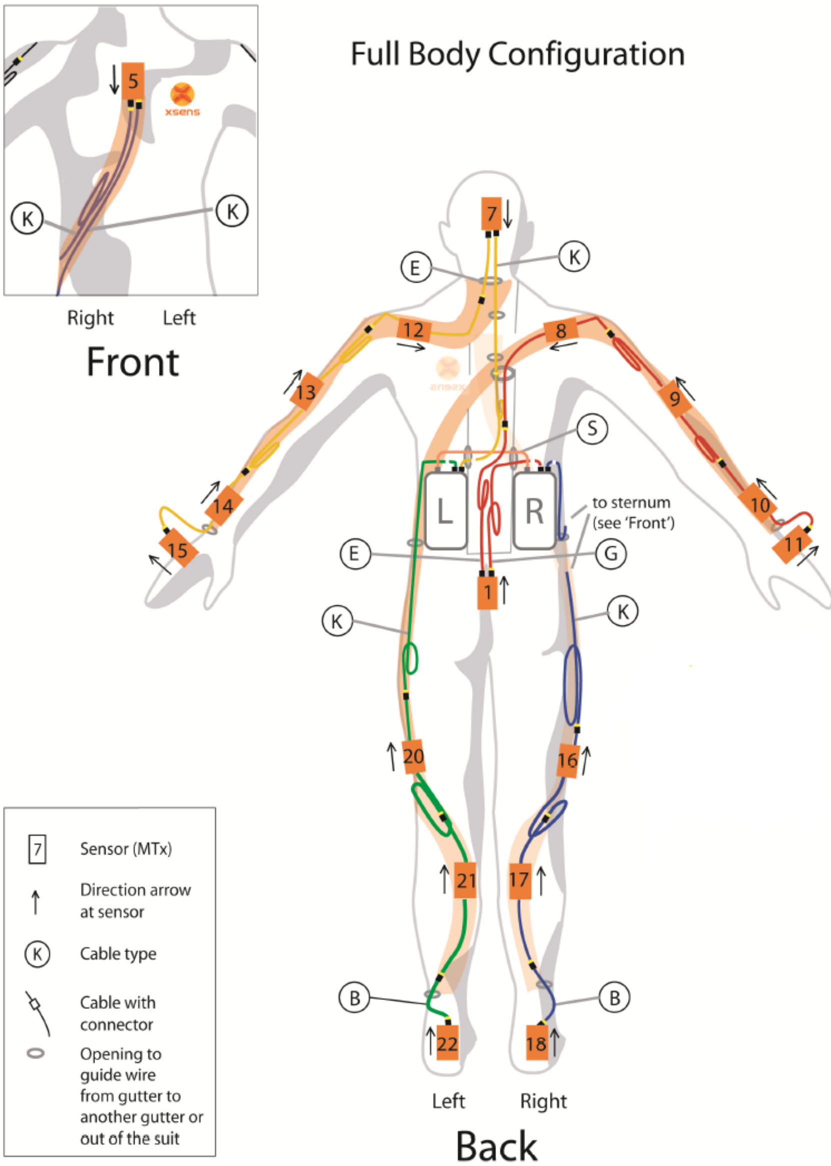


Figure 4.1: Locations of the 17 inertial sensors of the Xsens MVN motion capture suit. The sensor location ID numbers are listed in Table 4.1. Different lengths of cables are represented by "cable types" (besides the lengths, the cables all are identical except for the sync cable (S) which has four pins instead of five). Adapted from *Xsens MVN full body configuration sheet* [21].

MATLAB for further analysis.

From the walking trials the last frames are removed because of ending effects, i.e. some of the subjects were turning around at the end and started walking back. The first frames, where the subject is standing still are used for determining the initial sensor orientations (by measuring the gravitational accelerations), so these are kept.

Table 4.1: Measured body segments and their location ID numbers. See Figure 4.1 for a visualization. An alternative numbering, from 1 till 17, can also be used (see Table B.1 in Appendix B).

Location ID	Body segment
1	Pelvis
5	Sternum
7	Head
8	Right shoulder
9	Right upper arm
10	Right forearm
11	Right hand
12	Left shoulder
13	Left upper arm
14	Left forearm
15	Left hand
16	Right upper leg
17	Right lower leg
18	Right foot
20	Left upper leg
21	Left lower leg
22	Left foot

4.2 Methods

4.2.1 Preprocessing

The rotation of the sensor data to the global frame and the subtraction of the gravitational acceleration is done as described in Section 2.1.2, first the initial orientation of the sensors is estimated using the accelerometer and next the change of orientation is estimated by integration of the angular velocity. These are combined to come to a 3D rotation matrix which can be used to express the accelerations and angular velocities in global coordinates (see Figure 4.2 for an example of the change of coordinates for the sensor on the right foot).

After this, the heading (i.e. the angle about the vertical or z-axis) has to be aligned between the subjects, because not all the subjects are walking in the same direction. This is done by aligning the walking direction with the positive x-axis. The walking direction is obtained by integrating the acceleration in the global frame, yielding the change of velocity. This is done using trapezoidal numerical integration. See Figure 4.3(a) for an example of the velocity of the sensor on the pelvis. From the velocity, the x and y components are used to obtain the angle with the x-axis (in the horizontal plane). Because a lot of drift is showing up after integrating the accelerations, the average of the velocity of the first full walking cycle is used to estimate the walking direction. This is done using the peak detection function of MATLAB.

The angle θ (in the horizontal plane) between the velocity vector $\mathbf{v}$ and the positive x-axis (a vector $\mathbf{x}$ from the Origin to the point (1,0), in the horizontal plane, is used) can be obtained using:

$$\theta = \arccos \frac{\mathbf{x} \cdot \mathbf{v}}{\|\mathbf{x}\| \|\mathbf{v}\|}. \tag{4.1}$$

This angle is then used to obtain the rotation matrix in Equation 4.2, which can be used to rotate the accelerations, angular velocities and angular accelerations of all the sensors counterclockwise about the z-axis, so that all sensors are aligned.

$$R_z(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.2}$$

From the 3D angular velocities (in the global frame) the angular acceleration is calculated simply by differentiating the x-, y-, and z-components to the sample time ($1/F_s$). This is done because it gives (new) information about the change of angular velocity. So we now have, 3D accelerations, 3D angular velocities, and 3D angular accelerations. From these signals the magnitudes are calculated as already described in Chapter 3 and shown in Figure 3.1.

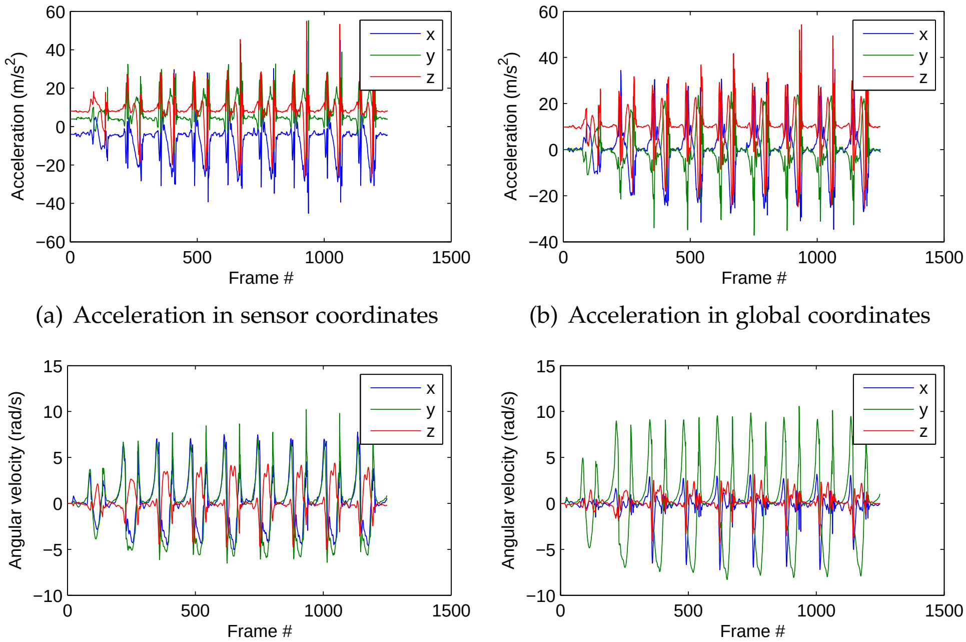
4.2.2 Feature extraction

Features are extracted with MATLAB, from both magnitudes as well as from the x-, y-, and z-components of the 3D accelerations, angular velocities and angular accelerations.

The features that are extracted are:

- Mean
- Variance
- Correlation coefficients between (components of) sensors
- Inter-axis correlation coefficients

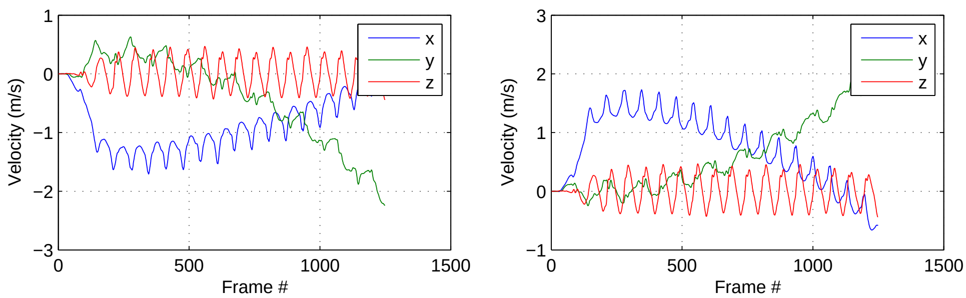
The mean and variance were already explained in Section 3.3, they are used in the same way here (from the x-, y-, and z-components the root mean square values are used now). Because the correlation coefficients are two dimensional, i.e. they are calculated between two sensors, they can not be inserted directly as features (because the location of the sensors is unknown). This is why as features, the sum of the cc's (of the magnitude, x-, y-, or z-component) of a sensor with (magnitudes, x-, y-, or z-components of) all other sensors is used and the maximum value of the cc's (of the magnitude, x-, y-, or z-component) of a sensor with (magnitudes, x-, y-, or z-components of) the other sensors. So from the correlation matrix the sums of the rows and the maximum values of each row (when neglecting the autocorrelations, i.e. the



(a) Acceleration in sensor coordinates (b) Acceleration in global coordinates

(c) Angular velocity in sensor coordinates (d) Angular velocity in global coordinates

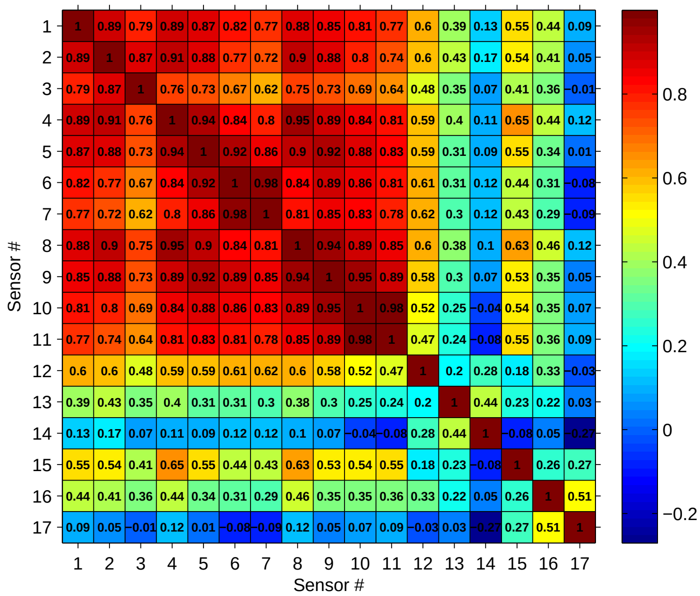
Figure 4.2: Accelerations (a and b) and angular velocities (c and d) of the sensor an the right foot during walking, expressed in sensor coordinates (a and c) and global coordinates (b and d). From the acceleration in the global frame (b), the gravitational component can be subtracted easily from the z-component.



(a) Before aligning walking direction (b) After aligning walking direction

Figure 4.3: Change of velocity of the sensor on the pelvis before and after aligning the walking direction with the x-axis. The change of velocity is obtained by integrating the measured acceleration (after rotating it to the global frame and subtracting the gravitational acceleration). As can be seen, there is a lot of drift showing up after integrating the accelerations. This is why only the first full walking cycle is used for determining the walking direction.

diagonal of the correlation matrix) are used as features. This is done to get an impression of the correlation of a sensor with all other sensors (see also Section 3.4.3 in the pilot study, were it turned out to be a useful feature). To visualize this, the correlation matrix of Figure 3.5(b) is repeated in Figure 4.4, with the extracted features in a Table next to it.



#	Sum	Max
1	10.54	0.89
2	10.56	0.91
3	8.92	0.87
4	10.94	0.95
5	10.62	0.94
6	10.19	0.98
7	9.78	0.98
8	11.00	0.95
9	10.67	0.95
10	10.26	0.98
11	9.81	0.98
12	7.25	0.62
13	4.78	0.44
14	1.75	0.44
15	6.76	0.65
16	5.48	0.51
17	1.41	0.51

Figure 4.4: Correlation coefficients of the magnitude of the sensor acceleration of subject 1, while walking at normal speed (left). In the Table (right) the features that are used as an input for the Weka classifier are shown, i.e. the sum and the maximum value of each row.

Because the inter-axis correlation coefficients are between axes of one sensor these features can be used directly. Because the x-axis is aligned with the walking direction (so the y-axis is pointing to the left) these features can be useful for the left and right identification. The idea behind it is that because the walking directions are aligned with the x-axis, segments on one lateral side of the body show a higher correlation between the accelerations in (or angular velocities about) the y-axis (pointing to the left) and the x- or z- axis than the same segment on the contralateral side. The right hand for example, moves (when walking) a bit to the left (positive y-axis) when swinging forward (in positive x-direction), while the left hand moves to the right (negative y-axis) when swinging forward. In the ideal case, when walking is symmetrical, the inter-axis correlation coefficients are opposite in sign for contralateral segments. Because in reality walking is not symmetrical, a ranking is applied.

All correlation coefficients are calculated using Equation 3.3 in Section 3.3.

4.2.3 Weka inputs and settings

The extracted features are saved as comma-separated value (CSV) files, which are then included in an *Attribute-Relation File Format ARFF*. These ARFF files can be opened with *Weka Explorer* and the data can be classified easily. An example of an ARFF file is shown in Listing 4.1. The dataset consists of 17 instances (rows), with 17 classes (the 17 sensors of a walking trial of one subject) and four features (columns).

Listing 4.1: Example of an arff-file

```
@RELATION features

@ATTRIBUTE meanAcc          REAL
@ATTRIBUTE meanAngVel       REAL
@ATTRIBUTE varianceAcc      REAL
@ATTRIBUTE varianceAngVel   REAL
@ATTRIBUTE class {Pelvis,Sternum,Head,RightShoulder,RightUpperarm,
    RightForearm,RightHand,LeftShoulder,LeftUpperarm,LeftForearm,LeftHand,
    RightUpperleg,RightLowerleg,RightFoot,LeftUpperleg,LeftLowerleg,
    LeftFoot}

@DATA
0.62045,0.19154,0.098683,0.032244,Pelvis
0.6023,0.16805,0.10002,0.037977,Sternum
0.60179,0.13426,0.09548,0.0612,Head
0.60642,0.17405,0.086246,0.04128,RightShoulder
0.61391,0.3252,0.082787,0.063497,RightUpperarm
0.63318,0.49069,0.069966,0.11761,RightForearm
0.64761,0.51977,0.076807,0.12213,RightHand
0.60887,0.16814,0.090881,0.040926,LeftShoulder
0.61647,0.38079,0.07918,0.066509,LeftUpperarm
0.653,0.62723,0.065441,0.15744,LeftForearm
0.68585,0.68282,0.079303,0.17788,LeftHand
0.68469,0.52385,0.14933,0.086714,RightUpperleg
0.81388,0.80248,0.34584,0.41,RightLowerleg
1,0.96028,0.91929,1,RightFoot
0.69063,0.54148,0.16018,0.10126,LeftUpperleg
0.81744,0.79278,0.32029,0.37025,LeftLowerleg
0.99914,1,1,0.96884,LeftFoot
```

The datasets that are used for developing the decision tree contain instances of 31 walking trials, so 17*31=527 rows. The 57 features that are used are listed in Table 4.2. All the 57 features are given as input to the decision tree learner, because the C4.5 algorithm automatically chooses the features that split the data most effectively (no dimensionality reduction is needed). So the dataset is a matrix of 527 rows and 57 columns. The features are ranked, to create ordinal features, so they are invariant for different walking speeds between trials (using threshold values instead of ranked features caused problems in the pilot study, see Section 3.4.2). For example, for one trial, the accelerations and angular velocities of the sensors on the feet always

Table 4.2: Features used for identifying the inertial sensors. All the (19×3=) 57 features are given as input to the decision tree learner, because the C4.5 algorithm automatically chooses the features that split the data most effectively.

	Feature name		
Description	Acceleration	Angular velocity	Angular acceleration
Mean of the			
-magnitude	MeanAccNorm	MeanAngVelNorm	MeanAngAccNorm
-x-xomponent	MeanAccX	MeanAngVelX	MeanAngAccX
-y-xomponent	MeanAccY	MeanAngVelY	MeanAngAccY
-z-xomponent	MeanAccZ	MeanAngVelZ	MeanAngAccZ
Variance of the			
-magnitude	VarianceAccNorm	VarianceAngVelNorm	VarianceAngAccNorm
-x-xomponent	VarianceAccX	VarianceAngVelX	VarianceAngAccX
-y-xomponent	VarianceAccY	VarianceAngVelY	VarianceAngAccY
-z-xomponent	VarianceAccZ	VarianceAngVelZ	VarianceAngAccZ
Sum of the Pearson correlation coefficients (cc's) of a sensor with all other sensors of the			
-magnitude	CcAccOtherNorm	CcAngVelOtherNorm	CcAngAccOtherNorm
-x-xomponent	CcAccOtherX	CcAngVelOtherX	CcAngAccOtherX
-y-xomponent	CcAccOtherY	CcAngVelOtherY	CcAngAccOtherY
-z-xomponent	CcAccOtherZ	CcAngVelOtherZ	CcAngAccOtherZ
The maximum value of the cc's of a sensor with all other sensors of the			
-magnitude	CcAccMaxNorm	CcAngVelMaxNorm	CcAngAccMaxNorm
-x-xomponent	CcAccMaxX	CcAngVelMaxX	CcAngAccMaxX
-y-xomponent	CcAccMaxY	CcAngVelMaxY	CcAngAccMaxY
-z-xomponent	CcAccMaxZ	CcAngVelMaxZ	CcAngAccMaxZ
The inter-axis cc's of a sensor between the			
-x- and y-axes	CcAccXY	CcAngVelXY	CcAngAccXY
-x- and z-axes	CcAccXZ	CcAngVelXZ	CcAngAccXZ
-y- and z-axes	CcAccYZ	CcAngVelYZ	CcAngAccYZ

have the largest mean and variance (see Figures 3.2 and 3.3), but this is not always the case when comparing the mean and variance of sensors of other trials. This ranking process of categorizing the features is a form of classification and can only be used when the number of sensors is known beforehand (in this case it is known that 17 inertial sensors are used). A drawback of this ranking process is that the distance between the feature values (and thus the physical meaning) is removed.

The identification is split into two steps. In the first step the body segments are classified **without** looking at left or right (or contra-/ipsilateral). In the second step the left and right indentification is done, because this caused some problems with the upper arms. This is resolved by identifying the left and right upper arms at the end, by means of the correlation with the forearms.

After opening the ARFF file in *Weka explorer*, the J4.8 decision tree classifier is chosen. Most of the options of this classifier, e.g. different options for pruning are unchanged. The only option that is changed in some cases is the *minNumObj* parameter. This parameter sets the minimum number of instances permissible at a leaf. In some cases when the tree became too large, i.e. when the tree had leaves that only classified two or three instances, the *minNumObj* parameter is changed from 2 (the default value) to 20.

As a test option a 10-folds cross-validation is chosen because in literature it is proven to be a good estimate of predicting the error rate for many problems (it has become the 'standard') (see Section 2.2.2).

4.2.4 Weka outputs

Weka produces a lot of output information. First a summary of the input dataset and the chosen test mode is given. After this, Weka lists the classifier model in textual form. This decision tree classifier can also be visualized graphically. The features that split the data and the class labels for each leaf are shown, followed by the number of instances that reach a leaf. The number of incorrectly classified instances also show up in the leafs.

The next part of the weka output gives estimates of the classifier's predictive performance. The number (and percentages) of correctly and incorrectly classified instances are shown, followed by a detailed accuracy by class and a *confusion matrix*. This is a matrix with a column and a row for each class. Each element shows the number of test examples for which the actual class is the row, and the predicted class is the column. Ideally, the elements on the diagonal are large numbers and the other elements are zero.

4.2.5 Other sensor configurations

In some cases it is not necessary (or even unwanted) to measure a subject using the full body configuration. In these cases lower- or upper body configurations can be used.

For the lower body configuration only 7 sensors are used: the sensor on the pelvis and the sensors on the left and right upper- and lower legs and the feet (sensor numbers: 1, 12, 13, 14, 15, 16, and 17). For the upper body configuration 11 sensors are used: the sensors on the pelvis, sternum and head and the sensors on the left and right shoulders, upper- and forearms and hands (sensor numbers: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, and 11).

Two separate decision trees will be derived for these configurations. For deriving these trees, the methods used for deriving the tree for the full body configuration are used (but the rankings are now from 1 to 7 and from 1 to 11 instead of from 1 to 18 and the correlation coefficient matrices are also smaller). Before starting a new measurement it has to be clear which configuration is used, so the corresponding tree for sensor identification can be used.

Chapter 5

Results

In this Chapter the results of the identification of the inertial sensors are presented. First the full body configuration is described, followed by the upper- and lower body configurations. These results are discussed in the next Chapter.

5.1 Full body configuration

As described in Section 4.2.3, the identification is split into two steps. In the first step the body segments are classified without looking at left or right. These results are shown in Section 5.1.1. The results of the second step, the distinction between left and right, are shown in Section 5.1.2.

5.1.1 Identifying the sensors

The first step is to identify the sensors without making a distinction between left or right. A J4.8 decision tree classifier is developed with Weka and shown in Figure 5.1. The corresponding confusing matrix is shown in Table 5.1. As testing option 10 fold cross validation is used. From the (31*17=) 527 inertial sensors, 523 are correctly classified (99.24%).

Because the features are ranked, the decision making is based on these rankings. For example, when looking at the top of the decision tree (at the first split) the 7 sensors (of each trial) with the smallest mean magnitude of the angular velocity (MeanAngVelNorm) are separated from the rest. These are the sternum, head, shoulders, pelvis and upper arms. So the other 10 sensors of each walking trial are the hands, forearms, upper legs, lower legs, and feet. In Figure 5.2 box plots of the values of this feature for the sensors of all the 31 walking trials are shown.

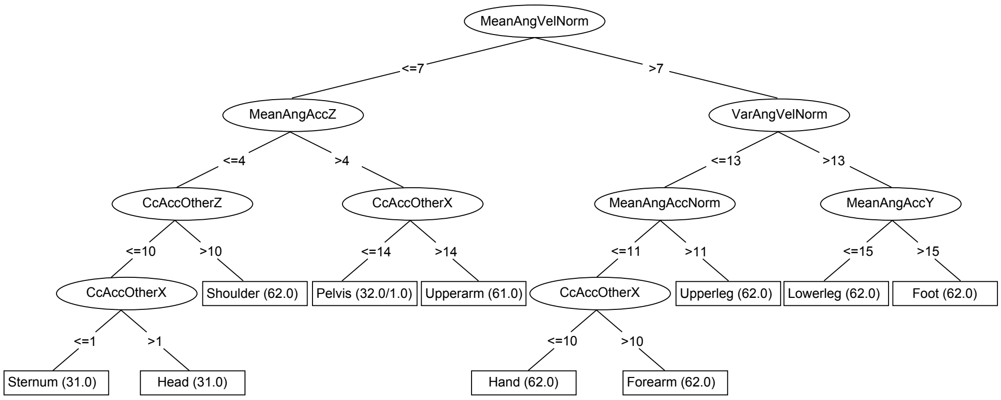


Figure 5.1: Decision tree created with the J4.8 algorithm of Weka. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. From the (31*17=)527 inertial sensors, 523 are correctly classified (99.24%).

Table 5.1: Confusion Matrix of the J4.8 algorithm of Weka. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. From the (31*17=)527 inertial sensors, 523 are correctly classified (99.24%).

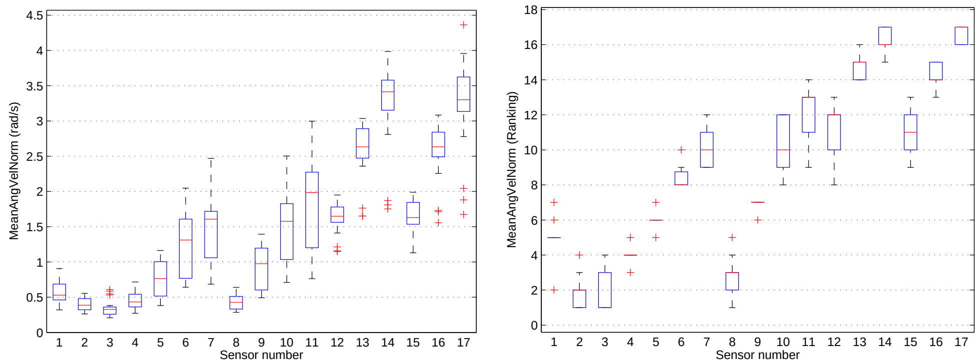
a	b	c	d	e	f	g	h	i	j	<— classified as
31	0	0	0	0	0	0	0	0	0	a = Pelvis
0	31	0	0	0	0	0	0	0	0	b = Sternum
0	0	31	0	0	0	0	0	0	0	c = Head
0	0	1	61	0	0	0	0	0	0	d = Shoulder
1	0	0	0	61	0	0	0	0	0	e = Upperarm
0	0	0	0	0	62	0	0	0	0	f = Forearm
0	0	0	0	0	2	60	0	0	0	g = Hand
0	0	0	0	0	0	0	62	0	0	h = Upperleg
0	0	0	0	0	0	0	0	62	0	i = Lowerleg
0	0	0	0	0	0	0	0	0	62	j = Foot

Box plots of the other features that are used by the decision tree are shown in Figure 5.3. The CcAccOtherX-feature is used three times.

5.1.2 Left and right identification

The next step is the left and right identification for which inter-axis correlation coefficients are used. This is also done using Weka. In Figure 5.4(a) the decision trees used for left and right identification for all the segments, except the upper arms, are shown. As testing option a 10 fold cross-validation is used. For all these segments all sensors were identified correctly (100% accuracy).

For the upper arms the correlations with the forearms are inserted in an



(a) Actual feature values (b) Ranked features

Figure 5.2: Box plots of the MeanAngVelNorm features of the sensors of all the 31 walking trials. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers. Outliers are plotted as red crosses. Left are the actual feature values. As can be seen, there is a lot of overlap in the values of the features. On the right the features are ranked (see Section 4.2.3 for details) and it can be seen that now there is a possible split between ranking 7 and 8, which is also the first split used by the decision tree in Figure 5.1.

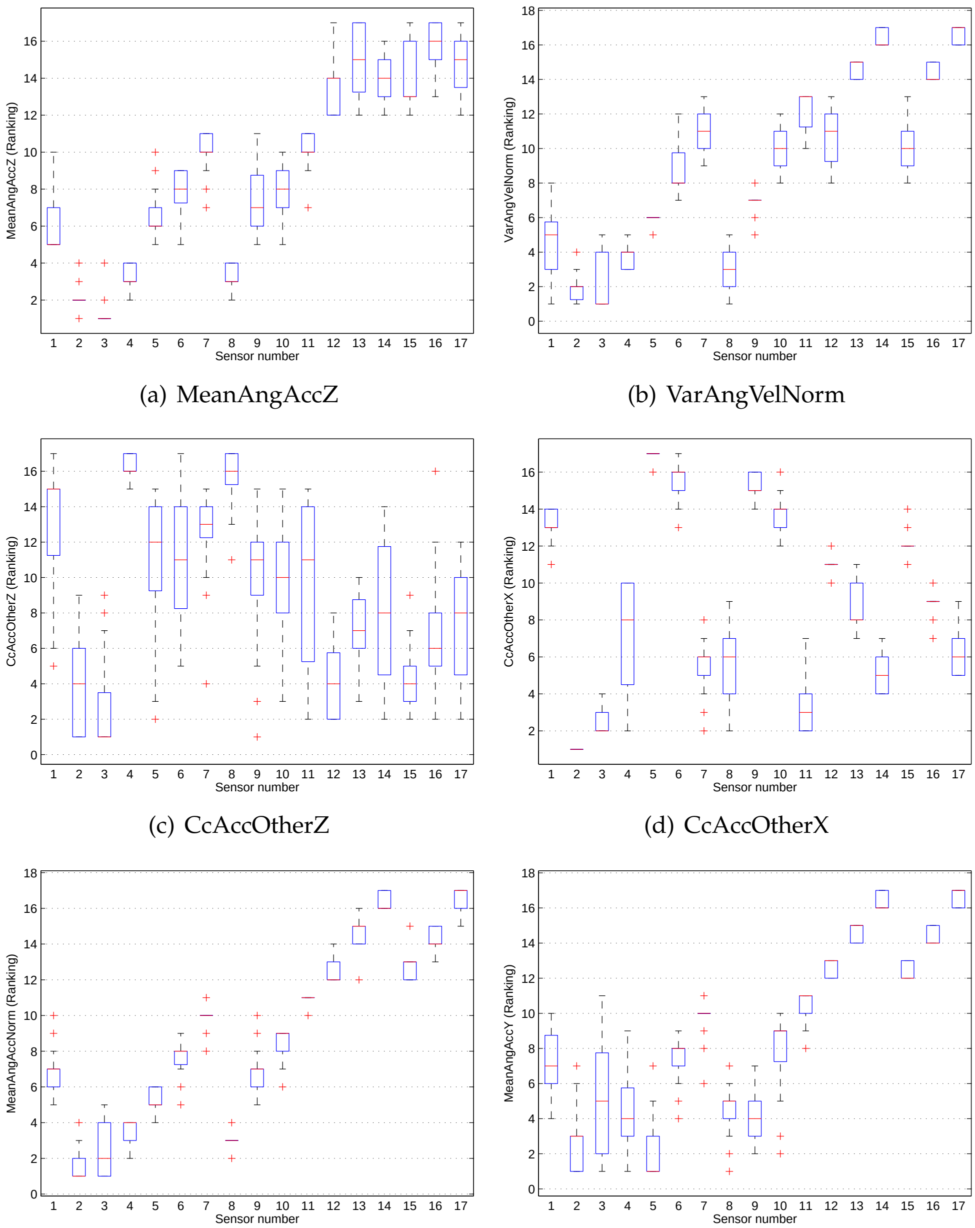
ARFF file and classified with the decision tree in Figure 5.5. This can only be done when the left and right forearm are identified, so an extra (third) step is required.

5.2 Upper body configuration

The decision tree for identifying the inertial sensors of an upper body configuration is shown in Figure 5.6. All sensors were identified correctly, so the confusion matrix is not shown. For the left and right identification the Figures 5.4(a), 5.4(b), 5.4(c) and Figure 5.5 can be used.

5.3 Lower body configuration

The decision tree for identifying the inertial sensors of a lower body configuration is shown in Figure 5.7. For this configuration also all sensors were identified correctly. For the left and right identification the Figures 5.4(d), 5.4(e), 5.4(f) can be used.



(a) MeanAngAccZ

(b) VarAngVelNorm

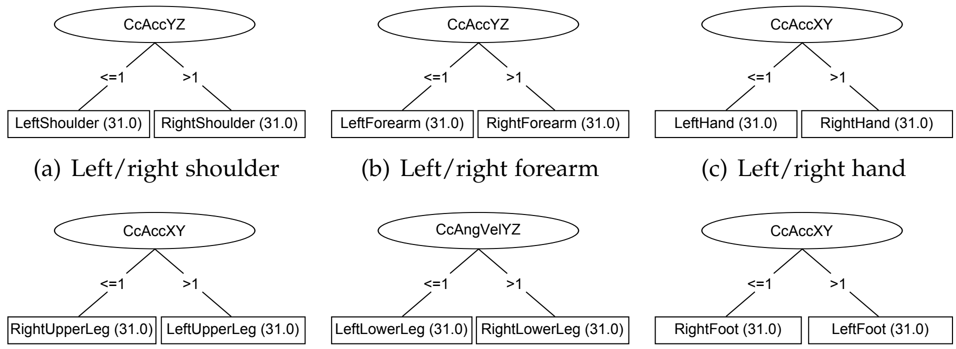
(c) CcAccOtherZ

(d) CcAccOtherX

(e) MeanAngAccNorm

(f) MeanAngAccY

Figure 5.3: Box plots of all the features used by the decision tree in Figure 5.1. The CcAccOtherX-feature (Figure 5.3(d)) is used three times.



(a) Left/right shoulder (b) Left/right forearm (c) Left/right hand

(d) Left/right upper leg (e) Left/right lower leg (f) Left/right foot

Figure 5.4: Decision trees for identifying the left and right body segments, using inter-axis correlation coefficients. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. For all these segments all sensors were identified correctly (100% accuracy). The idea behind this is that because the walking directions are aligned with the x-axis, segments on one lateral side of the body show a higher correlation between the accelerations in (or angular velocities about) the y- and the x- or z- axis than the same segment on the contralateral side, as described in Section 4.2.2. In the ideal case, when walking is symmetrical, the inter-axis correlation coefficients are opposite in sign for contralateral segments.

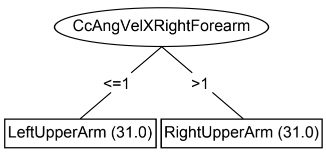


Figure 5.5: Decision tree for identifying the left and right upper arms. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. All sensors were correctly classified. As can be seen, the upper arm with the largest correlation with the (angular velocity about the x-axis of the) right forearm (which is identified in the previous step, see Figure 5.4(b)) is classified as a right upper arm.

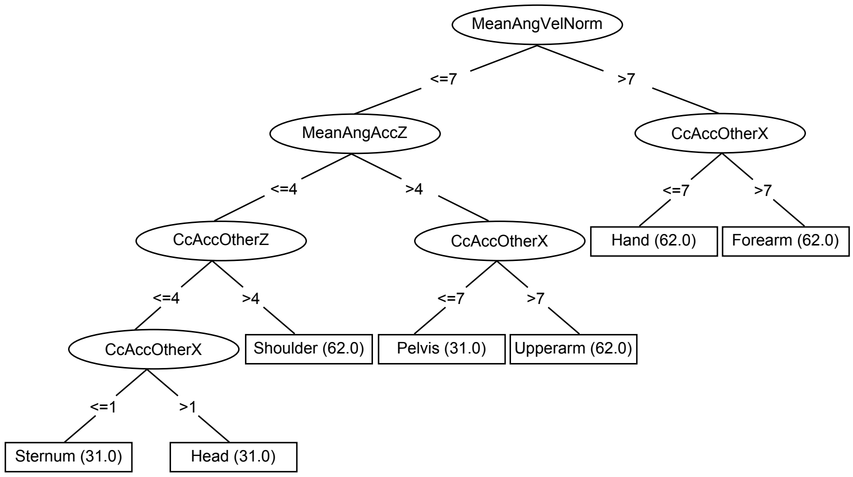


Figure 5.6: Decision tree for identifying the inertial sensors using an upper body configuration. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. All sensors were identified correctly.

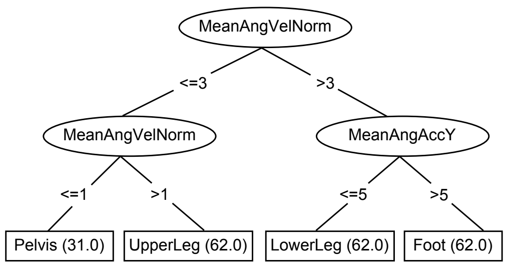


Figure 5.7: Decision tree for identifying the inertial sensors using a lower body configuration. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. All sensors were identified correctly.

Chapter 6

Discussion

In this Chapter the results presented in the previous Chapter are discussed.

6.1 The features chosen by the J4.8 algorithm

The features that are used to most effectively split the data were shown in the Figures 5.2 and 5.3. As described in Section 2.2.2, these features are chosen based on information gain. Ranking the features is necessary, because there is a lot of overlap in the features between different walking trials, as was shown in Figure 5.2 and explained in detail in Section 4.2.3. A drawback of this ranking process is that the number of sensors has to be known beforehand.

6.2 Accuracy of the classifier

For trials starting from standing still and ending with normal walking, the accuracy of the decision tree for a full body configuration is 99.24 %, including left and right identification. For upper- or lower body configurations all the sensors were identified correctly (100% accuracy).

In Weka Explorer, other classifiers can be chosen easily. Results for a decision table and a neural network are shown in Appendix D. The decision table performs less then the decision tree presented in the previous Chapter. With use of a neural network, all sensors are identified correctly, however the result is very complex and the computation time is very high[1], i.e. about 88 seconds, compared to 0.2 to 0.3 seconds for training and testing a decision tree; furthermore, a neural network gives no insight in the rules that are used. This is why these two methods are not suitable for identifying the inertial sensors.

[1]Calculations were performed using an Intel Core i7-720QM quad core processor (1.6 GHz with Turbo Boost Technology to 2.8 GHz).

Comparing more classifiers might give other insights, however, due to time limitations this is recommended for future work. The results for the decision tree classifier are over 99 % and the computation time is between 0.2 and 0.3 seconds. To implement this signal classification scheme only 10, relatively easy to extract, signal features need to be compared.

6.3 Other test-train options

Instead of using the 10 fold cross-validation, different training options can be used. For instance a different number of folds or a percentage split can be used (see Section 2.2.2). Results of the decision tree classifier when using different numbers of folds for cross-validation are shown in Table 6.1. Using

Table 6.1: Accuracy of the decision tree classifier when using different numbers of folds for cross-validation. The total number of instances to classify is 527.

Folds	Correctly classified instances
2	519 (98.48 %)
3	519 (98.48 %)
4	525 (99.62 %)
5	523 (99.24 %)
6	525 (99.62 %)
7	525 (99.62 %)
8	525 (99.62 %)
9	523 (99.24 %)
10	523 (99.24 %)
11-20	525 (99.62 %)

a Wilcoxon rank sum test with a 5 % significance level, it is calculated that the accuracy of the decision tree classifier is significantly worse when using the 2 or 3 fold cross-validation (98.48 %) then when using an other number of folds. There is no significant difference between the 99.24 and the 99.62 % accuracy values, so using a number of folds between 4 and 20 does not result in significantly different performances (which was described in Section 2.2.2).

When a percentage split is used, with 66 % of the data used for training and the remainder for testing, 176 instances (of a total of 179) are correctly classified (98.32 %). Altough the percentage is lower, the number of incorrectly classified instances is the same as when using the cross-validation method with different numbers of folds.

Compared to the result when using all the data for training the tree, (526 instances (of a total of 527) are correctly classified (99.81 %), these result are all relatively accurate, so it is likely that the decision tree performs well on new data.

6.4 Left and right identification

The left and right identification requires a second step in the classification process, because of the rankings of the inter-axis correlation coefficients. After the first step, a new ranking has to be determined for identifying the left and right of each body segment (see Figure 5.4). This is done because the inter-axis correlation coefficients are overlapping between different subject when using all the sensors, but for one segment as identified in step 1 (for example the hands) the ranking between left and right is clear, e.g. the correlation coefficient between the acceleration of the x- and y-axis is always larger for the right hand, than it is for the left hand (see Figure 5.4(c)). In the ideal case, when walking symmetrically, the inter axis correlation coefficients are opposite in sign for contralateral segments (see explanation in Section 4.2.2).

The upper arms caused some problems, because the inter-axis correlation coefficients were not sufficient to make a distinction between left and right. Therefore the correlation with the forearms is used here. This results in an extra step in the identification process, because after all other segments are identified, a final step is needed to identify the upper arms.

6.5 Comparison with the Pilot Study

Compared to the Pilot Study in Chapter 3 the classifier developed with Weka performs better. 1 of the 527 instances is incorrectly classified (0.19 % error), compared to 18 of the 187 instances in the Pilot Study (9.6 % error), which is also without left and right identification. Because 10 fold cross-validation is used, the prediction error will be even better when testing the tree with a new dataset.

The tree developed with Weka performs better, because 3D information is used, instead of only the magnitudes of the 3D values. Also inter-axis correlation coefficients were used, so the left and right identification is possible now, while in the Pilot Study only contra-/ipsilateral identification was possible.

When the same features that were used in the Pilot Study are entered in Weka, so features extracted only from magnitudes of accelerations and angular velocities, 95.07 % of the sensors is classified correctly (no left and right identification).

6.6 Accuracy of the change of coordinates

To obtain the rotation matrix $\boldsymbol{R}_s^g(t)$, angular velocities were integrated. To verify that no (significant) errors were made here, $\boldsymbol{R}_s^g(t)$ was plot over time. No significant trends were found, so it is assumed that the change of coordinates did not cause any errors in the sensor identification.

The walking direction is estimated using the average velocity of the first walking cycle. Because the change of velocity (the measurements are started when standing still, so this gives an indication of the walking speed) is obtained by integrating the acceleration, integration drift shows up. Because only the first walking cycle is used the velocity might be accurate enough, however, a recommendation would be to look into different techniques of walking direction estimation.

6.7 Varying sensor positions

In the user manual of the Xsens MVN system [22] the optimal positioning of the sensors, to minimize soft tissue artifacts, is described. This sensor position is also used in the measurements that are used for training the decision tree. But what is the influence of the sensor positions on the accuracy of the decision tree? Will the sensors be classified correctly if they are positioned on different positions?

To answer this question a decision tree without the (translational) acceleration features might be helpful, because on a rigid body the angular velocities (so also the angular accelerations) are considered to be the same everywhere on that rigid body. This tree and the corresponding confusion matrix are shown in Figure 6.1 and Table 6.2.

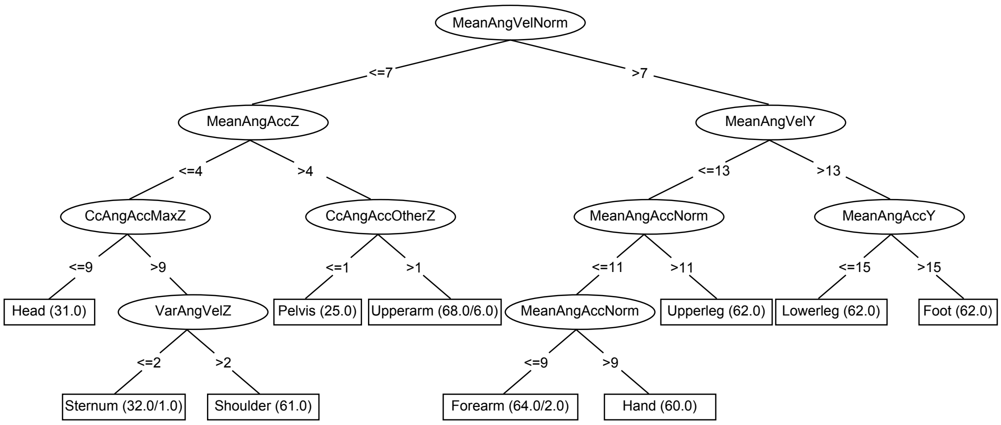


Figure 6.1: Decision tree created with the J4.8 algorithm of Weka when only angular velocity and angular acceleration features are used. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. 511 of 527 instances were classified correctly (96.96 %)

It can be seen that the distinction between pelvis and upper arm is causing a lot of problems.

For the left and right identification inter-axis acceleration features are

Table 6.2: Confusion Matrix of the J4.8 algorithm of Weka when only angular velocity and angular acceleration features are used. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. From the (31*17=)527 inertial sensors, 511 are correctly classified (96.96%).

a	b	c	d	e	f	g	h	i	j	<— classified as
23	0	0	0	8	0	0	0	0	0	a = Pelvis
0	30	0	1	0	0	0	0	0	0	b = Sternum
0	0	30	1	0	0	0	0	0	0	c = Head
0	1	0	61	0	0	0	0	0	0	d = Shoulder
3	0	0	0	59	0	0	0	0	0	e = Upperarm
0	0	0	0	0	62	0	0	0	0	f = Forearm
0	0	0	0	0	2	60	0	0	0	g = Hand
0	0	0	0	0	0	0	62	0	0	h = Upperleg
0	0	0	0	0	0	0	0	62	0	i = Lowerleg
0	0	0	0	0	0	0	0	0	62	j = Foot

used for the shoulders, forearms, hands, upper legs and feet. When these inter-axis acceleration features are left out of the datasets used for generating these decision trees, this caused no problem, except for the forearms, where 2 of the 62 sensors were incorrectly classified and for the hands, where 9 of the 62 instances were incorrectly classified. For these segments the correlations with the shoulders or the legs can then be used to distinguish left from right.

Overall the performance is still relatively good, because this is the minimum performance, where the positioning of the sensor is on a 'not-optimal' position on the body segment. To get a better impression of the influence of the sensor positions, additional measurements are required. This is another recommendation for future work.

6.8 Missing sensors

When some of the sensors are missing (and it is unknown which of the sensors), they can not be identified using the decision tree from Figure 5.1, because then the ranking is incorrect. Also the correlation coefficients between sensors can not be used anymore, because there are some sensors missing (and it is unknown which sensors are missing). When these features are left out the tree in Figure 6.2 is trained with Weka. The corresponding confusion matrix is shown in Table 6.3. The accuracy of the classifier is a lot less now, which was also explained in Figure 5.2, the box plot of the unranked MeanAngVelNorm. There is more overlap in the unranked features because of different walking speeds and or arm movements between different walking trials.

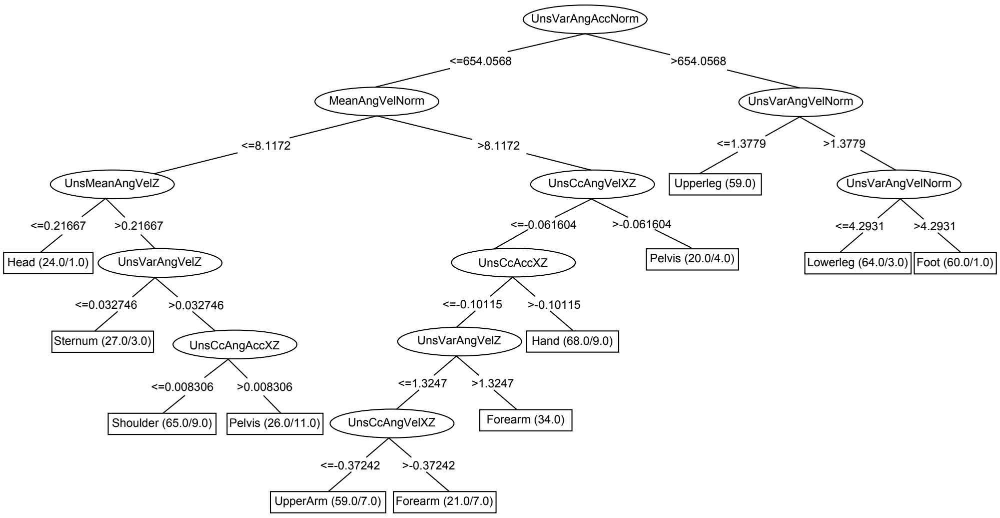


Figure 6.2: Decision tree for identifying the sensors, if not all the sensors are present. The values near the branches now represent the actual feature values. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. 418 of 527 instances were classified correctly (79.32 %).

If it is known which sensors are missing, another decision tree (without the missing sensors) can be used. These decision trees can be developed from the walking trials used in this study, a recommendation for future work.

6.9 Other daily-life activities

The decision tree created in the previous Chapter is based on walking trials (starting from standing still) only. Because it is the goal of the Fusion project to create a 'Click-On-and-Play' ambulatory 3D human motion capture system, it might be interesting to see the results for other daily-life activities. These activities could then be monitored using one of the techniques described in Section 2.3. Then, based on this information, the right decision tree for identifying the sensors can be chosen. The expectation is that the identification becomes more robust when combining the current classification method with other daily-life activities, for example when standing up from sitting the sensors on the upper legs rotate approximately 90°, which make these sensors easy to identify. Several new features might be needed when other activities are investigated.

Another, more preferable alternative, is to create one decision tree which can identify the inertial sensors for any arbitrary movement. This is specifically wanted for people with movement disorders, because in most cases they are not able to perform a certain predefined task, for example normal walking.

Table 6.3: Confusion Matrix of the tree in Figure 6.2. 31 walking trials of 10 different healthy subjects were used. As testing option a 10 fold cross-validation is used. From the (31*17=)527 inertial sensors, 418 are correctly classified (79.32%).

a	b	c	d	e	f	g	h	i	j	<— classified as
25	0	0	1	1	2	2	0	0	0	a = Pelvis
3	24	1	3	0	0	0	0	0	0	b = Sternum
1	8	16	6	0	0	0	0	0	0	c = Head
9	3	0	50	0	0	0	0	0	0	d = Shoulder
1	1	0	2	41	15	2	0	0	0	e = Upperarm
3	0	0	0	17	35	7	0	0	0	f = Forearm
3	0	0	0	1	3	54	0	1	0	g = Hand
1	0	0	0	0	2	0	59	0	0	h = Upperleg
0	0	0	0	0	0	0	1	57	4	i = Lowerleg
0	0	0	0	0	0	0	0	5	57	j = Foot

Due to time limitations, the results for other daily-life activities could not be investigated. This is one of the recommendations to investigate in future work.

6.10 Use in rehabilitation

The decision tree is trained with features extracted from walking trials of healthy subjects. It is not expected that the identification of the inertial sensors is also working for people with movement disorders or other uses in rehabilitation.

Another recommendation for future work is to test the decision tree on more subjects, also on subjects with movement disorders. But as described in the previous Section, ideally the inertial sensors are identified automatically during any arbitrary movement.

It is not always necessary (or even unwanted) to use a full body configuration. In these cases an upper- or lower body configuration can be used. The identification performance is even better in these cases: all sensors were correctly identified. This is because the sum of the correlation coefficients of a sensor with the other sensors (Section 4.2.2) is now only calculated from a subset of the sensors, so there is less influence/noise from the other (unused) segments.

Chapter 7

Conclusions and recommendations

7.1 Conclusions

In this master thesis a method for automatic identification of inertial sensors on the human body segments, i.e. the assessment of the body segment to which each inertial sensor is attached to, is presented. By comparing 10, relatively easy to extract features, the inertial sensors can be identified with an accuracy of 99.24 % given the following constraints:

- The subject is standing still for a moment, so the initial orientation in the global frame can be obtained, and then starts walking 'normally' and more or less straight on.
- All 17 inertial sensors are present and attached to (or near) the optimal positions.

The features can be extracted from magnitudes and 3D components of the measured accelerations and angular velocities and from the calculated angular acceleration, after rotating all signals to the global frame and aligning the walking directions with the positive x-axis.

Left and right identification is done using inter-axis correlation coefficients, except for the upper arms. These are identified with use of the correlation with the forearms.

Because 10 fold cross validation is used, it is likely that the decision tree performs well on new testing data. 10 folds are used because this became the standard, but different number of folds and a simple percentage split were also investigated. These other training options resulted in similar error rates (98.32-99.62 % correctly classified instances).

When sensors are missing or the sensors are not attached to the optimal body positions, identification is still possible, however the performance will be worse. When sensors are missing and it is unknown which sensors are missing the rankings cannot be used. Also the correlation coefficients between sensors can not be used anymore. The tree from Figure 6.2 is used in this case and 79.32 % of the sensors are identified correctly. If it is known which sensors are missing another decision tree, without the missing sensors, can be used (section 6.8). When the sensors are not attached to the optimal body positions, the performance can drop to a minimum of 96.96 %, because otherwise the decision tree which only uses the features extracted from the angular velocities and the angular accelerations can be used instead (see Figure 6.1).

If a full body configuration is not necessary, a lower- or upper body configuration can be used instead. In these cases all the sensors are correctly identified.

In a first Pilot Study (Chapter 3), a decision tree was developed by comparing the features manually and with trial and error. In this first study only magnitudes of accelerations and angular velocities were used and left and right identification appeared to be impossible. The accuracy of this first decision tree was 90.4 %. Because all data was used to train the tree, the performance on new data is probably worse.

Over all, the project goal, *develop a new method to automatically identify human body segments to which inertial sensors are attached during walking,* has been achieved (with an accuracy of 99.24 %), given the constraints mentioned before.

7.2 Recommendations

For future work it is recommended to improve the current method of identification of the inertial sensors, by creating a classifier for other daily life activities or even for random movements. Also the influence of the sensor position has to be investigated by measuring additional walking trials with varying sensor positions. Instead of healthy subjects, also the identification of the sensors on subjects with movement disorders needs some attention.

Several extra trees with missing sensors can be created to use instead of the 'normal' tree, when some of the sensors are missing and it is known which sensors are missing.

Another recommendation is to compare the results of this decision tree identification algorithm with other types of classification algorithms.

The integrated acceleration is used to estimate the walking direction of the subject. It is recommended to look into the influence of integration drift, or to look into other methods of estimating the walking direction.

Appendix A

Search databases and keywords

Table A.1 lists the search terms used for verifying that there are no publications on the automatic identification of inertial sensors so far. The search terms are inserted in the following databases:

- Google Scholar
- Scopus
- Web of Science database
- IEEE/IET Electronic Library (IEL)

Most of the results are about activity classification techniques. There is one hit on the automatic position recognition of sensors on the human body (Winter et al, 2008 [19]), but this method is not based on automatic classification using inertial sensor data, but on position recognition units which are placed on pre-determined body segments.

Table A.1: Search terms used in Google Scholar. Search is for articles and patents, since 2008.

"inertial sensor" "body segment"
"inertial sensor" "body segment" classification
"inertial sensors" "body segment" classification
"inertial sensor" "human body" classification
"inertial sensors" "human body" classification
"inertial sensor" "body segment" identification
"inertial sensors" "body segment" identification
"inertial sensor" "human body" identification
"inertial sensors" "human body" identification
"inertial sensor" "position recognition"
"inertial sensors" "position recognition"
"sensor position recognition"
"inertial sensors" "position estimation"
"sensor position estimation"
"inertial sensors" "human body" configuration
"inertial sensors" "body segment" configuration
"inertial sensors" "human body" self-configuration
"inertial sensors" "body segment" self-configuration
"inertial sensors" "human body" auto-configuration
"inertial sensors" "body segment" auto-configuration

Appendix B

MVN Biomechanical model and measured segments

In MVN a biomechanical model consisting of 23 segments is used (Figure B.1). These segments are considered to be rigid bodies and are connected by joins. The coordinate systems of each segment are defined as follows, assuming standing in the anatomical pose [8, 11]:

Segment 1: Pelvis The pelvis has its origin in the midpoint between the right and left hip rotation centers (Figure B.2(a)). The X-axis is pointed to the front, while the y-axis points upwards, towards the joint that connects L5 (Lumbar 5) to S1 (Sacrum 1) (this joint is abbreviated as jL5S1, from now on this abbreviation will be used). The z-axis is pointing to the right.

Segments 2-5, Spinal segments: L5, L3, T12, T8 There are four spinal segments:

- segment 2: L5-L4 (Lumbar vertebrae)
- segment 3: L3-L1 (Lumbar vertebrae)
- segment 4: T12-T9 (Thoracic vertebrae)
- segment 5: T8-T1 (Thoracic vertebrae)

Not all of these segments are measured directly. Using a certain model of the spine they are interpolated between the pelvis, sternum and head sensors of the MVN suit. In Figure B.2(b) two of the segments of the spine are shown. The origins of the segments are in jL5S1, jL4L3, jL1T12, jT9T8 respectively. The x-axis is pointing to the front, the y-axis points upwards (to the next joint) and the z-axis is pointing to the right.

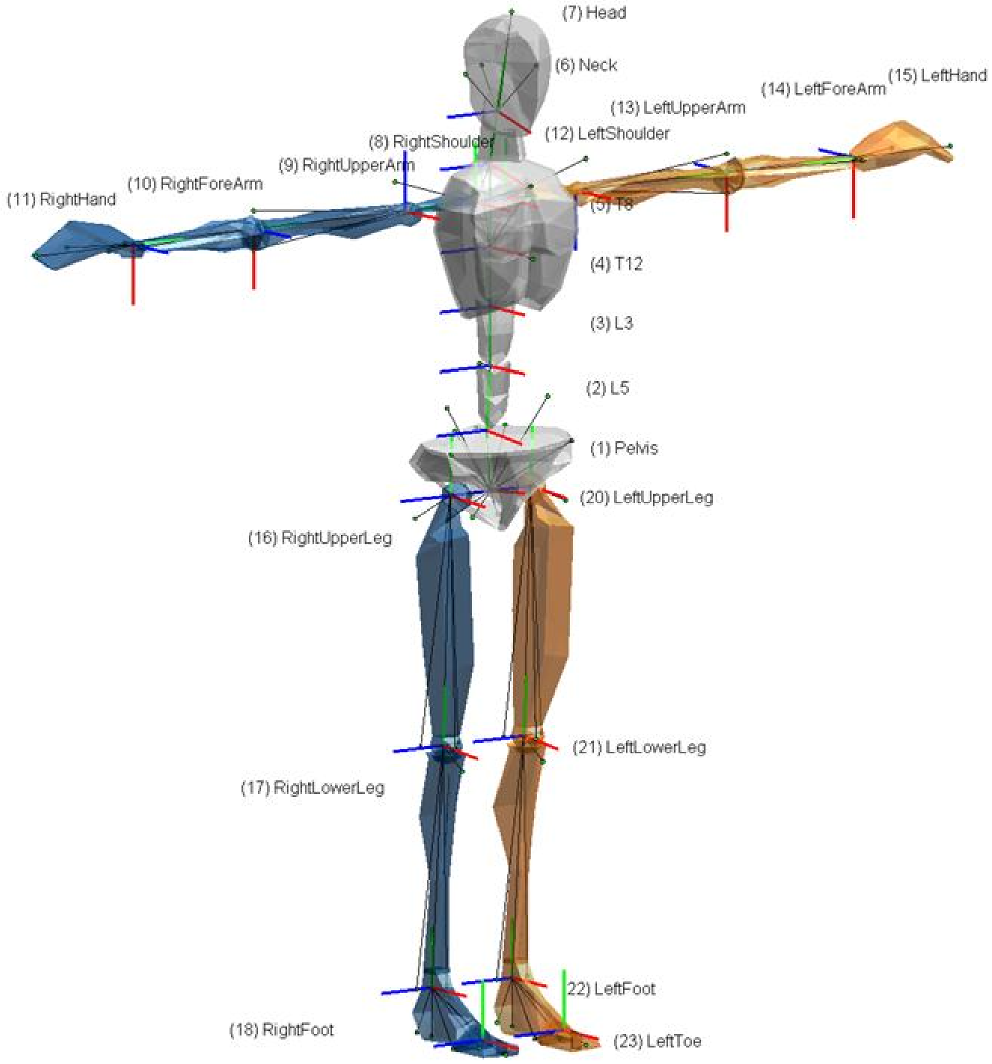


Figure B.1: The 23 segments of the biomechanical model that is used in MVN. The segment coordinate systems are indicated in the segment origins (x is red, y is green and z is blue) (from Xsens MVN BIOMECH User Manual [22]).

Segment 6: Neck The neck contains the Cervical vertebrae (C1-C7) (Figure B.2(c)). The origin is in jT1C7, the x-axis is pointing to the front, the y-axis points upward (towards jC1Head), and the z-axis is pointing to the right. The neck is not measured directly in MVN, but calculated from a model of the neck.

Segment 7: Head The origin is in jC1Head, the x-axis point to the front, the y-axis point towards the top of the head, and the z-axis is parallel to the line connecting both ears and pointing to the right. (Figure B.2(d))

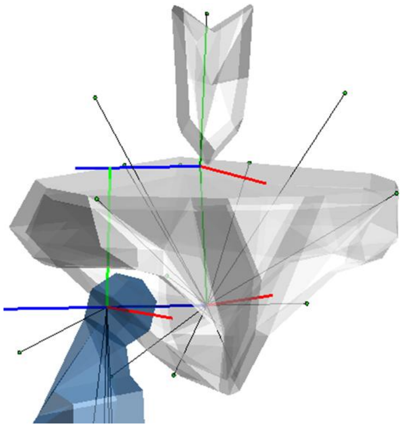
Segment 8 Right and Segment 12 Left: Shoulder The shoulder is a more complicated segment. The movement of the shoulders is measured by two sensors on the scapulae. The origin is on the midpoint between the sternum and jT8T9, The x-axis is pointing to the front, the y-axis point upwards and the z-axis points to the glenohumeral joint. The clavicles are not measured in MVN (Figure B.2(e)).

Segment 9 Right and Segment 13 Left: Upper Arm (Humerus) The origin is in the glenohumeral joint. The x-axis is pointing forward, the y-axis points from the elbow joint towards the glenohumeral joint, and the z-axis points to the right (perpendicular to the x and y-axis) (Figure B.2(f)).

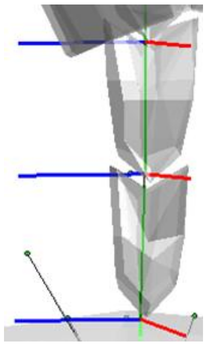
Segment 10 Right and Segment 14 Left: Forearm (Radius/Ulna) The forearm has its origin in the elbow joint. The x-axis is pointing forward, the y-axis points from the wrist to the elbow joint, and the z-axis points to the right (Figure B.2(g)).

Segment 11 Right and Segment 15 Left: Hand The origin is in the wrist, the x-axis points medially in N-pose (for a description of the different poses in MVN see the “Segment Calibrations” paragraph below) and forward in the anatomical position, the y-axis points from the top of the hand towards the wrist, and the z-axis is pointing perpendicular to the x and y-axis (pointing to the right during the anatomical pose, to the front during N-pose) (Figure B.2(h)).

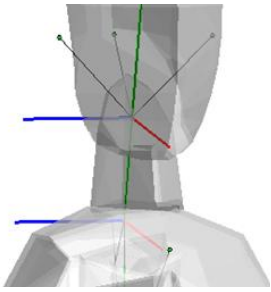
Segment 16 Right and Segment 20 Left: Upper Leg (Femur) The origin is in the hip joint, the x-axis is pointing forward, the y-axis is pointing from the right knee towards the hip joint, and the z-axis is pointing to the right (Figure B.2(i)).



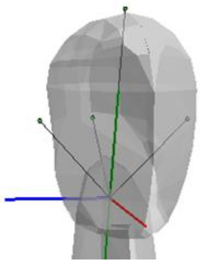
(a) Pelvis



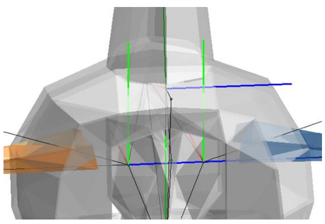
(b) 2 spinal segments



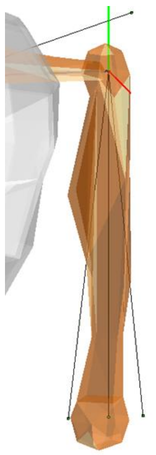
(c) Neck



(d) Head

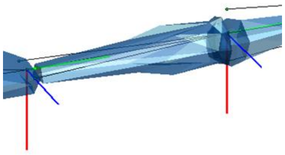


(e) Shoulders

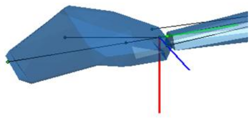


(f) Upper arm

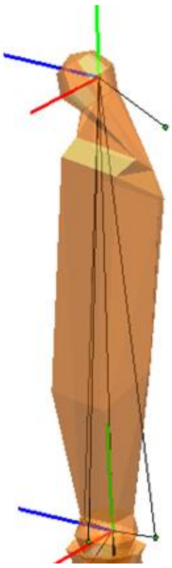
Figure B.2: Different segments of the biomechanical model used in MVN. The segment coordinate systems are indicated in the segment origins (x is red, y is green and z is blue) (from Xsens MVN BIOMECH User Manual [22]).



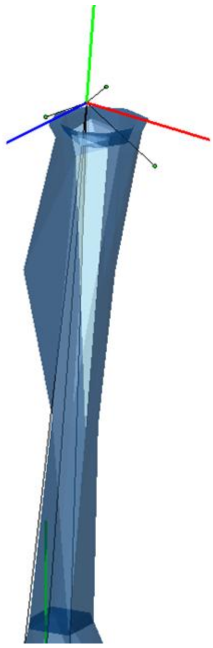
(g) Forearm



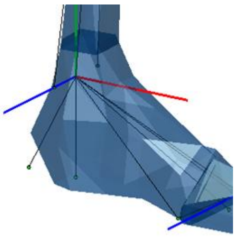
(h) Hand



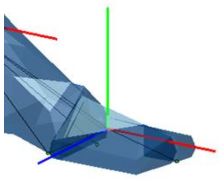
(i) Upper leg



(j) Lower leg



(k) Foot



(l) Toe

Figure B.2: (continued)

Segment 17 Right and Segment 21 Left: Lower Leg (Tibia/Fibula) The origin of the lower leg is in the knee joint, the x-axis is pointing forward, the y-axis points from the ankle joint towards the knee joint, and the z-axis is pointing to the right (Figure B.2(j)).

Segment 18 Right and Segment 22 Left: Foot (Calcaneus) The origin is in the ankle joint, the x-axis is pointing forward, the y-axis is pointing vertical (aligned with gravity), and the z-axis is perpendicular to the x and y-axis pointing to the right (Figure B.2(k)).

Segment 19 Right and Segment 23 Left: Toe The origin is in the ball joint of the right/left toe, the x-axis is pointing forward, the y-axis points upwards (aligned with gravity), and the z-axis points to the right. In MVN the Toes are not measured, but based on foot kinematics and contact detection (Figure B.2(l)).

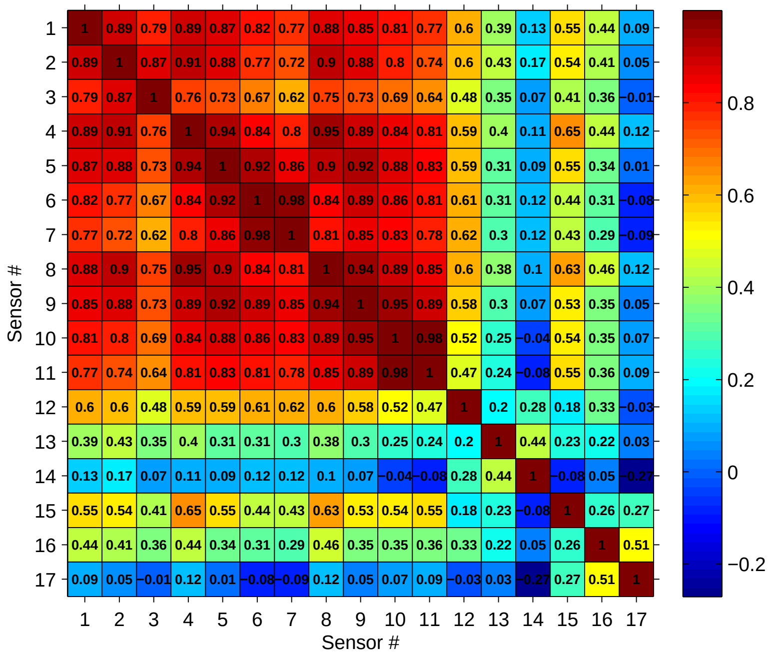
Inertial sensors used So as described above, not all the segments are directly measured in MVN, only 17 inertial sensors are used. From these sensors, the positions of the other segments can be calculated. To summarize, the sensor numbers and their corresponding segments are listed in Table B.1.

Table B.1: Sensor numbers and their corresponding body segments (from Xsens MVN BIOMECH User Manual [22]).

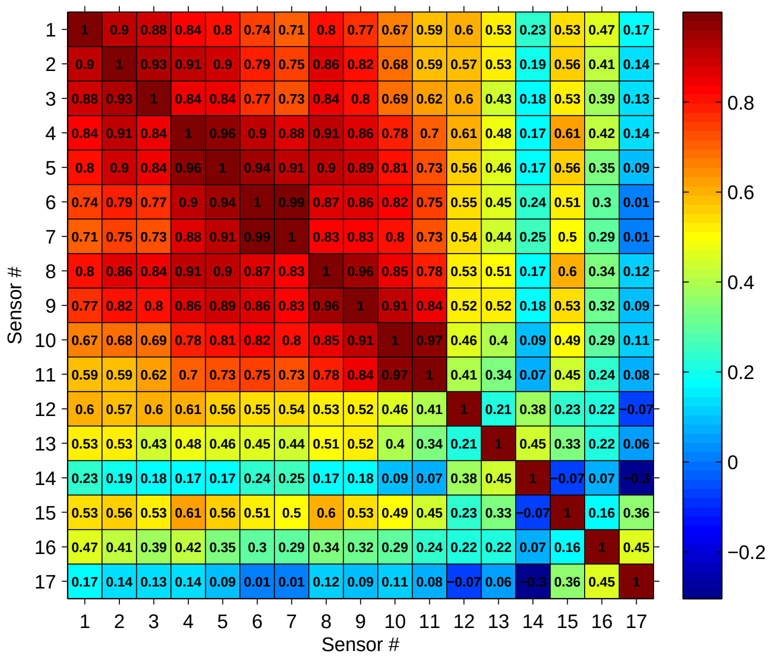
#	Body segment	Sensor	Joint
1	Pelvis	Pelvis	jL5S1
2	L5	Sternum	jL4L3
3	L3	Head	jL1T12
4	T12	Right shoulder	jT9T8
5	T8	Right upper arm	jT1C7
6	Neck	Right forearm	jC1Head
7	Head	Right hand	jRightC7Shoulder
8	Right Shoulder	Left shoulder	jRightShoulder
9	Right Upper Arm	Left upper arm	jRightElbow
10	Right Forearm	Left forearm	jRightWrist
11	Right Hand	Left hand	jLeftC7Shoulder
12	Left Shoulder	Right upper leg	jLeftShoulder
13	Left Upper Arm	Right lower leg	jLeftElbow
14	Left Forearm	Right foot	jLeftWrist
15	Left Hand	Left upper leg	jRightHip
16	Right Upper Leg	Left lower leg	jRightKnee
17	Right Lower Leg	Left foot	jRightAnkle
18	Right Foot		jRightBallFoot
19	Right Toe		jLeftHip
20	Left Upper Leg		jLeftKnee
21	Left Lower Leg		jLeftAnkle
22	Left Foot		jLeftBallFoot
23	Left Toe		

Appendix C

Correlation coefficients during walking

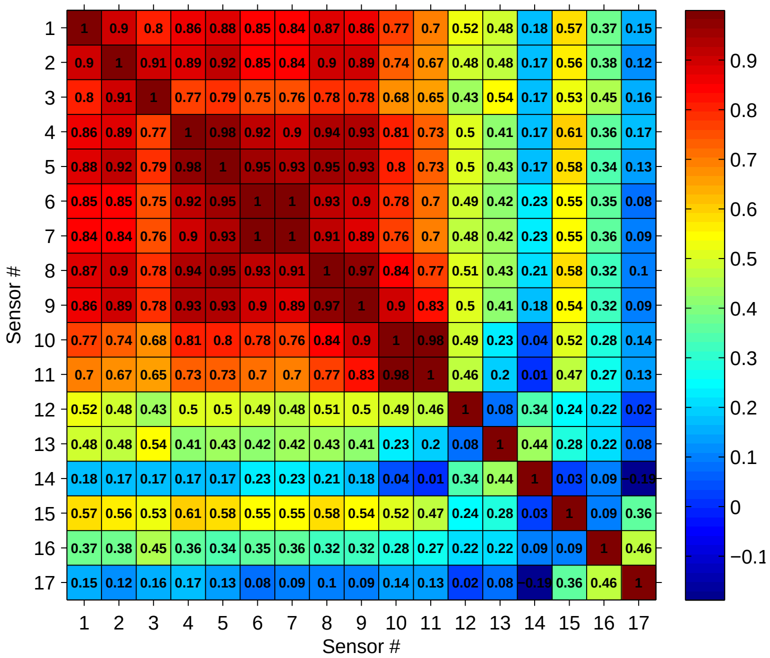


(a) Subject 1



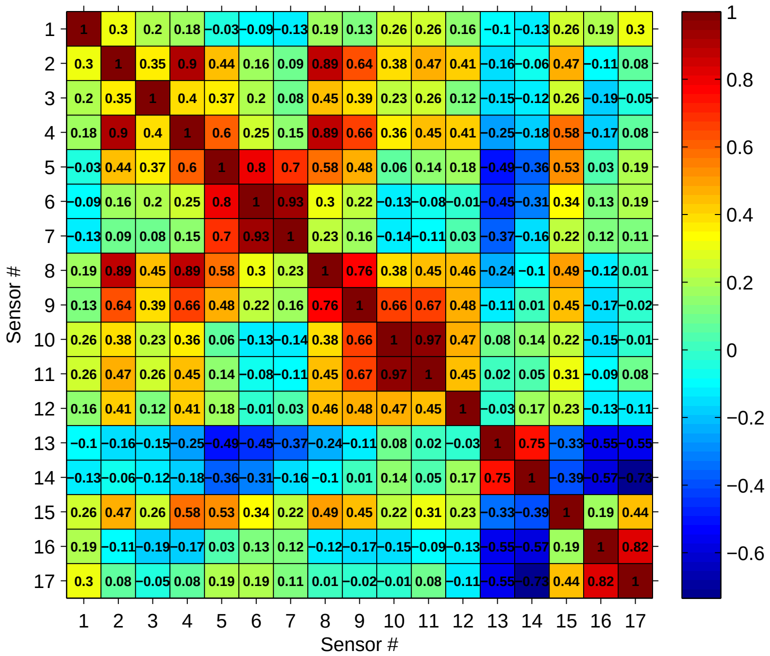
(b) Subject 2

Figure C.1: Correlation coefficients of the magnitude of the sensor acceleration of the subjects 1, 2, and 3 while walking at normal speed. (continued...).



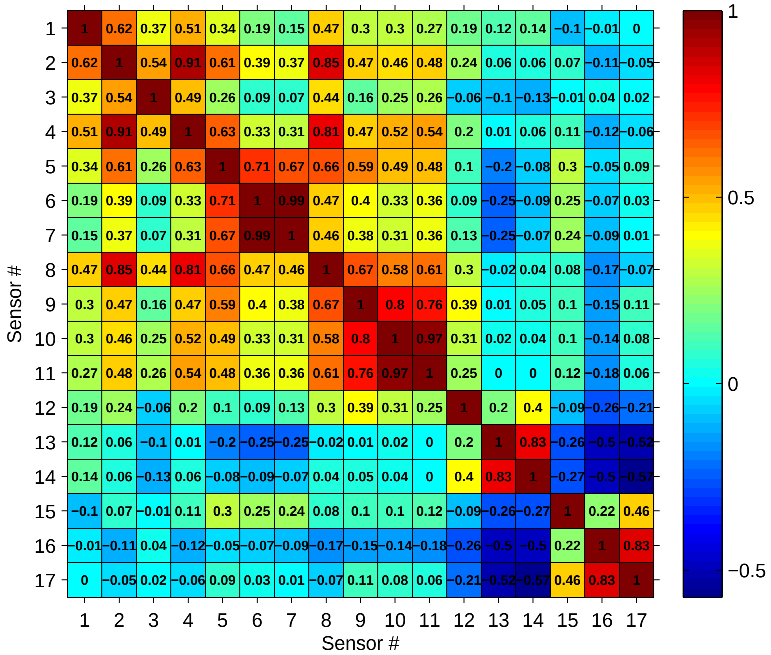
(c) Subject 3

Figure C.1: (...continued). Correlation coefficients of the magnitude of the sensor acceleration of the subjects 1, 2, and 3 while walking at normal speed.

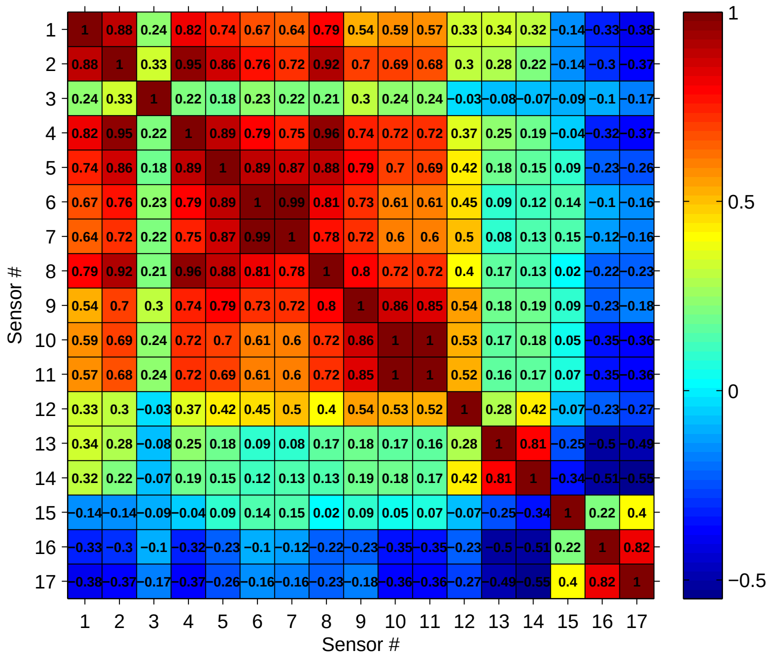


(a) Subject 1

Figure C.2: Correlation coefficients of the magnitude of the sensor angular velocity of the subjects 1, 2, and 3 while walking at normal speed. (continued...).



(b) Subject 2



(c) Subject 3

Figure C.2: (...continued). Correlation coefficients of the magnitude of the sensor angular velocity of the subjects 1, 2, and 3 while walking at normal speed.

Appendix D

Other Weka classifiers

In this Appendix, several other classifiers that can be chosen in Weka Explorer and can be used to identify the inertial sensors are listed.

In `Weka Explorer` several classifiers can be chosen easily. Using default parameter settings in Listing D.1 the Weka decision table is shown. From the 527 instances, 477 instances were identified correctly (90.5123 %) using 10 fold cross validation. The UnsCcAngVelMaXNorm-feature that is used is the same as CcAngVelMaXNorm, but no ranking is used.

Listing D.1: Weka decision table

```
Rules:
=============================================================
MeanAngAccNorm        UnsCcAngVelMaXNorm    class
=============================================================
(7.5-9.5]           (0.946145-inf)      Forearm
(9.5-11.5]          (0.946145-inf)      Hand
(4.5-7.5]           (0.946145-inf)      Forearm
(-inf-1.5]          (0.872005-0.946145] Sternum
(9.5-11.5]          (0.872005-0.946145] Hand
(7.5-9.5]           (0.872005-0.946145] Forearm
(4.5-7.5]           (0.872005-0.946145] Forearm
(1.5-4.5]           (0.872005-0.946145] Shoulder
(-inf-1.5]          (0.78042-0.872005]  Sternum
(9.5-11.5]          (0.78042-0.872005]  Hand
(7.5-9.5]           (0.78042-0.872005]  Forearm
(11.5-13.5]         (0.78042-0.872005]  Lowerleg
(15.5-inf)          (0.78042-0.872005]  Foot
(13.5-15.5]         (0.78042-0.872005]  Lowerleg
(4.5-7.5]           (0.78042-0.872005]  Upperarm
(1.5-4.5]           (0.78042-0.872005]  Shoulder
(-inf-1.5]          (0.6717-0.78042]    Sternum
(1.5-4.5]           (0.6717-0.78042]    Shoulder
(4.5-7.5]           (0.6717-0.78042]    Upperarm
(15.5-inf)          (0.6717-0.78042]    Foot
(13.5-15.5]         (0.6717-0.78042]    Lowerleg
(11.5-13.5]         (0.6717-0.78042]    Upperleg
(7.5-9.5]           (0.6717-0.78042]    Upperarm
(15.5-inf)          (0.583295-0.6717]   Foot
```

```
(13.5—15.5]         (0.583295—0.6717]   Lowerleg
(1.5—4.5]           (0.583295—0.6717]   Shoulder
(—inf—1.5]          (0.583295—0.6717]   Pelvis
(7.5—9.5]           (0.583295—0.6717]   Pelvis
(9.5—11.5]          (0.583295—0.6717]   Pelvis
(11.5—13.5]         (0.583295—0.6717]   Upperleg
(4.5—7.5]           (0.583295—0.6717]   Upperarm
(15.5—inf)          (—inf—0.583295]     Pelvis
(13.5—15.5]         (—inf—0.583295]     Pelvis
(11.5—13.5]         (—inf—0.583295]     Upperleg
(1.5—4.5]           (—inf—0.583295]     Head
(9.5—11.5]          (—inf—0.583295]     Pelvis
(4.5—7.5]           (—inf—0.583295]     Pelvis
(—inf—1.5]          (—inf—0.583295]     Head
(7.5—9.5]           (—inf—0.583295]     Pelvis
=============================================================
```

Also a MultiLayerPerceptron neural network classifier is generated, but because of space limitations not shown here. All instances were classified correctly, but the time taken to build the model was about 88 seconds.

References

[1] Weka, the Waikato Environment for Knowledge Analysis (`http://www.cs.waikato.ac.nz/ml/weka/`).

[2] Xsens Technologies B.V. (`http://www.xsens.com`).

[3] A. Avci, S. Bosch, M. Marin-Perianu, R.S. Marin-Perianu, and P.J.M. Havinga. Activity Recognition Using Inertial Sensing for Healthcare, Wellbeing and Sports Applications: A Survey. 2010.

[4] W. Dubitzky. *Data mining techniques in grid computing environments*. Wiley-Blackwell, 2008.

[5] J.B. Kuipers. *Quaternions and rotation sequences: a primer with applications to orbits, aerospace and vitual reality*. Princeton University Press, 1999.

[6] H. Liu and H. Motoda. *Feature extraction, construction and selection: A data mining perspective*. Kluwer Academic Publishers, 1998.

[7] A. Mannini and A.M. Sabatini. Machine Learning Methods for Classifying Human Physical Activity from On-Body Accelerometers. *Sensors*, 10(2):1154–1175, 2010.

[8] E.N. Marieb and K.M.D. Hoehn. *Human Anatomy & Physiology*. Pearson Education, Inc., 7th edition, 2007.

[9] G.J. Miao and M.A. Clements. *Digital signal processing and statistical classification*. Artech House Publishers, 2002.

[10] P. Pellikaan. *What is the effect of wearing 'unstable' Masai Barefoot Technology (MBT) shoes on the walking motion, balance and posture of a person compared to conventional 'stable' shoes.* Bachelor thesis, University of Twente, 2010.

[11] W. Platzer. *Atlas van de anatomie. 1. Bewegingsapparaat*. SESAM/HBuitgevers, 21st edition, 2006.

[12] D. Roetenberg. *Inertial and magnetic sensing of human motion*. PhD thesis, University of Twente, 2006.

[13] D. Roetenberg, H. Luinge, and P. Slycke. Xsens MVN Technical Whitepaper: Full 6DOF Human Motion Tracking Using Miniature Inertial Sensors. April 2009.

[14] H.M. Schepers. *Ambulatory assessment of human body kinematics and kinetics*. PhD thesis, University of Twente, 2009.

[15] R. Shiavi. *Introduction to applied statistical signal analysis: guide to biomedical and electrical engineering applications*. Academic Press, third edition, 2007.

[16] P. Veltink, F. van der Helm, B. Koopman, and A. Nene. *Dictaat Biomechatronics*. University of Twente, 2008.

[17] P.H. Veltink, H.B.J. Bussmann, W. de Vries, W.L.J. Martens, and R.C. Van Lummel. Detection of static and dynamic activities using uniaxial accelerometers. *IEEE Transactions on Rehabilitation Engineering*, 4(4):375–385, 1996.

[18] R.G.V. Wassink, C. Baten, P.H. Veltink, R.N.J. Veldhuis, and J.H. Smeding. Monitoring of human activities using a trainable system based on Hidden Markov modelling technology. *Gait and Posture*, 24(Suppl 2):S109–S110, 2006.

[19] S. Winter, J. Te Vrugt, R.D. Willmann, G. Lanfermann, and E.G.J.M. Bongers. System and method for automatic sensor position recognition, July 22 2008. US Patent App. 12/669,841.

[20] I.H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann Publishers, 2nd edition, 2005.

[21] Xsens Technologies B.V. *MVN Full Body Configuration Sheet*, May 2010.

[22] Xsens Technologies B.V. *MVN User Manual*, May 2010.

Index