• U C •

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

DEPARTAMENTO DE
ENGENHARIA MECÂNICA

Gesture Spotting from IMU and EMG Data for Human-Robot Interaction

Submitted in Partial Fulfilment of the Requirements for the Degree of Master in Mechanical Engineering in the speciality of Production and Project

Segmentação de Gestos a partir de Dados IMU e EMG para Interação Homem Robô

Author
João Diogo Faria Lopes

Advisor
Pedro Mariano Simões Neto

Jury

President	**Professor Doutor Cristovão Silva** **Professor da Universidade de Coimbra**
Vowels	**Professor Doutor Nuno Alberto Marques Mendes** **Professor Auxiliar da Universidade de Coimbra**
Advisor	**Professor Doutor Pedro Mariano Simões Neto** **Professor da Universidade de Coimbra**

Coimbra, Setembro, 2016

Acknowledgements

The work here presented was only possible thanks to the support and collaboration of some people, to whom I must pay my recognition.

To Professor Pedro Neto, for this given opportunity, as well as the support and guidance throughout the entire work.

To the colleagues at the Robotics laboratory, for their help and friendliness.

To the participants in this work, without whom this would not have been possible.

To my friends, thank you for supporting and cheering and creating a great environment both inside and outside of work.

To all my family, for all their support throughout the duration of the thesis, as well as the last 23 years, with special mention to my mother and father, without whom nothing would have truly been possible.

Abstract

Gesture spotting is an important factor in the development of human-machine interaction modalities, which can be improved by reliable motion segmentation methods. This work uses a gesture segmentation method in order to distinguish dynamic from static motions, using IMU and EMG sensor modalities. The performance of the sensors individually as well as their combination was evaluated, with thresholds and window size manually defined for each sensor modality, through 60 sequences performed by 6 users. The method which used the IMU alone obtained the best results in regards to the total segmentation error (11.88%), in comparison to the other two methods (EMG = 43.75% e IMU+EMG= 12.92%). When considering gestures which only contain arm movement, the best error obtained was 1.11% by the IMU method (EMG = 58.89% e IMU+EMG= 7.22%). However, when considering gestures which have only hand motion, the combination of the 2 sensors achieved the best performance, with an error of 10% (IMU = 30.83% e EMG= 17.5%). Results of the sensor fusion modality varied greatly depending on user, with segmentation errors varying between 1.25% and 26.25%, where users with more training obtained better results. Application of different filtering method to the EMG data as a solution to the limb position resulted in an error for the combination of sensors of 9.17%, with all gestures performing similarly or better than the IMU method but with an increased number of non-detected gestures.

Keywords EMG, gestures, IMU, human-machine interaction, motion, segmentation

Resumo

O reconhecimento de gestos é um fator importante no desenvolvimento de modalidades para interação homem-máquina, que podem ser melhoradas através de métodos fiáveis de segmentação de movimento. Esta tese usou um método de segmentação de modo a distinguir movimentos dinâmicos de estáticos, através do uso de sensores IMU e EMG. Foi avaliado o desempenho dos sensores individualmente e em combinação, com *thresholds* e tamanho de janela calculados manualmente para cada modalidade, através de 60 testes realizados por 6 utilizadores. O método que usou o IMU isoladamente obteve melhores resultados em relação ao erro total de segmentação (11,88%), comparativamente aos outros dois métodos (EMG = 43,75% e IMU+EMG= 12,92%). Quando considerámos os gestos que continham apenas movimento de braço, o melhor erro obtido foi de 1,11% para o método de IMU (EMG = 58,89% e IMU+EMG= 7,22%). No entanto, quando avaliámos os gestos apenas com movimento da mão a combinação dos dois sensores atingiu o melhor desempenho, com um erro de 10% (IMU = 30,83% e EMG= 17,5%). Os resultados da metodologia de combinação de sensores variaram consideravelmente dependendo do utilizador, com erros de segmentação entre 1,25% e 26,25%, em que os utilizadores com maior treino obtiveram os melhores resultados. A utilização de um método de filtragem diferente aos dados do sensor EMG, como solução para o problema da posição do membro, resultou em um erro para a combinação de sensores de 9,17%, com todos os gestos a terem um desempenho semelhante ou superior em comparação ao método que usou o IMU, mas com um número mais avultado de gestos não detetados.

Palavras-chave: EMG, gestos, IMU, interação homem-máquina, movimento, segmentação

Table of Contents

FIGURE INDEX

TABLE INDEX

SYMBOLOGY AND ACRONYMS

Symbology

a_x, a_y, a_z – Linear acceleration components

a_r –Euclidian distance of the acceleration components

g_x, g_y, g_z – Angular velocity

o_x, o_y, o_z – Euler orientation

q_x, q_y, q_z, q_w – Quaternions

t_{IMU} – IMU timestamp

t_{EMG} – EMG timestamp

$semg_i$ – Original data from EMG sensor i

T - Threshold

w – Window size

k – Sensitivity factor

R_{IMU} – Gestures which include arm motion

R_{EMG} – Gestures which include hand motion

O_{IMU} – Gestures which only include hand motion

O_{EMG} – Gestures which only include arm motion

do – Variation of orientation

T_{IMU} – Threshold for IMU features

T_{EMG} – Threshold for EMG features

$remg_i$ – rectified data from EMG sensor i

$femg_i$ – filtered data from EMG sensor i

F_c – Sampling frequency

F_e – Cut-off frequency

N – Butterworth filter order

A, B – Butterworth filter function coefficients

val – Function for base values of EMG

sum.v – Function output for sum of EMG values

wsum.v – Function output for weighted sum of EMG values

varsum – Function output for variance of sum of EMG values

var – Function output for variance of EMG values

S_{error} - Segmentation error

Acronyms

IMU – Inertial measurement unit

EMG – Electromyography

HMI – Human-machine interaction

FN – False negative

FP – False positive

SVM – Support Vector Machine

ANN – Artificial Neural Network

HMM – Hidden Markov Models

1. INTRODUCTION

Human-machine interaction (HMI) is an increasingly common occurrence in today's technological society.

Flexible work stations rely on a joint collaboration between humans and robots. One of the most intuitive methods for HMI is gesture spotting: robots performing defined movements based on gestures being performed by users. These movements are generally executed in sequence in order to perform complex tasks.

As such, there is the need for increasingly reliable mechanisms for a real-time interaction between both participants.

Towards HMI, multiple solutions have been presented for gesture spotting, such as gesture detection through body-worn sensors or using computer vision. In some cases, the solution includes a combination of multiple modalities.

Methodologies for gesture segmentation have been studied. (Simão, Neto, and Gibaru 2016) has tackled this problem, using a Cyber Data Glove in order to detect hand and arm gestures performed by the user. However, the equipment is not very practical, as it is wired, uncomfortable to wear and expensive.

In the search for more accessible and comfortable options, a solution was found in the MYO armband. This device, available to the general public, includes two sensors: the IMU (inertial measurement unit) and the EMG (electromyography) sensor.

Following the work developed in (Simão, Neto, and Gibaru 2016), this thesis aims to evaluate the performance of IMU and EMG sensors in regards to gesture segmentation, aiming to distinguish dynamic motions.

This work will start by a state of the art review, posteriorly analysing an alternative method to motion detection to justify the usage of the sliding window method in chapter 3. In chapter 4 the design of the algorithms for motion detection for the individual sensors and their combination is discussed, with an analysis of the obtained results being performed in chapter 5, based on types of errors, gestures and participants.

2. STATE OF THE ART

2.1. IMU

Inertial measurements units (IMUs) are devices used to measure linear acceleration and angular rate through the use of two different types of inertial sensors, accelerometers and gyroscopes. According to (Unsal and Demirbas 2012), "an accelerometer measures linear acceleration about its sensitivity axis and integrated acceleration measurements are used to calculate velocity and position", whereas "a gyroscope measures angular rate about its sensitivity axis and gyroscope outputs are used to maintain orientation in space". When using the IMU in a tridimensional space, a total of 3 accelerometers and 3 gyroscopes are used, both with orthogonally distributed axis as referred by (King 1998).

More recently, IMU sensors have been integrated with magnetometers (Brunner et al. 2015; Fourati et al. 2014) to measure the local magnetic field vector in sensor coordinates and thus allow the determination of orientation relative to the vertical axis as mentioned by (Caruso 2000). (Brunner et al. 2015) noted that if the magnetic field is not disturbed, it corresponds to the Earth's magnetic field.

2.1.1. Applications of IMU

(Unsal and Demirbas 2012) states that due to recent technological advances, associated with improved calibration algorithms and error calibration models, inertial and magnetic sensors have become available at low cost, with small size and low energy consumption. This allowed to build small-sized and cheap IMU modules, comparable to other commonplace devices, as suggested by (Verplaetse 1996), which led to them being used commonly, for example, in smartphones, which have IMUs or 3-axis accelerometers integrated as seen in (del Rosario, Redmond, and Lovell 2015).

IMU have been used as core tools in inertial navigation, in conjunction with GPS as studied by (King 1998). They have also been increasingly used for motion sensing in applications involving relative motion, such as handwriting recognition, for example by

placing a sensor on the tip of a pen, or retrieval of data on sports equipment, both referred by (Verplaetse 1996).

Related to the present work, usage in the replication of human movements by machines has become a common field of study. Some examples of applications of IMU devices on wearable body sensors include (Ganesan, Gobee, and Durairajah 2015), in which an upper limb exoskeleton relying on both data from IMU and EMG sensors for the rehabilitation of neurological or musculoskeletal diseased patients was studied.

(Junker et al. 2008) presents a study where 5 inertial sensors were placed on the upper body for the detection of sporadic occurring activities in a continuous signal stream.

(Jung et al. 2015) refers the use of IMU sensors on ROBIN-H1, a lower limb exoskeleton which "was developed as a walking rehabilitation service for stroke patients" and requires data from IMUs placed on the right and left trunk segments.

2.1.2. IMU Benefits

According to (Fourati et al. 2014), IMUs have some associated benefits in comparison to other sensors. The main advantages mentioned are that "there is no inherent latency associated with this sensing technology and all delays are due to data transmission and processing". The authors also mention another benefit to be "its lack of necessary source, whereas electromagnetic, acoustic, and optic devices require emissions from a source to track objects", becoming a more advantageous option on non-controlled environments.

2.1.3. IMU Disadvantages

Inertial and magnetic sensors have shown to have certain drawbacks as well. According to (Fourati et al. 2014), accelerometers measure the sum of linear acceleration and gravity. In quasi-static situations, where there is no linear acceleration present, measuring the gravity in the sensor coordinate frame allows for an accurate estimation of orientation relative to the horizontal plane. However, in a dynamic situation, it is not easy to dissociate these two physical quantities, becoming difficult to measure the orientation with accuracy. A method for separating acceleration from the gravity component has been studied in (Neto, Pires, and Moreira 2013).

Another the major issue mentioned in IMU related bibliography is drift. According to (Neto, Pires, and Moreira 2013), due to the sensor calculating its position based

on previously calculated positions, any existing errors in measurement, no matter how small, are accumulated with every calculation, leading to an increasing difference between the calculated position of the sensor and the actual truth, not allowing for an accurate position estimation for long periods. A major factor to this is derived from the double integration of acceleration data in order to obtain position as suggested by (Neto, Pires, and Moreira 2013), but gyroscopes have also been mentioned by (Bortz 1971) to be prone to drifting over time due to the build-up of various errors when estimating changes in orientation.

Magnetometers are relatively drift-free and are therefore used to cancel out any possible drift errors present in the previous sensors according to (Roetenberg, Luinge, and Veltink 2003). However, the main problem with magnetometers identified by the authors is the influence of ferrous material in the surroundings of the sensor, which disturb the orientation measurement.

2.1.4. IMU Specific Errors

According to (Unsal and Demirbas 2012), errors present on IMU sensors can be defined under two categories: deterministic and stochastic errors. According to the authors, deterministic errors are those that, given a certain defined input and known error, will always provide the same output. They can be estimated by laboratory calibration tests and can be used as input for error compensation algorithms. Stochastic errors, on the other hand, are associated with random variations of bias or scale factor over time, as well as random sensor noise.

2.1.4.1. Deterministic Errors

Bias is defined by (NovAtel 2014) as the offset value output by the sensor measurement for a given physical input. The bias for the accelerometer or the gyroscope can be calculated according to (Unsal and Demirbas 2012) as the measured value when no input acceleration or angular rate is applied to the sensor respectively. (NovAtel 2014) divides the bias error into two components: bias repeatability, which refers to different initial bias with every power up of the IMU; and bias stability, associated with the change of the initial bias over time.

Scale factor error is defined by (Titterton and Weston 2004) as the error in the ratio of a change in an output signal relative to a change in the input signal, be it either linear

acceleration or angular rate. The major parts responsible for scale factor error suggested by (Unsal and Demirbas 2012) are fixed terms and temperature induced variations.

Misalignment of sensors is associated by (Unsal and Demirbas 2012) to a scale factor error on measurements due to a non-orthogonality between the IMU axes, which results from IMU mechanical components not being produced and mounted perfectly. As such, any movement in an axis causes a change in the other axes depending on the magnitude of the misalignment.

G-dependency is related to the effect of acceleration on the output signal. According to (NovAtel 2014), it is most commonly seen on Micro Electrical Mechanical Systems gyroscopes, when the mass undergoes acceleration along its sensing axis. The g-dependent bias coefficient is referred by (Unsal and Demirbas 2012) as the relation between the acceleration magnitude and the gyroscope measurements.

2.1.4.2. Stochastic Errors

A random noise in the measurement is always present when measuring a constant signal (NovAtel 2014). The sources of these errors are flicker noises in the electronics or interference effects on signals.

To reduce the effect of sensor noise, (Unsal and Demirbas 2012) suggests either applying a vast number of processes for modelling stochastic errors, or applying a filter to the signal.

2.2. EMG

According to (Carpi and Rossi 2006), electromyography (EMG) is a method for recording and analysing electric signals resulting from neuromuscular activity, also known as electromyograms. (Raez et al. 2006) indicates that the muscle tissue conducts electrical potentials in similarity to nerves, which are named muscle action potentials, whose information the EMG is used for recording. According to (Alkan and Günay 2012), since each movement of the muscles corresponds to a specific pattern of activation of several muscle fibres, using multi-channel EMG recordings it is possible to identify the movement being performed.

According to (Raez et al. 2006), two types of electrodes can be used to acquire muscle signal: invasive and non-invasive electrodes. Invasive EMG relies on using wire or

needle electrodes placed directly in the muscle. In the case of non-invasive electrodes, also known as sEMG, the EMG signal is acquired from electrodes mounted directly on the surface of the skin. As such, the signal is a composite of all muscle fibres' action potentials occurring in the muscles beneath the skin as stated by (Raez et al. 2006). Since these action potentials occur at random intervals, the EMG signal can either be positive or negative voltage.

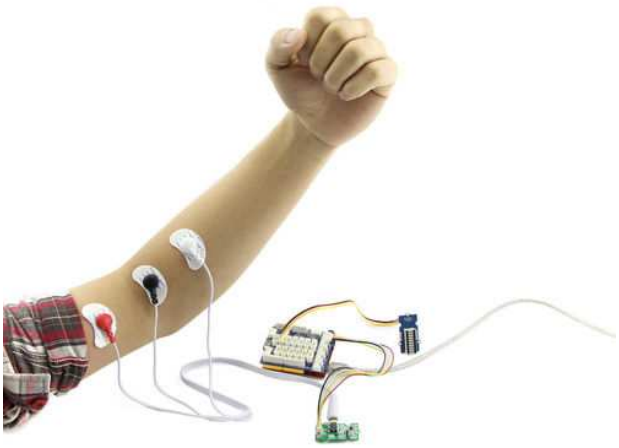

Figure 2.1 – EMG detector circuit with 3 electrodes placed on the forearm (Seeed Studio 2015)

2.2.1. Applications of EMG

While EMG is mainly used in clinical applications, in the context of human motion, cases where EMG sensors have been used include (Al-Angari et al. 2016), where the classification performance of EMG features for hand and arm movements was studied using data from 15 EMG sensors placed on the forearm.

(Kawasaki et al. 2014) presents a system for prosthetic hand control which was studied for forearm amputees based on EMG sensors, also placed on the forearm.

2.2.2. Types of Error of EMG

2.2.2.1. Quality of signal

(Raez et al. 2006) has identified the two main issues that influence the quality of the signal to be the signal to noise ratio and the distortion of the signal.

According to (Raez et al. 2006), signal to noise ratio refers to the ratio of energy in the EMG signals to the energy in the noise signals. Noise are electrical signals that are not part of the desired EMG signal. They can be the result of inherent noise in electronics

equipment, ambient noise due to electromagnetic radiation, motion artefact associated with faulty design of electrode components such as the interface or cable, and inherent instability of the signal, given how the EMG is random in nature. The authors claim that, to obtain a good EMG signal, the signal-to-noise ratio should contain the highest amount of information from EMG possible while keeping the amount of noise contamination to a minimum.

(Raez et al. 2006) also defines the distortion of the signal means that the relative contribution of any frequency component in the EMG signal should not be altered. As such, the distortion of the signal should be kept to the required minimum, avoiding unnecessary filtering and the distortion of signal peaks and notch filters.

2.2.2.2. Problems with muscle information extraction

In regards to issues pertaining the retrieve of information from the musculature by the EMG, (Scheme and Englehart 2011) identifies 3 major issues.

Due to the region of muscle activity recorded by a single EMG, the activity measured by the EMG may include the contribution of more than one muscle, an issue which has been defined as EMG cross talk.

Similarly, muscle co-activation, related to the presence of multiple EMGs, occurs when a muscle registers activity due to the activity of another, which "complicates the task of resolving the intended force about a joint".

Limited muscle sampling depth limits the measurement of muscle activity to only those close to the surface of the skin.

2.2.2.3. Issues with EMG misuse

During the usage of EMG, one must be aware that ideal conditions do not exist in practical use. Issues related to misuse of EMG mentioned by (Scheme and Englehart 2011) include electrode shift, variation in force, variation in position of the limb, and transient changes in EMG.

The electrode shift is associated with the possibility that, whenever a user places the device, "the electrodes will likely settle in a slightly different position, relative to the underlying musculature" (Scheme and Englehart 2011).

Pattern recognition control "relies on clustering repeatable patterns of EMG activity into discernible classes. Contractions performed at different force levels may be very

different from one another and therefore present a challenge to a pattern classifier." (Scheme and Englehart 2011), challenge which the work identifies as variation of force.

The variation of limb position, according to (Radmand, Scheme, and Englehart 2014), refers to "the degradation of myoelectric pattern recognition performance when the classifier is trained with limb in one fixed position but is tested or used with limb in other positions. This degradation is due to the impact of arm position variation on the muscular activation pattern when performing activities". In this regard, (Liu et al. 2014) has studied the effect of arm movements in EMG pattern recognition, including both static and dynamic arm motions, and concluded "that dynamic change of arm position had seriously adverse impact on sEMG pattern recognition"

Transient changes are defined in (Scheme and Englehart 2011) as "additional factors that confound the use of EMG and are a result of short- and long- term variations in the recording environment during use". These changes include external interference, electrode impedance changes, electrode shift, electrode lift (loss of contact between electrode and skin), and muscle fatigue.

2.2.2.4. Issue for amputees

According to (Scheme and Englehart 2011), another major issue which complicates the task of obtaining information from EMG for an appropriate dexterous control occurs when the user is an amputee and does not have the appropriate musculature to estimate the intended motion, with the issue being more severe the larger the limb deficiency of the user is.

(Scheme and Englehart 2011) provides the example that, in the case of transradial amputation, since many of the muscles responsible for the control of the wrist and the hand are present in the forearm, it would still be possible for the user to obtain a dexterous control of the hand. However, with a more severe deficiency, such task would be far more difficult, as the functionality of the hand becomes dependent on less physiologically appropriate sites.

2.3. MYO Armband

The MYO armband, as described by (Thalmic Labs 2016), is a device, meant to be worn on the forearm, whose purpose is to detect hand gestures and wrist and forearm

movements by using 8 stainless steel sEMG muscle sensors, combined with a nine-axis IMU, containing a three-axis accelerometer, a three-axis gyroscope and a three-axis magnetometer. Developed by Thalmic Labs, the armband uses an ARM Cortex M4 Processor and communicates the data to the computer through Bluetooth Smart Wireless technology.

As mentioned by (Thalmic Labs 2016), the MYO armband provides two kinds of data to an application: spatial data and gestural data.

According to (Thalmic Labs 2016), spatial data provides the application with data regarding the orientation and movement of the user's arm, obtained by IMU. This kind of data includes orientation data which indicates which way the MYO armband is pointed; raw acceleration data which represents the acceleration the MYO armband is undergoing at any given time, in the form of 3-dimensional vector; and angular velocity data provided by the gyroscope, also in the format of a vector.

The raw data from the accelerometer measures the linear acceleration of the armband, with its units being in g, the gravitational constant, of roughly 9.8 m/s^2, according to (Thalmic Labs 2016). The consequence of this measurement is that, when the user is stationary, a value of 1 should be noticed in the vertical direction, due to Earth's gravity. The limit of this measurement has been indicated to be around 8 g by (Thalmic Labs 2016).

Gyroscope data measures the angular acceleration of the armband. The data units used are $^o/s$, degrees per second, and are limited at approximately 16 rad/s, according to (Thalmic Labs 2016).

(Weili 2014) argues that, while each component of the data alone is not of great use in most scenarios, their combined effect by the calculation of the square root of the sum of the squares allows to obtain the magnitude of the linear or angular acceleration of the arm, which, as stated by the author, "are very effective indicators of the intensity of the arm movement, which in turn contains emotional or rhythmical information of the performance".

The orientation data is presented in 2 different fashions: in the form of quaternions and the Euler angles, which are yaw, pitch and roll.

The orientation data is calculated using the raw data from the accelerometer and gyroscope of the IMU. However, (Thalmic Labs 2016) mentions that, in order to obtain position data, double integration of the input data would be required. Such a method is bound to introduce a significant amount of error, and therefore the developers of the armband chose

not to offer such data as a standard output, claiming that “the MYO armband is better suited to getting the relative orientations of the arms rather than the absolute position”.

According to (Thalmic Labs 2016), gestural data provides the data to the application in order to recognize gestures performed by the users with their hands. The MYO provides gestural data in the form of one of several pre-set poses, which represent a particular configuration of the user's hand. The pre-determined gestures able to be detected by the device mentioned by (Thalmic Labs 2016) are: (i) fist, (ii) waving in, (iii) waving out, (iv) fingers spread and (v) thumb to pinky, as well as a “rest” gesture, indicative of no other gesture being detected.

The hand gesture data are provided by the proprietary EMG muscle activity sensors. The EMG data provided by the device is claimed by (Thalmic Labs 2016) to be “unitless”, representing activation, resulting from an unknown conversion from mV. This is due to the fact that the actual EMG units in voltage are extremely small, in microvolt range, with its limits ranging from -127 to 127 according to (Arief, Sulistijono, and Ardiansyah 2015). There are 8 EMG sensors mounted to the device, whose data obtained corresponds to the sensors presented in figure 2.2.

Figure 2.2 - EMG channel assignments to each sensor (Thalmic Labs 2015)

The IMU data has a sampling frequency of 50 Hz and the EMG of 200 Hz. However (Nyomen, Romarheim Haugen, and Jensenius 2015) has shown, when evaluating the sensor data provided by the MYO, that the MYO data stream had lower frame rate than the specified 50 Hz. According to (Thalmic Labs 2016), this issue is due to noisy environments which causes packet loss on transmitted data through Bluetooth.

(Weili 2014) claims that the hand gesture data is not as useful as it may appear. First, given that “the hand gesture is calculated from the EMG data measured on the skin of

the forearm, which is a side effect of the muscle movement", there is a possibility that "the calculated gesture may not loyally indicate the actual gesture of the hand". Second, when exterior forces are applied to the muscles or there is some other interference with the EMG readings such as tight clothes, the accuracy of the measurement can be vastly degraded, to the point where the gesture data may not be usable at all.

The MYO armband has been in the centre of some studies. For example, (Nyomen, Romarheim Haugen, and Jensenius 2015) has studied the potential of the MYO armband for the application on New Interfaces for Musical Expression, namely a MuMYO prototype for the production and modification of sounds with arm movements and hand gestures.

2.4. Pattern recognition process

For pattern recognition from wearable sensor data, the process includes several different modules, here mentioned based on gesture recognition sequences from (Carpi and Rossi 2006) and (Fida et al. 2015) and shown on figure 2.3:

1) Data acquisition from the sensors;
2) Pre-processing of the signal, which includes both data filtering and motion segmentation;
3) Feature extraction;
4) Pattern classification, in this case related to recognising gestures based on chosen features;

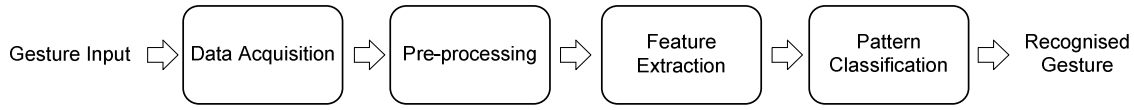

Figure 2.3 - Sequence required for gesture recognition from the gesture being performed by the user to it being recognised

2.4.1. Pre-processing

According to (Carpi and Rossi 2006), the purpose of the pre-processing stage is "to reduce noise artefacts and/or enhance spectral components that contain important information for data analysis. Moreover, it detects the onset of the movement and activates all the following modules". The first concept mentioned refers to filtering, whereas the

second refers to segmentation. According to (Attal et al. 2015), pre-processing also includes the step of feature selection and extraction, but this will not be considered in this work as a part of pre-processing.

2.4.1.1. Filtering

The purpose of filtering is, according to (Carpi and Rossi 2006), "to reduce noise artefacts and/or enhance spectral components that contain important information for data analysis".

Filters can be applied to both IMU and EMG data. In the case of EMG data, (Zecca et al. 2002) shows us an example of the processing of an EMG signal with data recorded from a biceps brachial muscle, in the upper arm, with data being treated through rectification of the EMG signal, removal of noise through low pass filter and then a process of segmentation with threshold-based detection of movement. Other filter possibilities can also be considered, such as in (Yang et al. 2015), where a band pass filter was additionally applied to the EMG signal.

In regards to IMU data, (Fida et al. 2015) presents us a study where the impact on classification by IMU data pre-processing is evaluated. The author mentions 2 common pre-processing steps: inclination correction, and signal filtering, with a low pass filter being applied. The study concludes however that these pre-processing stages have little to no impact on the average classification of the performed activities.

2.4.1.2. Segmentation

According to (Attal et al. 2015), segmentation is a technique used to extract features from input data which consists of dividing sensor signals into small time segments or windows, to which are then applied classification algorithms for gesture recognition.

There are 3 types of windowing techniques generally used according to (Attal et al. 2015): "sliding window where signals are divided into fixed-length windows; event-defined windows, where pre-processing is necessary to locate specific events, which are further used to define successive data partitioning and activity-defined windows where data partitioning is based on the detection of activity changes". The authors claim that the sliding window approach is well-suited to real-time applications since it does not require any pre-processing treatments.

The sliding window method to be used on this work is based on the approach by (Simão, Neto, and Gibaru 2016), where a gesture segmentation process using the sliding window method was used to segment continuous data obtained from a data glove and magnetic tracking device. (Simão, Neto, and Gibaru 2016) also includes references of existing gesture recognition techniques, with other examples of different windowing techniques being listed in (Fida et al. 2015).

The overview of the sliding window method presented in (Simão, Neto, and Gibaru 2016) considers that "there is motion if there are motion features above the defined thresholds", with the thresholds being calculated "for each motion feature using a genetic algorithm". A sliding window is composed of *w* consecutive frames, with *w* being the window size, and with each instant, the window slides forward one frame and is updated and evaluated. According to the authors, "a static frame is only acknowledged as such if none of the motion features exceed the threshold within the sliding window".

A major issue mentioned in (Simão, Neto, and Gibaru 2016) to segmentation is the existence of false positives and false negatives. According to the authors, false positives are false gestures, which may occur if the system is too sensitive to motion, and which hold no actual meaning. False negatives are associated with the identification of a motion as static when in truth it is not. This can result in data associated with a dynamic gesture being split into two different segments, which can lead to over segmentation and a gesture losing its meaning.

2.4.2. Feature Extraction

The goal of feature extraction has been defined by (Duda, Hart, and Stork 1999) as "characterize an object to be recognized by measurements whose values are very similar for objects in the same category, and very different for objects in different categories". As such, the authors affirm that the task of the feature extraction process is "seeking distinguishing features that are invariant to irrelevant transformations of the input".

2.4.2.1. Types of features

(Phinyomark, Phukpattaranont, and Limsakul 2012) has performed a listing of possible EMG features, which the author has referred to be divided into 3 main groups: time domain features, frequency domain features and time-frequency domain features. This classification is applicable to IMU features as well, mentioned by (Dargie 2009).

According to (Phinyomark, Phukpattaranont, and Limsakul 2012), time domain features refers to features which are calculated based on the variation of amplitude of the signal with time. They are generally quickly calculated since they do not require transformations. Major disadvantages are that time-domain features assume the data as a stationary signal, when it is in fact non-stationary, which may cause variations of time-domain features when recording through dynamic movements, as well as issues regarding the interference acquired through the recording when evaluating features extracted from energy property. However, time-domain features have good classification performances in low-noise environments and they have lower computational requirements.

Frequency domain features relates to features which analyse the system based on the frequency of occurring events. They are mostly used to study fatigue of the muscle and motor unit recruitment analysis according to (Phinyomark, Phukpattaranont, and Limsakul 2012).

According to (Zecca et al. 2002), "time–frequency representation can localize the energy of the signal both in time and in frequency, thus allowing a more accurate description of the physical phenomenon. On the other hand, time–frequency representation (TFR) generally requires a transformation that could be computationally heavy".

2.4.2.2. List of features

(Phinyomark, Phukpattaranont, and Limsakul 2012) includes a list of time and frequency-domain features for EMG sensor data classification, whereas (Dargie 2009) has a list for features which can be obtained from accelerometer sensors.

Mean absolute value is one of the most popular features in both EMG signal analysis (Phinyomark, Phukpattaranont, and Limsakul 2012) and accelerometer signal analysis (Dargie 2009). The mean absolute value feature, shown as *MAV* in equation 2.1 obtained from (Phinyomark, Phukpattaranont, and Limsakul 2012), is a time-domain feature defined by (Phinyomark, Phukpattaranont, and Limsakul 2012) as an average of absolute value of the signal amplitude in a segment.

$$MAV = \frac{1}{N}\sum_{i=1}^{N}|x_i| \quad \textbf{(2.1)}$$

Useful to analyse measurements affected by noise, the zero-crossing rate stands for the number of samples per second that cross the zero reference line (Dargie 2009), reported being used for both EMG (Phinyomark, Phukpattaranont, and Limsakul 2012) and IMU sensors (Dargie 2009). To avoid random noise such as low-voltage fluctuations or background noises, a threshold condition is implemented according to (Phinyomark, Phukpattaranont, and Limsakul 2012), which is expressed as *ZC* in equation 2.2 found in (Phinyomark, Phukpattaranont, and Limsakul 2012).

$$ZC = \sum_{i=1}^{N-1} [sgn(x_i * x_{i+1}) \cap |x_i - x_{i+1}| \geq threshold] \quad \textbf{(2.2)}$$

$$sgn(x) = \begin{cases} 1, & if\ x \geq threshold \\ 0, & otherwise \end{cases} \quad \textbf{(2.3)}$$

Waveform length is a measure of complexity of the EMG signal, which is defined by (Phinyomark, Phukpattaranont, and Limsakul 2012) as cumulative length of the EMG waveform over the time segment, shown in equation 2.3 as WL from (Phinyomark, Phukpattaranont, and Limsakul 2012).

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad \textbf{(2.4)}$$

The usage of plane acceleration in the study of heel strike was made by (Lee et al. 2015), in which 3-axis accelerations were considered, calculated using the Pythagorean theorem in each combination of two axes, in addition to the singular acceleration components. This resulted in 3 different plane accelerations, in the horizontal *xy* plane, in the sagittal *xz* plane and the coronal *yz* plane.

The Fast Fourier Transform is a frequency-domain feature for the IMU signal which, according to (Laudanski, Brouwer, and Li 2015), is a faster version of the Discrete Fourier Transform, which transforms a discrete signal in the time domain into its frequency domain representation. (Dargie 2009) refers also another frequency feature, the Short Time Fourier Transform, which is indicated to be the best performing amongst a set of selected features.

2.4.3. Classification

In the context of wearable sensors, the purpose of classification is to assign features retrieved from a segment of motion data to classify a pattern in order to recognize a gesture. (Duda, Hart, and Stork 1999) refers that "because perfect classification performance is often impossible, a more general task is to determine the probability for each of the possible categories".

2.4.3.1. Classification Techniques

For the task of pattern recognition, a classification method is required. Amongst many other developed techniques, three popular models have been identified: support vector machines (SVM), artificial neural networks (ANN), and hidden Markov models (HMM). Other classification methods include k-Nearest Neighbours, Random Forests, Gaussian Mixture Models and K-Means, identified in (Attal et al. 2015).

An example of the application of the SVM was shown in (Alkan and Günay 2012), where SVM was used to classify data from four different arm movements obtained by EMG sensors. Similarly, (Fida et al. 2015) uses the SVM method to classify selected features to evaluate the performance of activity recognition depending on different pre-processing operations, by analysing the data from an IMU sensor placed on the lower limb as subjects performed a predefined route.

The HMM is shown to be used in (Aoki, Venture, and Kulić 2013), which suggested an approach for the online segmentation of human body movements using IMU, where the segmented motions are then recognized using HMM models. Another example is in (Taborri et al. 2014), where a HMM-based classifier was studied for the purpose of gait detection, based on data from 3 IMU sensors placed on the lower limb.

The ANN, according to (Kriesel 2007), is a method for pattern classification whose development was motivated by the "similarity to successful working biological systems", namely the nervous system. Examples of application of this method can be seen in (Neto et al. 2013), a paper which suggests an approach at real-time hand gesture spotting which utilized the ANN classifier to recognize gestures made using a Data Glove. (Jung et al. 2015) also proposed 2 neural network-based classifiers for the classification of gait phases for lower limb exoskeletons.

2.5. Sensor Fusion

According to (Novak and Riener 2015), “multimodal sensor fusion combines information from different sensor modalities to overcome the shortcomings of each sensor”, with the fusion of EMG and IMU sensors, present in this work, the most explored. An example of this application can be seen in (Georgi, Amma, and Schultz 2015), where a system for recognizing hand and finger gestures is presented, and the achieved conclusion is that the system benefits from the combination of both sensor modalities. An example of this application can also be seen in (Fougner et al. 2011), where an accelerometer is used in conjunction with EMG in order to improve classification results of arm movements.

The author in (Novak and Riener 2015) mentions 4 different approaches to performing the fusion of sensors: single fusion algorithm; unimodal switching; multimodal switching; and mixing.

3. INPUT DATA PRE-ANALYSIS

In a first approach to the problem, the abilities of the MYO armband were evaluated, by analysing the output data obtained. The purpose of this first evaluation is to detect the existence of motion using no segmentation method, analysing solely the performance of the data from the sensors on the detection of dynamic movements. The two sensors were studied separately.

3.1. IMU

A sequence of movements was determined and reproduced wearing the MYO armband. That sequence combination was selected with the purpose of triggering all the possible signals associated with the IMU, namely linear acceleration, angular velocity and orientation. The sequence was performed, with the following stages observable in the sequence data:

1º) Sequence of undefined movements, associated with the setup of the planned sequence;

2º) Static gesture, with the arm resting in the horizontal direction and the palm facing downwards, for approximately 20 seconds;

3º) 5 consecutive horizontal movements, turning the arm around 90º degrees, rotating in the yaw angle;

4º) 5 consecutive movements with rotation of the hand around 90º degrees, rotating in the roll angle;

5º) 5 consecutive vertical movements, rotating 90º in the pitch angle from the pose described in the 2nd step to a pose with the arm pointed upwards in the vertical direction;

6º) 5 consecutive vertical movements, rotating 180º in the pitch angle, from a pose with the arm pointed upwards to a pose with the arm pointed downwards;

7º) Static gesture similar to the 2nd step;

8º) Sequence of undefined movements, associated with the ending of the sequence.

Note that the purpose and placement of the static gestures was not only to include static rest in the sequence but to distinguish the relevant from the irrelevant data obtained in the first and eighth steps. The ideal output to this sequence would be uninterrupted dynamic motion detected between the two static gestures, with the undefined movements related to setup not being considered.

From this sequence, data pertaining to both IMU and EMG sensors was obtained, with the IMU data having a sampling rate average of 50 Hz and EMG data of 200 Hz.

The graphic described in figure 3.1 represents the behaviour of the 3 components of linear acceleration throughout the sequence. It is quite distinguishable the various steps in the analysis of the graphic. However, it is also possible to notice the random natural noise in the static poses, resulting from human shaking.

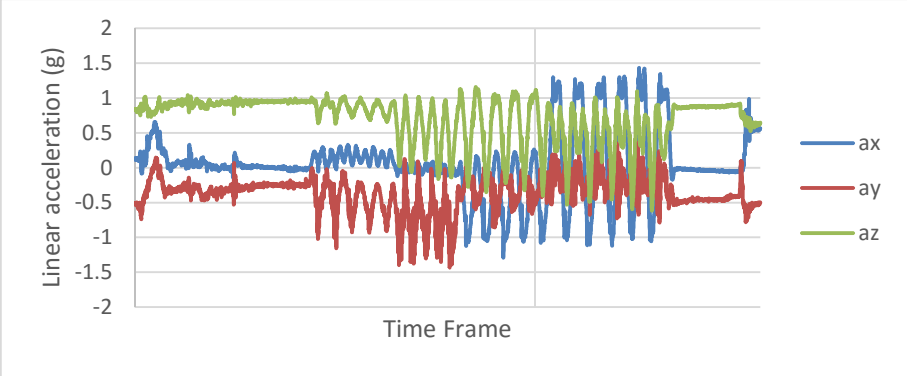

Figure 3.1 - Behaviour of components of linear acceleration thorughout the sequence

Obtained the behaviour of the acceleration components, the Euclidian distance is calculated, as described in equation 3.1, for each time frame.

$$a_r = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{3.1}$$

From this results the graphic in figure 3.2, where the behaviour of the resulting acceleration can be viewed:

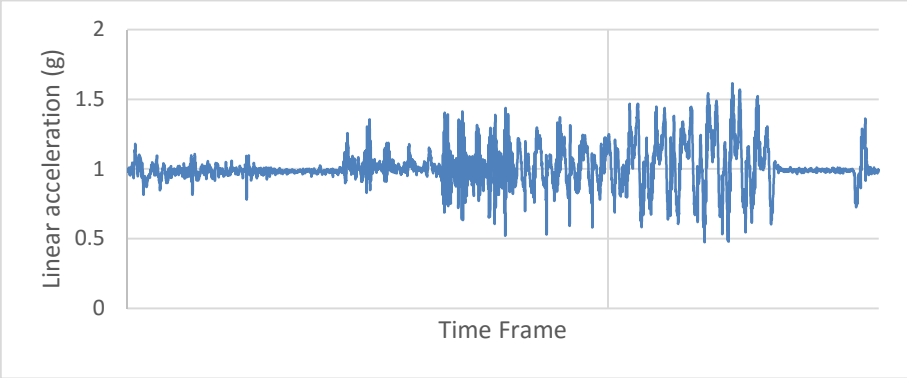

Figure 3.2 - Resulting acceleration throughout the sequence

In graphic 3.2 it is noticeable the presence of gravity throughout the entire sequence, which is responsible for an approximate average of 1 on the acceleration value. Using function *VAR.A* in Excel, the variance of the resulting acceleration for each time frame *var* is obtained, allowing for a comparison between consecutive values of acceleration In this case only 2 consecutive values of acceleration were compared for variance.

In order to distinguish between static and dynamic moments, the values *var* obtained in the variance graph shown in figure 3.3 were compared with a threshold *T*. That threshold was obtained through approximations by trial and error and the value which allowed for a fine distinction between rest and motion was estimated to be around 0,0015.

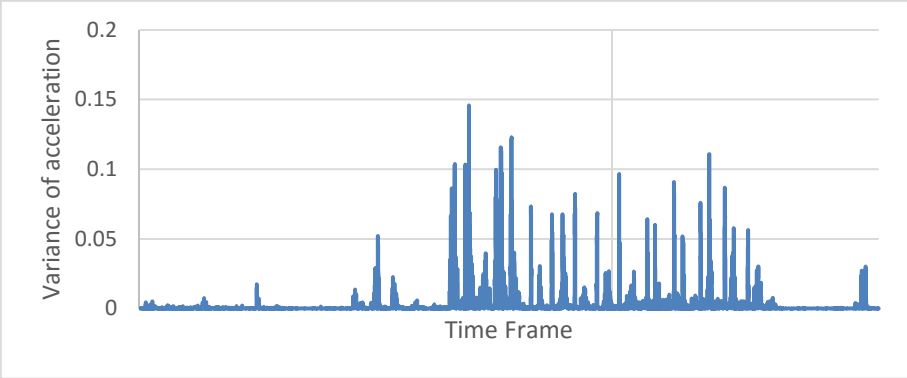

Figure 3.3 - Variance of acceleration

Following the function *s* in equation 3.2, a graphic describing the frames associated with hand motion is shown in figure 3.4.

$$s = \begin{cases} 1, & if\ variance > T \\ 0, & if\ variance < T \end{cases} \quad \textbf{(3.2)}$$

Figure 3.4 – Dynamic motion frames in the sequence based on linear acceleration

Through analysis of figure 3.4 and comparison with its input data, one can see that there are no motions detected during static rest with the acceleration feature, however there are multiple static motions detected throughout the dynamic part of the gesture sequence, which results in a multitude of false negatives when evaluating gestures. Such false negatives can be attributed to multiple causes:

Moments where the movement was being performed and there was no significant change of velocity during the interval of time frames, even though a noticeable movement is being performed. This leads back to (Simão, Neto, and Gibaru 2016), where the inverse occurs: in the cases of steady velocity, there are no variations of acceleration, which can trigger false negatives.

Transitions between different movements, where small pauses might occur.

The graphic in figure 3.4 shows that linear acceleration feature by itself is not a valid solution, therefore other features should be investigated. The same procedure can be applied towards the other features of the IMU.

In the case of angular velocity, the same process used for the linear acceleration feature was applied. The threshold value used for angular velocity was 25. Similarly to linear acceleration, angular velocity appears to show a vast number of false negatives in its sequence, shown in figure 3.5.

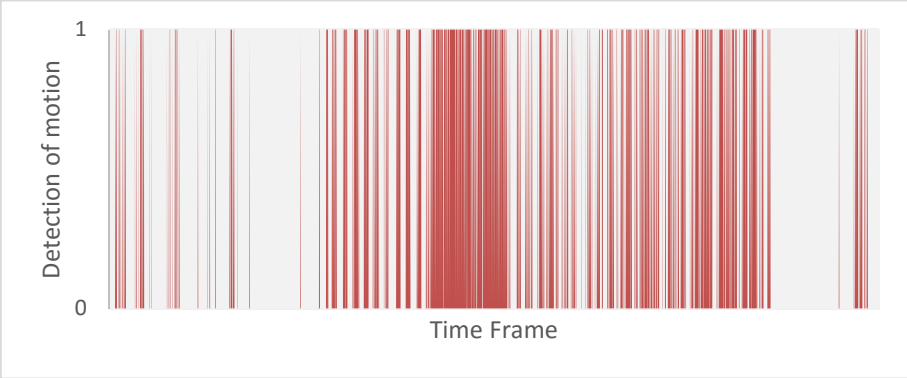

Figure 3.5 - Motion frames on the sequence based on angular velocity

Orientation can also be evaluated as a feature for motion detection. While it is generally not used as a feature in literature, its performance for motion detection was tested in this case, using the same process as for linear acceleration.

The major difference from the other features comes from the fact that orientation itself represents position. As such, to consider orientation for motion detection, one must analyse the differences between consecutive values, rather than the values themselves.

The obtained graphic in figure 3.6 appears to be very effective in the discerning of motion, with fewer false negatives during the dynamic phase of the motion when compared to other features. However, using a threshold of 0.01, the number of false negatives is still high.

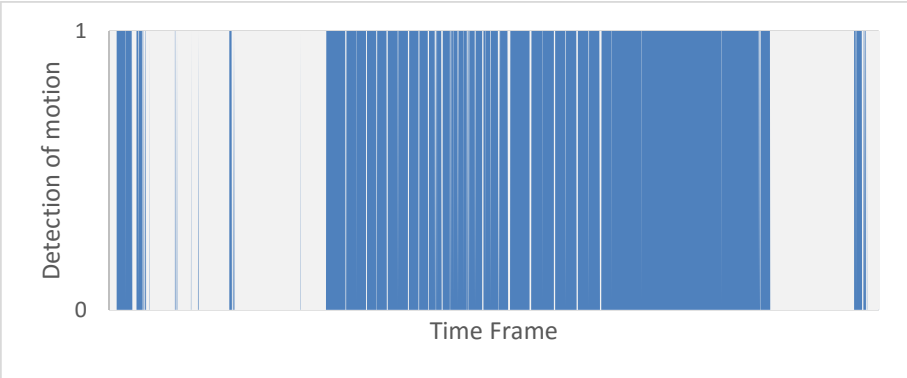

Figure 3.6 - Motion frames on the sequence based on variation of orientation

While the 3 different features may not be effective alone, using the 3 together for motion detection may present better results. The 3 obtained motion evaluations are merged together, as to analyse the presence of false negatives in the dynamic segment, as shown in figure 3.7. While this method allows to reduce the number of false negatives in the

sequence, it can however add the false positives from the individual graphics, increasing the error in the static phase.

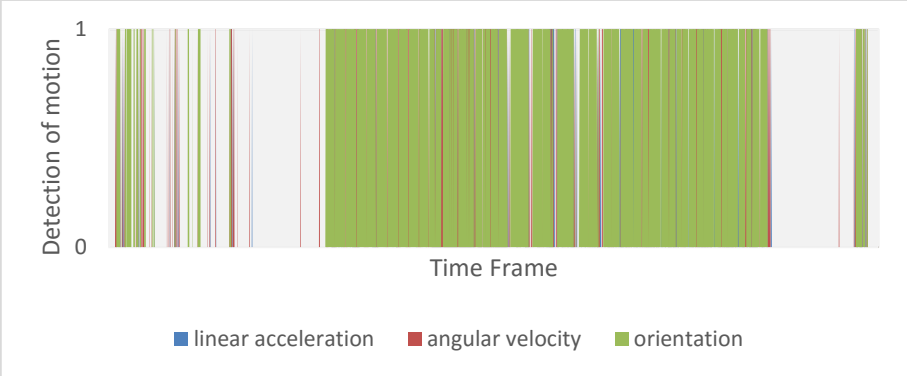

Figure 3.7 - Motion frames on the sequence based on all 3 IMU features

Even after merging the 3 graphics, some false negatives can still be seen in the dynamic phase in figure 3.7. Coupled with the present false positives, it is concluded that this method is not effective in dealing with motion detection of the IMU sensor, and therefore a sliding window algorithm is recommended.

3.2. EMG

To study EMG signals, 8 graphics were created for the 8 different EMG signals, in order to attempt to identify motion. However, using the data from the previous sequence, consistent values throughout the sequence for the EMG signals were obtained on all 8 graphics, with some exceptions due to random movements and noise. This is due to the fact that, throughout the entire sequence, the hand gesture performed was always a stretched palm. As such, the values obtained from the EMG signals refer to the muscle activations required to maintain that gesture, as well as the muscle strength required to counter both gravity and the inertia resulting from the arm movements performed in the sequence.

Given such, a new trial was performed. The arm stood in a horizontal position, as to consider the noise read by the EMG from the arm necessary to counter gravity and inertia.

The hand gestures performed in the sequence were:

1º) 5 successive movements of opening and closing of the fist

2º) 5 successive motions of stretching apart and joining the fingers

3º) 5 successive motions of waving up and down

In figure 3.8 the data output from EMG sensor 1, can be observed with noticeably higher values during dynamic motion.

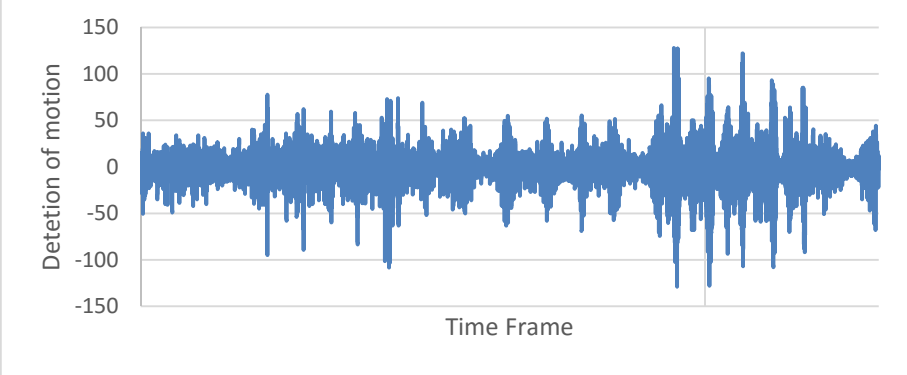

Figure 3.8 - Behaviour of EMG data during the sequence for EMG sensor 1

Using the same method as for other features, the variance of the EMG signal from each of the 8 sensors was analysed and, as shown in figure 3.9, a frame was considered to show dynamic motion if any of the 8 sensors' variance data showed dynamic motion. A manually defined threshold of 10000 was used for this case.

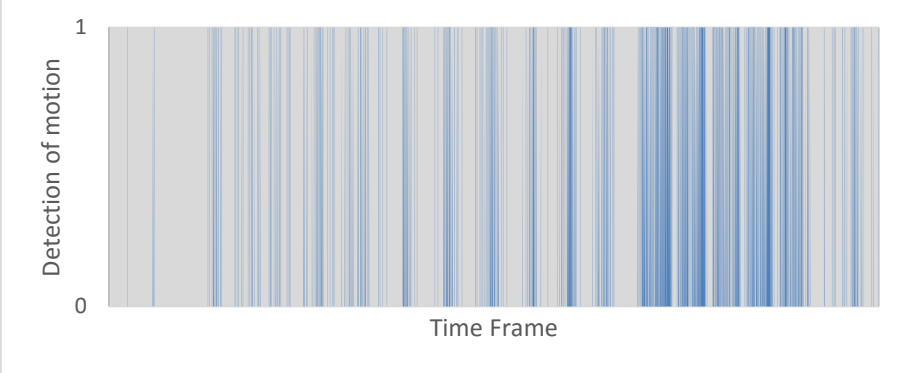

Figure 3.9 - Motion frames for variance of EMG data

As it can be observed in figure 3.9, the EMG sensor method appears to be capable of distinguishing dynamic gestures, however they are too over segmented for gestures to be recognized. The data in figure 3.9 will not be merged with data from figure 3.7 since the sampling rates are not similar.

Given this, a method to compensate for the various false negatives must be used if gestures are to be successfully segmented, and a solution for this issue can be found in the sliding window method.

4. SLIDING WINDOW METHOD

The sliding window method is based in (Simão, Neto, and Gibaru 2016). The sliding window method relies on input data from the MYO armband as well as calculated thresholds in order to output:

1) A motion vector, which detects in each timeframe whether the movement is identified as dynamic or static.
2) A feature vector, which provides the data on the features for the classifier to recognize gestures.

The input data available for the sliding function from the IMU sensor in the MYO armband are the linear acceleration components *a*, the angular velocity components *g*, the sensor orientation, both in quaternions *q* and Euler angles *o*, and the timestamp associated with the time occurrence of the frames t_{IMU}. The 8 EMG signals *semg* are obtained by the EMG sensors, as well as the timestamp associated with the EMG data frames, t_{EMG}, different from the IMU timestamp.

$$Input_{IMU} = [a_x, a_y, a_z, g_x, g_y, g_z, o_x, o_y, o_z, q_x, q_y, q_z, q_w, t_{IMU}]$$

$$Input_{EMG} = [semg_1, semg_2, semg_3, semg_4, semg_5. semg_6, semg_7, semg_8, t_{EMG}]$$

Additional information must be acquired for the method as described in (Simão, Neto, and Gibaru 2016): the thresholds for each feature *T*; the sensitivity factor *k*; and the window size *w*.

In an initial attempt, a sliding window function which merged both input data from IMU and EMG was attempted. However, this soon brought in some issues:

The gestures evaluated by the sensors are different. While this would be beneficial in evaluating the existence of any kind of gesture, issues before mentioned regarding high EMG values during static gestures could compromise the detection of arm movements.

The sampling rates for both sensors are also different, with values of IMU and EMG sampling being of 50 Hz and 200 Hz respectively, which would require a method to correlate.

Considering the present issues, the solution was to start by studying the performance of the sensors individually, by building a sliding window function for each of the sensors.

4.1. Sequence for motion detection

In order to study both arm movements detected by IMU and hand movements detected by EMG, a sequence containing both was needed. The motion sequence shown in figure 4.1 and performed in (Simão, Neto, and Gibaru 2016) was chosen, given its variety of movements, as well as to allow a comparison between both works.

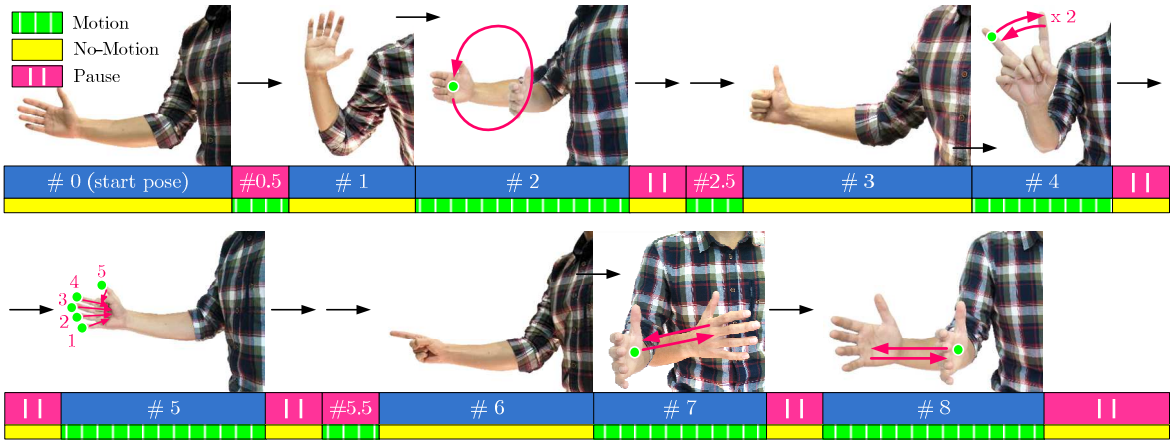

Figure 4.1 - Performed gesture sequence (Simão, Neto, and Gibaru 2016)

The sequence is composed of 8 different dynamic movements, including both arm and hand movements, which are signalled in green. While some are clearly identified by numbers – #2, #4, #5, #7 and #8 – the other 3 have been identified in (Simão, Neto, and Gibaru 2016) as movement epenthesis. They will throughout the work be identified with a number according to their position in the sequence - #0.5, #2.5 and #5.5.

An important note to take from this sequence is that not all gestures are guaranteed to be detected by both the sensors of the armband, as some correspond to only arm gestures and some to only hand gestures, with the other sensor being an auxiliary source of information in those cases.

Out of the 8 gestures, the ones which are expected to be detected by the IMU are gesture #0.5, which includes the lifting of the arm; gesture #2, with a clockwise motion of the arm; gesture #4 which includes the rotation of the arm to aid in performing the hand gesture; gesture #5 includes a small arm rotation since a rotation of the wrist is performed during the transition from gesture #4 to #5; and gestures #7 and #8, which include motion of the arm to the side. However, the other gestures, depending how they are performed, may be detected as well, since gestures #2.5 and #5.5 may include additional small movements of the arm, as the arm is likely to shake when performing the gesture.

Regarding EMG data, the ones expected to be detected are gesture #2.5, which includes the clenching of 4 fingers; gesture #4 where a transition gesture from the hand pose from #3 to #4 is performed; gesture #5 which includes both the transition from #4 to the initial pose in #5 and the clenching of the fist; gesture #5.5, where the index finger is stretched; and gesture #7, which includes the transition from the pose in #6 to the gesture in #7. It is important to notice that, similarly to IMU detection, all other gestures may include involuntary hand movements when performed or muscle co-activation when performing arm gestures, as well as detecting the force required for the hand to maintain a similar pose during the gestures.

In summary, the group of relevant gestures to be captured by the IMU sensor is R_{IMU} = [#0.5, #2, #4, #5, #7, #8] and the group of relevant gestures which are to be captured by the EMG sensor is R_{EMG} = [#2.5, #4, #5, #5.5, #7]. Gestures which only rely on arm gesture are O_{IMU} = [#0.5, #2, #8] and those which only rely on hand movement are O_{EMG} = [#2.5, #5.5].

4.2. Sliding Window for IMU

4.2.1. IMU features for motion detection

In regards to the features for motion detection, the initial choice for features of the IMU data were the features which are directly obtained from the data, namely linear acceleration, angular velocity and orientation.

While acceleration and velocity indicate by themselves the existence of movement, orientation is a variable which represents position. As such, similar to the case in (Simão, Neto, and Gibaru 2016) where the features are joint angles, for orientation to be

used as indicator of movement, the differences between frames of the feature would have to be considered. As such, the feature to be used is variation of orientation *do*, presented in equation 4.1, as in the difference between consecutive values of orientation.

$$do(i) = o(i) - o(i-1) \quad \textbf{(4.1)}$$

Since each of the features is composed of 3 different individual values, dependent on their respective axis, we reach an issue mentioned in (Simão, Neto, and Gibaru 2016), where "a motion pattern with a direction oblique to an axis would have lower coordinate differences compared to a pattern parallel to an axis with similar speed, thus producing different results." As such, a similar solution will be used, with the 3 coordinate components replaced by the respective Euclidian distance, for all 3 features, equal to what was used in the initial analysis.

Another issue faced was that the Euclidian distance for linear acceleration included also the gravity component. A method for removing the gravity component is discussed in (Neto, Pires, and Moreira 2013), using the orientation data to build the rotation matrix. A solution found was to build the rotation data based on quaternion data available from the sensor, but that proved to be an unreliable method as the quaternion use as origin not the Earth frame but the origin frame when the MYO armband is connected. No other solutions were found to calibrate the initial frame and as such, this method was abandoned. The linear acceleration feature was therefore not changed, with its gravity component always present.

4.2.2. Selection of threshold for IMU

The threshold used in the sliding window method is an important factor for a correct segmentation of the motion sequence. While (Simão, Neto, and Gibaru 2016) proposes an automatic optimization of thresholds using a genetic algorithm, this work opts for a simpler solution.

A static motion sequence has obtained, in which the user was sat down, with the arm at rest and supported by the chair. A small time sample from the sequence was extracted, in which there was no clear intention of motion. The time sample has a length of

approximately 3 seconds, as completely static poses are hard to maintain for long periods of time without unwanted motions.

It is taken into account that the ideal mean value in a perfectly static stance for linear acceleration should be 1 g, due to the consistent effect of gravity, and 0 for both angular velocity and variation of orientation since ideally these values remain unchanged. As such, the threshold calculation consisted on finding the value which resulted in the largest difference between the ideal value and itself in the entire static sequence.

Using this method, the obtained values for the thresholds of linear acceleration, angular velocity and variation of orientation were respectively shown in equation 4.2.

$$T_{IMU} = [0{,}00075 \quad 3{,}165364 \quad 0.001802] \tag{4.2}$$

However, the obtained thresholds showed to not be very reliable in later stages. This was concluded to be due mainly to the fact that during the rest position defined in the sequence, the arm had no support as above, so the threshold values did not take into account the arm shaking necessary to counter gravity. New thresholds values were obtained using the above method but this time with the arm standing horizontally with its user standing up. As such, new threshold values were obtained in equation 4.3.

$$T_{IMU} = [0{,}037140 \quad 9.029033 \quad 0.044568] \tag{4.3}$$

4.2.3. Orientation: a redundant feature

While the study of orientation as a feature has been performed in the initial analysis, according to MYO developers, the orientation is derived from angular velocity and liner acceleration data. As such, despite being used in the initial analysis, its redundancy has always been questionable, since in no other known article is orientation used as a feature.

Variation of orientation was compared to angular velocity and linear acceleration in figure 4.2 and it was confirmed that, due to the vast similarities between the data, that variation of orientation is indeed derived from angular velocity and therefore the information provided by this feature is redundant, with no benefit in applying it to motion detection.

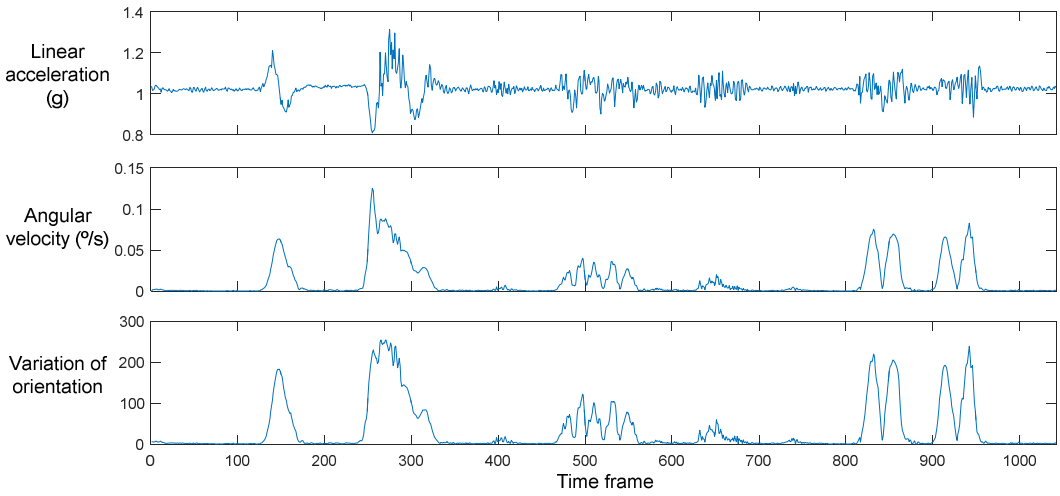

Figure 4.2 - Representation of motion features - linear acceleration, angular velocity and variation of orientation - from a sample of the sequence

4.2.4. Window Size

To find the ideal window size, analysis was made using different sensitivity factors for motion. Initially using a sensitivity factor of 3, same as in (Simão, Neto, and Gibaru 2016), the minimum window size necessary for no false negatives to occur was calculated to be of 9 frames, with a false negative occurring when the window size was lower. When increasing the window size above 9, the only noticeable difference was an increase in the sizes of the windows up to 39 frames, when the window became too large and different gestures, specifically gestures #7 and #8, were no longer discernible.

To confirm results, two other sensitivity factors were used, with values 2 and 5. In these cases, the minimum window size required is 10. In light of these results, the window size of 10 was defined as the minimum ideal value for the IMU sensor. This represents a window size with a time length of 200 milliseconds, given the 50 Hz sampling rate of the IMU.

4.2.5. Sensitivity Factor

Using the previously defined window size of 10, an ideal sensitivity factor was also sought after, by analysing the evolution of the sensitivity factor as it increased which is shown in table 4.1. Starting with a sensitivity factor of 1, which included far too many false positives, the value was gradually increased up to 1.2, where a clear distinction between different gestures was capable of being made, and where all gestures were being detected.

As can be seen in table 4.1, as it further increased, false positives in between gestures were gradually eliminated, but as the factor reached a value of 1.6, the IMU sensor was shown to no longer being able of detecting gesture #5.5. Since this is a gesture which relies mainly on hand movement, this non-detection is not problematic. With a factor of 1.7, all clear false positives were eliminated.

Table 4.1 - Result of segmentation method depending on sensitivity factor k for the IMU method, with w of 10 at 50 Hz sampling rate

k	Observations
< 1.2	Far too many errors
1.2	FP before #2, FP after #4
1.3	FP before #2 deleted
1.6	No longer detects #5.5
1.7	FP after #4 deleted
2.1	No longer detects #2.5, FN #5
2.2	FN in #5 deleted
3.5	FN in #5
3.8	FN in #5 deleted
5.4	FN in #4

The possible acceptable values for the sensitivity factor varied on a range from 1.7 to 5.4 according to table 4.1. Since the IMU was only able to detect the gesture #5.5 up to a value of 1.6 and gesture #2.5 up to 2.1, it is concluded that the IMU sensor may not be reliable for reading the motion generated from hand movement and therefore the methodology may rely on additional assistance for these movements to be successfully read.

4.2.6. Sliding Window Algorithm Design

By studying the code provided in (Simão, Neto, and Gibaru 2016), an algorithm for the extraction of motion features and applying the sliding window method was designed for the IMU sensor data. The code is mostly similar, albeit adaptations had to be done, including the data treatment to obtain the chosen motion features.

One of the most noticeable changes is in the comparison of the linear acceleration feature to its defined threshold. While all features obtain positive values at all times due to Euclidian distance calculation, since the linear acceleration values can be either superior or inferior to 1, a modification of that process had to be performed, starting with the threshold value, which is shown in the segment of code in figure 4.3. After being multiplied by the sensitivity factor *k* in lines 1-3, a value of 1 is added to the linear acceleration threshold in line 4, in order to consider the persistent effect of gravity. In line 9, an additional condition

is added for the motion condition not only to detect acceleration values above the sum of threshold and unitary value but also below the value equal to 1 minus the acceleration threshold.

```
Algorithm SlidingWindowIMU(n,T,k,w,O)
Inputs:   n timestamp
          T threshold vector
          k threshold sensitivity factor
          w window size
          O observation matrix
Output:   M sliding window motion function

1:  for i ∈ [1, LENGTH(T)] do ► Apply sensitivity factor
2:     t(i) ← k · t(i)
3:  end for
4:  T(1) ← T(1) + 1 ► Add gravity to acceleration threshold
T(1)
5:  F ← GetFeatures(O) ► Calculate features using equation
    of eclidian distance shown in equation 3.1
6:  for i ∈ [1, LENGTH(n)] do ► Obtain the motion binary
    function
7:  m(i) ← 0
8:     for j ∈ [1, LENGTH(T)] do
9:        m(i) ← (F(i,j) ≥ T(i)) V (F(i,1) ≤ (2 - T(1)) V
    m(i)
10:    end for
11: end for
12: for i ∈ [w, LENGTH(n)] do ► Apply sliding window func-
    tion for motion detection
13:    for j ∈ [1, w-1] do
14:       M(i) ← m(i) V m(i + j)
15:    end for
16: end for
```

Figure 4.3 – Pseudocode for sliding window motion function for the IMU method

4.3. Sliding Window for EMG

Similar to the algorithm for IMU data, the EMG sensor data algorithm was also based on (Simão, Neto, and Gibaru 2016).

4.3.1. EMG Data Filtering

In the treatment of data, rectification of the data from the 8 EMG sensors was applied using the *abs* function. This was followed by a low-pass Butterworth filter using function *butter* in order to obtain the filtering coefficients, which are then applied to a zero phase digital filter using the function *filtfilt*. The parameters used in the *butter* function were a sampling frequency of 200 Hz, in accordance to the EMG data frequency; a cut-off frequency of 3 Hz, and a filter order of 4, both which were compared with different values and provided a decent signal which avoided significant signal peaks due to noise while avoiding excessive smoothing of the data.

In the code shown in figure 4.4, s*emg* refers to the original data, *remg* refers to data after rectification and *femg* refers to filtered data.

```
% Signal rectification
remg1 = abs(emg1');

% Butterworth filter design
Fe=200; % Sampling frequence
Fc=3; % Cut-off frequency
N=4; % Filter order
[B, A] = butter(N,Fc*2/Fe, 'low'); % Filter's parameters

% Zero-lag filter for off-line treatment
femg1 = filtfilt(B, A, remg1);
```

Figure 4.4 - Matlab code segment for design of low pass filter

The application of the EMG filter can be seen in figure 4.5, with the data from EMG sensor number 1 (identified according to figure 2.2) retrieved from a sample of the gesture sequence.

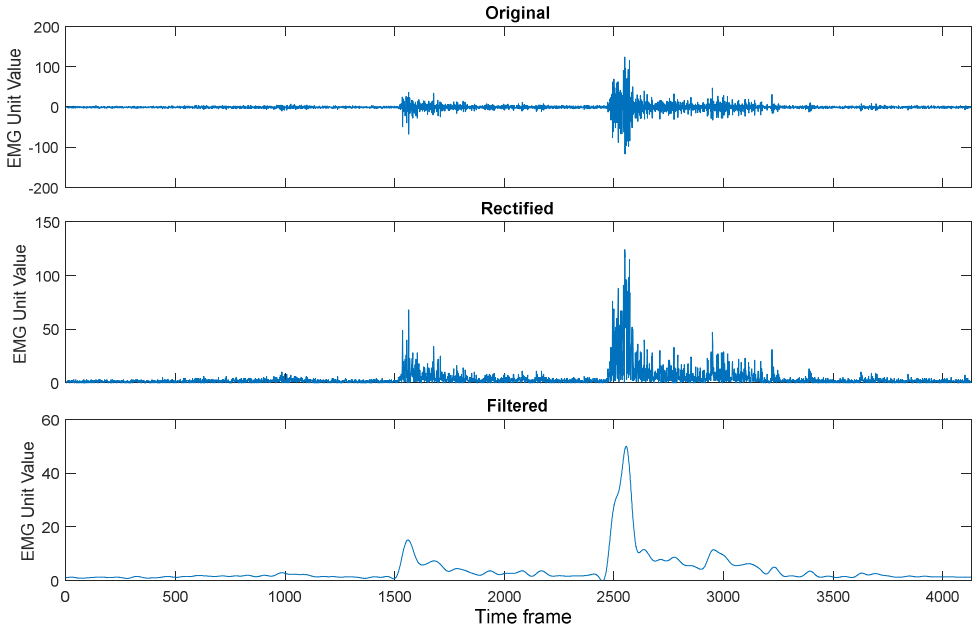

Figure 4.5 - Treatment of EMG signal data obtained from EMG sensor 1 in the initial sample: Original data is rectified and then filtered

A comparison between the filtered signal and the original one is generally required, to take into account possible lag that may occur from the filtering process and compensate for the error if necessary. Analyzing figure 4.6, it is visible that the data shows no significant delay error, due to the application of the *filtfilt* function.

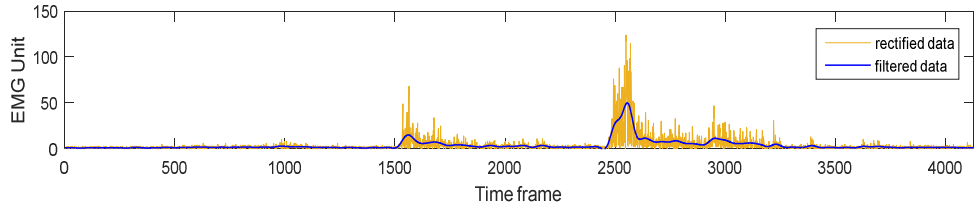

Figure 4.6 - Comparison of rectified data and filtered data from EMG sensor 1 data

4.3.2. EMG feature for motion detection

Multiple parameters were tested to detect and discern dynamic from static hand motion. The main objectives in looking for a good parameter were:

1) A method which did not discern the relevance of different EMG signals. The analysis of different EMG sensors with different thresholds is an issue given the fact that these sensors do not have fixed positions, which can shift with every new usage of the armband.

2) A method which allowed for a clear distinction between static and dynamic gestures even when performing hand poses. Even when static, a certain muscle strength is required to maintain the gesture. As such, some EMG data values may be consistently high depending on the gesture being performed.

For this end, multiple parameters were tested and for each parameter, a threshold was manually defined through trial and error.

For this analysis, a new sequence was performed, with multiple stages which involved doing hand poses followed by small pauses, and with the arm always standing horizontally. The sequences is as follows:

1º) Start with hand stretched

2º) Clench fist

3º) Stretch index finger

4º) Stretch hand

5º) Stretch fingers apart from each other

6º) Clench fist again

7º) Raise thumb

8º) Relax hand

4.3.2.1. Base Values of EMG

This feature, presented on equation 4.4, relies on directly analysing the EMG values obtained from the filtering stage, with *i* identifying the respective EMG sensor and *n* the time frame for equations 4.4 to 4.10.

$$val_i(n) = femg_i(n);\ i = 1, \dots, 8 \quad \textbf{(4.4)}$$

However, this feature did not allow for an indiscriminate distinction between different EMG values, with weak overall performance. Using the same threshold value, it has to be adjusted towards sensors which output larger EMG signal values, which makes sensors with weaker signals irrelevant, whereas using different threshold values results in issues due to EMG not having fixed positions.

4.3.2.2. Sum of values

This feature consists on adding the 8 EMG signals into a single feature, as shown in equation 4.5.

$$sum.v(n) = \sum_{i=1}^{8} femg_i(n) \tag{4.5}$$

It did provide good results with the sequence, but in the cases where the fingers were stretched both together and apart, it failed to distinguish the dynamic from the static gesture without selecting a threshold which would harm detection of other gestures.

Yet some sensors' values showed clear increases during dynamic gestures but their exhibited values were overshadowed by other sensors, whose values were significant in comparison even when performing static poses.

4.3.2.3. Weighted sum of values

This feature is a modification of the previous feature, but different signals were given values based on a comparison to the maximum value obtained throughout the sequence for that sensor, as shown in equation 4.6.

$$wsum.v(n) = \sum_{i=1}^{8} \left(\frac{femg_i(n)}{\max(femg_i)} \right) \tag{4.6}$$

It presented results similar to the sum of values feature in terms of performance, however it faced an issue when analysing a data sample with no dynamic gestures or very strong movements, as the method no longer had an efficient value for comparison.

4.3.2.4. Variance of the sum of EMG signal

The variance method did not perform well in the initial analysis, but with filtered data the results were different. The variances of the sum of the 8 EMG signals were analysed, as described in equation 4.7.

$$varsum(n) = \left(\sum_{i=1}^{8} femg_i(n) - femg_a(n)\right)^2 \left(\sum_{i=1}^{8} femg_i(n-1) - femg_a(n)\right)^2 \quad \textbf{(4.7)}$$

$$femg_a(n) = \frac{\sum_{i=1}^{8} femg_i(n) + \sum_{i=1}^{8} femg_i(n-1)}{2} \quad \textbf{(4.8)}$$

However, variance of the sum faced a similar issue to the sum of values, with multiple values being deemed insignificant in comparison.

4.3.2.5. Variance of the individual EMG signals

This feature studied the variance of each signal individually, as seen in equation 4.9, resulting in 8 different features.

$$var_i(n) = \left(femg_i(n) - femg_a(n)\right)^2 (femg_i(n-1) - femg_a(n))^2;\ i = 1, \dots, 8 \quad \textbf{(4.9)}$$

$$femg_a(n) = \frac{femg_i(n) + femg_i(n-1)}{2} \quad \textbf{(4.10)}$$

It showed the best performance of all studied features and was therefore chosen as the feature for motion detection.

The threshold chosen for the used sample when using this feature was 0.01, as it provided the most accurate solution for distinguishing dynamic and static motions.

4.3.3. Sensitivity Factor

To find the ideal sensitivity factor for the EMG threshold, in this approach we used the window size defined for the IMU sensor times the ratio between the EMG signal frequency and the IMU signal frequency. This resulted in an initial window size of 40. Similarly to the approach for the IMU, the evolution of the sensitivity factor was studied in table 4.2. The iteration here was with increments of 0.5.

Table 4.2 - Result of segmentation method depending on sensitivity factor k for the EMG method, with w of 40 at 200 Hz sampling rate

K	Observations
4	Too many errors
5	FP after #2, FN in #2.5, FN in #7, FP after #8
5.5	FP after #8 deleted
6	Another FN in #7
7.5	FP after #2 deleted
8	#0.5 no longer detected
9	#5.5 not detected
12	

Analysing table 4.2, it was concluded that EMG is not as reliable as IMU data at first sight, with far more errors and with no sensitivity factor which provides an error free solution. In fact, some errors, namely in gesture #2.5 and #7, persist even after relevant hand motion data from gesture #5.5 is no longer detectable.

The values for the sensitivity factor must be inferior to 9, according to the sequence, but it is also shown that a minimum value of 5.5 should be imposed, as it avoids a false positive after gesture #8.

4.3.4. Window Size

For the study of the window size, two sensitivity factors were used based on the previous calculations, with values 6 and 8, shown respectively in tables 4.3 and 4.4.

Table 4.3 - Result of segmentation method depending on window size w for the EMG method, with k of 6 at 50 Hz sampling rate

W	Observations
10	Too many errors
30	FP after #2, FN in #2.5, FN in #4, FN in #5.5, FN in #7, FN in #8
35	Deleted FN #4, FN #5.5, and FN #8
45	FN #7 deleted
50	Deleted FN #2.5, and FN #7
80	FP after #2 merged with #2
95	Fusion of #7 and #8

Table 4.4 - Result of segmentation method depending on window size w for the EMG method, with k of 8 at 50 Hz sampling rate

W	Observations
35	FN in #2.5, FN in #4, two FN in #7
40	FN in #4 deleted
60	FN in #2.5 and one of FN in #7 deleted
85	Second FN in #7 deleted
100	Fusion of #7 and #8

According to the results, the minimum window size when using a factor of 6 is 50, and when using a factor of 8 is 60 to avoid errors in #2.5. Errors in #7 can be ignored as limits since it is mostly an arm movement, with the initial transition of the arm detected, and these errors are likely to be solved when merging the sensors together.

In the light of these results, the chosen sensitivity factor for the EMG sensor was 7, as it represents an intermediate value between the imposed limits. In regards to the window size to be used for this sensitivity factor, the chosen value was 50. Albeit it still showed errors, these are expected to be deleted when evaluating the combination of sensors and also takes into account the erratic nature of EMG.

4.4. Sliding Window for both sensors

As before mentioned, due to connection issues when transmitting data by Bluetooth from the armband to the computer, it is hard for the MYO armband to have a steady sampling rate for any of the sensors. However, this error was ignored, since it depends on environment conditions and noise, and it was therefore assumed for the IMU to have a steady 50 Hz sampling frequency and the EMG to have a consistent 200 Hz sampling frequency.

Data was also aligned according to the timestamp values. When analysing data, it was concluded that, when using the MYO data capture software provided by Thalmic Labs, the initial timestamps of IMU and EMG data would not match, with IMU occurring first, but when the data capture was terminated, they would be terminated in the same instant, with never a time difference in the final frames superior to 1 milliseconds in the 4 samples analysed.

Assuming that the data provided steady sampling rates, which is not always true according to (Nyomen, Romarheim Haugen, and Jensenius 2015), the data from the sensors was adjusted in order to end in the same final frame. However, the initial frames always included a certain difference. By analysing 4 samples, the time difference found between sensors' initial frame varied between 22 to 38 milliseconds, which is the equivalent of 4,4 to 7,6 frames, with EMG data length being the shortest in all cases. This is explained due to packet loss in the communications between the computer and the armband, which causes

some frames of the EMG signal to drop and not be recorded in the data, causing the misalignment when considering steady sampling rates.

When using both sensors, a sensor fusion modality based on a single fusion algorithm was used. As such, should one of the sensors present a false negative in their motion sequence, this error can be covered by the detection of motion by the other sensor. Some false negatives are expected to be eliminated in the middle of gestures, but errors associated with false positives outside of gestures are likely to pile up.

4.4.1. Parameters for the EXP method

When performing the sliding window method, values before calculated of *k* and *w* had to be chosen for IMU and EMG sensors. Albeit a window size of 10 at a 50Hz sampling rate was defined as ideal for IMU, which would correspond to 40 Hz at 200 Hz sampling rate to maintain the equal time length of 200 milliseconds, the 50 Hz minimum required by the EMG sensor was chosen as the window size used for the method, with a time length of 250 milliseconds.

Sensitivity factors, on the other hand, could be chosen separately, with *k* of 2 for the IMU features and *k* of 7 for the EMG features.

4.4.2. Choice of sampling rate

In order to relate data from the two sensors which is provided with differing sampling rates, two different methods for sensor fusion were designed.

The first function was labelled *expand*, whose duty was to obtain an IMU signal which repeats 4 times to match EMG signal rate. This would assume the timestamp to use a sampling rate of 200 Hz.

The other function, named *minimize*, did the inverse of *expand*: obtaining an EMG signal for the 50 Hz rate with each frame being a result of the average of 4 distinct signals, merged to adjust to the smaller rate. However there is a risk that the EMG signals might lose significance.

For the sensor fusion, the above mentioned functions *expand* and *minimize* had to be analysed.

Expand function provided good results, allowing to replicate the results obtained from the tests with the sensors individually. Using values obtained in previous tests allows for a successful segmentation process with no errors when combining the sensors.

Minimize function however required new calculation for EMG sensitivity factor, but at first sight provided a good solution as well.

Expand function appears to be more heavy computational wise, however that is not an issue in an offline approach. On an online approach, where response time is crucial, it might be relevant. *Minimize* function, while it appeared to show a good solution, was not seen as reliable as the averaging of EMG signals after data having already been filtered could cause the EMG data to lose a lot of significance.

Expand function was therefore chosen for this offline approach, as it avoids distortion of the EMG data and allows to use values calculated before and therefore the EXP method was designed, resulting from the combination of both IMU and EMG methods previously designed.

4.4.3. Analysis of features on the EXP method

The features used in the method were analysed and compared with the motion segment output. In figures 4.7 and 4.8 it is easy to see its importance of the IMU sensor for the detection of all R_{IMU} gestures. In the case of gestures #2.5 and #5.5, the IMU features are less noticeable, with the angular velocity feature barely surpassing the threshold in gesture #2.5 and detecting a motion frame.

In the case of EMG features, it is possible to see in figure 4.9 that the most significant values are obtained at the beginning of gestures, with transitions in gestures #4 and #7 showing to be more significant than hand gestures#2.5 and #5.5, and start of gestures which only include arm motion being detected as well. Gesture #5 seems to be the most easily recognised, with multiple EMG sensors' data values surpassing the defined threshold throughout the gesture.

Note that there is a false positive present in the beginning of the sequence, originated due to setup and is not related to any gesture. Another trait visible from this analysis is that, in various gestures, there is hand motion detected before arm motion, referring to transitions between gestures, in the case of gestures #4 and #7, but also noise,

noticeable in the case of gesture #2, which may hamper the gesture classification and recognition.

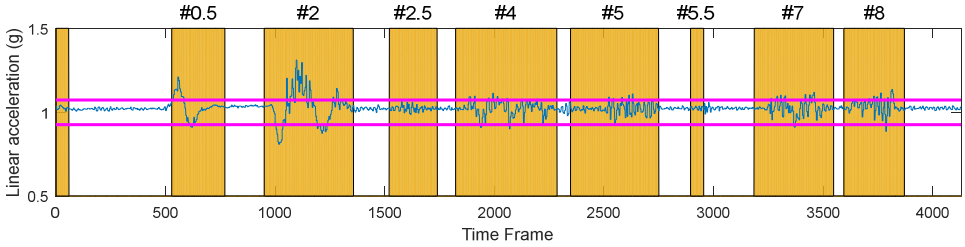

Figure 4.7 - Linear acceleration feature in motion segmentation

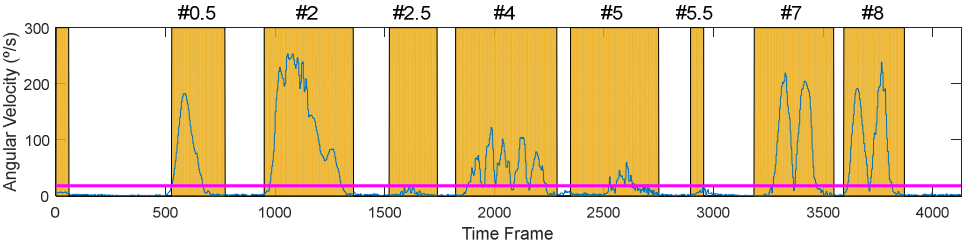

Figure 4.8 - Angular velocity feature in motion segmentation

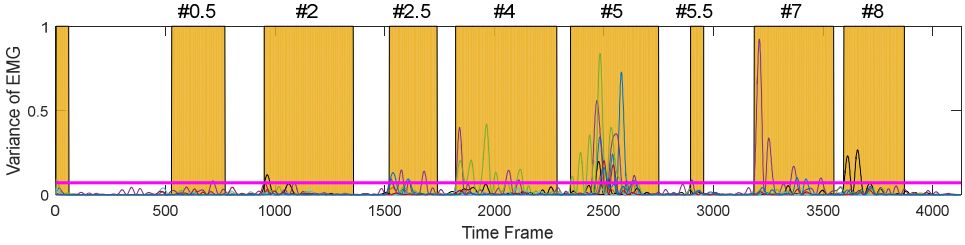

Figure 4.9 - Variance of EMG signals in motion segmentation. Features from signals of EMG sensors 1 to 8 included

4.5. Motion Dataset and analysis

The analysis was made by observing the motion output obtained by the EXP method, which is the combination of EMG and IMU signals, and the EMG and IMU methods alone. The purpose of this process is to better detect which motion segments correspond to each gesture, as the sequences were executed at different paces and with the presence of errors, it is often difficult to discern between gestures, as well as to achieve conclusions regarding the overall performance of each method.

4.5.1. Subject Recording

For the validation of the proposed segmentation method, the sequence was performed 10 times by 6 different participants, resulting in 60 test sequences, with a total of 480 gestures being evaluated.

The participants were requested to perform the sequence in figure 4.1 after a session of training, and pauses between tests were done in order to avoid the deterioration of data due to arm fatigue. Some participants did perform more than 10 sequences, but those tests included obvious errors when performing the sequence or had the user in a state of fatigue, which severely harmed the quality of the data.

4.5.2. Analysis of segmentation accuracy

In order to analyse the accuracy performance of the method, the chosen parameter for evaluation is segmentation error. Segmentation error is, according to (Simão, Neto, and Gibaru 2016), "the fraction between the number of segmentation errors (the sum of the number of times a gesture is over split and false segments of motion) and the number of samples". It is expressed in percentage in equation 4.11.

$$S_{error} = \frac{number\ of\ segmentation\ errors}{number\ of\ samples} * 100\ [\%] \quad \textbf{(4.11)}$$

Other parameters used in (Simão, Neto, and Gibaru 2016) were average start delay, average end delay, and extend level. However, they will not be evaluated in this work due to this study not being performed in real-time, with no method of comparing tests with ground truth.

5. RESULT ANALYSIS

5.1. Analysis of the sliding window methods

5.1.1. Approach to data analysis

An important factor to take into account is the evaluation of individual EMG or IMU data performance when considering non-relevant gestures. In the data analysis in this work, should a single segment be detected in the EMG data alone within a gesture period, regardless of whether it supposedly contained hand motion or not, it was assumed that the EMG had detected that gesture. The same process is used for the analysis of IMU data.

However, there is a clear difference in EMG segments from different samples for gestures which include arm movement. In the example of gesture #7, while in some cases the EMG segment would only include the initial moment where the hand transitions from gesture #5.5 to #7, in other cases it includes the entirety of the gesture, with length comparable to IMU, since there is noise from the hand being detected as the arm motion is performed. While this factor can be either beneficial - EMG sensor noise which nullifies IMU false negatives - or harmful - larger EMG segments which go beyond the gesture - in the combination of sensors, in the case of the individual EMG sensor this can greatly affect results in gesture recognition.

5.1.2. Types of Errors found

Multiple errors were found throughout the analysis of data by the 6 participants, with the following described errors found for the 3 modalities.

5.1.2.1. False positives due to setup

In the moments the recording of the sequence begins or ends, it was commonly registered for one or more false positives to occur. These were related to random movements performed by the users in preparation for the sequence or after the sequence was performed. Out of the 60 sequences analysed for testing, 44 showed setup errors, most commonly associated with random hand noise detected by the EMG sensors in the beginning of the

sequence. For example, figures 5.1, 5.2, 5.4 and 5.5 all include a false positive due to a setup error.

They can be identified as errors related to the recording rather than the actual gesture sequence, and analysis of segmentation error will be performed both ignoring and taking into account this type of error due to its low relevance.

5.1.2.2. False negatives due to differing arm and hand motion

With the combination of sensors, for gestures #4 and #7, in the beginning of the gesture there is a transition motion of the hand as it changes its pose from the previous gesture, which is included within the gesture. In certain cases, this motion was performed before the arm movement, distant enough for there to be a discernible separation in the gesture. This error is therefore classified as a false negative.

An example of this error can be seen in figure 5.1, where in gesture #7 the segment detected by the EMG ends before the start of the IMU detected segment, resulting in a false negative for gesture 7.

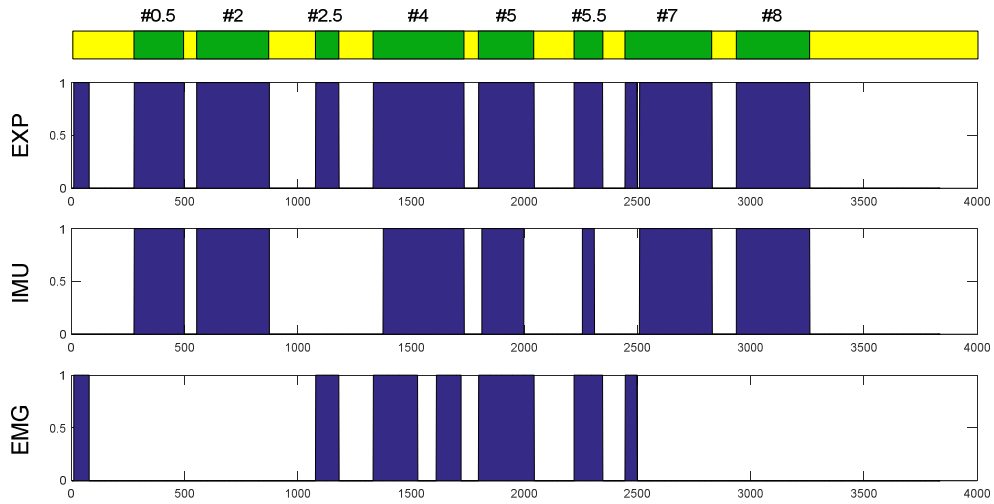

Figure 5.1 - Sample from participant [B] of the segmentation with the 3 methods: EXP (top), IMU (middle) and EMG (bottom)

5.1.2.3. False negatives mid-motion

In motion #4, #7 and #8, false negatives were found during the performing of gestures. This error occurs due to these gestures relying on inversions of the direction of movement, which resulted in the decrease of velocity when performing the gesture, with

possibly even small pauses when inverting movement for some cases. This conclusion was reached since the false negatives are easily discernible when analysing the IMU signal data alone.

This type of error can be observed when analysing the sequence in figure 5.2, where a false negative in the middle of gesture #4 can be seen in both the IMU sensor segmentation and the combined segmentation. Similarly, gesture #4 shows a false negative with the EMG method.

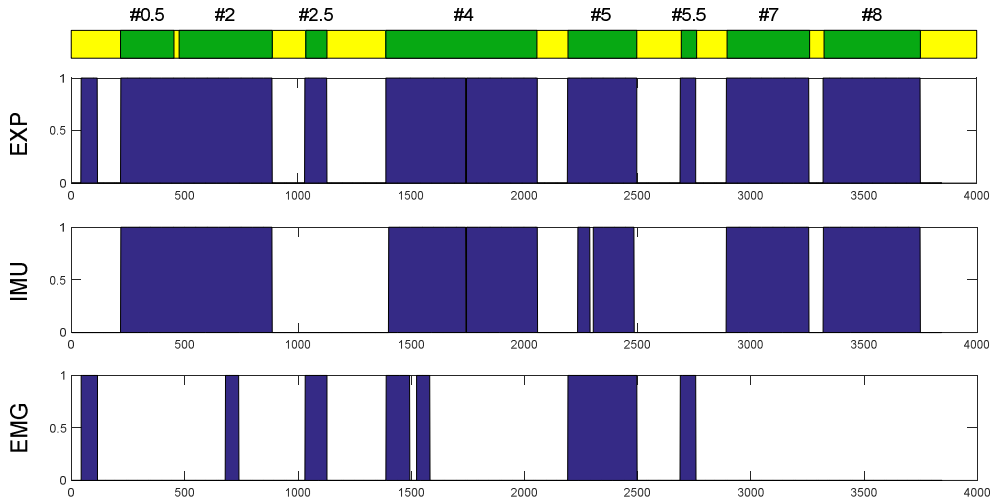

Figure 5.2 - Sample from participant [C] for the segmentation with the 3 methods

5.1.2.4. False positives in between gestures

In between gestures, some dynamic segments were detected by the sliding window function. These motions were concluded to be random natural shaking which was detected due to muscle fatigue or due to certain small hand gestures which were performed unconsciously by the participants in preparation for the different gestures.

In figure 5.3, a segment was detected by the EMG sensor in between gestures #2.5 and #4, which is identified as a false positive.

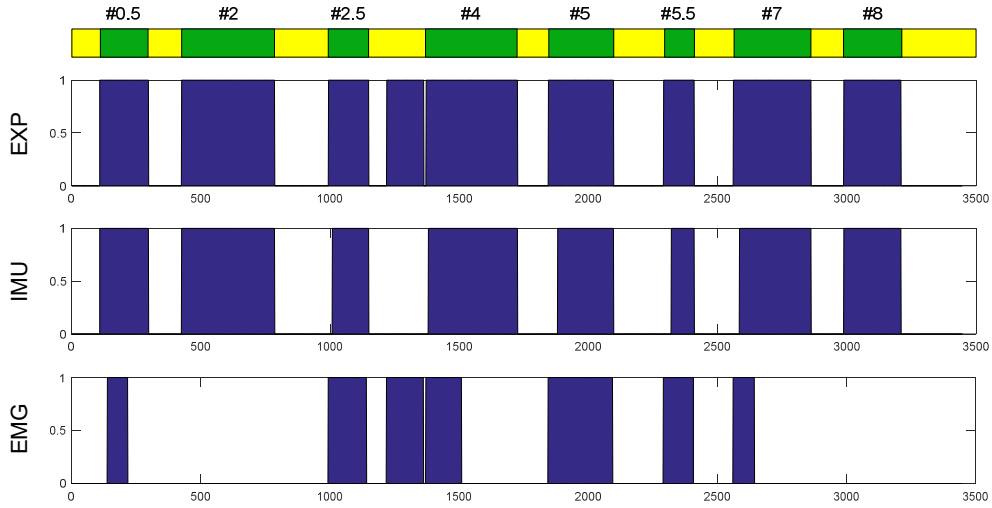

Figure 5.3 - Sample from participant [E] for the segmentation with the 3 methods

5.1.2.1. Undersegmentation

Certain gesture segments have no breaks between one another, with motion being continuously detected. Undersegmentation, also known as segment fusion throughout this work, can be seen as a false positive which occurs in between two segments. This error is assumed to not affect the first gesture performed, since its start can still be detected through pattern classification. However, the following gestures included in the fusion error cannot be discerned.

Many IMU data samples show that some gestures may be too proximate with one another, revealing that in some cases there may not have been a decent pause period in between gestures, due to the proximity of some segments, but yet only a single IMU sample shows two continuous gestures, shown in figure 5.2.

EMG data shows to be more frequent in this regard, with a total of 8 occurrences of fusion when analysing EMG sensors alone. This is due to a combined effect of small pauses in between gestures with noise, which resulted in false positives which could not be discerned from the real gesture segment due to the window size.

However, when combining both sensors, the piling of data from both has also in some situations led to gestures being indiscernible. Besides the factors mentioned above, the fact that motions from hand and arm can start at different times can further contribute for this error.

In figure 5.4, gestures #4 and #5 have merged when using the EXP method, despite being separated in each of the sensor methods. This occurs since there is motion associated with gesture #5 detected by the EMG sensor before gesture #4 is completed in the IMU method. In this case, it is suspected that the EMG segment of gesture #5 is extended due to random noise occurring before the gesture #5.

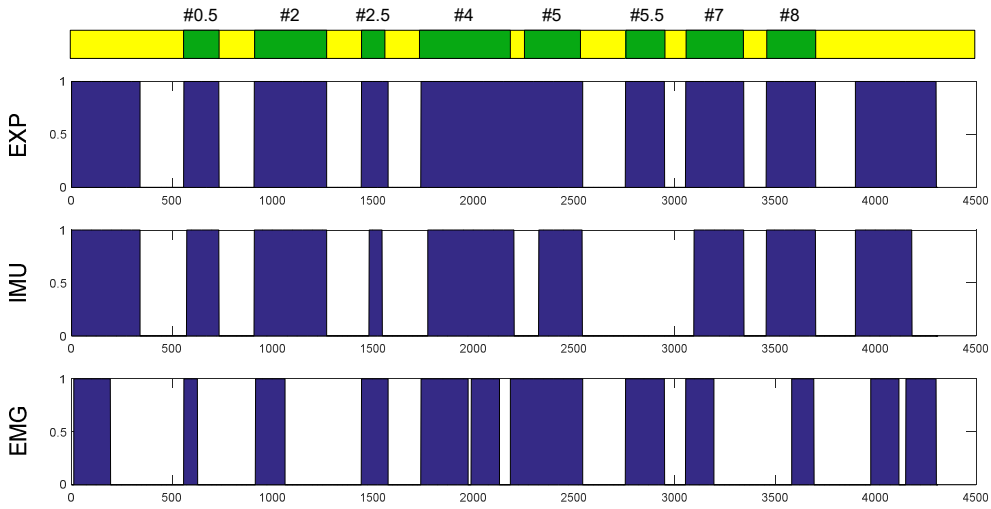

Figure 5.4 - Sample from participant [A] for the segmentation with the 3 methods

5.1.2.2. Gestures not being detected

In five sequence samples, the gesture #2.5 was not detected at all using the initial parameters with the combination of sensors. These can be explained as cases where the users not only did not perform an arm motion swift enough to be detected by the IMU but neither did apply sufficient strength to the fist grip for it to be well recognized by the EMG. The non-detection of gestures is considered a false negative.

In the case of the individual sensors, the non-detection of gestures was more usual, with it occurring for each sensor often when attempting to detect non-relevant gestures, as it was expected. However, EMG still shows a fair number of non-detection errors when considering EMG relevant gestures.

Figure 5.5 shows a clear example of this. Gesture #5.5 is undetected by the IMU but the EMG does detect it; whereas both gestures #7 and #8 are undetected by the EMG but the IMU sensor is capable of detecting them. Gesture #2.5 however is not detected by either sensor, and as such is unrecognized in the combination of sensors.

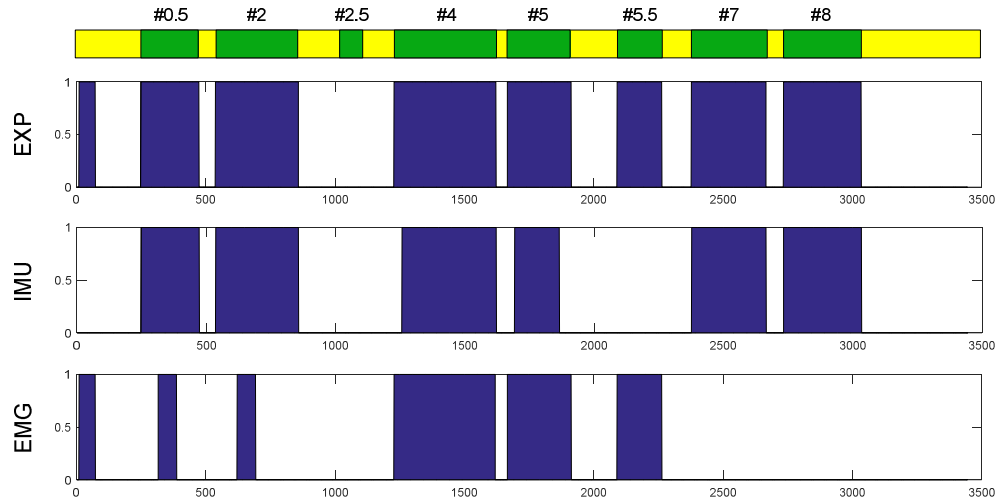

Figure 5.5 – Another sample from participant [B] for the segmentation with the 3 methods

5.2. Comparison between methods

After analysing the sequences, the overall performance of the sensors was concluded to be as shown in table 5.1, with the IMU method being the best performing regardless of setup errors.

Tabela 5.1 - Overall segmentation error (%) of methods non-including and including setup errors

Method	S_{ERROR}	S_{ERROR} with setup errors
EXP	12.91	27.08
IMU	11.88	16.67
EMG	43.75	57.92

5.2.1. EXP method

In the end, the resulting segmentation error for the combination of both sensors, assuming both false positives, false negatives and non-detection, is 12.92%. If setup errors were to be included, the segmentation error would increase to 27.08%.

The total number of errors present in the study when using the combination of sensors is 62. The most frequent type of error occurring is undersegmentation, responsible for 40,3% of the existing errors.

While this error could be justified by gestures not having significant pauses between each other, the main issue stands that, in the case of EMG, many segments often include some noise before and after the actual gesture which, due to the size of the window of 250 milliseconds, are not detected as false positives but instead are identified as part of the gesture.

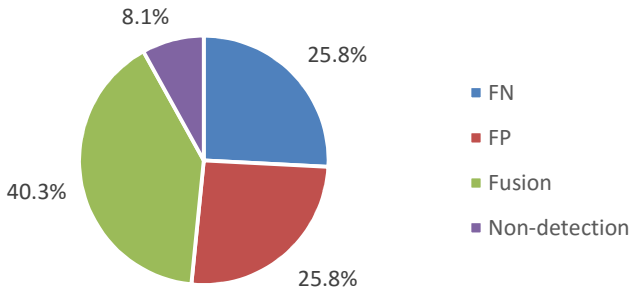

Figure 5.6 - Occurrence of each type of error for all gestures when considering combination of IMU and EMG sensors

5.2.2. Individual IMU

Evaluating IMU sensor alone, the segmentation error for all gestures is 11.88%, slightly lower than the total error. Including setup errors, the segmentation error is increased to 16.67%.

The number of errors that occurred using only IMU is 57, with a vast majority due to non-detection of gestures (59,6%). Notably, the only two gestures the IMU failed to detect were gestures #2.5 and #5.5, which are hand-only gestures and explains the big decline in the segmentation error when considering only relevant gestures.

Important notes to take are that the only false positive detected by the IMU sensor was the one that resulted in the fusion of the segments of gestures #0.5 and #2, and which can be explained by not enough time in between gestures. No false positives occurred, which indicates good distinction by the IMU method of dynamic gestures.

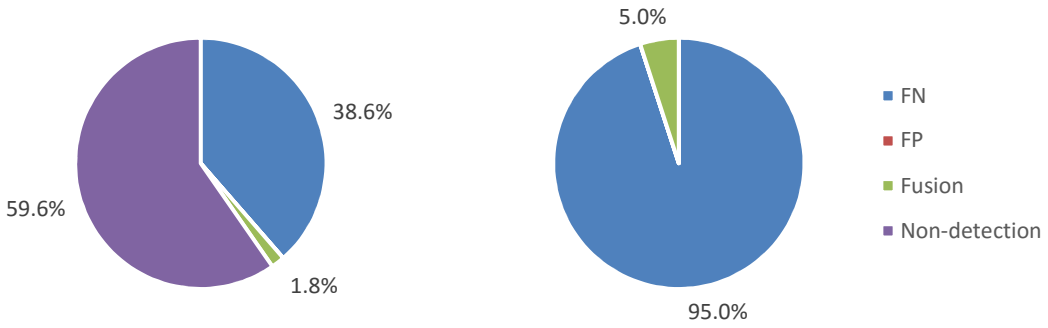

Figure 5.7 - Occurrence of each type of error when considering only IMU sensors for all gestures (left) and for R_{IMU} gestures (right)

5.2.3. Individual EMG

Evaluating EMG sensor alone, the segmentation error obtained for all gestures is 43.75%, even without including setup errors, which increases the error to 57.92%.

The total number of errors that occurred using the EMG sensor data, excluding setup errors, were 233. There is a large number of errors due to non-detection of gestures. In fact, should only R_{EMG} gestures be considered, the number of non-detections lowers from 91 to 20. Yet, the issue is that in 11 different situations, the EMG was unable to detect either hand-only gesture #2.5 or #5.5.

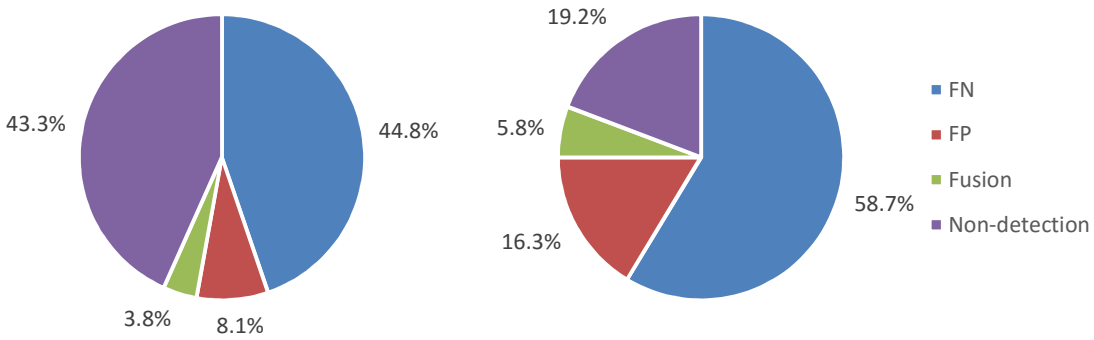

Figure 5.8 - Occurrence of each type of error when considering only EMG sensors for all gestures (left) and for R_{EMG} gestures (right)

5.3. Gesture Comparison

When analysing performance based on gestures, it is possible to observe through table 5.2 that EXP obtained better results for gestures #2.5, #4, #5.5; IMU for gestures #0.5, #2, #7, #8 ; and EMG for gestures #5.

Table 5.2 - Segmentation error (%) based on gesture

Method	#0.5	#2	#2.5	#4	#5	#5.5	#7	#8	FP	Total
EXP	1.67	6.67	10.0	5.0	16.67	10.0	13.33	13.33	3.81	12.92
IMU	0.0	1.67	16.67	13.33	10.0	45	6.67	1.67	0	11.88
EMG	51.67	55.0	20.0	56.67	8.33	15.0	45.0	70.0	4.05	43.75

Similarly, performance of the methods was also evaluated in table 5.3 based on the groups of gestures defined in chapter 4.1.

Table 5.3 - Segmentation error (%) based on group of gestures

Method	R_{IMU}	R_{EMG}	O_{IMU}	O_{EMG}
EXP	9.44	11	7.22	10
IMU	5.56	18.33	1.11	30.83
EMG	47.78	29	58.89	17.5

In the case of the IMU method, it is possible to see that gestures which featured arm motion performed better, with the segmentation error values for O_{IMU} being the lowest amongst all gesture errors, with only 1.67% for gestures #2 and #8 and no error for gesture #0.5, resulting in a S_{error} of 1.11% when considering O_{IMU}, and a S_{error} of 5.56% when considering R_{IMU} gestures, lower than the S_{error} of 11.88% achieved for the IMU method when considering all gestures.

Gesture #7 did perform better using the IMU method but the drawback being that the initial hand movement responsible for the hand pose in #7 was not recorded in this case, with the false negatives due to discerning arm and hand motion not being represented. Depending on the gesture intent, this can mean that the recorded segment may not represent truthfully the gesture if only using the IMU sensor. In the case of the EXP method, the existence of discerning segments of hand and arm motion, as seen in figure 5.1, was responsible for half the existing errors in all the samples of gesture #7.

In the case of EMG, it was not the best solution for O_{EMG} gestures, with an error of 17.5% compared to the 10% obtained with the EXP method. The vast segmentation error on O_{IMU} and R_{IMU} gestures by the EMG method can be explained by the non-detection of the gestures by the EMG.

The only best performing gesture was #5, which also includes arm rotation. According to the error count, both EMG and EXP presented 3 different false negatives on gesture #5 on all samples, with the major cause for better performance of the EMG method being due to fusion errors. Fusion errors for the EXP method included the fusion errors present in the analysis of the individual EMG, as well as the segment fusion due to discerning motion of arm and hand.

The lacklustre performance of the EMG method when considering gestures #4 and #7 versus #5, all which contain hand and arm motion, was concluded to be due to limb position effect, in which significant dynamic motions of the arm have been reported to affect pattern recognition using EMG data. As #5 includes a small motion of the wrist rather than the entire arm, it is less likely to be damaged due to arm inertia. IMU errors for gesture #5, on the other hand, have been noted to occur since, in some cases, the gesture did not only include an arm rotation but as a small lowering of the arm well as the hand motion was performed. This resulted in multiple segments being detected by the IMU sensor which, while undiscernible from hand motion in the EXP method, when analysing IMU sensor they were identified as errors. This is also an error which depends on user, as only participants [C] and [D] registered this error.

In the case of the EXP method, the O_{EMG} and R_{EMG} gestures performed best for this method. When a hand gesture was not detected by the EMG sensor, it could be in some situations compensated in the EXP method by IMU data when it registered an arm movement from the gestures. The EXP method did perform well on O_{IMU} gestures with a S_{error} of 7.22%, even achieving an error of 1.67% on gesture #0.5, with segment fusion being the main factor for the difference in performance compared to the IMU method.

In the case of gesture #4, in the cases where errors occurred using the IMU method due to false negatives mid-motion, in some situations the detection of motion by the EMG sensor, associated with hand contractions during movement inversions, would cover errors from the IMU, and therefore obtain an error-free gesture segment in the EXP method.

The overall better performance of the IMU method when compared to the EXP method can be explained also due to the fact that there are 3 gestures which include only arm motion, that present very good performances within the IMU method, versus 2 gestures which only include hand motion, where the IMU method registers a large number of non-detections.

5.4. Comparison between participants

The amount and types of errors were greatly dependent on the participant performing them. The segmentation error for combined sensors, when setup errors are not considered, varied from a range of 1.25%, obtained by participant [E] which includes a single non-setup error, to 26.25%, by participant [D], as shown in table 5.4.

In regards to tables 5.4, 5.5 and 5.6, which indicate results from the EXP, IMU and EMG method respectively, the colours identify the group in which each gesture is included in, with light blue representing O_{IMU} gestures, light orange O_{EMG} gestures and grey the other gestures which include contributions from both sensor sources. The data regarding FP and total error segmentation does not include setup errors.

Table 5.4 - Segmentation error (%) based on gesture and participant for the EXP method

User	#0.5	#2	#2.5	#4	#5	#5.5	#7	#8	FP	Total
A	10	0	0	0	30	20	0	0	10	16.25
B	0	0	40	0	0	0	20	10	0	8.75
C	0	10	10	20	10	10	20	10	4.29	15
D	0	30	0	10	30	20	20	50	7.14	26.25
E	0	0	0	0	0	0	0	0	1.43	1.25
F	0	0	10	0	30	10	20	10	0	10
Total	1.67	6.67	10.00	5.00	16.67	10.00	13.33	13.33	3.81	12.92

Table 5.5 - Segmentation error (%) based on gesture and participant for the IMU method

User	#0.5	#2	#2.5	#4	#5	#5.5	#7	#8	FP	Total
A	0	0	10	0	0	30	0	0	0	5
B	0	0	50	0	30	70	0	10	0	20
C	0	10	40	20	30	90	0	0	0	23.75
D	0	0	0	60	0	70	40	0	0	21.25
E	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	10	0	0	0	1.25
Total	0.0	1.67	16.67	13.33	10.0	45	6.67	1.67	0	11.88

Table 5.6 - Segmentation error (%) based on gesture and participant for the EMG method

User	#0.5	#2	#2.5	#4	#5	#5.5	#7	#8	FP	Total
A	30	50	0	20	20	20	20	50	10	35
B	80	60	70	80	0	0	40	70	0	50
C	90	70	20	90	0	20	50	50	4.29	52.5
D	30	50	10	10	20	20	90	80	8.57	46.25
E	60	70	10	50	0	20	10	100	1.43	41.25
F	20	30	10	90	10	10	60	70	0	37.5
Total	51.67	55.0	20.0	56.67	8.33	15.0	45.0	70.0	4.05	43.75

Information regarding the time duration of the sequences was also obtained in table 5.7, with the time being counted from the initial frame of the first gesture to the last frame of the eighth gesture. The average time duration of the sequence based on all samples is 16 seconds.

Table 5.7 - Average time duration for each participant

User	A	B	C	D	E	F	Total
Time (s)	15.4	15.41	18.2	19.09	15.59	12.32	16

Depending on the participant, the EXP or the IMU method showed the best performance, with EXP being the better choice for participants [B] and [C] and IMU for others according to tables 5.4 and 5.5. Analysing the errors of all users based on method, it can be seen that the EMG method is underperforming for every user according to table 5.6.

The best performing participant for the EXP method is [E], who registered only a false positive during #3, originated from the EMG sensor, besides 2 setup errors, resulting in a 1.25% segmentation error. The IMU sensor alone shows no errors performed at all, with all gestures detected.

On the other hand, participant [D] obtained the largest non-setup segmentation error, of 26.25% for the EXP method, with an equal number of false negatives, mid motion false positives and fusion errors.

Participant [A], who was responsible for the initial sample through which the function parameters were calibrated, showed the lowest segmentation error for the EMG method, with 35%, and a 5% segmentation error with the IMU method. However, when merging the data, the error was 16.25%. This is in stark contrast with participant [B], who

showed higher errors with the individual sensor methods - 20% for IMU method and 50% for EMG method - but with the EXP method obtained an error of only 8.75%. This is due to participant [B] presenting a large number of non-detection errors, 20 in the IMU method and 30 in the EMG method, which were reduced in the EXP method to only 4.

It is important to notice that the conditions under which the tests were performed and the quantity of training were different for the users. The tests were performed in different days, with multiple external conditions which could have affected the state of the user and partially explain the difference in results' quality.

In the same way, the amount of training done by each user was also different. Participants [E] and [F] had more training than other users, having performed the sequence for (Simão, Neto, and Gibaru 2016) but with a Data Glove instead. Participants [A] and [B] had worn the armband prior to the training session using the MYO armband, and participants [C] and [D] were using the armband for the first time. While not clearly, the segmentation error shows a tendency for users who show more training or more comfort to have better results.

Also associated with the amount of training done, the focus given to each sequence could have been a factor in some users having performed better or worse, by avoiding certain unwanted movements in between gestures which can be the source of false positives, especially undersegmentation in the sensor combination case.

The musculature for all individuals was not similar, and therefore the ability for a consistently reliable contact between the armband's sensors and the skin could have decreased for individuals with thinner arms, harming the EMG signal.

While the position of the armband along the arm and the position of the IMU sensor relative to the arm were mostly similar, certain differences between users may have occurred, resulting in electrode shift in between participants. The MYO armband has a mechanism which invalidates the test in the case of electrode shift or lift within the same gesture sequence. However, when comparing different samples, whether the armband was located closer to the hand or the position of the sensor was different are likely scenarios. These different positions would result in EMG cross talk differing between users or between sequences from the same user, resulting in different muscular situations being evaluated by the sensors.

The variation of force between users may have also been a factor in the performance. When considering hand gestures, the variation in speed at which gestures were performed as well as the strength used could have resulted in variations of force between users. This is exemplified by participant [C] and [D], who performed the sequence at a slower speed according to notes taken from the recording sessions, but also according to the average time length of the sequences of these users shown in table 5.7, which were substantially higher than others', and obtained the highest segmentation errors when evaluating the IMU method according to table 5.5.

Similarly, the gestures performed may be different to some extent depending on user. From notes taken regarding participants' performance, user [F] has been noted to perform noticeable arm motion when performing O_{EMG} gestures, which could explain the good results obtained by the IMU sensor in detecting the user's gestures, with no non-detection errors.

5.5. Application of different filter for EMG data

Given the low performance of the EMG signal, new pre-processing options were studied. One of the solutions found was the application of a bandpass filter prior to the application of the already existing filter, in an attempt to remove motion artefact which causes errors related to limb position. The application of the filter can be seen in figure 5.9, in which 2 stages of filtering can be observed: the resulting data from the application of the bandpass filter and rectification, and the data after the filtering process.

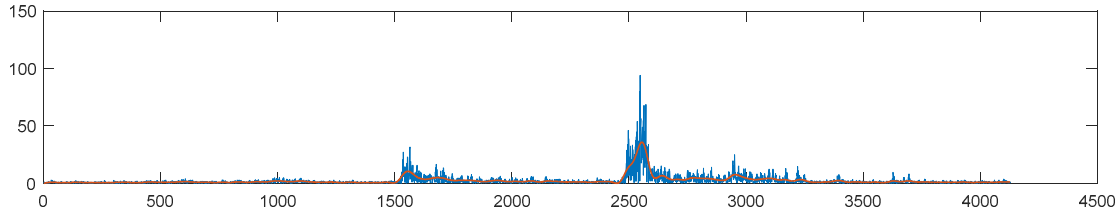

Figure 5.9 - Data treated with bandpass filter, rectification, and with lowpass filter, from EMG sensor 1

In the analysis of this filter, using the initial sequence, it was noted that the new method using both sensors could not detect gesture #5.5, but presented smaller segments than in the former method, as seen in figure 5.10. When altering the sensitivity factor, the

detection of the gesture could not be made without damaging the remaining gestures. The defined parameters in chapter 4.4.1 were used in this approach.

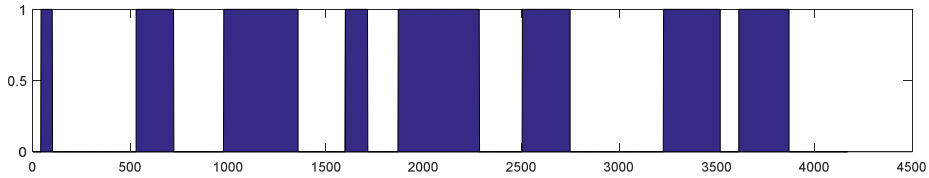

Figure 5.10 - Resulting sequence segmentation from the application of the modified EXP method

The filter was then applied to the sequences from the 6 participants, in order to obtain a comparison between performances of filters. After the process, results were obtained regarding comparison of gestures between methods in table 5.8 and comparison between participants in table 5.9.

As observed in table 5.8, the EXP method improved with the new filter, with a new total segmentation error of 9.17%, compared to the unchanged segmentation error of the IMU sensor of 11.88%. The noticeable change is that all gestures, with the exception of gesture #8, obtained the same of better results with the EXP method compared to the IMU method. This is due to a decline in the number in the number of undersegmentation errors in the sequences from 24 to 4, due to the removal of low-frequency noise from EMG sensor.

However, the visible drawback of this filter is the decline in performance of the O_{EMG} gestures' detection in comparison to the previous filter, with 19 non-detections, as the EMG filter had more difficulty identifying the hand gestures. The EMG filter is still capable to detect a majority of the hand gestures, which can be seen since the EXP method presents better results than the IMU method when considering hand gestures. However, the results are still worse than the ones present for the old filter, with errors of 10% for both gestures as seen in table 5.2.

The EMG shows an increase in the segmentation error from 43.75%, shown in table 5.1, to 55.63% observed in table 5.8, mostly due to increased non-detection of O_{IMU} gestures. Other gestures however also show increased errors, especially gesture #5.5. When considering the EMG method, the non-detection of O_{EMG} gestures increased from 11 to 40, a third of all O_{EMG} gestures.

Table 5.8 - Segmentation error based on gesture with modified filter

	#0.5	#2	#2.5	#4	#5	#5.5	#7	#8	FP	Total
EXP	0	1.67	13.33	10	3.33	21.67	6.67	3.33	1.90	9.17
IMU	0.0	1.67	16.67	13.33	10.0	45	6.67	1.67	0	11.88
EMG	70	75	33.33	61.67	11.67	41.67	51.67	88.33	1.67	55.63

When comparing participants, in table 5.9 it can be seen that the EXP method performed best for users [A], [B], and [C], but overall was an improvement compared to the values in table 5.5. The only exception was with participant [B], as the detection of hand gestures for this user was mainly dependent on the EMG sensor.

Table 5.9 - Segmentation error based on participant

	A	B	C	D	E	F	Total
EXP	2.5	10	13.75	22.5	1.25	5	9.17
IMU	5	20	23.75	21.25	0	1.25	11.88
EMG	47.5	70	68.75	56.25	57.5	33.75	55.63

Overall, the performance of the sensor fusion method has improved with the application of a different pre-processing method. Study into other filtering options could further improve the segmentation, as well as the estimation of new parameters for segmentation, possibly using different methods such as the genetic algorithm suggested in (Simão, Neto, & Gibaru, 2016), since the current method uses values defined for the previous method.

5.6. Comparison to the previous work

When comparing with the results obtained in (Simão, Neto, and Gibaru 2016), it is possible to see that the combination of IMU and EMG sensors is not as effective as using

a data glove. The average oversegmentation error obtained in the other work of 2.70% is inferior to segmentation errors achieved any method in this work. It is therefore concluded that the use of IMU and EMG sensors within a MYO armband, while a more accessible option, does not provide a motion segmentation as accurate as the one obtained with a Cyber Data Glove.

6. CONCLUSION

The sliding window method is a necessary method when attempting to identify segments for gesture recognition. Three sliding window methods were used to analyse data from a number of sequences, one relying on data from the IMU, one on data from the EMG and a third one relying on data from both sensors. Parameters such as the thresholds and window sizes were manually calculated and applied to the segmentation methods.

In a first approach, IMU is the best option of the three methods using the defined sequence, mainly when considering a segmentation error of 1.11% for arm only gestures, with the segmentation error for this method for all gestures being 11.88%. However, the combination of sensors appears to show better results than the individual sensor if hand motion is included in the gesture, depending on the intensity of arm motion.

Segmentation based on EMG, on the other hand, proves to not be a very effective method using the planned methodology, with a vast segmentation error percentage of 43.75% when used alone. When considering gestures which contain hand movement however, it is still an important tool to improve the detection of gestures alongside IMU.

With a second approach aimed at solving the error from limb position using a different filter, the combination of sensors achieved a lower segmentation error of 9,17%, with the drawback of fewer gesture detections by the EMG sensor.

Future work with this solution will be dedicated to integrating the proposed solution using IMU and EMG sensors to an online analysis. This work was performed offline, and could not be verified online, with the ground truth not being recorded. As such, errors like start delay, end delay and extend error could not be evaluated and a full comparison to (Simão, Neto, and Gibaru 2016) cannot be made.

No classification was performed, however features for classification were studied. Classification is an important process to evaluate the quality of the segments obtained to later be correctly identified and used in a HMI scenario.

Additional efforts to this work could be dedicated to further improving EMG motion segmentation by exploring other pre-processing methods. Similarly, an adaptive threshold for gesture segmentation was not used in this work. Quality of motion detection

for both IMU and EMG methods could possibly be improved by studying the application of a genetic algorithm as done in (Simão, Neto, and Gibaru 2016).

BIBLIOGRAPHY

Al-Angari, H. M., Kanitz, G., Tarantino, S., and Cipriani., C. 2016. "Distance and Mutual Information Methods for EMG Feature and Channel Subset Selection for Classification of Hand Movements." *Biomedical Signal Processing and Control* 27: 24–31. http://linkinghub.elsevier.com/retrieve/pii/S1746809416300040.

Alkan, A., and Günay, M. 2012. "Identification of EMG Signals Using Discriminant Analysis and SVM Classifier." *Expert Systems with Applications* 39(1): 44–47. http://dx.doi.org/10.1016/j.eswa.2011.06.043.

Aoki, T., Venture, G., and Kulić, D.. 2013. "Segmentation of Human Body Movement Using Inertial Measurement Unit." *2013 IEEE International Conference on Systems, Man, and Cybernetics*: 1181–86. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6721958.

Arief, Z., Sulistijono, I. A., and Ardiansyah., R. A. 2015. "Comparison of Five Time Series EMG Features Extractions Using Myo Armband." In *2015 International Electronics Symposium (IES)*, IEEE, 11–14. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7380805 (May 11, 2016).

Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L, and Amirat, Y. 2015. "Physical Human Activity Recognition Using Wearable Sensors." *Sensors* 15(12): 31314–38. http://www.mdpi.com/1424-8220/15/12/29858 (March 15, 2016).

Bortz, J. E. 1971. "A New Mathematical Formulation for Strap-down Inertial Navigation." *IEEE Trans. Aerosp. Electron. Syst.* 7(1): 61–66.

Brunner, T., Lauffenburger, J. P., Changey, S., and Basset, M. 2015. "Magnetometer-Augmented IMU Simulator: In-Depth Elaboration." *Sensors (Switzerland)* 15(3): 5293–5310.

Carpi, F., and Rossi, D .D. 2006. "Non Invasive Brain-Machine Interfaces." *ESA Ariadna Study* 05/6402.

Caruso, M. J. 2000. "Applications of Magnetic Sensors for Low Cost Compass Systems." In *Proceedings IEEE Position Location and Navigation Symposium*, San Diego, CA, 177–84.

Dargie, W.. 2009. "Analysis of Time and Frequency Domain Features of Accelerometer Measurements." In *Proceedings - International Conference on Computer Communications and Networks, ICCCN*, IEEE, 1–6. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5235366 (May 17, 2016).

Duda, R. O., Hart, P. E., and Stork., D. G. 1999. *Pattern Classification*. John Wiley & Sons, Inc.

Fida, B., Bernabucci, I., Bibbo, D., Conforto, S., and Schmid, M. 2015. "Pre-Processing

Effect on the Accuracy of Event-Based Activity Segmentation and Classification through Inertial Sensors." *Sensors (Switzerland)* 15(9): 23095–109.

Fougner, A., Chan, A. D. C., Englehart, K., and Stavdahl, Ø. 2011. "A Multi-Modal Approach for Hand Motion Classification Using Surface EMG and Accelerometers." (Grant 192546): 4247–50.

Fourati, H., Manamanni, N., Afilal, L., and Handrich, Y. 2014. "Complementary Observer for Body Segments Motion Capturing by Inertial and Magnetic Sensors." *IEEE/ASME Transactions on Mechatronics* 19(1): 149–57.

Ganesan, Y., Gobee, S., and Durairajah, V. 2015. "Development of an Upper Limb Exoskeleton for Rehabilitation with Feedback from EMG and IMU Sensor." *Procedia Computer Science* 76(Iris): 53–59. http://dx.doi.org/10.1016/j.procs.2015.12.275.

Georgi, M., Amma, C., and Schultz, T. 2015. "Recognizing Hand and Finger Gestures with IMU Based Motion and EMG Based Muscle Activity Sensing." *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing*: 99–108. http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0005276900990108 (April 29, 2016).

Jung, J. Y., Heo, W., Yang, H., and Park, H. 2015. "A Neural Network-Based Gait Phase Classification Method Using Sensors Equipped on Lower Limb Exoskeleton Robots." *Sensors (Switzerland)* 15(11): 27738–59. http://www.mdpi.com/1424-8220/15/11/27738/ (March 14, 2016).

Junker, H., Amft, O., Lukowicz, P., and Tröster, G.. 2008. "Gesture Spotting with Body-Worn Inertial Sensors to Detect User Activities." *Pattern Recognition* 41(6): 2010–24.

Kawasaki, H., Kayukawa, M., Sakaeda, H., and Mouri, T. 2014. "Learning System for Myoelectric Prosthetic Hand Control by Forearm Amputees." *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication* 2014–Octob(October): 899–904.

King, A. D. 1998. "Inertial Navigation - Forty Years of Evolution." *Gec Review* 13(3): 140–49.

Kriesel, D. 2007. *A Brief Introduction to Neural Networks*. http://www.dkriesel.com/en/science/neural_networks.

Laudanski, A., Brouwer, B., and Li, Q.. 2015. "Activity Classification in Persons with Stroke Based on Frequency Features." *Medical Engineering and Physics* 37(2): 180–86. http://linkinghub.elsevier.com/retrieve/pii/S1350453314002963 (July 21, 2016).

Lee, Y., Ho, C., Shih, Y., Chang, S., Róbert, F. J., and Shiang, T. 2015. "Assessment of Walking, Running, and Jumping Movement Features by Using the Inertial Measurement Unit." *Gait & posture* 41(4): 877–81. http://linkinghub.elsevier.com/retrieve/pii/S0966636215000764 (May 3, 2016).

Liu, J., Zhang, D., Sheng, X., and Zhu, X. 2014. "Quantification and Solutions of Arm Movements Effect on sEMG Pattern Recognition." *Biomedical Signal Processing and Control* 13(1): 189–97. http://dx.doi.org/10.1016/j.bspc.2014.05.001.

Neto, P., Pereira, D., Pires, J. N., and Moreira, A. P. 2013. "Real-Time and Continuous

Hand Gesture Spotting: An Approach Based on Artificial Neural Networks." *2013 IEEE International Conference on Robotics and Automation*: 178–83. http://arxiv.org/abs/1309.2084 (March 11, 2016).

Neto, P., Pires, J. N., and Moreira, A. P. 2013. "3-D Position Estimation from Inertial Sensing: Minimizing the Error from the Process of Double Integration of Accelerations." *IECON Proceedings (Industrial Electronics Conference)*: 4026–31.

Novak, D., and Riener, R. 2015. "A Survey of Sensor Fusion Methods in Wearable Robotics." *Robotics and Autonomous Systems* 73: 155–70. http://dx.doi.org/10.1016/j.robot.2014.08.012.

NovAtel. 2014. "IMU Errors and Their Effects. Report APN-064 (Rev A)." : 1–6. http://www.novatel.com/assets/Documents/Bulletins/APN064.pdf (July 13, 2016).

Nyomen, K., Haugen, M. R. and Jensenius, A. R. 2015. "MuMYO — Evaluating and Exploring the MYO Armband for Musical Interaction." *Proceedings of the International Conference on New Interfaces for Musical Expression*: 215–18. https://nime2015.lsu.edu/proceedings/179/0179-paper.pdf (April 7, 2016).

Phinyomark, A., Phukpattaranont, P., and Limsakul, C. 2012. "Feature Reduction and Selection for EMG Signal Classification." *Expert Systems with Applications* 39(8): 7420–31. http://dx.doi.org/10.1016/j.eswa.2012.01.102.

Radmand, A., Scheme, E., and Englehart, K. 2014. "A Characterization of the Effect of Limb Position on EMG Features to Guide the Development of Effective Prosthetic Control Schemes." In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*, IEEE, 662–67. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6943678 (May 24, 2016).

Raez, M. B. I., Hussain, M. S., and Mohd-Yasin, F. 2006. "Techniques of EMG Signal Analysis: Detection, Processing, Classification and Applications." *Biological procedures online* 8(1): 11–35. http://www.ncbi.nlm.nih.gov/pubmed/16799694 (May 3, 2016).

Roetenberg, D., Luinge, H., and Veltink, P. 2003. "Inertial and Magnetic Sensing of Human Movement near Ferromagnetic Materials." In *Proceedings - 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2003*, IEEE Comput. Soc, 268–69. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1240714 (August 31, 2016).

del Rosario, M., Redmond, S., and Lovell, N. 2015. "Tracking the Evolution of Smartphone Sensing for Monitoring Human Movement." *Sensors* 15(8): 18901–33. http://www.mdpi.com/1424-8220/15/8/18901/ (March 29, 2016).

Scheme, E., and Englehart, K. 2011. "Electromyogram Pattern Recognition for Control of Powered Upper-Limb Prostheses: State of the Art and Challenges for Clinical Use." *Journal of Rehabilitation Research and Development* 48(6): 643–60. http://www.rehab.research.va.gov/jour/11/486/pdf/scheme486.pdf.

Simão, M. A., Neto, P., and Gibaru, O. 2016. "Unsupervised Gesture Segmentation by Motion Detection of a Real-Time Data Stream." *IEEE Transactions on Industrial*

Informatics, IEEE: 1–11.

Taborri, J., Rossi, S., Palermo, E., Patanè, F., and Cappa, P. 2014. "A Novel HMM Distributed Classifier for the Detection of Gait Phases by Means of a Wearable Inertial Sensor Network." *Sensors (Switzerland)* 14(9): 16212–34. http://www.mdpi.com/1424-8220/14/9/16212/ (July 18, 2016).

Thalmic Labs. 2016. "MYO Developer." https://developer.thalmic.com/ (August 30, 2016).

Titterton, D. H., and Weston, J. L. 2004. *Strapdown Inertial Navigation Technology*. American Institute of Aeronautics and Astronautics.

Unsal, D., and Demirbas, K. 2012. "Estimation of Deterministic and Stochastic IMU Error Parameters." *Record - IEEE PLANS, Position Location and Navigation Symposium* (2): 862–68.

Verplaetse, C. 1996. "Inertial Prioceptive Devices: Self-Motion Sensing Toys and Tools." *IBM Systems Journal* 35(NOS 3&4): 639–50.

Weili, S.. 2014. "Using MYO Armband in Performances." http://shi-weili.com/using-myo-armband-in-performances/ (August 30, 2016).

Yang, C., Liang, P., Li, Z., Ajoudani, A., Su, C., and Bicchi, A. 2015. "Teaching by Demonstration on Dual-Arm Robot Using Variable Stiffness Transferring." : 1202–8.

Zecca, M., Micera, S., Carrozza, M. C., and Dario, P. 2002 "On the Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal." *Crit Rev Biomed Eng* 30(4–6): 459–85.

Figure Bibliography

Seeed Studio, 2015 "Grove – EMG Detector", last viewed 1 September, 2016 (http://www.seeedstudio.com/wiki/Grove_-_EMG_Detector)

Thalmic Labs, 2015 "MYO Developer FAQ", last viewed 1 September 2016 (https://developer.thalmic.com/forums/topic/255/)

Simão, M. A., Neto, P., and Gibaru, O. 2016. "Unsupervised Gesture Segmentation by Motion Detection of a Real-Time Data Stream.", *IEEE Transactions on Industrial Informatics, IEEE,* 8