

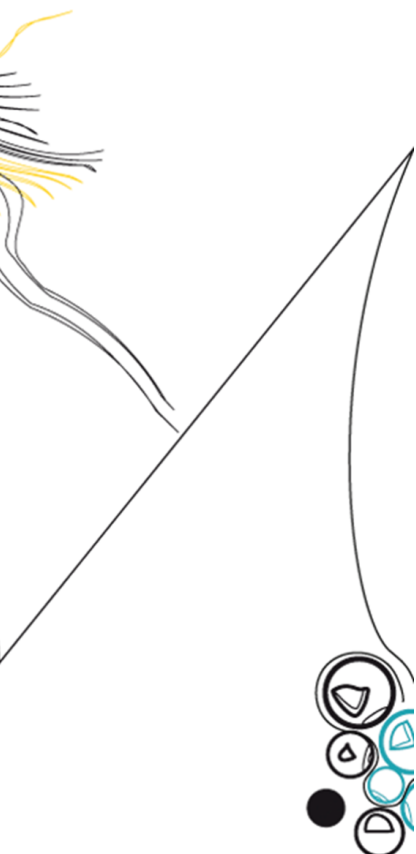


UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science

Comparison of common signal pre-processing techniques for long-term sEMG-based knee angle estimation

Jeroen Casper Ritmeester
Master Graduation Thesis
August 23, 2022



Supervisors:
dr. M. Poel
R. Schulte, MSc
dr. Ing. G. Englebienne

MSc Interaction Technology
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente

CONTENTS

Acknowledgements	iii
Abstract	v
1 INTRODUCTION	1
1.1 Motivation	1
1.2 Problem statement	2
1.3 Scope	2
1.4 Research question	3
2 BACKGROUND	4
2.1 Anatomy of the knee	4
2.2 sEMG signals and disturbances	5
2.2.1 Obtaining measurements	5
2.2.2 Sources of noise	5
2.2.3 Non-stationarity	7
2.3 Advanced filtering techniques	7
2.3.1 Wavelet Transform	7
2.3.2 Empirical Mode Decomposition	9
2.3.3 Independent Component Analysis	10
2.4 Machine learning approaches	12
2.4.1 Autoencoders	12
2.4.2 Recurrent Neural Networks	13
2.4.3 Long short-term memory	14
2.4.4 Model performance	14
3 RELATED WORK AND STATE OF THE ART	15
4 METHODS AND TECHNIQUES	17
4.1 Pre-processing	17
4.2 Segmentation and decomposition techniques	17
4.3 Envelope smoothing techniques	18
4.4 Model architecture	19
4.5 Model training	20
5 RESULTS AND DISCUSSION	22
5.1 Preliminary grid search	22
5.1.1 LSTM-AE and ICA	24
5.2 Same-day performance	24
5.3 Multi-day performance	25
6 CONCLUSION	30
BIBLIOGRAPHY	32

ACKNOWLEDGEMENTS

This thesis was started in February 2022 and concluded in August 2022 for the MSc programme Interaction Technology, led by my supervisors Dr. M. Poel and R. Schulte, MSc from Roessingh Research & Development, as well as critical observer Dr. Ing. G. Englebienne. Great thanks goes out to each of them for their constant support over the last six months.

A heartfelt thanks goes out to family and friends who have helped me through the coronavirus pandemic, during which I started this master's programme, as well as everything that came in its wake. Although difficult at times, their support and reaching out of others allowed me to get far enough to conclude my time at the University of Twente with this thesis. In the words of a good friend, writing a graduation thesis is done individually, but not alone.

Jeroen Ritmeester
August 23rd, 2022, Enschede

This page has been left intentionally blank.

ABSTRACT

The rapid advancement of technology over the past decades have made it increasingly possible to help people through the use of advanced devices. One example of this are the microprocessor controlled knees, that allow people that have undergone a transfemoral amputation to regain the use of their leg. This prosthesis is driven by surface EMG (sEMG) signals that originate from the remainder of the limb after amputation. Unfortunately, sEMG signals are easily distorted by a number of sources, most of which are located inside the body. Numerous prior works have focused on improving the accuracy of these prostheses by applying various filtering pre-processing steps, but no consensus has been reached on what combination of steps performs best. Furthermore, the performance in terms of accuracy is also dependent on the signal-to-noise ratio in the signals that are used to train the machine learning models that the prosthesis uses to control the knee. As it stands now, the performance deteriorates so quickly that the knee requires constant retraining, with the data acquisition process that comes along with, making it impractical for widespread adoption. This thesis focuses on combining signal pre-processing techniques that are commonly found in scientific literature, and aims to find the best combination thereof with regards to multi-day model performance. The wavelet transform, empirical mode decomposition, and independent component analysis are compared, as well as the effect of differentiation of the signal, and two alternative ways to modify the signal by calculating the RMS in sub-windows and smoothing the envelope of the signals. Additionally, an LSTM-based autoencoder is proposed with the aim of reducing the dimensionality of the input signals, whilst simultaneously reducing the noise present in the signals. Through a process of gradual elimination, a number of best practices in terms of sEMG signal pre-processing were isolated, and subject for further research are proposed.

INTRODUCTION

Most people are fortunate enough to be able to use both their legs, which is something many might take for granted as they go about their days, not paying much attention to their own mobility. Yet there are many reasons why anyone could lose a part of one or both of their legs. Vascular disease, cancerous tumours, infection, or severe trauma to the leg among many other things can create the medical necessity to have the leg amputated. Most leg amputations are performed below-knee, called a transtibial (Lat. *through the shin bone*) amputation, meaning the patient will keep their knees intact. In a smaller number of cases, this is not possible, requiring an above-knee amputation, also called transfemoral (Lat: *through the thigh bone*) amputation, leaving only a part of the patients' upper leg intact. For transfemoral amputees, rehabilitation is even more difficult than for transtibial amputees, as the former have to regain control of not only their lower leg, ankle, and foot, but also an artificial knee.

To aid in this rehabilitation process, a number of types of prostheses has been developed to allow the user to adapt their new limb to the day-to-day requirements, flexibility and cost as required. These types can range from making use of manual or weight-activated locking systems, to constant or variable friction joints, and even pneumatic or hydraulic fluid control systems [1]. The latest development are the micro-processor controlled knees (MPK), which use on-board sensors to detect movement and adjust the knee joint accordingly. These computerised knee prostheses have already been around since the late 1990s [2]. They may use different sensors of various types, allowing the user to control the angle of the knee with physiological signals. In combination with machine learning techniques, it is possible to conduct sEMG measurements on the remaining part of the leg, and process these signals to estimate the knee joint angle, angular velocity, or torque.

An example of MPKs is the MyLeg [3], an osseointegrated (*integrated with bone*) transfemoral prosthesis developed by a collaboration of universities and research organisation such as Roessingh Research and Development, who are collectively trying to improve the reliability and robustness of the MyLeg prosthesis. The prosthesis is controlled by an array of surface electromyography (sEMG) sensors, that measure the electrical muscle activity under the skin. The electrodes are marked in red in Figure 1. Unfortunately, due to the noisy nature of sEMG signals and their measurements, the accuracy of the knee joint in the prosthesis wanes in a matter of days [4]. This thesis will hopefully aid the MyLeg research project investigating which combination of commonly pre-processing techniques has the best effect on the multi-day stability of the accuracy of the prostheses. The goal is to reduce the frequency with which the prosthesis has to be retrained in order to maintain acceptable levels of error in knee's final angle as predicted by the underlying machine learning model.

1.1 MOTIVATION

Roessingh Research and Development is a research group that focuses on people and their needs, working in the field of rehabilitation technology and e-health [5]. One of the research topics involves computerised knees and is in need of a way to preprocess the raw sEMG data for the MyLeg research project [3], and suspects that there is much to be gained when a robust data pre-processing pipeline and a reliable dimensionality reduction technique are applied to the sEMG measurements, hopefully decreasing the complexity of the high dimensional signal and improving the robustness and stability of the machine learning models that drive the movement of the knee joint.

The goal of the MyLeg project is to "let transfemoral amputees better accept the prosthesis, improving their quality of life, and allowing them to reach a higher degree of self-reliance [...]" [5]. Additionally, the project aims to reduce the support needed by formal and informal caregivers, reducing the burden on these groups and society as a whole. This is an ambitious goal, but an important one, as a lot of people can be helped with the improvement of robustness in computerised knees, now and in the future.

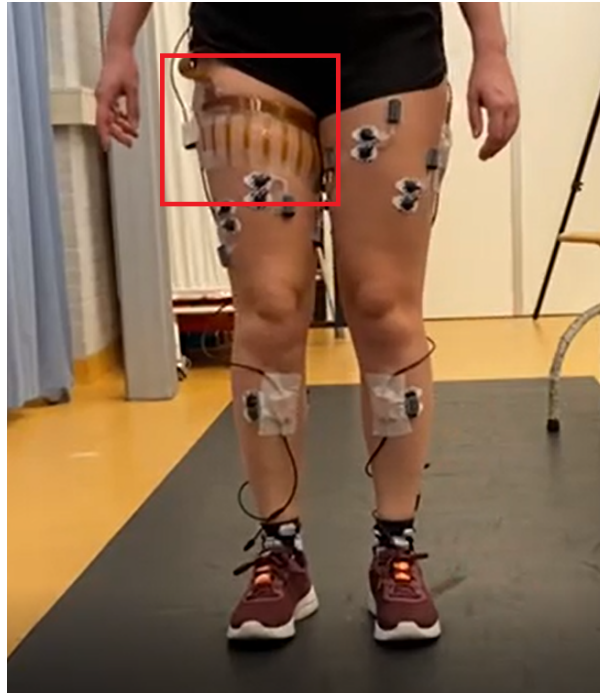


Figure 1: Photo of a healthy user wearing a 16x4 array of sEMG electrodes on the thigh, marked in red.

1.2 PROBLEM STATEMENT

Raw sEMG signals are filled with noise and contamination. Sub-optimal contact of the electrodes with the skin's surface, movement of the electrodes, power line interference, cross-talk between muscles and multiple electrodes, and numerous other reasons can cause the intended signal - that of the muscle groups underneath the skin - to be obfuscated. As such, if this data is used in a machine learning model, tuning this model will likely require a great amount of effort with no guarantee of successful training. Even if the training is completed with an acceptable accuracy, its performance may be short-term, degrading quickly over time [4]. The details of this performance decay is further explored in [chapter 3](#).

The signal acquisition currently consists of long measurements at a high sample rate of 1000 Hz of 64 electrodes, during which a user performs simple activities, such as walking, sitting, and standing. This yields 64 long signals for each trial. After removing as many unwanted contaminations as possible, the data can be used in a number of pre-processing and/or dimensionality reduction techniques to compress the signals into more information-rich signals. Then, from these signals, a regression model can be trained in order to estimate the angle of the knee joint, so that when used, the sEMG signals can be used to control the knee. Currently, it is difficult to state a definitive best combination of pre-processing techniques with regards to long-term stability of the model accuracy. As such, in the scientific literature on this topic, a wide range of techniques is used without consensus on a best practice.

1.3 SCOPE

Within the MyLeg research project there are many facets already in place in order to make the MPK work. This spans from from the physiological measurements to the final prediction and control of the joint. This, then, creates a large scope in which the topic of this thesis could be placed. As a consequence, it is important to delineate the boundaries of this research. The goal is not to research ways to generally improve the accuracy of the knee joint, but specifically to investigate the effects of dimensionality reduction and commonly used signal filtering techniques on the predictive instability over multiple days which currently poses a problem in the use of computerised knees. Furthermore, throughout the research, the input data used will have been prerecorded, which means that the data acquisition is not considered. Raw sEMG signals are provided by Roessingh Research & Development at the start of the development

of the models in this research. Since the focus lies on sEMG here, no other sensor data such as inertial measurement unit (IMU) will be taken into account. Finally, although of great significance in real-life use of filtering techniques, the speed and computational requirements required for the pre-processing techniques in this research are not taken into account when comparing the performance and results of each approach.

1.4 RESEARCH QUESTION

Combining the problem at hand with the intended outcome yields the following main research question:

What combination of commonly used signal filtering and dimensionality reduction techniques can best increase the multi-day performance stability of an sEMG-based computerised knee joint?

In order to work towards answering the main research question, two sub-questions can be posed. Although the second question is similar to the main research question, it is important to explicitly state the first question, as the result are not necessarily be the same.

1. What combination of surface electromyography filtering techniques found commonly in literature yield the highest accuracy when the training and evaluation occurs on the same day?
2. What impact do these selected techniques have on the multi-day accuracy of predicted angle of the knee joint compared to the same-day accuracy?

BACKGROUND

This chapter will lay a theoretical foundation on the workings of muscles and sEMG measurements. An overview of the most effective ways of filtering and reducing the sEMG signals is created by comparing various techniques found in scientific literature.

2.1 ANATOMY OF THE KNEE

The knee is the largest and most complex joint in the human body [6], creating a hinge joint between the femur (thigh bone) with the tibia (shin bone) and the fibula (calf bone). Technically, the tibia and fibula have a joint that is separate from the the main joint between the femur and tibia, but this second joint has very limited movement and is generally not considered part of the main knee joint [7]. Together with the patella (knee cap), these four bones make up the knee joint. For the knee to work maintain its effectivity and longevity, a lot of additional tissues assist the rotational movement. Articular cartilage covers the areas of the femur, tibia, and patella that move along each other, Synovial fluid in between the bones helps to lubricate the movement, two C-shaped menisci spread the force over a larger area to reduce damage to the cartilage, and the tendons and ligaments limit the knee's range of motion.

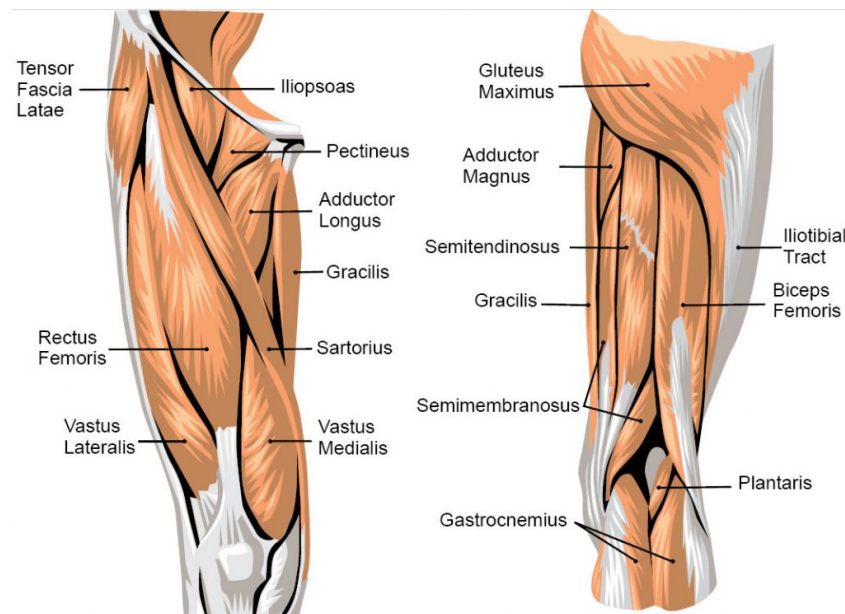


Figure 2: The muscles of the upper leg. The left-hand image is the anterior view showing the quadriceps (*rectus femoris*, *vastus lateralis*, *vastus medialis*, *vastus intermedius*), and the right-hand image is the posterior view, showing the hamstrings (*semimembranosus*, *semitendinosus*, *biceps femoris*). Image source unknown.

There are many muscles involved in moving the knee joint. The two main muscle groups are called the quadriceps and the hamstrings [6]. The *quadriceps femoris*, or simply quadriceps, are located in the front of the upper leg. Flexion of this muscle group causes the knee to extend and the leg to straighten. The quadriceps consist of four muscle heads: the *rectus femoris* (RF) muscle occupying the middle of the thigh, the *vastus lateralis* (VL) muscle on the outer side of thigh, the *vastus medialis* (VM) muscle on the inner side of the thigh, and the *vastus intermedius* (VI) muscle [8] between the VL and VM muscle, deeper inside the thigh. The hamstrings are located at the back of the thigh and consists of three muscle heads: the *semimembranosus* (SM), *semitendinosus* (ST), and the *biceps femoris* (BF). Flexion of the hamstrings bend the knee joint. All the mentioned muscles are shown in Figure 2, except the VI, as it is located underneath the RF.

2.2 SEMG SIGNALS AND DISTURBANCES

As the amount of unwanted noise is usually too large for the sEMG measurements to be used directly, there has been a lot of research in the area of sEMG filtering to allow the obtained signals to be reliable and useful in various applications, from analysis to prediction. On top of the many sources of noise, EMG signals are also highly complex, non-linear, and non-stationary, which means that the frequencies that make up the signal are not constant over time [9]. In order to utilise the obtained signals, a high quality pre-processing step is generally considered especially important.

2.2.1 Obtaining measurements

EMG signals can be obtained in two ways, using either surface electrodes or intramuscular electrodes. The former uses electrodes on the skin with a conductive gel in order to minimise the electrical impedance between the skin and the electrode, whereas the latter uses needle-like electrodes, called monopolar needle electrodes, that are placed in between the muscles. This makes intramuscular EMG acquisition much more invasive than surface EMG measurements. The most important reason for using intramuscular measurements is that it allows you to reach deeper muscles with much more precision, as there is no skin and fat to distort the signal and the electrode is much closer to the source of the electrical activity. For the use in prostheses, these benefits would be very convenient to have, but the insertion of needles is simply not feasible for various reasons, such as causing pain, the danger of having to insert and remove the needles yourself, and the need to repeat this procedure many times during the use of prosthesis, to name a few. This leaves surface EMG (sEMG) signals. These have been used for prosthetic control since at least the 1980's [10]. Using conductive gel for better electrical contact, electrodes are placed along the muscles to pick up their electrical activity when flexed.

2.2.2 Sources of noise

In order to understand how sEMG signals can best be filtered and used for the training of models, it is crucial to understand the things that contaminate the desired signals. sEMG signals are very noisy signals, and by consequence it is not possible to directly find the muscle activations through raw sEMG measurements. In Figure 3, the most prominent of these sources of noise are shown schematically.

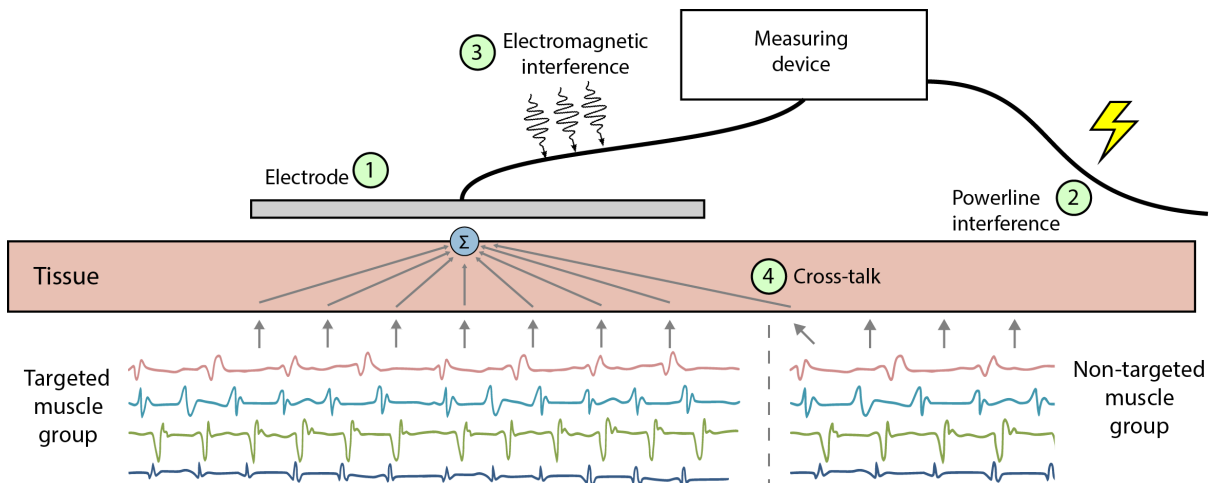


Figure 3: Motor unit action potentials have to move through the body tissue, where they add up before being picked up by the electrode. There are various sources of noise that come into play. (1) Electrode noise, like movement artefacts, and possible imperfect contact with the skin. (2) Cross-talk from untargeted muscle groups. (3) Electromagnetic interference from other devices nearby at PLI frequency, and other noise. (4) Power line interference.

The electrode itself can be a source of several types of disturbances. Improper contact with the skin can cause differences in impedance and change the measured voltage (1). Movement of the electrode on

the skin, or of the subcutaneous tissue when the underlying muscles move, can cause the same kind of artefacts. The frequency range of those artefacts lie in the low frequency range of 1-10Hz with a voltage amplitude comparable to that of the targeted signal [11]. In that range there is also the disturbance of the firing rate of the motor units that carry the electrical signal to and throughout the muscles, namely 0-20 Hz. This means that filtering around the 20 Hz mark is a good idea. This was confirmed by [12], which state that for most purposes, 20 Hz is a good value to filter at, balancing information loss and removing noise.

On top of that, the electrode does not only pick up electrical signals directly underneath it. This effect, called cross-talk (2), occurs when an EMG signal from a non-monitored muscle group is detected, causing an incorrect interpretation of the signal information [13]. There are ways to reduce cross-talk by use of different types of electrodes [14], but post-hoc, there appears to be little that can be done. [15] states that the cross-correlation of the captured signals is neither a quantitative nor a qualitative measure of the level of cross-talk [11].

Perhaps the simplest type of noise to recognise and remove is power line interference (3) and electromagnetic radiation interference (4). In most parts of the world, the power grid runs at either 50 or 60 Hz. Devices connected to the grid will usually give off a faint magnetic field at that frequency, and since we are surrounded by such devices, this field will - in Europe - cause 50 Hz interference on the metal leads and internal parts of the EMG device. Luckily, this is easy to remove using a notch filter. A commonly used type of filter is a Butterworth notch filter, also called a band-pass filter (BPF). With this filter there is an accompanying quality factor Q , which is unfortunately usually not reported in other research. A value that was reported by [16] was $Q = 3$. This choice for this value is not supported, though, and appears rather arbitrarily chosen. A low value of Q means the notch filter is not very specific and attenuates surrounding frequencies more, whereas a higher value means the filter is more precise. This is visualised in Figure 4. It is readily apparent that the light green part of the signal, with $Q = 20$ filter out much less of the frequencies between 40 Hz and 60 Hz, thus distorting the underlying data much less than the dark green/blue part of the signal. Thus it appears that a value higher than 3 is more appropriate. Most studies disregard the filtering of its harmonics, i.e. frequencies of integer multiples of the power line frequency (100 Hz, 150 Hz, 200 Hz, et cetera).

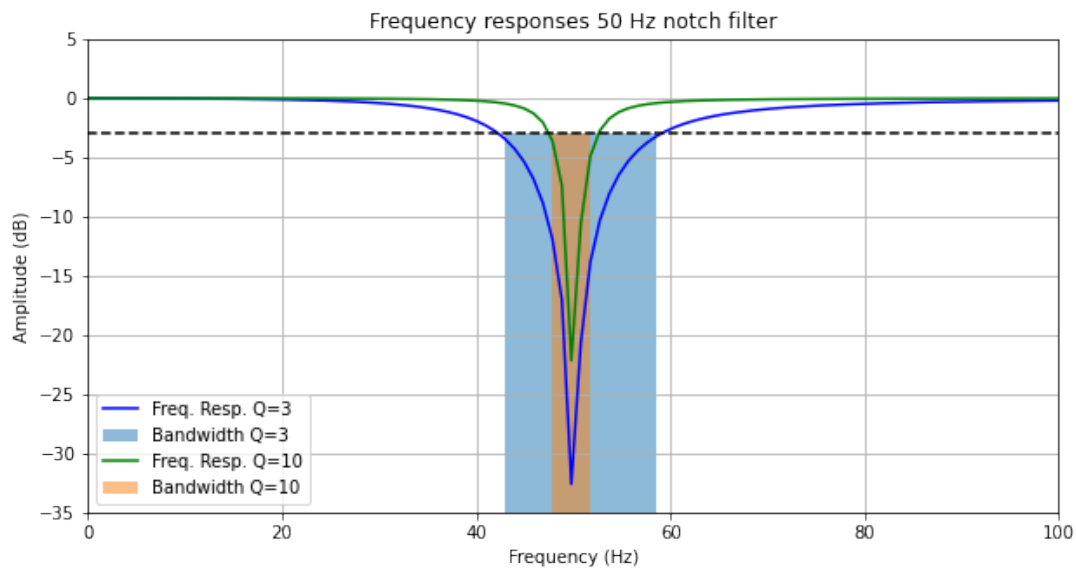


Figure 4: Frequency response of two 50 Hz notch filters with different quality factors Q . The black dashed line indicates the -3dB point, the intersection with which indicates the start and end of the respective bandwidth, highlighted by the shaded areas.

2.2.3 Non-stationarity

A large part of the reason why sEMG models are unstable over time is because of non-stationarity. Formally, a signal is considered stationary if it meets three criteria: the signal should have a constant mean over time, as well as a constant variance over time. Together, these criteria imply the third criterion, namely that the distribution of any two equally sized windows taken from the signal must be identically distributed.

2.3 ADVANCED FILTERING TECHNIQUES

After the filtering steps are done and the signal is free from easy to remove noise, a number of methods can be used to further clean up the signal. Two very popular approaches found in the literature are the wavelet transform (WT) and empirical mode decomposition (EMD). Both approaches attempt to decompose the signal into sub-signals, after which according to some criterion some sub-signals can be kept and others discarded. The remaining sub-signals are used to reconstruct the signal, having removed unwanted temporal features in the process.

2.3.1 Wavelet Transform

The wavelet transform can be thought of as an extension of the Fourier transform. In order to best understand the wavelet transform, a comparison can be made with the Fourier transform. The goal of the Fourier transform is to decompose a time signal into oscillating basis functions, namely sines and cosines. This allows you to find the amplitudes of the frequencies of oscillating signals that are present in the signal. If there is only one frequency, the plot of the transform would give a single peak at that one frequency value. In this case, this frequency persists over a long time. It is also possible for a signal to be localised in time, i.e. the signal is not infinite. In this case, the Fourier transform still finds the frequency, but with less precision in the frequency domain. Even more localisation in time gives less frequency resolution still. This is precisely why the Fourier transform is poorly suited for analysing sEMG measurements: the abrupt changes in signal cannot be described by a sum of sinusoidal waves [17].

A significant step forward was the use of the Short Time Fourier transform (STFT), which divides the signal into time windows, and take the Fourier transform of each window, thus giving you a group of frequencies for every interval. There is a trade-off to be made there. The longer the windows, the more resolution you get in frequency space, but less in temporal space. Using shorter windows will give you a better idea of which frequencies occur when, but the exact value of this frequency is less well known.

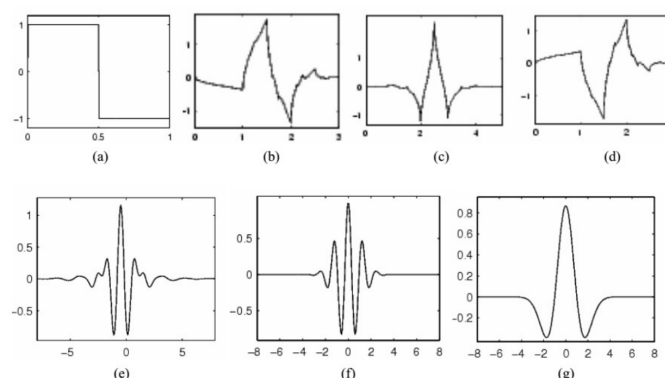


Figure 5: Various types of wavelets. (a) Haar, (b) Daubechies, Coiflet (d) Symlet (e) Meyer (f) Morlet (g) Mexican Hat. Source: [18].

This is then improved upon by the wavelet decomposition. It does not use sines and cosines as its basis functions, but wavelets, examples of which are shown in Figure 5. They are specific signals of finite length

- that is, a specific value within a given range, and zero everywhere else - that can be scaled in time, and shifted along the time axis, as shown in Figure 6. In doing so, the wavelet transform can calculate the similarity of the signal at given time steps at varying resolutions. It can pass over the same signal multiple times with the same wavelet, but scaled, so that you get multiple resolutions. The rationale behind this is that low frequency parts of the signal tend to sustain longer, allowing to analyse low frequencies with lower temporal resolution in favour of frequency accuracy, and high frequency signals tend to be shorter lived, requiring a high temporal resolution, at the cost of knowing the precise values of these frequencies. This is visualised in Figure 7. In a time series signal, you have no resolution in frequency, but a lot in time. In the Fourier transform of the signal, you have no resolution in time, but a lot in frequency. The STFT gives a general compromise between time and frequency, and WT gives a more structured compromise.

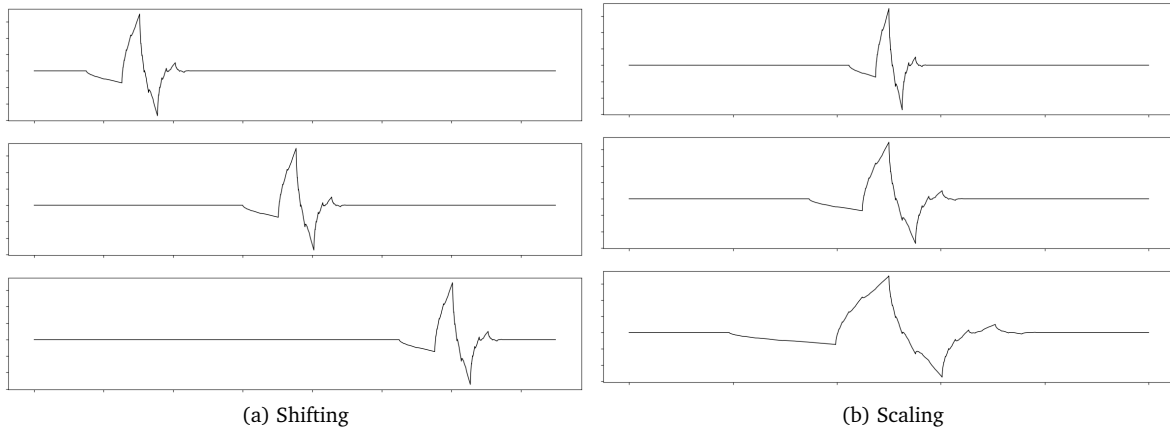


Figure 6: A Daubechies 2 (db2) wavelet, shifted over time (left) and scaled along time (right).

Where the output of a Fourier transform gives you a set of coefficients corresponding to different frequencies, the output of a wavelet transform yields a two-dimensional set of coefficients corresponding to different scales of the wavelet at different positions along the signal. From these coefficients, a three-dimensional plot can be created in order to visualise the amplitudes at each scale and at each point in time.

As there are a large number of wavelet families, each with different levels for more or less detail, it is important that an appropriate wavelet is chosen. The shape of each wavelets can drastically alter its efficacy for a given applications. In this case, working with highly variable and stochastic data, a wavelet should be chosen that reflects these characteristics. Phinyomark et al. [20] compared 53 wavelets from the Daubechies (db), Symlet (sym), Coiflet (coif), BiorSpline (bior), ReverseBior (rbio) and discrete Meyer (dmey) wavelet families. After rigorous testing, the authors recommend to use a db2, db7, sym2, bior5.5, or rbio2.2 wavelet.

In order to use the transform as a denoising technique, the resulting coefficients have to be altered, so that the noisy components are not reconstructed. This threshold can be selected based on a number of criteria. Donoho and Johnstone [21] proposed a thresholding technique, called the universal fixed form threshold, in 1994. If any given coefficient has a smaller value than that threshold value, it should be reduced to 0 before the signal is reconstructed. The universal threshold value is calculated as follows:

$$THR_{uni} = \sigma \sqrt{2 \log(N)}$$

Here, N is the number of samples in the signal, and σ is the standard deviation of the noise. σ can be approximated by $\sigma = \text{median}(|cD_j|)/0.6745$, where cD_j is the detail wavelet coefficients at scale level j and 0.6475 is a numerical normalisation factor. Phinyomark et al. [20] compared adaptations of this universal threshold. They found that the Log Scale Modified Universal Rule (LSMU) yielded the lowest MSE compared to other threshold rules. LSMU is described by the following equation:

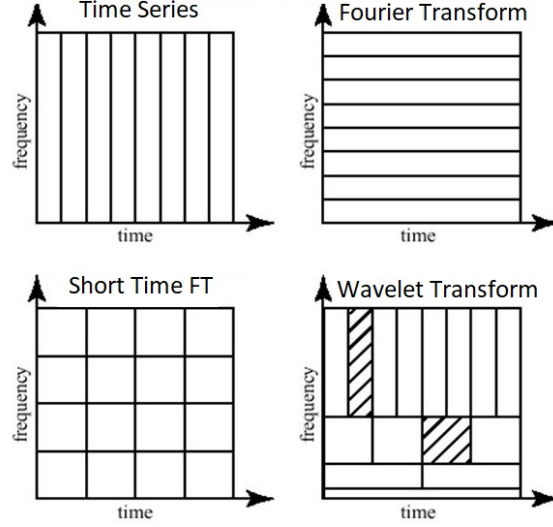


Figure 7: Comparison of the resolution in frequency and time of various techniques. Source: [19].

$$THR_{LSMU,j} = \frac{THR_{uni}}{\log(j+1)} = \frac{\sigma\sqrt{2\log(N)}}{\log(j+1)}$$

The same authors also investigated adaptations of threshold functions to see what works best in the context of sEMG signals. It was found that a rule called the adaptive denoising function (ADP) yields the lowest mean square error, and thus maintaining the signals characteristics the best while removing the most noise. ADP updates the found coefficients as follows:

$$cD_j = cD_j - THR_{LSMU,j} + \frac{2THR_{LSMU,j}}{1 + e^{2.1cD_j/THR_{LSMU,j}}}$$

Here, THR_j is the threshold value for that scale level as given by the selected threshold rule. Although quite a large expression, the main benefit is that, as opposed to other threshold rules, the right-hand side of the expression is a continuous function, rather than a hard threshold. As such, there is no sharp drop-off when the a given coefficient falls just underneath the threshold value THR_j .

In summary, the update rule that is found to work best for sEMG data, yielding the lowest mean square error values, is described as follows:

$$cD_j = cD_j - THR_{LSMU,j} + \frac{2THR_{LSMU,j}}{1 + e^{2.1cD_j/THR_{LSMU,j}}}$$

After the coefficients have been updated, they are multiplied with their respective wavelets to reconstruct a filtered version of the original signal.

2.3.2 Empirical Mode Decomposition

The EMD follows a different approach, visualised in Figure 8. The decomposition process starts by identifying the local minima and maxima, collectively called the extrema, of the signal, shown in (a). The minima and maxima are each interpolated by a cubic spline to create a continuous signal for each (b). The average of these two signals is computed (c). Finally, this average of the envelope is subtracted from the original signal (d). The remaining signal is called a residual. The goal of EMD is to identify intrinsic mode factors, or IMFs.

The residual is considered an IMF if it meets two criteria. Firstly, the number of extrema can only differ from the amount of zero-crossings by at most one. This implies that there can be no two consecutive ex-

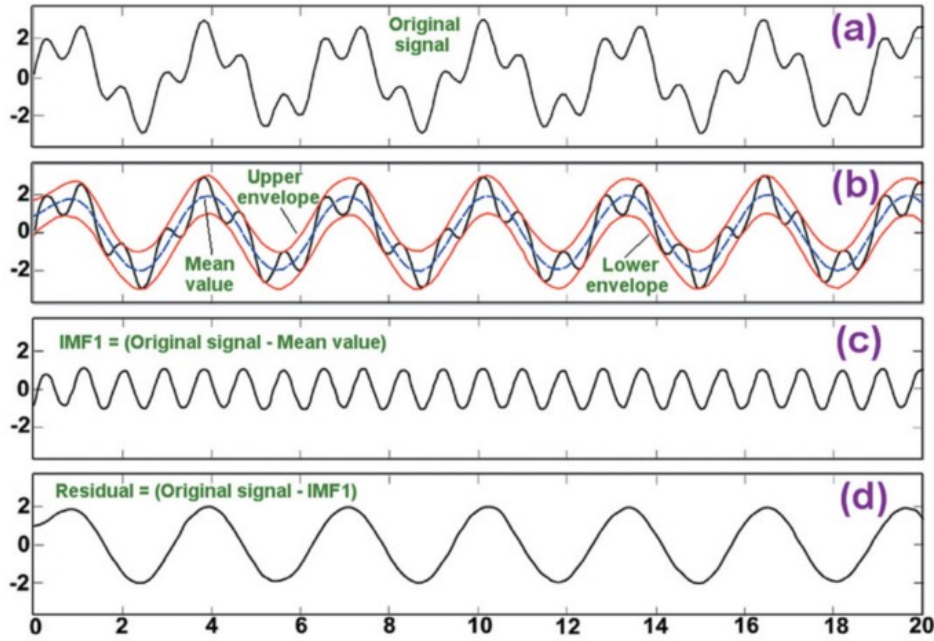


Figure 8: (a) Original signal with multiple frequencies. (b) Creating an envelope by interpolation of the extrema. (c) Subtracting the mean of the envelopes yields an IMF. (d) If the remaining signal meets the criteria, all IMFs and the residual have been found. Source: [22].

rema before crossing the x-axis again. Secondly, the signal must have a local mean of zero everywhere. This implies that the mean of the two envelopes created by the extrema must be (negligibly close to) zero. The residual meets these criteria, as shown in Figure 8 (d) (one extremum per zero-crossing) and (e) (the mean of the envelopes is zero), so the residual is an IMF. This process repeats until no IMF can be extracted. The resulting set of IMFs will have decomposed the original signal into a finite number components, which in turn form a nearly orthogonal basis for the original signal. This means that a linear combination of these signals can reconstruct the original signal, and removal of unwanted IMFs can filter the signal before reconstruction.

Similar to the wavelet transform, the IMFs can be altered based on a threshold value. However, they are not removed or scaled as a whole. Rather, depending on the type of thresholding, the smaller IMFs are squashed towards zero only at the parts of the IMF that are small enough. This type of thresholding is referred to as Smoothly Clipped Absolute Deviation, or SCAD. This has recently become the more popular threshold rule, over the other two often-used threshold, soft and hard thresholding. The scaling factors are visualised in Figure 9. Because SCAD is has a smoother transition between the different piece-wise functions, a more continuous scaling transition occurs when applying the threshold to a given IMF. It's non-linear region of influence is determined by a coefficient α , which, according to Kopsinis and McLaughlin [23] is optimal at 3.7. The width of the non-linear part in all three methods is also determined by the threshold value T , which is calculated in the same way as the wavelet transform's universal threshold value [24]. This value determine how small parts of an IMF must be to be subjected to change. As can be seen in the figure, the SCAD thresholding rule has increasing steepness when moving away from 0, and becomes linear afterwards. These piece-wise functions allow a signal to be "stretched" towards 0, while allowing sufficiently large values to remain unchanged. As such, noise signals, which tend to be relatively small in amplitude, can be squashed while preserving the signal as much as possible.

2.3.3 Independent Component Analysis

Another commonly used technique to separate the signals into its components in independent component analysis (ICA), which is a technique that separates multivariate signals into additive components, by assuming that the components are potentially non-Gaussian signals and that they are statistically independent from each other. ICA is known as a blind source separation technique. Its goal is to separate

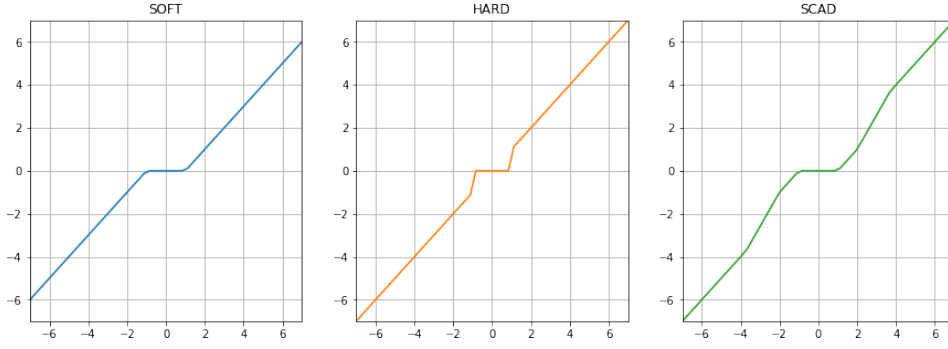


Figure 9: Three common threshold methods for EMD: Soft, hard, and SCAD ($\alpha = 3.7$) with $T = 1$. The x-axis shows the input coefficient cd_j , and the y-axis shows the coefficient value after applying the respective thresholding technique.

the original sources into a linear mixture of the original sources, see Figure 10. A common illustration for this is a person trying to filter out a single person’s voice in a crowded room, which requires the independent signal (the other person’s voice) to be isolated from the other sources (other people talking or noise). ICA is generally applied post-hoc, allowing larger temporal structures to be revealed. This may pose a problem when using it on windowed or live data.

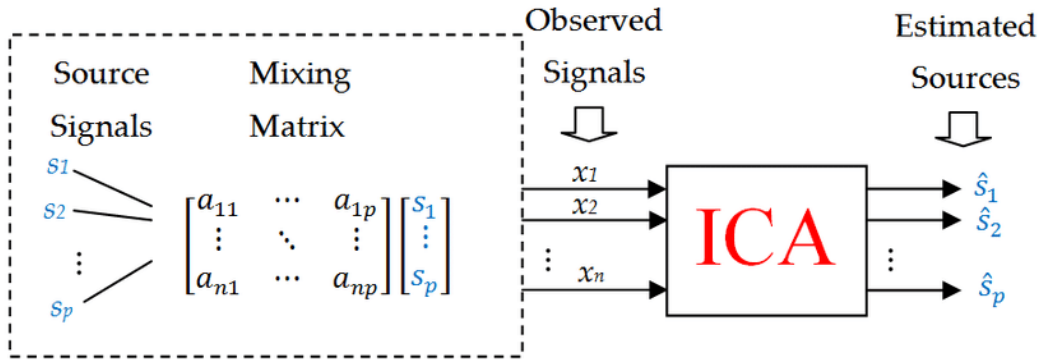


Figure 10: Schematic overview of the purpose of ICA. Source: [25]

To use ICA, multiple recordings from different spatial locations have to be available, which is the case for multi-channel sEMG measurements. One parameter that has to be picked manually, is the number of source signals N that ICA will try to find. Given that there may be multiple sources of noise and multiple muscles - targeted or untargeted - that are producing cross-talk artefacts, an estimate of N is required. Generally, ICA implementations assume that $N = M$, where M is the number of recordings. This allows ICA to simplify the estimation of the square mixing and unmixing matrix, since square matrices are invertible. Naik and Kumar [26] developed an estimate for undercomplete mixtures, that is, when the number of components N is less than the number of source signals M , as is likely the case given the 64 sEMG channels available for this research. The proposed measure to identify the appropriate number of independent component is calculated in two steps. First, ICA is performed on all M measurements, using N expected ICs. This generates a global matrix $\mathbf{G} = \mathbf{AW}$, which are the mixing and unmixing matrices, respectively. The determinant of \mathbf{G} is an indication of the effectiveness of the unmixing, and the Frobenius norm of \mathbf{G} can be used to obtain a new estimate for the number of ICs, K . This value is substituted in place of N . This can be repeated until K doesn’t change, at which point a good estimate for the number of sources should have been found.

2.4 MACHINE LEARNING APPROACHES

Neural networks are computational models that can take numerical data as an input and reshape this input to match a desired target value. It does this by adapting the weights that connect the nodes of network. The nodes are arranged in layers, organised from start (the input layer) to end (the output layer). The data is said to flow through the network one layer at a time, until it reaches the end. Here the output is compared to the target output. In this case, the input data is a set of sEMG signals, and the target output is the correct knee angle at the same moment the sEMG signals were measured. The similarity between the output value and target value can be measured according to any criterion that suits the application. This criterion is called a loss function and calculates the accuracy of the model for that prediction. For predicting one or more scalar numbers, a process called regression, the mean square error (MSE), the square root thereof (RMSE), or the absolute difference are commonly used.

Throughout the literature, there are several types of neural network architectures that have been used for the estimation of position, velocity, acceleration, and torque in upper and lower limbs. However, the majority of the literature encountered has focused on the shoulders [27, 28], elbows [28, 29, 30], or hands [31, 32], and less on the knee or ankles. This is likely because regression on the muscles of the legs is more challenging, as the muscles are located farther away from the skin [33]. Furthermore, the focus generally lies in pattern recognition, whereby actions are grouped into categories, making it a classification problem, whereas the intended outcome of this thesis is to create a pre-processing pipeline for a regression model in order to estimate a continuous numerical value. To illustrate the difference, a multi-branch neural network by Wang et al. [33] uses time domain and frequency domain features, as well as the cleaned sEMG signals to estimate the gait phase, which is categorical prediction, as well as the knee joint angle, which is a continuous prediction. What this means, is that sEMG data can aid in both classification and regression models, despite the majority of the research focusing on classifiers. Unfortunately this specific study made use of additional sensors other than sEMG measurements, which is a common occurrence: many approaches to joint angle estimation are based on so-called synergy or fusion methods, often making use of inertial measurement unit (IMU) data to directly measure the acceleration [34].

2.4.1 Autoencoders

Autoencoders are a family of neural network architectures with the aim of compressing the input data into a more compact form, called the latent space, after which it tries to restore the input from that compressed latent space. The goal of doing this can be different according to the application; saving memory by only storing the data in the latent space, removing noise by forcing the model to only retain crucial information, or reduce the dimensionality of the input data. Generally, an autoencoder consists of several hidden layers, gradually reducing in size, until it reaches the bottleneck of the architecture. This allows the model to gradually learn an encoding for the input data, and is thus called the encoder. The shape of the decoder is usually the exact mirror architecture of the encoder, so that the data in the latent space can be decoded into an approximation of the input signals. An example of an autoencoder is shown in Figure 11. If successfully trained, an autoencoder combines the tasks of dimensionality reduction and denoising simultaneously, as in order to successfully bring the loss function to acceptable levels, the noise will also have been consistently and significantly reduced.

Similar to other neural network architectures, this architecture can only use cross-sectional data. In other words, they cannot effectively work with time series data, as each input vector represents only one instant in time. If time series data was given as an input, this would require the time series to be split up so that each instance in time is a separate training example. The downside of this is that any temporal correlation is lost in the process. The changes in the input, and the rate at which those changes occur, cannot be taken into account with a simple autoencoder.

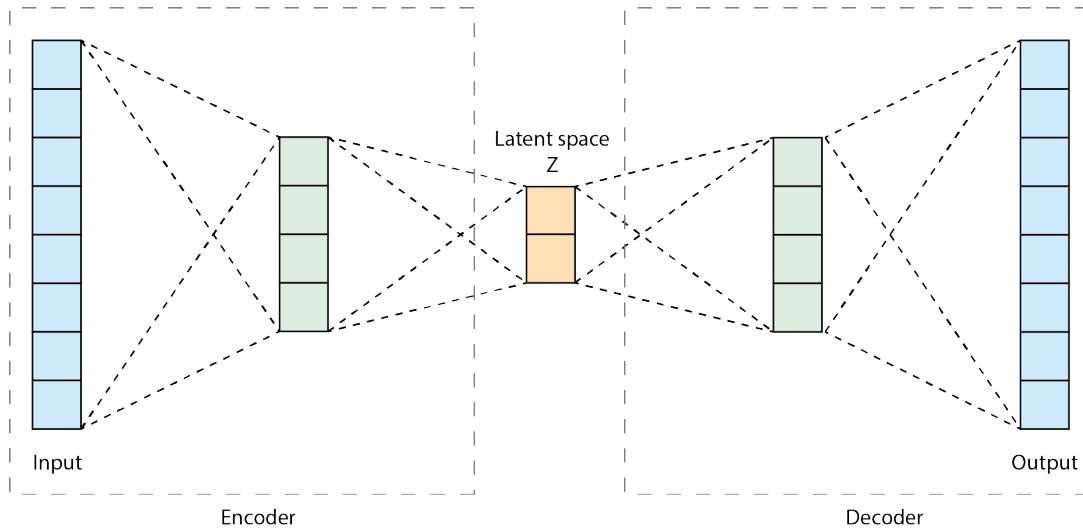


Figure 11: A very simple autoencoder, compressing eight input channels to a two-dimensional latent space.

2.4.2 Recurrent Neural Networks

To overcome the loss of temporal structure in the data, a more advanced family of architectures, called recurrent neural networks (RNN), were developed. These models don't take cross-sectional data as an input, but rather entire time series. Each time step is then put through the network sequentially, whereby the output of a node can be fed back into itself, thus allowing to retain information for the next step. This means that the output of a node can be based on its old output and new input. There are four general shapes of RNNs, shown in Figure 12. The applications of RNNs range from music generation (one-to-many), which takes one input, often called a seed, and generates a long sequence of outputs, in this case music notes that have some temporal structure, to sentence translation (many-to-many), to voice recognition (many-to-one). Note that a one-to-one RNN is really just a feed-forward neural network.

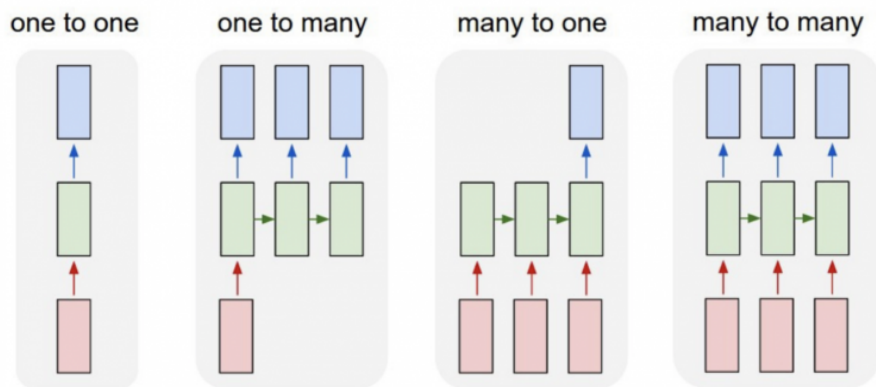


Figure 12: The four general shapes of RNN models when unrolled in time. Each colour represents one layer (model depth), and each sideways step represents a step in time. Source: [35]

To use RNNs for regression, taking sEMG signals as input in order to predict a single value, a many-to-one architecture could be used directly. However, this means that the model will have to learn from all measurements simultaneously, requiring a very large model with many hidden layers, only to parse large amounts of redundant information. Instead, RNNs and autoencoders could be combined to compress the time series into a lower dimension. If this recurrent autoencoder is successfully trained, it could be possible to train a recurrent regression model on this smaller amount of output signals. The downside of

RNNs is that the backpropagation (BP), the algorithm that adapt the weights of the connections between each node in the model in order to try to reduce the loss value, is that the gradients of the error values used to calculate the change in each weight, can either vanish or explode due to the large number of times this gradient is calculated. Making very deep RNNs may very well create weights that approach to infinity, or reduce to near-zero values, causing them to destroy any value that is multiplied by them.

2.4.3 Long short-term memory

One step further, the long short-term memory (LSTM) architecture was designed in 1997 by Sepp Hochreiter and Jürgen Schmidhuber [36] in order to prevent the exploding and vanishing gradient problem. The idea behind an LSTM gate is that it can remember long-term dependencies in the input data, and to retain or forget its own internal state. Additionally, a chain of LSTM gates can form a sort of highway for the backpropagating gradient. Note that LSTM gates do not loop back on themselves. This allows the time series data to be input sequentially while still allowing the model to learn the temporal dependencies that are required for effective training with sEMG signals.

A long short-term memory autoencoder (LSTM-AE) can thus allow for time series compression, creating the lower-dimensional input for a recurrent regression model. LSTM-AEs are a relatively new type of architecture. The first use of such a model is often attributed to a study by Srivastava [37] from 2015, where it was used to predict the next frames of a video sequence. In this case, the video frames are sequentially compressed into the latent space representation, and two decoders were used: one to decode the image and reconstruct it to match the input image, and one to predict what the next frame would look like. Today, the most ubiquitous applications of LSTM-AEs are short-term forecasting of sequences like stocks prices, and anomaly detection, whereby an outlier of a complex sequence can be automatically detected. Given its capabilities, this thesis will look into using it as a compression model for time series data.

2.4.4 Model performance

Many studies report the regression accuracies of their models in term of the average root mean square error (RMSE) of the predicted angle and the actual angle. The RMSE is calculated as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\theta_i - \hat{\theta}_i)^2}$$

Here, n is the number of data points in a signal, θ_i is the i th data point in that signal, and $\hat{\theta}_i$ is the predicted value for that data point. In words, the RMSE takes the difference of the true value and the predicted value and squares the difference, so that all difference values become positive. These values are then averaged in order to remove the effect of the length of the sequence. This produces a single number, of which the square root is taken to undo the squaring. The final number gives an alternative kind of "average" in terms of how many degrees the model is off. All the RMSE values for each exercise or sequence can be averaged over to give the average RMSE.

RELATED WORK AND STATE OF THE ART

The research into limb control using sEMG has been a field of interest in the scientific community since at least the 1960s [38]. Unfortunately, a large part of this research has gone into other joints than the knees. However, since sEMG signals have similar characteristics regardless of the joint that is being researched, one can assume that inspiration and knowledge can be transferred, at least to some extent, to apply on analysis and control of the knee joint.

Ma et al. [39] have used a custom combination of autoencoders and an LSTM in order to estimate the angle of the shoulders, leveraging the autoencoder part of the model to extract common mode information, similar to the EMD and ICA techniques, and the LSTM part in order to decode the autoencoder's latent space. What appears to be crucial to the functioning of this model is the custom signal envelope they created. Unfortunately, the creation of the signal envelope is poorly explained and can therefore not be reproduced. The study also applies the raw and processed data to multi-layer perceptrons and convolutional neural networks, but both perform worse than their LSTM-AE architecture counterpart. Rather than presenting an RMSE value, the authors reported the accuracy in terms of the correlation with the ground truth, yielding a correlation of 0.90 for the MLP, 0.91 for the CNN, and 0.97 for the LSTM-AE. Aung and Al-Jimaily [40] used a simple neural network with only four inputs, one for each measured muscle and its resulting sEMG signal, and one hidden layer with 20 hidden nodes to predict the shoulder angle. As their sampling rate was 2048 Hz, it allowed them to calculate the RMS value of the signal over time, and use those values as inputs to their models. Unfortunately no RMSE value or other performance metrics were reported.

Cheron et al. [41] used a dynamic RNN model with one hidden layer to predict the hip, knee, and ankle angles. The data was preprocessed using a full wave rectification and a third-order averaging filter with a time constant of 20 ms. This was presumably also done to remove the high frequency components of the envelope. Liu et al. [42] tried to improve upon the existing CNN approaches for knee angle estimation by using time-domain features as inputs, rather than the original sEMG signals. These features are the mean absolute value, RMS, waveform length, and energy percentage. They used six muscles, the RF, VL, VM, BF, ST, and the *medial gastrocnemius* (MG), which is the inner calf muscle, which is not present anymore in transfemoral amputees. Together, the six muscles and four features create 24 time-domain input features. The raw signals were only treated with a 20-460 Hz band-pass filter, a 50 Hz notch filter, and full wave rectification. Finally, the amplitude of each signal was normalised, so that the maximum absolute value in each signal becomes 1. The final RMSE that was achieved was $5.88^\circ \pm 0.22^\circ$.

Zhang et al. [43] looked at the continuous joint angle estimation of the legs with healthy users and users with spinal cord injury. They report average RMSE values for leg extension exercises and treadmill exercises of 9° and 6° , respectively. The study used a backpropagation neural network, without the use of recurrent elements, in order to predict the angles of the hip, knee, and ankle. After filtering their seven sEMG signals with a band-pass filter between 20-500 Hz and a 50 Hz notch filter, they rectified the wave, making all parts of signal positive, followed by sub-sampling in order to make the sample rate of sEMG match their angle sensors. Finally, they apply a first order low-pass filter of 5 Hz to gradually and increasingly reduce the high frequency components. This smoothens the envelope of the final signal. Instead of using temporal structure, they use 20 consecutive data points of this filtered signal as an input. Using more than 20 data points only increased their RMSE. They also found that rapid limb movement yielded better accuracy than slow movement.

Finally, Truong et al. [44] created a model to estimate the knee angle and torque using an LSTM-based architecture. After applying a second-order Butterworth band-pass filter (20-500 Hz), they calculated the derivative (i.e. the difference for each data point with respect to the previous data point) of the sEMG signals in order to reduce the non-stationarity [45]. Then, feature extraction was performed by finding the coefficients of a 6th-order autoregressive model, while also calculating the RMS values over

a time window. Together, the autoregressive coefficients, which remain constant, and the RMS values, which constantly change, over a 100 ms time window with 50 ms overlap form the input features. With this, their best model achieved an RMSE of roughly 10° .

The amount of publications investigating or reporting on the long-term stability of sEMG-based models is not very high. Kaufmann et al. [46] looked into the drifting of classification accuracy in hand gesture recognition for various model architectures. They gathered 21 days' worth of test trials and compared three scenarios. The first scenario allows the model to be constantly retrained on all preceding trials up to that time, thus having access to all labelled sEMG data gathered so far. The second scenario restricts the amount of training data to only the five most recent trials, discarding the older sEMG signals. The final scenario allowed the models to only train on the first five trials, creating a larger gap in time as more data is gathered. They found that for the last scenario the accuracy almost always drifted, from the 80-90% accuracy range, to the 60-70% range or worse. Four out of five models could not keep up the accuracy, often seeing significant deterioration in accuracy in only 20 trials, which corresponds to about half a week. This means that every three days, the model has to be retrained in order to keep functioning optimally.

Similarly, Amsüss et al. [47] investigated hand gesture classifier stability over five days by gathering the sEMG data each day, splitting it up into five train sets and five test sets, and combining them to create 25 sets of train and test data. As they expected, they saw the highest accuracies when the train and test data sets originated from the same day, with accuracy generally decreasing monotonically as the time between the train set and test set increased. On average, a 4.2% decrease in accuracy was found over the range of only five days. They argued that this is the consequence of many stochastic processes involving a time component, like the muscle displacing underneath the skin. Although 4.2% may not seem much, the span of five days is very short compared to the long-term use of the prosthesis that the classifier would be used for, considering it would be used for multiple years. This is further supported by the work of He et al. [48], which tracked hand gesture recognition over 12 days, seeing an increase in classification error rates from 7-16% to 35-42%, and a reduced intra-class distance for experiments that were further spaced apart, meaning that the classes tend to overlap more in their sEMG characteristics as time goes on.

From these related works, a number of best practices can be derived that can be implemented into the method of this research. Regardless of the model architecture or advanced filtering technique, a universal preprocessing step should be applied to all EMG signals available. In terms of frequency domain filters, each signal window will be filtered by a 20 Hz high-pass filter, a 450 Hz low-pass filter, and a 50 Hz notch filter with a high precision to remove powerline interference. Furthermore, LSTM-based architectures have been employed before, which has generally kept the overall architecture quite simple, as opposed to the commonly used deep learning models in other application fields of machine learning. This means any model used in this research should not have to be very deep or complex. It is important to note that little of the related works employed advanced filtering techniques like wavelet transforms, EMD, or ICA. This implies a significant contribution can be made by comparing these techniques in the preprocessing phase. In terms of model performance evaluation, the RMSE of the knee angle is most commonly used to reflect the deviation of the predicted knee angle with respect to the ground truth angle, and will also be used as the performance metric here.

METHODS AND TECHNIQUES

In this section, the method of research is discussed, explaining what techniques and approaches are used in order to answer the research questions. As there exists a wide variety of techniques to process sEMG signals and create regression models, multiple data sets will be created from the original, each with their own pre-processing steps. This allows the model proposed in this section to be trained on these data sets in order to look more closely into the effect the pre-processing has on the long-term stability of the accuracy of each model.

The given data exists of 160 trials, each with 64 channels originating from the sensor's 16 by 4 grid, placed as shown in [Figure 1](#). The 160 trials are the total of four days of 40 trials each, those days being the first, second, third, and seventh day of the experiment. The model described below will be trained on the first day only, so that the effect of the delay between training and evaluation on the accuracy can be evaluated.

4.1 PRE-PROCESSING

In order to ensure that the applied frequency domain filters and subsequent decompositions have the same effect during training time as during the inference time, each signal is first segmented into 256 ms windows with a moving window of 32 ms, thus creating 224 ms overlap between subsequent windows. They full set of windows will still be referred to as a single signal or channel, collectively. Each of the 64 channels is then individually band-pass filtered from 20-450 Hz to limit the high frequency noise, and reduce low frequency artefacts. A 50 Hz notch filter with a high quality factor of $Q = 20$ is applied to remove powerline interference. All filters are fourth-order Butterworth filters. In the figure below, both the individual frequency response of the filters as well as the total compounded frequency response are shown in [Figure 13](#). Additionally, this preprocessing step is shown schematically in [Figure 14](#).

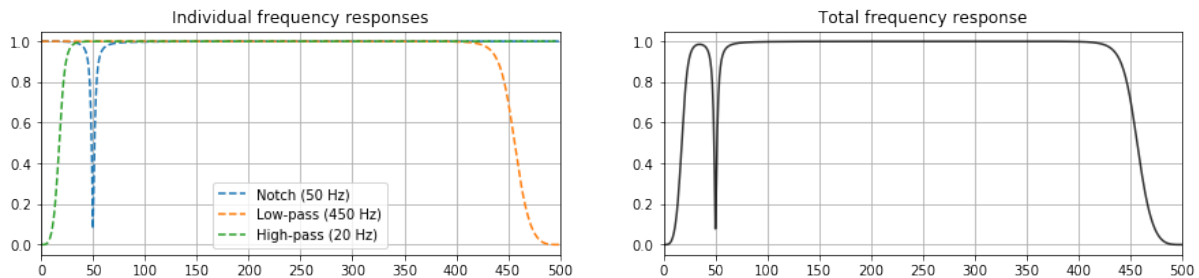


Figure 13: Frequency response for each filter (left) and all filters combined (right). The values shown indicate the factor by which the frequency will be attenuated after filtering.

4.2 SEGMENTATION AND DECOMPOSITION TECHNIQUES

In order to investigate the effect of differentiation, a copy of the windows is made and differentiated. This is done for all data points by taking its current value and subtracting the value of the data point before it: $y_t = y_t - y_{t-1}$. For each technique discussed earlier - Wavelet transform, EMD, and ICA - a copy of the original signal, referred to as *Base*, and its derivative, referred to as *Diff*, is made, so that the techniques can be applied separately. The first copy of *Base* and *Diff* will have the wavelet transform applied. Based on the best thresholding criteria and update rule reported by Phinyomark [20], each coefficient will be adjusted by calculating the Log Scale Modified Universal threshold value and updating it according to the Adaptive Denoising Function (see 2.3.1). Using the adapted coefficients, the signals will be reconstructed. The second technique, EMD, is applied in a similar fashion to the wavelet transform. EMD is applied to every window separately and the SCAD threshold is used to minimise noise. The signals are

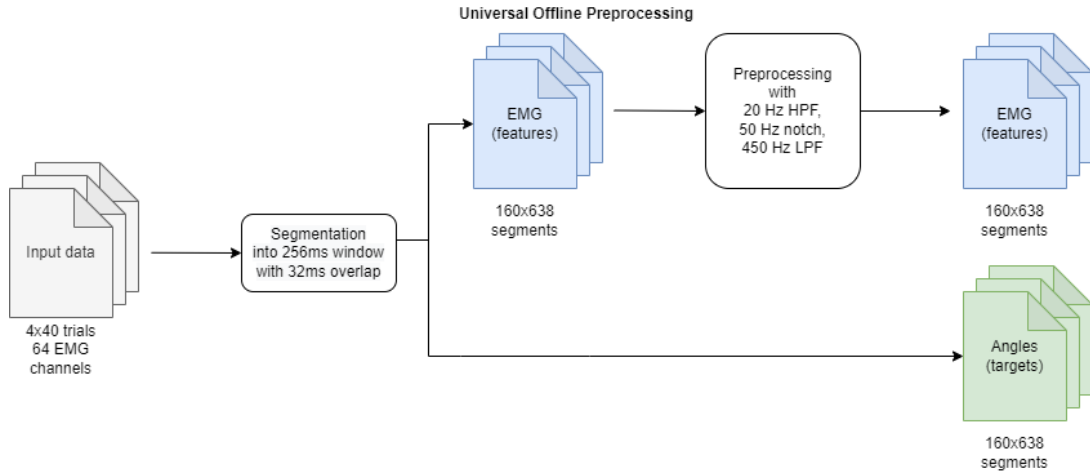


Figure 14: The first step of preprocessing the data consists of segmentation and frequency domain filters on each window.

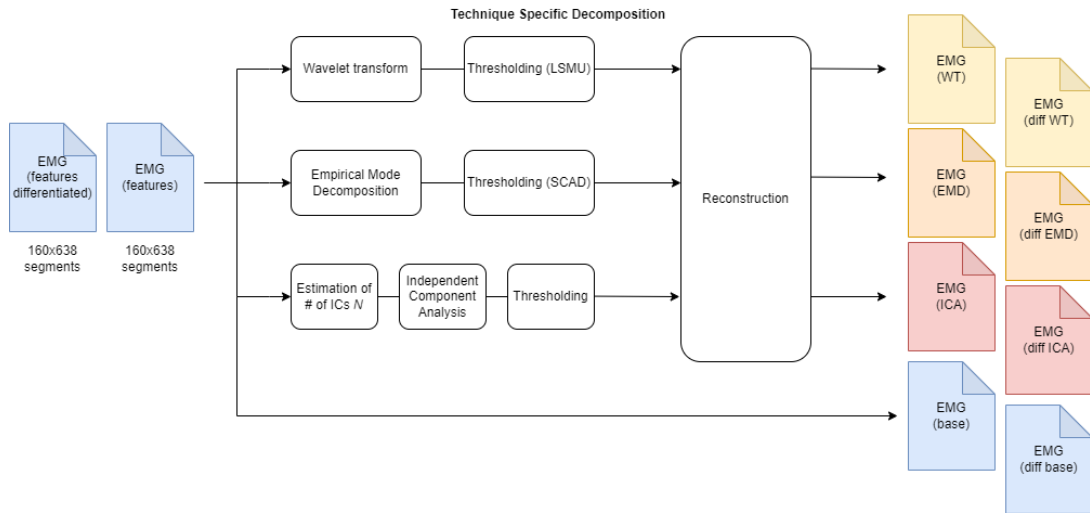


Figure 15: The second step of preprocessing consists of creating three more copies: one for the wavelet transform, one for empirical mode decomposition, and one for independent component analysis.

then reconstructed.

To be able to use ICA, a small grid search will be performed in order to establish the number of sources. The estimate for the number of ICs by Naik and Kumar (see 2.3.3) will be run in parallel to see if the found value matches. Based on this number, the ICA will be run on all trials. This value will also help to inform the size of the latent space in the LSTM-AE model, given that, if the method works, this method should yield the true amount of sources present, which is also the goal of the LSTM-AE. As ICA acts as a dimensionality reduction technique on its own, there will be no need for the autoencoder. Therefore, the ICs will be passed directly into the regression model, where the number of nodes per layer corresponds to the number of ICs N . These steps, referred to as technique specific decomposition, are shown schematically in Figure 15. For each of the four datasets, another copy is made in which the signals are differentiated in order to reduce the effect of the non-stationarity.

4.3 ENVELOPE SMOOTHING TECHNIQUES

In a final adaptation step, the root mean square value is calculated over time by taking sub-windows of 8 ms, which is close to the often used 10 ms [49]. This allows all root mean square values to be calculated

based on the same number of samples, yielding $256/8 = 32$ values per window. The goal of combining multiple measurements in a single prediction is to smoothen out any leftover disturbances and increase the overall stability by reducing the direct effect of the sEMG signal amplitude. These RMS windows will each form a set of inputs, that can be fed into the regression model simultaneously. This means that there will be a predicted value once every eight milliseconds, rather than every single millisecond. If successful, the number of predictions per second should still be high enough to not interfere with the joint's update rate. An alternative to using the RMS approach is to use a first-order low-pass filter at 5 Hz, as done by Zhang et al. [43]. This will gradually and increasingly attenuate higher frequencies, which should also smoothen the envelope of the sEMG signal. In summary, there exists three separate steps that can be combined with each of its options, as illustrated in Figure 16.

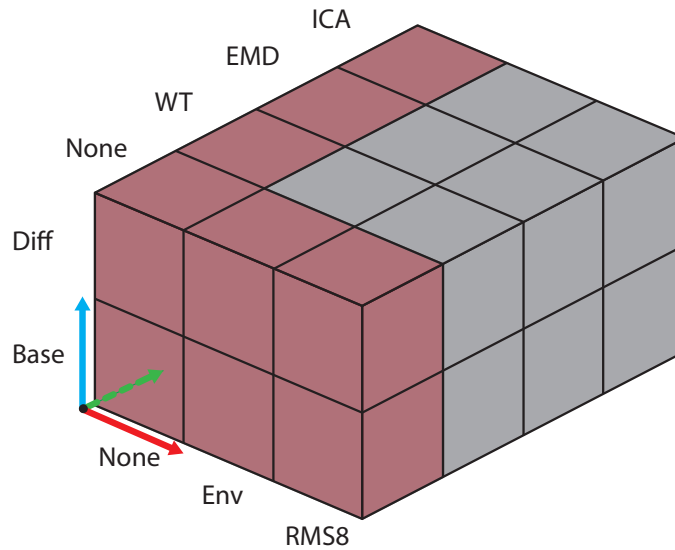


Figure 16: At this stage, there exists a three-dimensional space of possible pre-processing steps, allowing 24 total possible datasets. The three axes are characterised as differentiation (blue), envelope smoothing techniques (red), and decomposition methods (green).

4.4 MODEL ARCHITECTURE

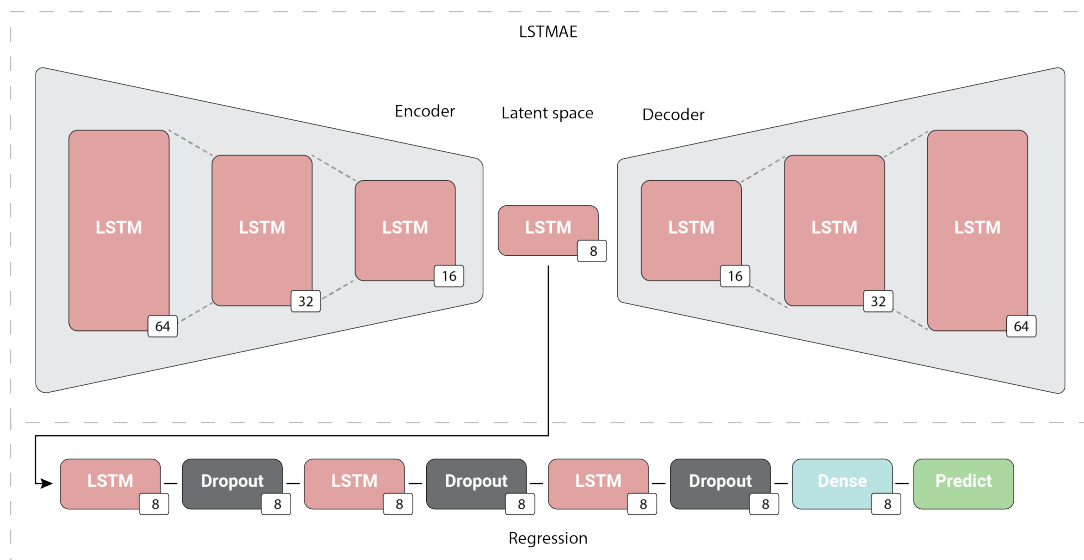


Figure 17: Proposed LSTM-AE architecture with the regression model.

The proposed model consists of two parts, namely the LSTM-AE and a regression model, as shown in Figure 17. Two classes of models will be trained and compared, namely one with and one without the LSTM-AE. The LSTM-AE consists of an encoder and decoder, each with three hidden layers, halving or increasing in size every next layer to gradually reduce or restore the dimensionality of the input signals. The initial sizes of the encoder, and thus also the decoder but in reverse, will be 64:32:16:8. This number of hidden layers is based on the four layers (3 + 1 for the latent space) by Ma et al. [39]. The size of the latent space is based on the expected number of major muscle groups that contribute to the bulk of the sEMG signals. The definitive size of the latent space is to be determined by repeated training and searching for the best performing model. After the autoencoder is trained, the decoder is decoupled, and the encoder is used as an online dimensionality reduction step before the input to the regression model. This part of the model is based on the architecture found in [44]. This should finally result in a single number, the knee joint angle, for every window in the signal. The angle is chosen over the velocity and torque as it is more easily comparable to the results of other works.

4.5 MODEL TRAINING

For all data sets except the ICA sets, two variants of the regression model can be trained. One variant use the LSTM-AE to compress its input space, reducing the dimensionality, and the other variant keeps the original 64 channels. This means that the regression model has two different versions, where only the number of nodes per layer changes. This is not needed for ICA, as it has no need for the LSTM-AE; the number of channels is already reduced. As such, the ICA dataset will not make use of the LSTM-AE and go straight into a regression model. For that dataset, the number of nodes per layer in that model will be the number of independent components N found by the earlier grid search.

The regression models will be trained on a subset of the complete dataset due to time constraints and the large volume of models that require training. The models are trained on a random 10% subset of the first day data, which amounts to four complete trials, or roughly 10,000 windows. For reference, compensating for the added data points created by the overlap in the windows, this equals about 38 minutes of continuous sEMG measurements.

In the interest of time, not all 24 datasets can be created and used to train two models each (or one if ICA is applied). This search space is too large for the timespan in which this research sits. As such, the differentiation option (blue axis) and envelope transformations (red axis) will be searched through first. The best performing model in terms of same-day performance will be used to look into the decomposition options (green axis). This turns 42 (18x2 + 6 ICA) possible models into 14 (6 + 4x2) possible models, which are shaded red in Figure 16. This reduces the workload by two-thirds.

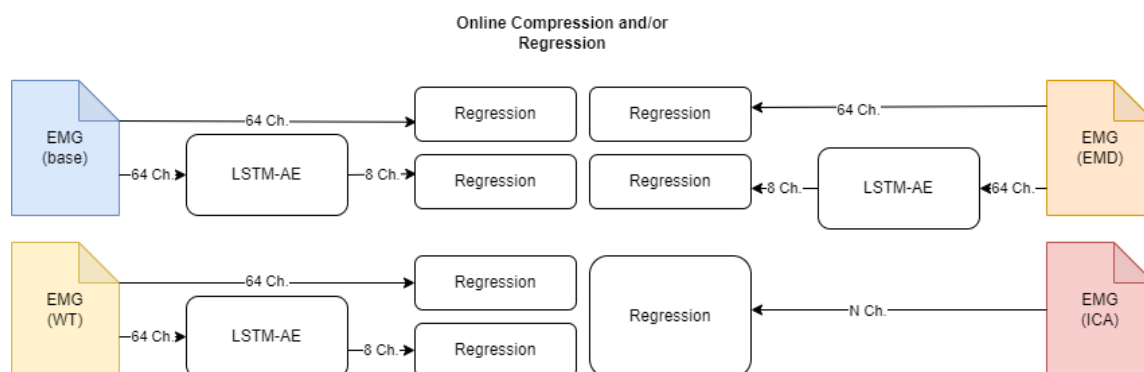


Figure 18: The third step of preprocessing consists of training two variants for the base, wavelet transformed, and EMD datasets, one with and one without the LSTM-AE as a compression layer. The ICA dataset will only be trained using a regression model, without the LSTM-AE. Note that in the diagram, the differentiated versions are omitted for clarity.

In order to select which models performed best out of the first six, the same-day performance will be compared. This is represented as the test loss value that is generated during the training of the models. The model with the lowest loss value, provided it is sufficiently consistent during training time, will be selected as the starting point for evaluating the decomposition methods. Based on these findings, one or more models and their accompanying datasets will be selected, and their multi-day performance compared. A randomised but constant testset will be used to calculate the distribution of RMSE values for each day, for each model. These daily model distributions and their means can be compared between models. There are two criteria which should ideally both be met by the same model: the distributions should be closely packed around the mean and have small tails on the right-hand side, the daily means should be as low as possible and increase as little as possible between subsequent days. Based on these criteria, the effects of each set of applied pre-processing steps should be comparable.

RESULTS AND DISCUSSION

In this chapter, the results of the experiments as described in the previous section are presented and interpreted.

5.1 PRELIMINARY GRID SEARCH

The first set of models consist of the envelope smoothing techniques, with and without differentiation. This created six datasets and accompanying models in total. Note that when referring to any model by its name, this in fact refers to the same model architecture trained on the dataset of that same name. The target values are obtained directly from the right-hand side IMU, which provides the knee angle. These values were only altered for the two RMS models, where the window of 256 samples was subdivided into 32 sub-windows of eight milliseconds each, of which the average is taken to be the new target value. There is an additional left-hand IMU on the same knee, which was disregarded.

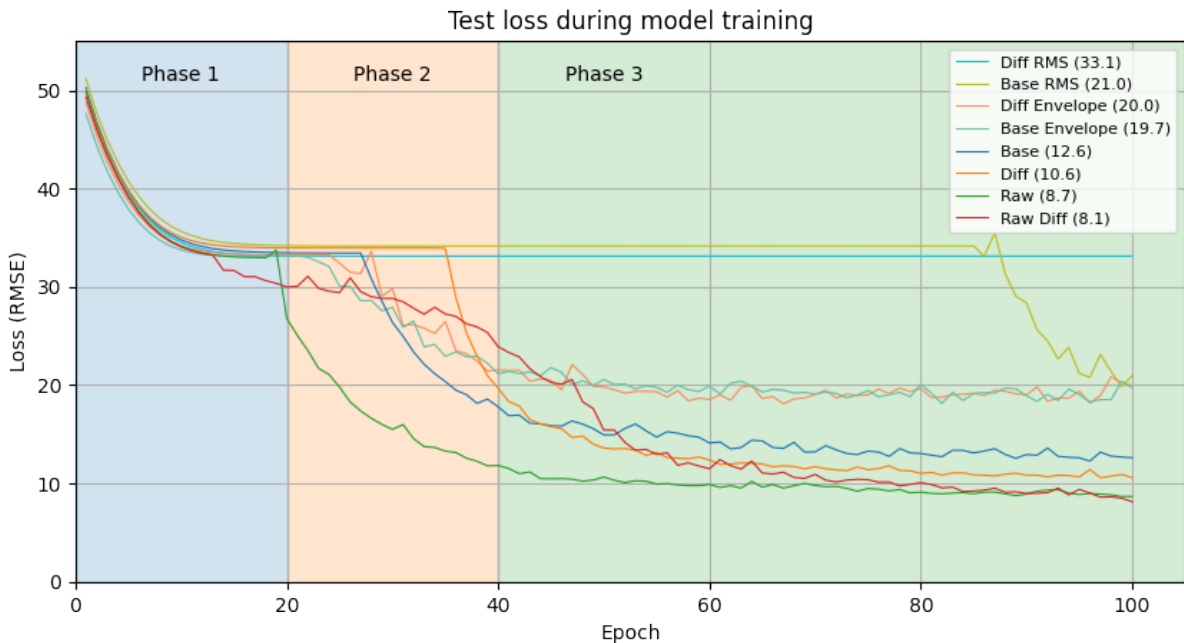


Figure 19: The test losses from the training of the initial six models. Lower is better.

In [Figure 19](#) the test loss plots are graphed, showing how the performance of the models change with more training iterations. A general pattern emerges that divides the graph into three phases. The first phase spans from the first epoch until epoch 20. Although it is not certain why the curves follow one another so closely, one possibility is that this is where the initial model weights are gradually adjusted to allow the predicted value to converge to a roughly equal value for all models. This value can be interpreted as a minimum performance threshold, which any model should evidently be able to obtain. In the second phase, five out of six models start their second convergence. This sudden drop-off after multiple epochs of plateau is likely due to the highly non-convex nature of the loss landscape. At the start of the second convergence, the combination of model weights has reached a point where the gradient of the loss function suddenly sharply decreases, pointing towards a new local minimum. As the model architecture stays constant, the difference in the moment of second convergence is caused by the difference in the input data. As such, this moment can be compared between models and provide an indication of the how well the input features are related to the target values. It is important to note that all hyperparameters, such the learning rate and optimisers are maintained constant between different datasets. Finally,

the third phase sees most models reducing their loss values until another plateau is hit.

It is clear that the RMS and envelope smoothing techniques, in both differentiated and undifferentiated forms, are outperformed by the simpler models, *Base* and *Diff*. The *RMS* models struggle to find a point in the loss landscape that allows them to converge further. In fact, after the initial drop from the randomised weights in the first ten epochs, the differentiated *RMS* model stops converging altogether. As the target value for the *RMS* models is not much more complex than the standard values, this indicates that the *RMS* of the eight millisecond sub-windows is not a good predictor of the knee angle. Although the *Base RMS* model may appear to be decreasing quite steeply, the approximately 65 epochs of plateau make this model an unlikely candidate to perform better when combined with the decomposition techniques.

The *Envelope* datasets do obtain their second convergence relatively quickly, even faster than the *Diff* model, but fail to get much farther than the 20.0 mark. The conclusion here is the same as for the *RMS* models, although there is a big difference in the loss landscape, which is much less flat and more easily traversed by the optimiser in this case. Further fine-tuning of the way the envelope of the signals are smoothed may provide a lower loss value still.

Finally, the *Base* and *Diff* models perform almost equally well, with the former converging earlier, and the latter obtaining a better final result of 10.6. This is an acceptable margin of error, falling approximately in line with the errors found in literature. The reason for the better performance may be attributed to the effects the differentiation has on the non-stationary nature of raw sEMG signals.

It is not clear from this experiment which method will perform better with respect to the multi-day performance. At this stage two additional models were added that used completely raw data, *Raw* and *Raw Diff*, without pre-processing it with the frequency domain filters described in [section 4.1](#). This adds a meaningful baseline which aids in the comparison between the models. In [Figure 20](#), the new model *Raw* and its differentiated version are compared against *Base* and *Diff*. Note that there are no versions that include the RMS and envelope smoothing techniques, as they are already disregarded at this point.

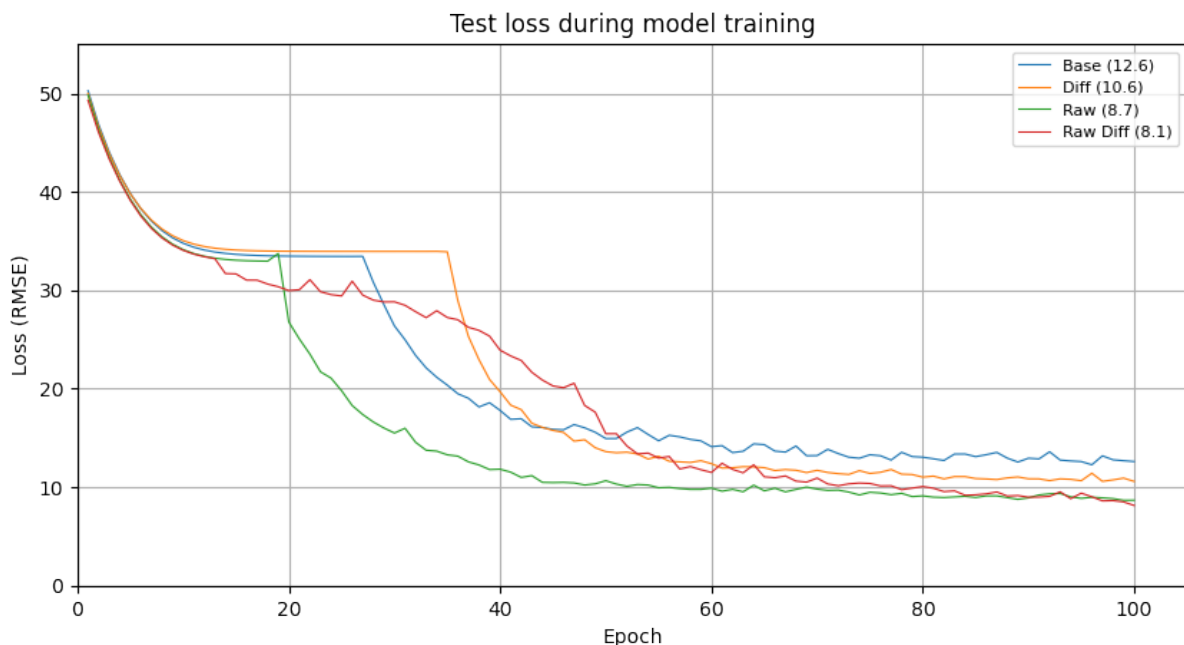


Figure 20: The test losses from the training of the raw data model, differentiated and unaltered, compared to the original *Base* and *Diff* models. Lower is better.

The most striking conclusion from this graph is that the raw data performs even better than the pre-processed data, which puts the frequency domain filtering in question. This difference may indicate that

the filters overlap and remove a salient part of the frequency spectrum of the sEMG signals. Furthermore, the difference between the differentiated version and standard version of the *Raw* models has become even smaller. *Raw Diff* starts its second convergence before the plateau in the second phase has started, but requiring almost 80 epochs to get mostly settled. It may even converge slightly further based on the average slope of the loss curve in the last 20 epochs. *Raw* converges more steeply and starts to plateau around epoch 50 already, indicating a better relationship between the input features and target values than the other three models. Based on these combined findings, the *Raw* and *Raw Diff* datasets were both selected out of the now eight datasets to be used in combination with the decomposition techniques described in [section 2.3](#). It is important to note that since the original *Base* and *Diff* models show that they can be trained properly, albeit with a larger margin of error, the results of the other four models in [Figure 19](#) and the conclusions based thereon are still considered valid.

In short, differentiated datasets seem to generally outperform non-differentiated counterparts, and the models that run on data that has not been pre-processed by frequency domain filters generally outperform those who do include that step. Combining these two facts then also yields the best performing model with respect to same-day performance: *Raw Diff*. Making use of RMS of sub-windows only increases the loss, and smoothing the envelope of the signal yields no benefits either. Based on the loss graphs, the datasets *Raw* and *Raw Diff* were chosen to be combined with the decomposition techniques.

5.1.1 LSTM-AE and ICA

At the same time, the two datasets were used to determine the viability and configuration of the LSTM-AE. Both datasets were used in the autoencoder as depicted in [Figure 17](#). Unfortunately, the loss value did not converge to any value much lower than the model when it uses its the randomly initialised starting weights. Alternative architectures with fewer layers were used, where the latent space was two and four times as large, creating models with layers of sizes 64:32:16 and 64:32 nodes, respectively. This was done to investigate the effect of the depth of the model on the training. This yielded no improvement, as the loss function stayed nearly constant for 100 epochs. This means that the proposed LSTM-AE model is not suitable to compress sEMG data into fewer dimensions. As such, the focus was put solely on the regression models.

In the case of ICA, the iterative estimation approach by [\[26\]](#) was used to estimate the required number of ICs, but this failed to converge to any non-zero value. The method was applied to numerous windows from various trials from the *Raw* and *Raw Diff* datasets, as well as complete trials. Unfortunately, it occurred only rarely that ICA managed to converge on any given number of desired independent components between 1 and 63. This is indicated by the determinant of the global matrix $\mathbf{G} = \mathbf{AW}$, where \mathbf{A} is the mixing matrix and \mathbf{W} is the unmixing matrix. The determinant is almost always zero for any number of ICs, meaning the unmixing of the original signal was not successful. The mathematical reasoning is beyond the scope of this research.

As ICA and LSTM-AE have the same goal, it stands to reason that their lack of convergence is caused by a shared root cause. A likely contender for this is the small area over which the 64 signals are recorded. Related works generally make use of a smaller number of electrodes which are spaced farther apart. This should make it easier to perform blind source separation by means of ICA or autoencoders. This in turn explains why related articles obtain better results when employing these techniques. It is important to emphasise that the LSTM(-AE) and ICA should not be disregarded as a viable technique for sEMG signals in general, as it has been demonstrated to work quite well [\[Naik2011AProcessing\]](#). Regardless, for the purpose of this research, the LSTM-AE and ICA are disregarded as viable approaches for reducing the dimensionality or denoising the signals.

5.2 SAME-DAY PERFORMANCE

Next, the wavelet transform and EMD were applied to both datasets, as shown in [Figure 21](#). A clear hierarchy emerges, at least in terms of final loss values. *Raw EMD* performs slightly worse than *Raw Wavelet*, which in turn performs slightly worse than *Raw*, and the same order is true for their differentiated coun-

terparts. The point of second convergence is approximately the same for all models, with all of them converging earlier than *Base* and *Diff*. As the difference between the two extreme loss values, ignoring *Base* for now, is only 3.6, it may be too early to disregard any of the models already, as this performance makes no promises on the long-term performance.

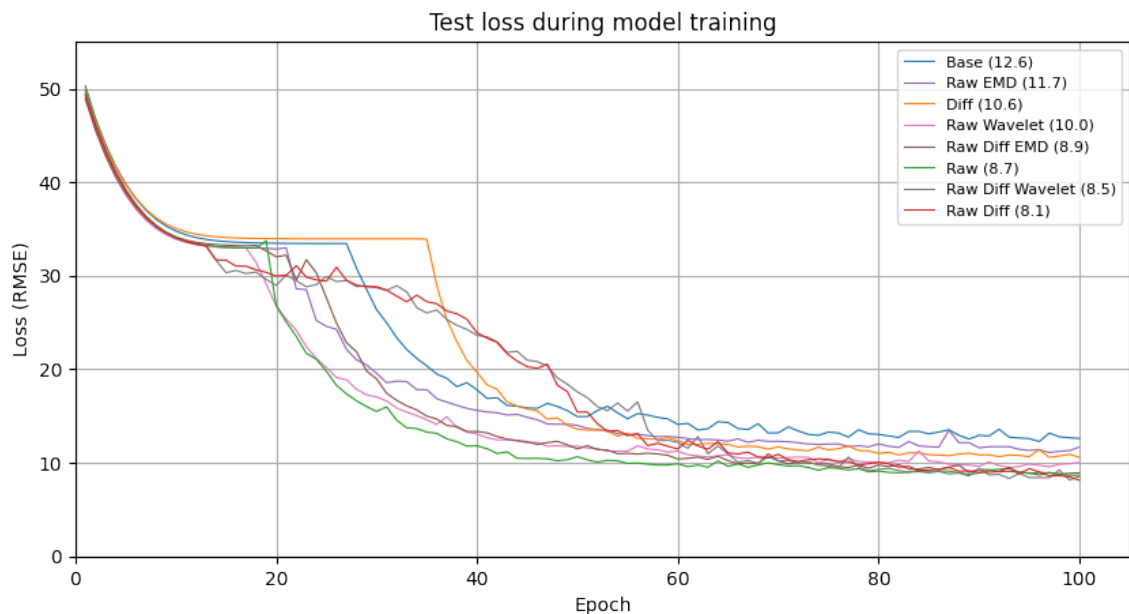


Figure 21: Test losses of the *Raw* and *Raw Diff* models, combined with the wavelet transform and EMD. *Base* and *Diff* are shown for comparison.

5.3 MULTI-DAY PERFORMANCE

With the grid search complete, the resulting six models were given data obtained from one, two, and six days later than the data that was used to train them. Ten percent of the total available data for each day was used to evaluate them. The distribution of RMSE values are as shown in Figure 22. The mean RMSE of each distribution is indicated with a dashed line in each plot, and is also shown in Figure 23.

One apparent inconsistency is that the second day means appear to be higher than expected, being higher than the third day means in all but one case. As it is expected that the performance decreases monotonically, this may point to a type of noise introduced during only the second day of the experiment, which could disproportionately affect the accuracy.

Despite this, there is a clear distinction visible between the three *Raw* non-differentiated models, indicated with dashed lines, compared to the three *Raw Diff* models, indicated with solid lines in Figure 23. The non-differentiated group performs structurally worse on all days than their differentiated counterparts, which in turn perform approximately the same as the *Base* and *Diff* models, which are also shown for comparison. These two models show a much better Day 1 performance, which contradicts the findings from the preliminary grid search. Since the conclusions drawn based on those performances are not based on the absolute value of the test loss, but rather on the relative differences between those values, this should have no significant effect the validity of the resulting conclusion.

One could argue that the models have been trained on too small a trainset, causing gaps in the model's ability to make good predictions. This could then be the cause for the inconsistent loss values between the same-day performance and the first day of the multi-day performance. To test effect that the volume of training data has on how well the model can , i.e. can maintain a consistent accuracy on the test data, the best performing model according to Figure 23, namely *Diff*, was retrained using increments of 10% more data, from 20% up to 100%. The mean RMSE values resulting from these models are shown in

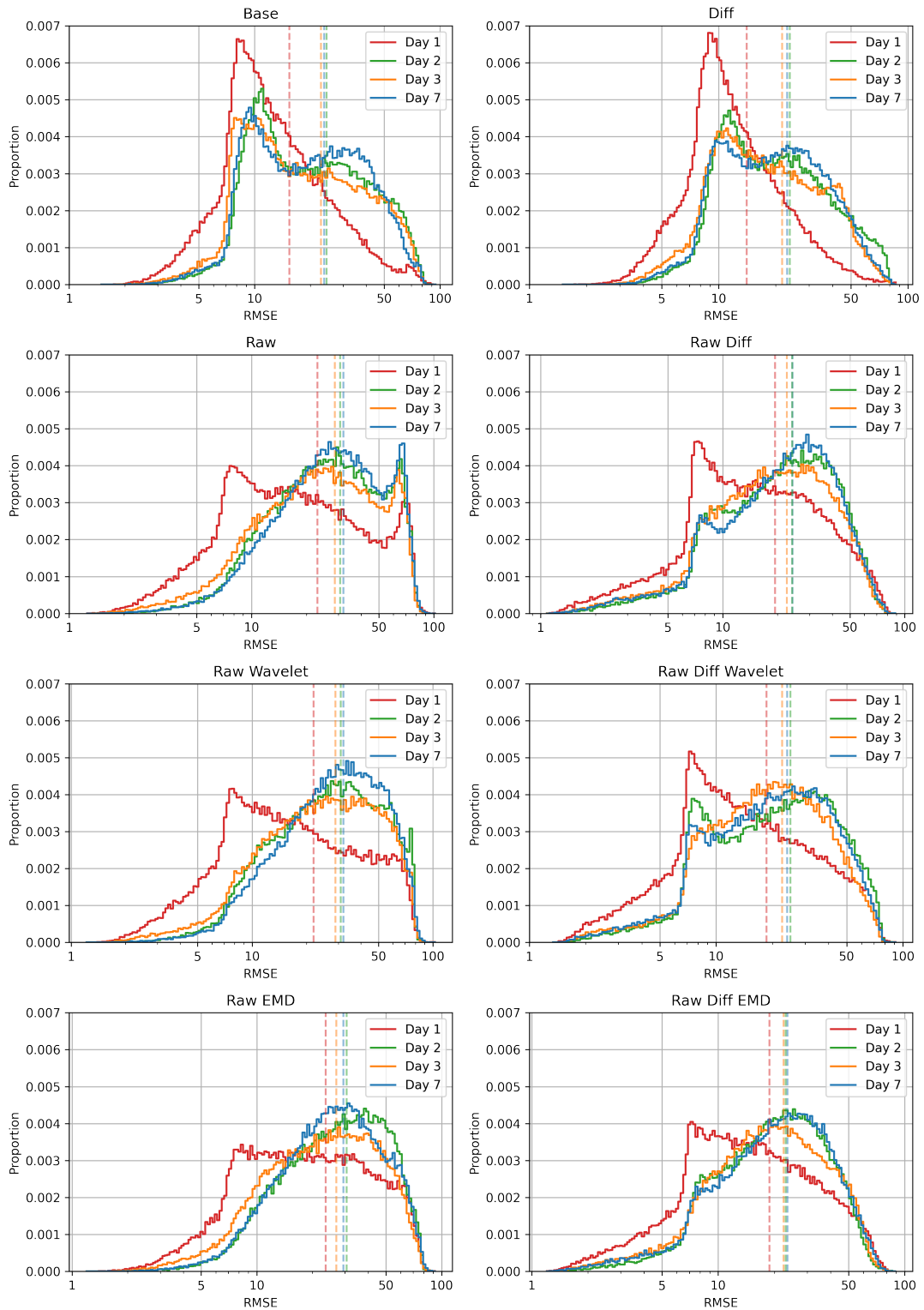


Figure 22: Multi-day performance of each model. The dashed lines indicate the mean RMSE of the respective model. Note that the x-axis is logarithmic to compensate for the right skew.

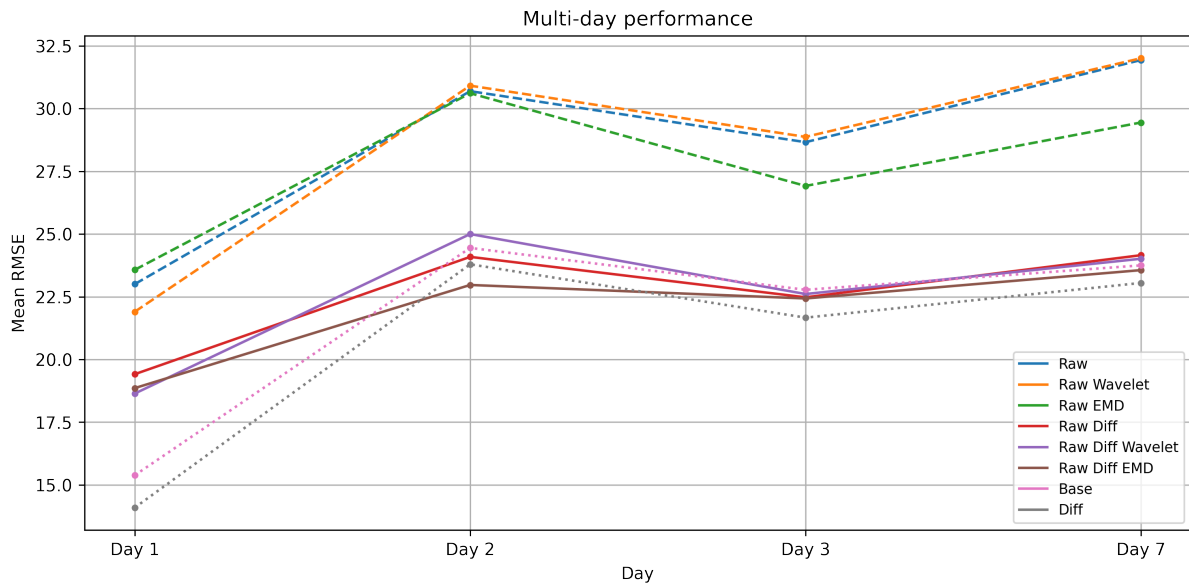


Figure 23: Multi-day mean RMSE of each model.

Figure 24. The first day performance should be ignored, as for this day the trainsets and testsets overlap, thereby influencing the prediction accuracy. Despite that, based on these findings, no amount of increase in training data allows the model to consistently outperform the models trained on less data. This is emphasised in Figure 25, where the ranking of these models is plotted for each day, where one is the best performing model on that day and ten is the worst performing model. As indicated by the many line crossings and lack of a clear trend, none of these models is consistently better than others. At first glance, it appears that 20% and 30% perform much better than 10%, but then the overall performance declines going up to 40%, 50% and 60%, rises again for 70% and 80% and finally drops severely for 90% and 100%.

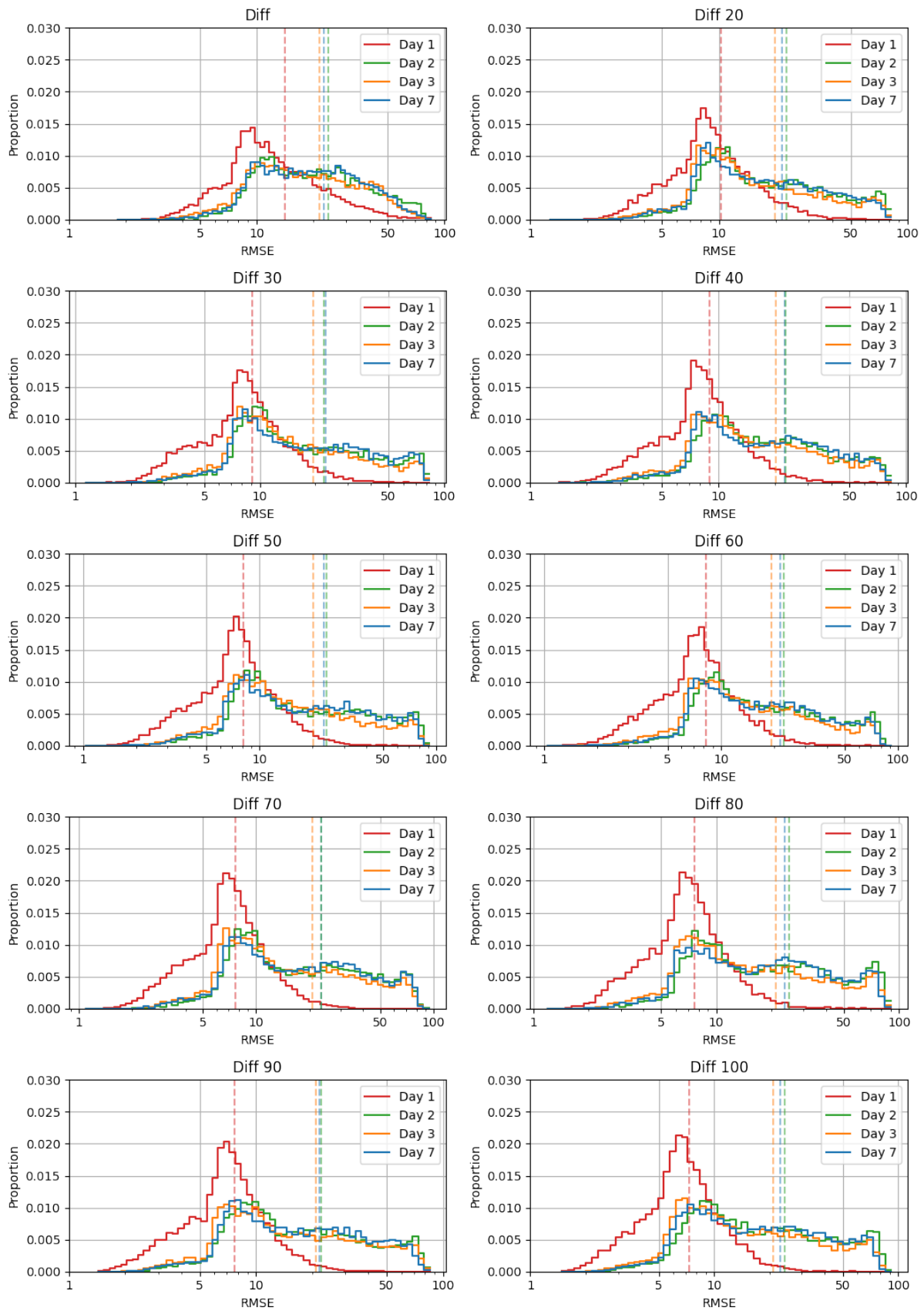


Figure 24: The multi-day performance of the *Diff* models trained on increments of 10% of the total available training data.

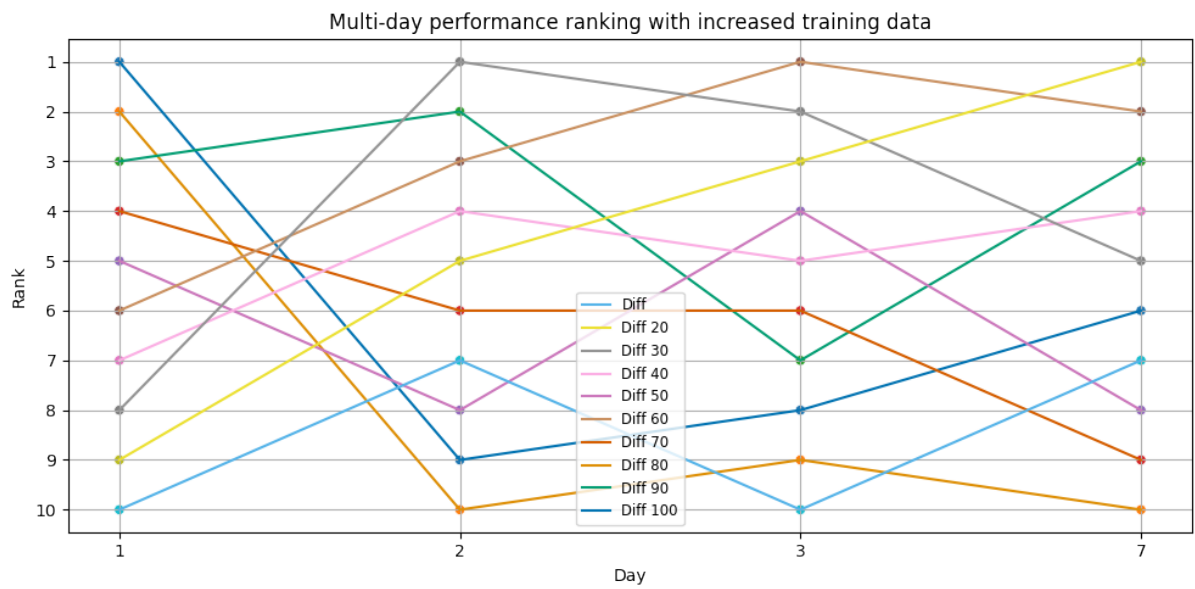


Figure 25: Multi-day ranking of each *Diff* model trained on increments of 10% of the total available training data. The number in each model name indicates the percentage of available data used for training. The rest is used for evaluating the model.

CONCLUSION

This research aimed to compare some of the most commonly used sEMG filtering techniques, specifically those obtained in order to control computerised knees, and see how the models trained on the data filtered with the various techniques perform in a multi-day setting. In order to do so, the following research question, and its two accompanying sub-questions, were posed:

What combination of commonly used signal filtering and dimensionality reduction techniques can best increase the multi-day performance stability of an sEMG-based computerised knee joint?

1. What combination of surface electromyography filtering techniques found commonly in literature yield the highest accuracy when the training and evaluation occurs on the same day?
2. What impact do these selected techniques have on the multi-day accuracy of predicted angle of the knee joint compared to the same-day accuracy?

To sum up, as there is a vast range of combinations of pre-processing steps, a methodology was created to make it possible to compare the effect of three decomposition techniques - the wavelet transform, empirical mode decomposition, and independent component analysis -, as well as the effect of differentiation, and the effect of envelope smoothing and using the root mean square of sub-windows, on top of applying frequency domain filters. Finally, an attempt was made to employ an LSTM autoencoder in a similar fashion to regular autoencoders with the aim to reduce the dimensionality and reduce the noise from the 64-channel sEMG measurements simultaneously.

As for the dimensionality reduction approaches, namely ICA and the LSTM-AE, neither approach succeeded to converge. The most likely cause for this is the small physical area of the leg over which the sEMG signals have been recorded, providing insufficient spatial separation between the signals to perform the unmixing. The methodology from this work could be applied to different datasets that are based on measurements that are farther apart spatially, like done in [39, 44], so that these techniques can be included in the final comparison.

Despite the inconsistencies found in the loss values during the training phases of the models, there are several key points that can be concluded from this experiment. Firstly, the two approaches of trying to smoothen the envelope of the input signal by taking the root mean square (previously referred to as "RMS") and low-pass filtering the signal to smoothen the envelope (referred to as "envelope") yield no beneficial results and are not suited to gain model performance in general. In the same vein, frequency domain filters may do more harm than good, as it may remove salient parts of the frequency spectrum, in turn losing valuable information. This is hinted at by the fact that applying the filters yields one of the best performing models in the multi-day performance comparison, but one of the worst in the same-day performance comparison. It has been shown that using no frequency domain filters may actually be a better option. A more in-depth look into what exact combinations of frequency domain filters provide the best signal-to-noise ratio, both in the general field of sEMG signal acquisition and for knee angle estimation specifically, would be a valuable addition to scientific literature.

Furthermore, differentiation helps to stabilise the signal and will generally provide better performance, both short-term and long-term, than the original signal. In all but two cases, namely using RMS and Envelope datasets, did the differentiated version perform better than its non-differentiated counterpart, regardless of the decomposition techniques used or whether or not the frequency domain filters were applied. Combining these findings, using the raw signal and taking its first derivative yields the best same-day performance. Based on the inconsistencies in the findings of the same-day performance and the first day of the multi-day performance, it may be wise for any further research to include a frequency-filtered dataset and taking its derivative and compare the results, as was also done in this work.

With regards to the decomposition techniques, no evidence was found that applying the wavelet transform or empirical mode decomposition yields any consistent performance gains in same-day or multi-day performance. Neither technique has changed the multi-day RMSE by a large enough amount to be considered effective. The means of day 2, 3, and 7 remain close together, indicating the decomposition techniques do not contribute to the mitigation of the cause of the increase in mean after the first day. The difference in in spread of the multi-day mean performances between every trained model and its differentiated counterpart is also negligible; differentiation does not contribute to this mitigation either, despite its positive effects on the same-day performance.

These findings go against the common idea that more advanced techniques are always better. In a pursuit to always improve the way humans and machines interact, one may find it encouraging to see that simple can outperform complex. More concretely, this has interesting consequences for real-life applications, as the need for fewer advanced and computationally heavy techniques would allow for less expensive and hopefully more widely available prostheses, as well as allowing others to more easily contribute to the research of this still growing scientific field.

BIBLIOGRAPHY

- [1] Bill Dupes. ‘Prosthetic Knee Systems - Amputee Coalition’. URL: <https://www.amputee-coalition.org/resources/prosthetic-knee-systems/>.
- [2] *Microprocessor controlled knees* | Ottobock UK. URL: https://www.ottobock.co.uk/prosthetics/info_for_new_amputees/prosthetic-technology-explained/computer_controlled_knees/ (visited on 09/02/2022).
- [3] ‘MyLeg’. URL: <http://myleg.eu/project-overview>.
- [4] Robert V. Schulte et al. ‘Multi-Day EMG-Based Knee Joint Torque Estimation Using Hybrid Neuromusculoskeletal Modelling and Convolutional Neural Networks’. In: *Frontiers in Robotics and AI* 9 (Apr. 2022), p. 107. ISSN: 22969144. DOI: [10.3389/FR0BT.2022.869476/BIBTEX](https://doi.org/10.3389/FR0BT.2022.869476/BIBTEX).
- [5] ‘Over ons - RRD’. URL: <https://www.rrd.nl/over-ons/>.
- [6] Jacob Kulowski. ‘THE CLASSIC: Flexion Contracture of the Knee’. In: *Clinical Orthopaedics & Related Research* 464 (Nov. 2007), pp. 4–10. ISSN: 0009-921X. DOI: [10.1097/BLO.0B013E31815760CA](https://doi.org/10.1097/BLO.0B013E31815760CA). URL: https://journals.lww.com/clinorthop/Fulltext/2007/11000/THE_CLASSIC__Flexion_Contracture_of_the_Knee__The.2.aspx.
- [7] ‘Knee Anatomy Video | Medical Video Library’. URL: <https://www.ypo.education/orthopaedics/knee/knee-anatomy-t194/video/>.
- [8] Barclay W. Bakkum and Gregory D. Cramer. ‘Muscles That Influence the Spine’. In: *Clinical Anatomy of the Spine, Spinal Cord, and ANS* (Jan. 2014), pp. 98–134. DOI: [10.1016/B978-0-323-07954-9.00004-9](https://doi.org/10.1016/B978-0-323-07954-9.00004-9).
- [9] Miguel Simao et al. ‘A Review on Electromyography Decoding and Pattern Recognition for Human-Machine Interaction’. In: *IEEE Access* 7 (2019), pp. 39564–39582. ISSN: 21693536. DOI: [10.1109/ACCESS.2019.2906584](https://doi.org/10.1109/ACCESS.2019.2906584).
- [10] Sukhan Lee and George N. Saridis. ‘The Control of a Prosthetic Arm by EMG Pattern Recognition’. In: *IEEE Transactions on Automatic Control* 29.4 (1984), pp. 290–302. ISSN: 15582523. DOI: [10.1109/TAC.1984.1103521](https://doi.org/10.1109/TAC.1984.1103521).
- [11] Rubana H. Chowdhury et al. ‘Surface Electromyography Signal Processing and Classification Techniques’. In: *Sensors 2013, Vol. 13, Pages 12431-12466* 13.9 (Sept. 2013), pp. 12431–12466. ISSN: 1424-8220. DOI: [10.3390/S130912431](https://doi.org/10.3390/S130912431). URL: <https://www.mdpi.com/1424-8220/13/9/12431/html><https://www.mdpi.com/1424-8220/13/9/12431>.
- [12] Carlo J. De Luca et al. ‘Filtering the surface EMG signal: Movement artifact and baseline noise contamination’. In: *Journal of biomechanics* 43.8 (May 2010), pp. 1573–1579. ISSN: 1873-2380. DOI: [10.1016/J.JBIOMECH.2010.01.027](https://doi.org/10.1016/J.JBIOMECH.2010.01.027). URL: <https://pubmed.ncbi.nlm.nih.gov/20206934/>.
- [13] Björn Gerdle et al. ‘Acquisition, Processing and Analysis of the Surface Electromyogram’. In: *Modern Techniques in Neuroscience Research* (1999), pp. 705–755. DOI: [10.1007/978-3-642-58552-4_{-}26](https://doi.org/10.1007/978-3-642-58552-4_{-}26).
- [14] J. P.P. Van Vugt and J. G. Van Dijk. ‘A convenient method to reduce crosstalk in surface EMG’. In: *Clinical Neurophysiology* 112.4 (Apr. 2001), pp. 583–592. ISSN: 1388-2457. DOI: [10.1016/S1388-2457\(01\)00482-5](https://doi.org/10.1016/S1388-2457(01)00482-5).
- [15] Madeleine M. Lowery, Nikolay S. Stoykov and Todd A. Kuiken. ‘A simulation study to examine the use of cross-correlation as an estimate of surface EMG cross talk’. In: *Journal of applied physiology (Bethesda, Md. : 1985)* 94.4 (Apr. 2003), pp. 1324–1334. ISSN: 8750-7587. DOI: [10.1152/JAPPLPHYSIOL.00698.2002](https://doi.org/10.1152/JAPPLPHYSIOL.00698.2002). URL: <https://pubmed.ncbi.nlm.nih.gov/12471047/>.
- [16] Chunfeng Wei et al. ‘Recognition of lower limb movements using empirical mode decomposition and k-nearest neighbor entropy estimator with surface electromyogram signals’. In: *Biomedical Signal Processing and Control* 71 (Jan. 2022), p. 103198. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2021.103198](https://doi.org/10.1016/J.BSPC.2021.103198).

- [17] Talon Garikayi. ‘Development of a robust myoelectric control architecture for lower limb robotic prostheses applications. | Semantic Scholar’. PhD thesis. Stellenbosch University, Dec. 2018. URL: <https://www.semanticscholar.org/paper/Development-of-a-robust-myoelectric-control-for-Garikayi/6545e1d3c1fafb758a80eb2ff010e356bc458771>.
- [18] Xinqi Louis Wang and J. Mikael Eklund. ‘Using Daubechies wavelet functions to generate masks for accurate QRS detection’. In: *Canadian Conference on Electrical and Computer Engineering* (June 2017). ISSN: 08407789. DOI: [10.1109/CCECE.2017.7946642](https://doi.org/10.1109/CCECE.2017.7946642).
- [19] ‘A guide for using the Wavelet Transform in Machine Learning – ML Fundamentals’. URL: <https://ataspinar.com/2018/12/21/a-guide-for-using-the-wavelet-transform-in-machine-learning/>.
- [20] Angkoon Phinyomark, Pornchai Phukpattaranont and Chusak Limsakul. ‘The Usefulness of Wavelet Transform to Reduce Noise in the SEMG Signal’. In: *EMG Methods for Evaluating Muscle and Nerve Function* (Jan. 2012). DOI: [10.5772/25757](https://doi.org/10.5772/25757). URL: <https://www.intechopen.com/chapters/25856>.
- [21] David L. Donoho and Iain M. Johnstone. ‘Ideal Spatial Adaptation by Wavelet Shrinkage’. In: *Biometrika* 81.3 (Aug. 1994), p. 425. ISSN: 00063444. DOI: [10.2307/2337118](https://doi.org/10.2307/2337118).
- [22] ‘Decomposing Signal Using Empirical Mode Decomposition — Algorithm Explanation for Dummy | by Muhammad Ryan | Towards Data Science’. URL: <https://towardsdatascience.com/decomposing-signal-using-empirical-mode-decomposition-algorithm-explanation-for-dummy-93a93304c541>.
- [23] Yannis Kopsinis and Stephen McLaughlin. ‘Development of EMD-based denoising methods inspired by wavelet thresholding’. In: *IEEE Transactions on Signal Processing* 57.4 (2009), pp. 1351–1362. ISSN: 1053587X. DOI: [10.1109/TSP.2009.2013885](https://doi.org/10.1109/TSP.2009.2013885).
- [24] Hassan Ashraf et al. ‘Threshold parameters selection for empirical mode decomposition-based emg signal denoising’. In: *Intelligent Automation and Soft Computing* 27.3 (2021), pp. 799–815. ISSN: 2326005X. DOI: [10.32604/IASC.2021.014765](https://doi.org/10.32604/IASC.2021.014765).
- [25] Dariusz Mika, Grzegorz Budzik and Jerzy Józwick. ‘Single channel source separation with ICA-based time-frequency decomposition’. In: *Sensors (Switzerland)* 20.7 (Apr. 2020). ISSN: 14248220. DOI: [10.3390/S20072019](https://doi.org/10.3390/S20072019).
- [26] Ganesh R. Naik and Dinesh K. Kumar. ‘Determining Number of Independent Sources in Undercomplete Mixture’. In: *EURASIP Journal on Advances in Signal Processing 2009 2009:1* 2009.1 (Sept. 2009), pp. 1–5. ISSN: 1687-6180. DOI: [10.1155/2009/694850](https://doi.org/10.1155/2009/694850). URL: <https://asp-urasipjournals.springeropen.com/articles/10.1155/2009/694850>.
- [27] Yongyu Jiang et al. ‘Shoulder muscle activation pattern recognition based on sEMG and machine learning algorithms’. In: *Computer Methods and Programs in Biomedicine* 197 (Dec. 2020), p. 105721. ISSN: 0169-2607. DOI: [10.1016/J.CMPB.2020.105721](https://doi.org/10.1016/J.CMPB.2020.105721).
- [28] Zhichuan Tang et al. ‘Effect of shoulder angle variation on sEMG-based elbow joint angle estimation’. In: *International Journal of Industrial Ergonomics* 68 (Nov. 2018), pp. 280–289. ISSN: 0169-8141. DOI: [10.1016/J.ERGON.2018.08.012](https://doi.org/10.1016/J.ERGON.2018.08.012).
- [29] Yanjiang Huang et al. ‘Motion estimation of elbow joint from sEMG using continuous wavelet transform and back propagation neural networks’. In: *Biomedical Signal Processing and Control* 68 (July 2021), p. 102657. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2021.102657](https://doi.org/10.1016/J.BSPC.2021.102657).
- [30] Longhai Lu et al. ‘Development of a sEMG-based torque estimation control strategy for a soft elbow exoskeleton’. In: *Robotics and Autonomous Systems* 111 (Jan. 2019), pp. 88–98. ISSN: 0921-8890. DOI: [10.1016/J.ROBOT.2018.10.017](https://doi.org/10.1016/J.ROBOT.2018.10.017).
- [31] José Jair A. Mendes Junior et al. ‘Feature selection and dimensionality reduction: An extensive comparison in hand gesture classification by sEMG in eight channels armband approach’. In: *Biomedical Signal Processing and Control* 59 (May 2020), p. 101920. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2020.101920](https://doi.org/10.1016/J.BSPC.2020.101920).
- [32] Naveen Kumar Karnam et al. ‘Classification of sEMG signals of hand gestures based on energy features’. In: *Biomedical Signal Processing and Control* 70 (Sept. 2021), p. 102948. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2021.102948](https://doi.org/10.1016/J.BSPC.2021.102948).

- [33] Xingjian Wang et al. 'sEMG-based consecutive estimation of human lower limb movement by using multi-branch neural network'. In: *Biomedical Signal Processing and Control* 68 (July 2021), p. 102781. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2021.102781](https://doi.org/10.1016/J.BSPC.2021.102781).
- [34] Feng Yan Liang, Fei Gao and Wei Hsin Liao. 'Synergy-based knee angle estimation using kinematics of thigh'. In: *Gait & Posture* 89 (Sept. 2021), pp. 25–30. ISSN: 0966-6362. DOI: [10.1016/J.GAITPOST.2021.06.015](https://doi.org/10.1016/J.GAITPOST.2021.06.015).
- [35] Andrej Karpathy. 'The Unreasonable Effectiveness of Recurrent Neural Networks'. URL: <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>.
- [36] Sepp Hochreiter and Jürgen Schmidhuber. 'Long short-term memory'. In: *Neural computation* 9.8 (1997), pp. 1735–1780.
- [37] Nitish Srivastava, Elman Mansimov and Ruslan Salakhutdinov. 'Unsupervised Learning of Video Representations using LSTMs'. In: *32nd International Conference on Machine Learning, ICML 2015* 1 (Feb. 2015), pp. 843–852. DOI: [10.48550/arxiv.1502.04681](https://doi.org/10.48550/arxiv.1502.04681). URL: <https://arxiv.org/abs/1502.04681v3>.
- [38] TJ Zebo et al. 'Myoelectric control of the Rancho electric arm'. In: *Proc of the 21st Annual Conference on Engineering in Medicine and Biology* November. 1968.
- [39] Chenfei Ma et al. 'Continuous estimation of upper limb joint angle from sEMG signals based on SCA-LSTM deep learning approach'. In: *Biomedical Signal Processing and Control* 61 (Aug. 2020), p. 102024. ISSN: 1746-8094. DOI: [10.1016/J.BSPC.2020.102024](https://doi.org/10.1016/J.BSPC.2020.102024).
- [40] Yee Mon Aung and Adel Al-Jumaily. 'sEMG Based ANN for Shoulder Angle Prediction'. In: *Procedia Engineering* 41 (Jan. 2012), pp. 1009–1015. ISSN: 1877-7058. DOI: [10.1016/J.PROENG.2012.07.276](https://doi.org/10.1016/J.PROENG.2012.07.276).
- [41] G. Cheron et al. 'A dynamic recurrent neural network for multiple muscles electromyographic mapping to elevation angles of the lower limb in human locomotion'. In: *Journal of Neuroscience Methods* 129.2 (Oct. 2003), pp. 95–104. ISSN: 0165-0270. DOI: [10.1016/S0165-0270\(03\)00167-5](https://doi.org/10.1016/S0165-0270(03)00167-5).
- [42] Geng Liu et al. 'sEMG-based continuous estimation of knee joint angle using deep learning with convolutional neural network'. In: *IEEE International Conference on Automation Science and Engineering* 2019-August (Aug. 2019), pp. 140–145. ISSN: 21618089. DOI: [10.1109/COASE.2019.8843168](https://doi.org/10.1109/COASE.2019.8843168).
- [43] Feng Zhang et al. 'sEMG-based continuous estimation of joint angles of human legs by using BP neural network'. In: *Neurocomputing* 78.1 (Feb. 2012), pp. 139–148. ISSN: 0925-2312. DOI: [10.1016/J.NEUCOM.2011.05.033](https://doi.org/10.1016/J.NEUCOM.2011.05.033).
- [44] Minh T.N. TRUONG et al. 'EMG-based Estimation of Knee Torque and Angle using Recurrent Neural Network'. In: *The Proceedings of JSME annual Conference on Robotics and Mechatronics (Robomec)* 2020.0 (2020), 1A1–F04. DOI: [10.1299/jsmermd.2020.1a1-f04](https://doi.org/10.1299/jsmermd.2020.1a1-f04). URL: <https://doi.org/10.1299/jsmermd.2020.1a1-f04>.
- [45] Angkoon Phinyomark et al. 'Feature extraction of the first difference of EMG time series for EMG pattern recognition'. In: *Computer Methods and Programs in Biomedicine* 117.2 (Nov. 2014), pp. 247–256. ISSN: 0169-2607. DOI: [10.1016/J.CMPB.2014.06.013](https://doi.org/10.1016/J.CMPB.2014.06.013).
- [46] Paul Kaufmann, Kevin Englehart and Marco Platzner. 'Fluctuating emg signals: investigating long-term effects of pattern matching algorithms'. In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference* 2010 (2010), pp. 6357–6360. ISSN: 2375-7477. DOI: [10.1109/IEMBS.2010.5627288](https://doi.org/10.1109/IEMBS.2010.5627288). URL: <https://pubmed.ncbi.nlm.nih.gov/21096692/>.
- [47] Sebastian Amsuss et al. 'Long term stability of surface EMG pattern classification for prosthetic control'. In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2013), pp. 3622–3625. ISSN: 1557170X. DOI: [10.1109/EMBC.2013.6610327](https://doi.org/10.1109/EMBC.2013.6610327).

- [48] Jiayuan He et al. 'Effects of Long-Term Myoelectric Signals on Pattern Recognition'. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8102 LNAI.PART 1 (2013), pp. 396–404. ISSN: 16113349. DOI: [10.1007/978-3-642-40852-6_40](https://doi.org/10.1007/978-3-642-40852-6_40). URL: https://link.springer.com/chapter/10.1007/978-3-642-40852-6_40.
- [49] Richardson N. Leao and John A. Burne. 'Continuous wavelet transform in the evaluation of stretch reflex responses from surface EMG'. In: *Journal of Neuroscience Methods* 133.1-2 (Feb. 2004), pp. 115–125. ISSN: 0165-0270. DOI: [10.1016/J.JNEUMETH.2003.10.003](https://doi.org/10.1016/J.JNEUMETH.2003.10.003).