TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

Electromyography control of robotic systems

Master Thesis

Study programme: N0714A150003 Mechatronics
Author: **Bc. Denis Griaznov**
Thesis Supervisors: Ing. Jan Koprnický, Ph.D.
Institute of Mechatronics and Computer Engineering

Liberec 2020

TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

Master Thesis Assignment Form

Electromyography control of robotic systems

Name and surname: **Bc. Denis Griaznov**
Identification number: M19000191
Study programme: N0714A150003 Mechatronics
Assigning department: Institute of Mechatronics and Computer Engineering
Academic year: **2019/2020**

Rules for Elaboration:

1. Analyze Myo Armband system
2. Investigate prospective application fields of the Myo Armband system
3. Design Matlab model of control system with using of Myo Armband
4. Test Myo Armband for control of Arduino-based robotic hand

Scope of Graphic Work:	by appropriate documentation
Scope of Report:	40-50 pages
Thesis Form:	printed/electronic
Thesis Language:	English

List of Specialised Literature:

[1] HASSAN, Hussein F., Sadiq J. ABOU-LOUKH a Ibraheem Kasim IBRAHEEM. Teleoperated robotic arm movement using electromyography signal with wearable Myo armband. Journal of King Saud University – Engineering Sciences. 2019. DOI: 10.1016/j.jksues.2019.05.001. ISSN 1018-3639. Available from: http://www.sciencedirect.com/science/article/pii/S1018363919300340

[2] VISCONTI, P., F. GAETANI, G. A. ZAPPATORE a P. PRIMICERI. Technical Features and Functionalities of Myo Armband: An Overview on Related Literature and Advanced Applications of Myoelectric Armbands Mainly Focused on Arm Prostheses. International Journal on Smart Sensing and Intelligent Systems. 2018, 11(1), 1?25. DOI: 10.21307/ijssis-2018-005. ISSN 1178-5608. Available from: https://www.exeley.com/in_jour_smart_sensing_and_intelligent_systems/doi/10.21307/ijssis-2018-005

[3] GERYES, M., J. CHARARA, A. SKAIKY, A. MCHEICK a J. GIRAULT. A novel biomedical application for the Myo gesture control armband. In: 2017 29th International Conference on Microelectronics (ICM). 2017, p. 1-4. DOI: 10.1109/ICM.2017.8268823.

[4] MORAIS, Gabriel Doretto, Leonardo C. NEVES, Andrey A. MASIERO a Maria Claudia F. CASTRO. Application of Myo Armband System to Control a Robot Interface:. In: 9th International Conference on Bio-inspired Systems and Signal Processing. SCITEPRESS – Science and and Technology Publications, 2016, p. 227-231. DOI: 10.5220/0005706302270231. ISBN 978-989-758-170-0. Available from: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0005706302270231

[5] ABDUO, Mahmoud a Matthias GALSTER. Myo Gesture Control Armband for Medical Applications. Engineering: Reports. University of Canterbury, 2015, 2015(693), 1-29. Available from: https://ir.canterbury.ac.nz/handle/10092/14449

[6] MUZUMDAR, Ashok. Powered Upper Limb Prostheses: Control, Implementation and Clinical Application. 1. Berlin Heidelberg: Springer-Verlag, 2004. ISBN 978-3-642-62302-8.

Thesis Supervisors:	Ing. Jan Koprnický, Ph.D. Institute of Mechatronics and Computer Engineering
Date of Thesis Assignment:	November 8, 2019
Date of Thesis Submission:	May 18, 2020

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
Dean

doc. Ing. Milan Kolář, CSc.
head of institute

Liberec November 8, 2019

Declaration

I hereby certify, I, myself, have written my master thesis as an original and primary work using the literature listed below and consulting it with my thesis supervisor and my thesis counsellor.

I acknowledge that my bachelor master thesis is fully governed by Act No. 121/2000 Coll., the Copyright Act, in particular Article 60 – School Work.

I acknowledge that the Technical University of Liberec does not infringe my copyrights by using my master thesis for internal purposes of the Technical University of Liberec.

I am aware of my obligation to inform the Technical University of Liberec on having used or granted license to use the results of my master thesis; in such a case the Technical University of Liberec may require reimbursement of the costs incurred for creating the result up to their actual amount.

At the same time, I honestly declare that the text of the printed version of my master thesis is identical with the text of the electronic version uploaded into the IS/STAG.

I acknowledge that the Technical University of Liberec will make my master thesis public in accordance with paragraph 47b of Act No. 111/1998 Coll., on Higher Education Institutions and on Amendment to Other Acts (the Higher Education Act), as amended.

I am aware of the consequences which may under the Higher Education Act result from a breach of this declaration.

May 29, 2020 Bc. Denis Griaznov

Electromyography control of robotic systems

Abstract

This work describes research of myoelectric interfaces and their application for controlling robotic systems. Hand gesture data collection software has been created. The neural network was designed and trained to recognize various gestures. The accuracy was 0.96 for four gestures and 0.925 for seven gestures. The prototype of myoelectric signals controlled robot with two degrees of freedom was created. Wireless direct control via bluetooth was implemented.

Keywords: electromyography, Myo Armband, robotic control, neural networks

Acknowledgements

I want to express my gratitude to my supervisor, dr. Jan Koprnický, for providing all the necessary conditions for comfortable scientific work, providing technical support, as well as for constant assistance in all matters.

I would also like to thank prof. Jaroslav Nosek for the development of the double degree program, thanks to which I was able to study at this university, as well as for help with adaptation and administrative matters.

Contents

List of abbreviations

CNN	Convolutional neural network
EMG	Electromyography
ID	Identifier
IMU	Inertial measurement unit
IoT	Internet of things
k-NN	k-nearest neighbors algorithm
LSTM	Long short-term memory
MU	Myo units (abstract units in which Myo Armband transmits voltage)
MF	Myo frequency (200 Hz, frequency in which Myo Armband record samples)
ML	Machine learning
NN	Neural network
PCA	Principal component analysis
PC	Personal computer
RNN	Recurrent neural network
SVM	Support vector machine
SDK	Software development kit

1 Introduction

1.1 Overview

Currently, there are many ways to control robotic systems using various input devices. Most of them are based on mechanical input through human fingers. However, in some situations, the optimal solution would be to avoid direct mechanical control in favor of some alternatives.

In this work, it is proposed to use myoelectric sensors as alternative input devices. Research motivation is based on two factors.

First, such systems have proven their applicability in important areas. To date, myoelectric sensors have been successfully used in hand prosthetics [1, 2, 3]. Figure 1.1 shows such example. There may also use in situations where the arm should be locked. For example, a surgical operation, when the doctor must hold the instrument with his fingers, but at the same time, with the movement of the hand, he can control the position of the camera or other instruments showing the patient's condition [4].

Secondly, modern technology allows to develop a similar system and achieve better performance. Largely due to advances in machine learning and neural networks.

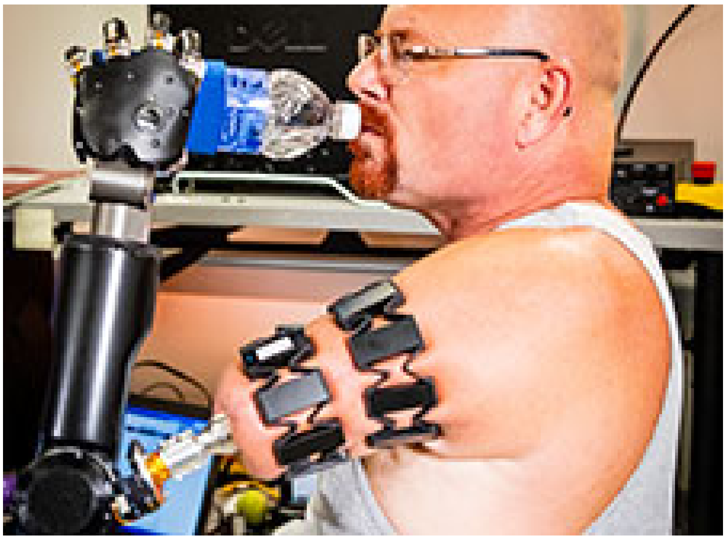


Figure 1.1: Example of using EMG control in hand prosthetics. Image is from [3].

1.1.1 Main tasks

The ultimate goal of the work is the creation of a robotic system that will be controlled by EMG signals using Myo Armband. In the process of achieving it, several other tasks must be completed. This is a study of the Myo Armband system and obtaining EMG data, an analysis of the possible applications of this system, as well as the development of a control model. During the development of the control model, it was decided to replace the Matlab environment with a combination of other programming environments.

The work was carried out in the environments of Jupyter Notebook (Python), Unity 3D (C#) and Arduino IDE (C). This decision was made for several reasons. First, Thalmic Lab (Myo manufacturing company) does not provide an official Matlab SDK. Secondly, it was decided to use ML and neural networks as the basis of the model. Python provides much more features and a wide international community. Thirdly, wireless control was implemented and for this it was necessary to compile the ML model into C code. This is much more convenient to do in selected development environments.

1.1.2 Thesis structure

The thesis report consists of several parts.

The following sections of the Introduction chapter describe the basic devices and phenomena that were used in the study. The Research part describes the methods of data analysis as well as the results of their application to the EMG signals obtained using Myo Armband. The chapter also describes machine learning methods and the models used. Further, in the Technical part, software development, model training, and a controlled robot circuit are described.

The Results chapter provide an analysis of the study and offer examples of the use of technology. The Conclusion chapter describes the overall results of each of the tasks.

1.2 Myoelectrical signals

Electromyography is a medical research method based on recording the electrical activity of human muscles. Electromyography can be divided into surface and intramuscular. Surface EMG records electrical signal from skin surface and allows only limited information about the muscle activity. Intramuscular EMG gives greater accuracy. But the implementation of this method is more complicated and requires the participation of professional doctors.

Further in this work only surface EMG will be borne in mind. Despite the less informative, this method is very affordable. Data collection is carried out using electrodes attached to the skin.

A raw surface EMG signal has the range of voltage 0–2 mV, and the range of frequency 0–1000 Hz.

EMG is widely used for the diagnosis of various diseases associated with impaired muscle activity [5].

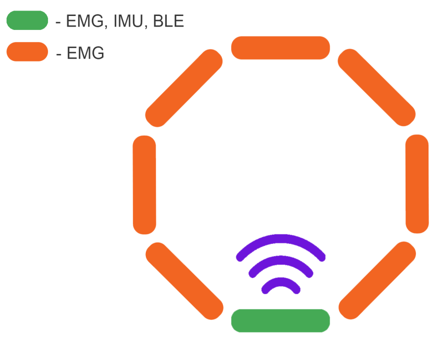
1.3 Myo Armband

Myo Armband was chosen as a device with EMG sensors. It is a bracelet with eight links. Each of them has a medical grade stainless steel EMG sensors. In addition, the bracelet is equipped with accelerometer, magnetometer and gyroscope. For wireless data transmission there is a Bluetooth Low Energy module.

Standard software allows to recognize five different hand gestures. It is possible also to get raw data from all sensors. The frequency of receiving EMG data is 200 Hz, for IMU data it is 50 Hz.

It has SDKs for popular operating systems on both computers and mobile devices (Windows, macOS, iOS, Android). Myo Armband production has officially ended as of Oct 12, 2018 and is no longer available for purchase. The view of Myo Armband is in Figure 1.2a and the link diagram is in Figure 1.2b.

(a) Photo of Myo Armband. Image is from [6].



(b) Scheme of Myo Armband.

Figure 1.2: Myo Armband.

There are other analogues of Myo Armband on the market. For example, gForce PRO is a similar bracelet with myoelectric sensors, but the cost is much more expensive [7]. Also on sale are separate EMG sensors that are not connected to the infrastructure (for example, sensors from MyoWare [8]). In terms of price and convenience, Myo Armband is the optimal solution for this research. But all the results can be easily repeated on other similar devices.

Myo Armband transmit signal with own units. It can be called Myo Units (MU). These are values in the *uint_8* format. There is no information on how they are converted to volts.

An example of the use of Myo Armband is the Adora project, which allows doctors to view patient information without assistants [4] (Figure 1.3).

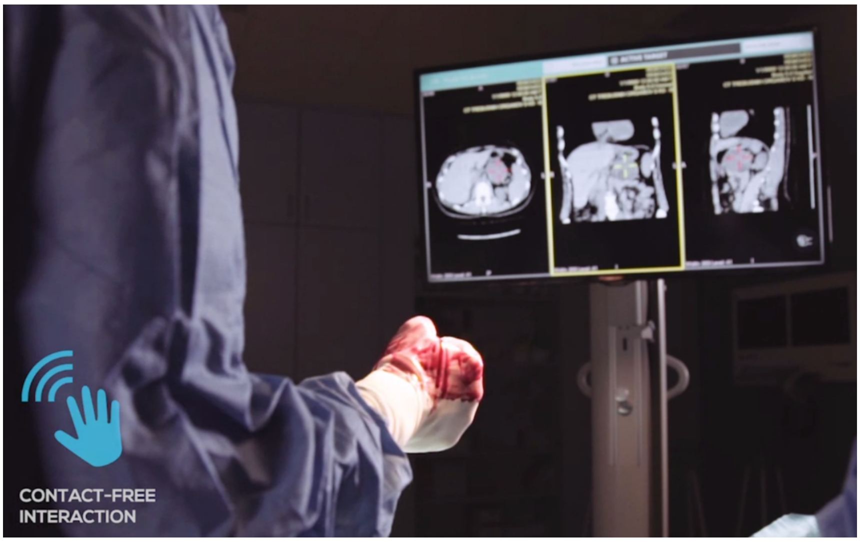


Figure 1.3: Example of using Myo Armband in surgery. Image is from [4].

Table 1.1: Myo Armband specifications summary

Size	Expandable between 19–34 centimetres
Weight	93 grams
Thickness	1.14 centimetres
Compatible devices	Windows, Mac OS X, macOS, iOS, Android
Sensors	EMG, IMU
Processor	ARM Cortex M4 processor
Communication	BLE
Charging	Micro-USB
Battery	Lithium-Ion battery
Performance	One full day of use in a single charge

1.4 Software and hardware review

In the work *C#* and Unity3D were used to create PC software. Unity3D is a popular multi-platform environment for creating games and applications. It has a convenient interface and allows quickly to try visual physical interactions.

It was used to collect data and visualize the results of a deep neural network before working with a microcontroller.

Date analysis and machine learning were done in Python in the Jupyter Notebook. To create and train the model, I used the Keras library with TensorFlow as a backend. This is a very powerful neural network solution that allows to work with LSTM, CNN, and other types of NN architectures. Also in the Jupiter Notebook provides the ability to build graphs. I have used many of them in this work.

I used MicroServo gg SG90 servomotor (Figure 1.5) and WeMos D1 Mini microcontroller (Figure 1.4) with the following specifications:

- Chip: ESP-8266EX
- GPIO pins: 11
- 1 pin ADC (0 V to 3.3 V)
- Max Input Voltage: 24 V
- Operating Frequency: 80/160 MHz
- Flash Memory: 4 MB
- Size: 34.2 × 25.6 mm

Also it has Wi-Fi and Bluetooth Low Energy modules for data transmitting.

Working with C code, writing libraries, and flashing the microcontroller was done through the Arduino IDE.

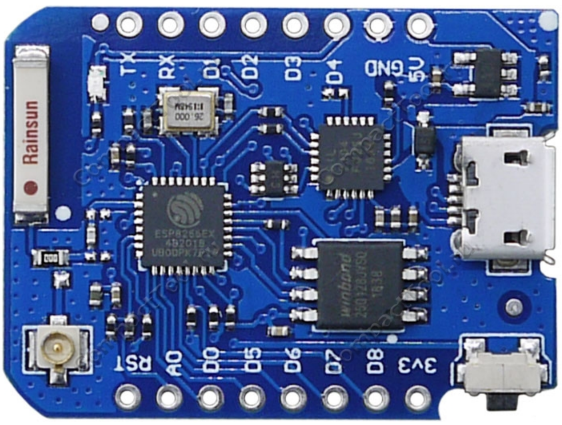


Figure 1.4: WeMos D1 Mini. Image is from [9].

Figure 1.5: MicroServo 9g SG90. Image is from [9].

2 Research part

2.1 Data structure

The data structure in experiments related to EMG is similar to other biological signals. Moreover, for the same gesture, the data can vary significantly. So, it's very difficult to build an ordinary mathematical model and understand significant features [10]. But machine learning models are well suited for such tasks. There are examples of the application of various ML methods for the task of classifying EMG signals [11, 12, 13]. Therefore, machine learning was used in this work.

2.2 Machine learning methods

The main problem of creating a myoelectric control system is gesture recognition. In primitive cases, it is possible to create a binary system by determining the threshold value of the total voltage. So when the tension of the hand control changes. However, for more complex control, it is necessary to recognize different gestures.

Below the main types of used ML models are described.

2.2.1 Artificial neural networks

Artificial neural networks (or simply Neural networks, NN) are computer systems based on biological neural networks that make up the brain of animals. Such systems can perform tasks, studying examples, as a rule, without programming and using rules for specific tasks. A neural network usually has at least two layers of artificial neurons: an input and an output. NN may have inner layers between input and output. This type of NN is called Deep Neural Network. Each artificial neuron in the layer connects to all neurons in the next layer. The neuron has an activation function that decides which value will go next, depending on the sum of the previous values. For example sigmoid function:

$$\phi(x) = \frac{1}{1 + e^{-x}} \tag{2.1}$$

Each connection (or synapse) has its own weight – a multiplier for transmitting values from one neuron to another. In fact, a neural network is a collection of such weights and can be described by a matrix of weights. When the input passes through the network, the weights are updated depending on the accuracy of the output (using gradient descent). One epoch of fitting ended when the entire dataset is passed through the neural network.

2.2.2 Long-term short-term memory

Long-term short-term memory (LSTM) is a form of neural network architecture that is great for time series analysis [16, 14, 15]. The network consists of LSTM modules. The LSTM module is a recurrent network module that can store values for both short and long periods of time. LSTM networks do a good job with speech recognition, classification of biological signals, recognition of human activity. LSTM is a good choice for recognizing hand gestures through EMG, with condition of using a sufficient time window.

2.2.3 Standard algorithms

The support vector machine (SVM) is a set of similar ML algorithms for teaching with a teacher [17]. It is widely used for classification and regression analysis. The main idea of the method is to increase the dimension and find the dividing hyperplane with the largest average gap.

The K-nearest neighbors (k-NN) algorithm is a metric algorithm for automatically classifying objects or regressing [18]. In the case of classification, the object is assigned the class most often found among the nearest neighbors.

2.3 Data analysis methods

Principal Component Analysis (PCA) is one of the main ways to reduce data dimension with minimal loss of information [19]. The method is used in machine learning to remove unnecessary functions and visualize high-dimensional spaces.

The PCA task has at least four basic versions, depending on the goal. In our case, the problem is reduced to finding a subspace of smaller dimension, in the orthogonal projection onto which the RMS distance between points (labeled classes) is maximum. As an example, there is case of a two-dimensional distribution with two different classes (Figure 2.1). It is needed to find a straight line so that after constructing the projections of all the points on it, the classes are spaced apart as much as possible. That is, to satisfy the condition:

$$\sum_{ij}(a_i - b_j) \Rightarrow max \tag{2.2}$$

where a and b are sets of point projections of first and second classes. Using the PCA, finding the most similar gestures could be visually.

Visualization of correlation helps to better understand how sensors work (Figure 2.2). The first graph (left graph in the Figure 2.2) was built on the basis of correlations of all EMG data for one Wave Out gesture. The gesture sample includes 90 measurements for each of 8 sensors. From the graph, it is probably to conclude that the same sensors are uniformly amplified and attenuated with a certain gesture. Next, a correlation between the sum of the signals from each sensor was built. The second graph (right graph in the Figure 2.2) shows smooth and isotropic transitions between the sensors.

The distribution of the sensor values also makes it possible to verify their suitability. For this, the sum of the values for all gestures was taken (Figure 2.3). The distributions were similar for all 8 sensors. The graph in Figure 2.3 shows that two different sensors the distribution is similar to Gaussian.

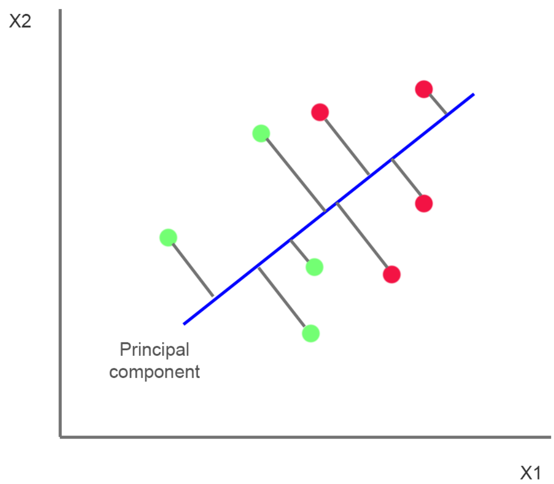


Figure 2.1: Example of PCA classification

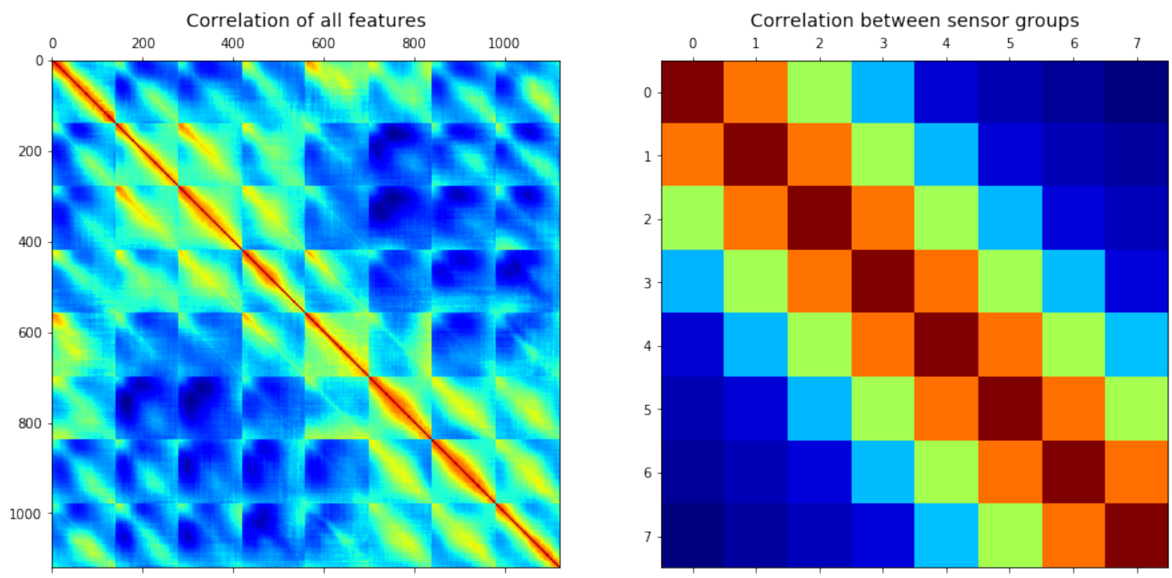


Figure 2.2: Correlation between sensors with Wave Out gesture on 90-points time window

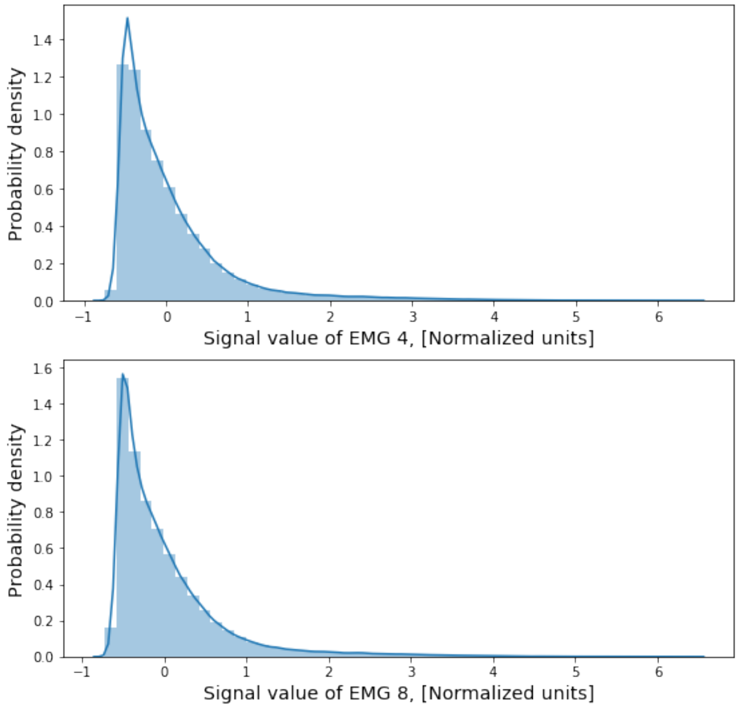


Figure 2.3: Examples of distribution of EMG values sum for all gestures dataset from different sensors

2.4 Dataset preparing

For better results, the data needs preprocessing. Original EMG signal has fast large differences of values (the Figure 2.4 shows raw data). At first absolute data values were taken. At second for its approximation Savitzkiy-Goley filter with 9 points and 2nd polynom order has been selected [20].

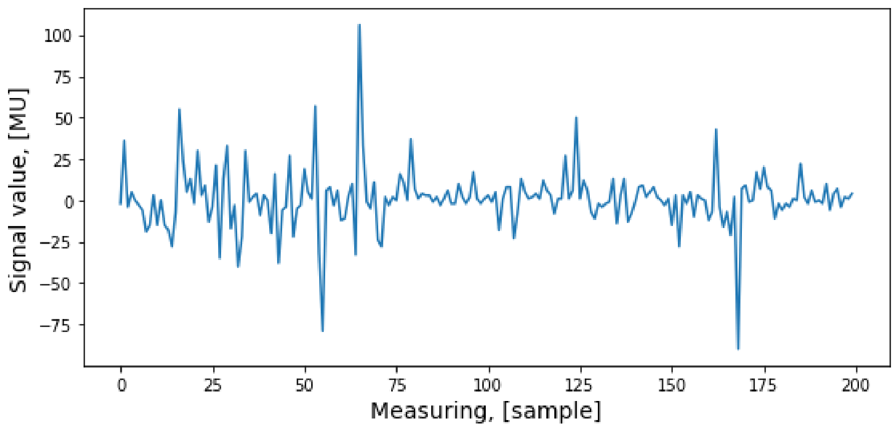


Figure 2.4: Example of raw EMG data.

Savitzky-Golay filtering is to perform a least squares fit of a set of consecutive data points to a polynomial and take the obtained middle point of the fitted polynomial curve as the new smoothed data point. Each next point y_k is calculated as:

$$y_k = \frac{\sum_{i=-n}^{n} y_{k+i} A_i}{\sum_{i=-n}^{n} A_i} \tag{2.3}$$

Where A_i are pre-calculated coefficients that are different for different smoothing orders. Depending on the required accuracy, number of points can be selected n. The Figure 2.5 shows the results of second-order filtering for a different number of points.

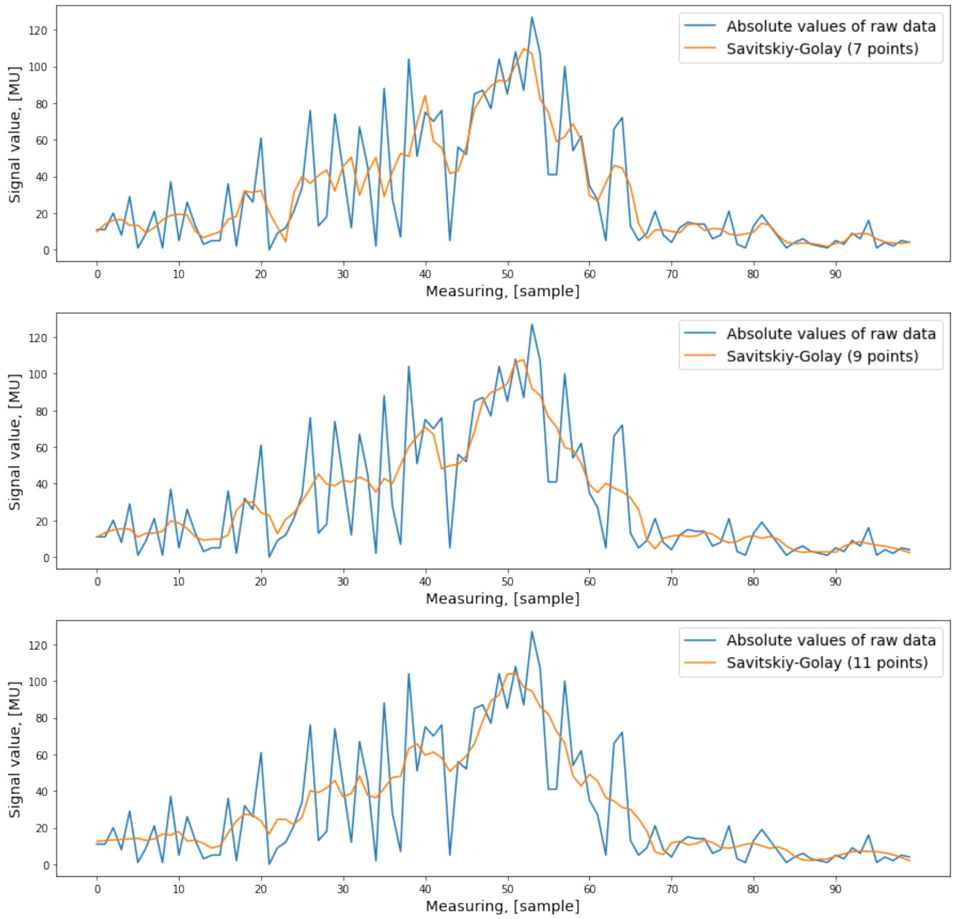


Figure 2.5: Savitzky-Golay smoothing results for different points numbers.

It was concluded that the option with 9 points is optimal since filtering with 7 points does not remove many interference, and filtering with 11 points does not retain significant signal features, which can be seen in the Figure 2.5.

But with comparison the results of this method and the moving average and noticed that they correlate, and for a different number of points (Figure 2.6). Consequently, the moving average method for our data will give satisfactory results at lower computational costs.

In result, I chose Moving-Average filter with 5 points because it acts on the raw data in much the same way as the selected Savitzkiy-Goley filter with 9 points (Figure 2.6 shows this), but requires less computation.

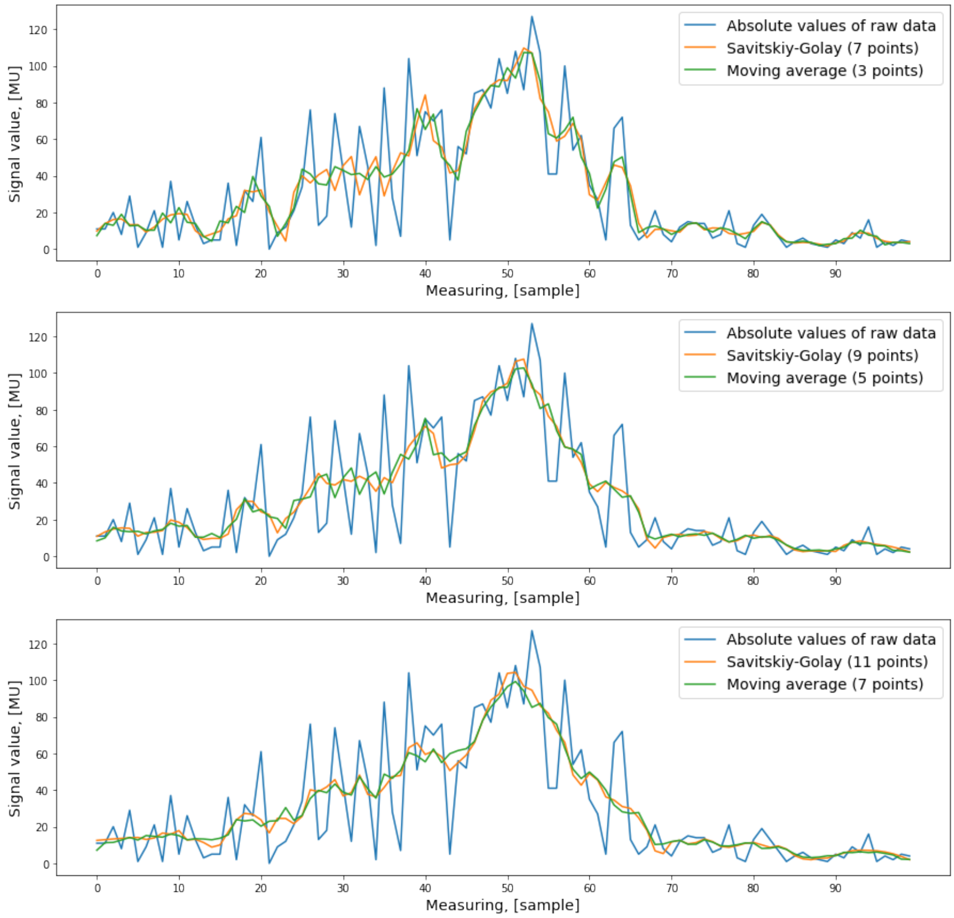


Figure 2.6: Comparison Savitzky-Golay and Moving-Average filtering.

Different people may have different powers of EMG arm signal. It is used normalized data for better fitting our model and decreasing of this error. It is necessary to normalize all array (8 EMG signals together) for save relationships between signal magnitude.

$$x_{mean} = \frac{\sum_{i=0}^{n} x_i}{n} \tag{2.4}$$

$$\sigma = \sqrt{\frac{\sum_{i=0}^{n} x_i^2}{n}} \tag{2.5}$$

$$x_{norm} = \frac{x - x_{mean}}{\sigma} \tag{2.6}$$

Where x and x_{norm} are original and normalized arrays respectively.

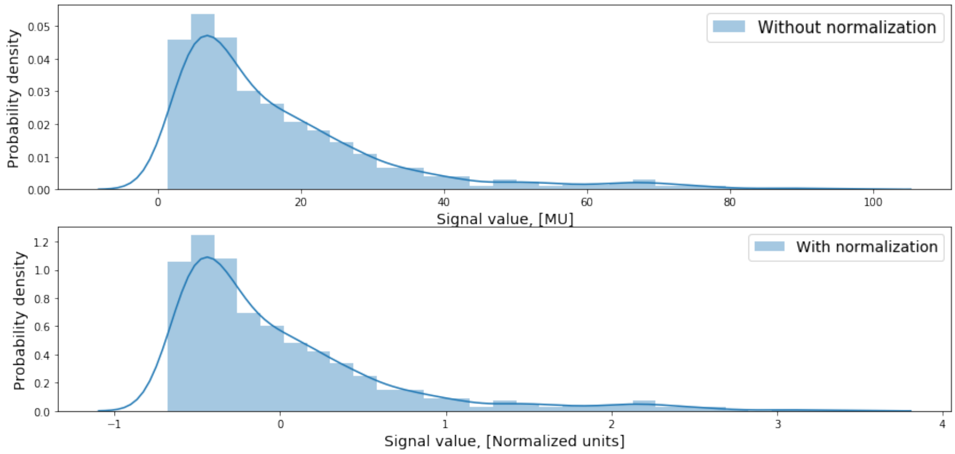


Figure 2.7: Values distributions for original and normalized signals.

2.5 Gesture selection

Data for 7 different gestures was collected through Arduino. Figure 2.8 shows their images and names. It is necessary to select 4 gestures from 7 to control the robot.

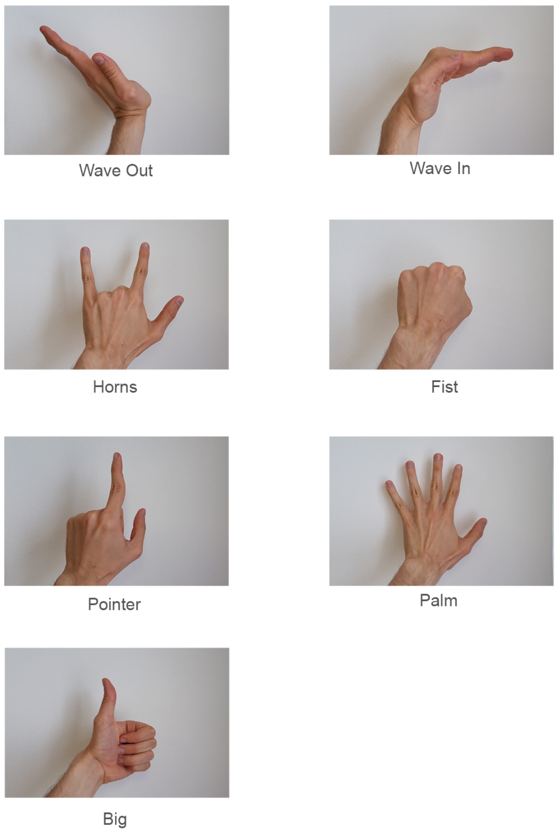


Figure 2.8: Images and names of used gestures.

It is important to choose the right gestures for training the model. The choice should be based on recognition accuracy. It depends on how understandable the control is with the selected set of gestures and how the gestures differ. It is necessary to get some metrics that could describe such characteristics mathematically. To do this, it is possible to calculate the average distance between each pair of classes. It is also possible to use PCA to visualize classes.

The average distances between each two gestures were calculated. The result can be seen in Table 2.1. Based on the obtained results and empirical observations, gestures Fist, Wave In and Wave Out are basic. Then charts with a decrease in dimension to two components were built for each of the remaining gestures (Figure 2.9).

After analyzing Table 2.1 and graphs from Figure 2.9, Pointer was chosen as the fourth gesture.

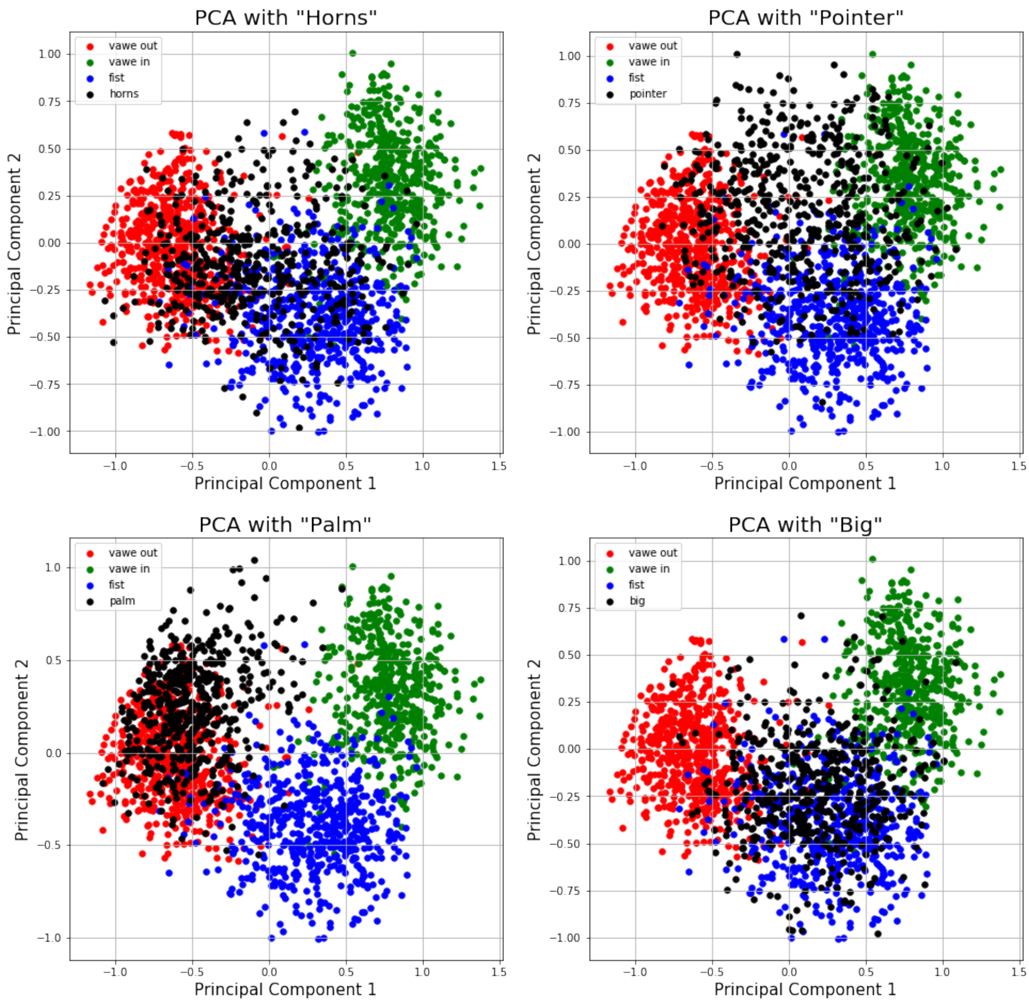


Figure 2.9: Result of PCA for four gestures.

Table 2.1: Average distances between each two classes

	Vawe out	Vawe in	Horns	Fist	Palm	Pointer	Big
Vawe out	0.000	1.450	0.719	1.000	0.403	0.844	0.919
Vawe in	1.450	0.000	1.087	0.966	1.336	0.889	0.945
Horns	0.719	1.087	0.000	0.700	0.634	0.421	0.485
Fist	1.000	0.966	0.700	0.000	1.043	0.727	0.439
Palm	0.403	1.336	0.634	1.043	0.000	0.625	0.840
Pointer	0.844	0.889	0.421	0.727	0.625	0.000	0.498
Big	0.919	0.945	0.485	0.439	0.840	0.498	0.000

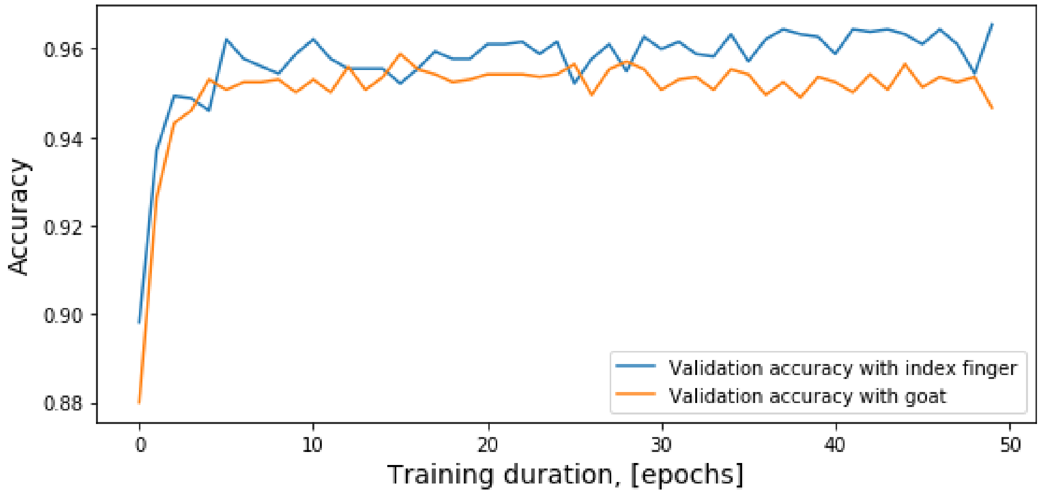


Figure 2.10: Comparison of fitting model with different fourth gesture.

2.6 Gesture detection and recognition

Two different ways to detect and recognize gestures were used in a PC program and in a microcontroller. In the PC software, there was more computing power, and it was used for a more detailed analysis of various gestures. Also the big ML model was used.

The first method (for PCs) is gesture detection with a large time window (about 700 ms). There is a sequence of 140 values for each sensor with a transmission frequency of 200 Hz. This provides high recognition accuracy, but makes the process very slow. Also, this method provides only individual gestures and is useless for constant control. But with setting connection between the gesture and the movement of the robot (even complex movement), it can be useful.

For each time moment, the program calculates the sum of 140 previous absolute values of EMG, as well as the derivative of the curve of this sum. To detect the gesture, I used two conditions: the equality of this derivative and zero and the threshold of the absolute value EMG. Thus, the software captures the window at the final moment. And the entire gesture curve is in the window. The recognition model uses all 1120 values (140 for each of 8 sensors).

The second method (for the microcontroller) is simpler. Only 16 values for each forecast were collected. This provides quick recognition. In fact, the gesture was recognized with the same frequency with which Myo Armband transmits new data (25 Hz). To detect a gesture, the simplest method is also used: threshold for the sum of absolute values.

This method can be used for continuous control of a robotic system, when each gesture is associated with one direction of movement.

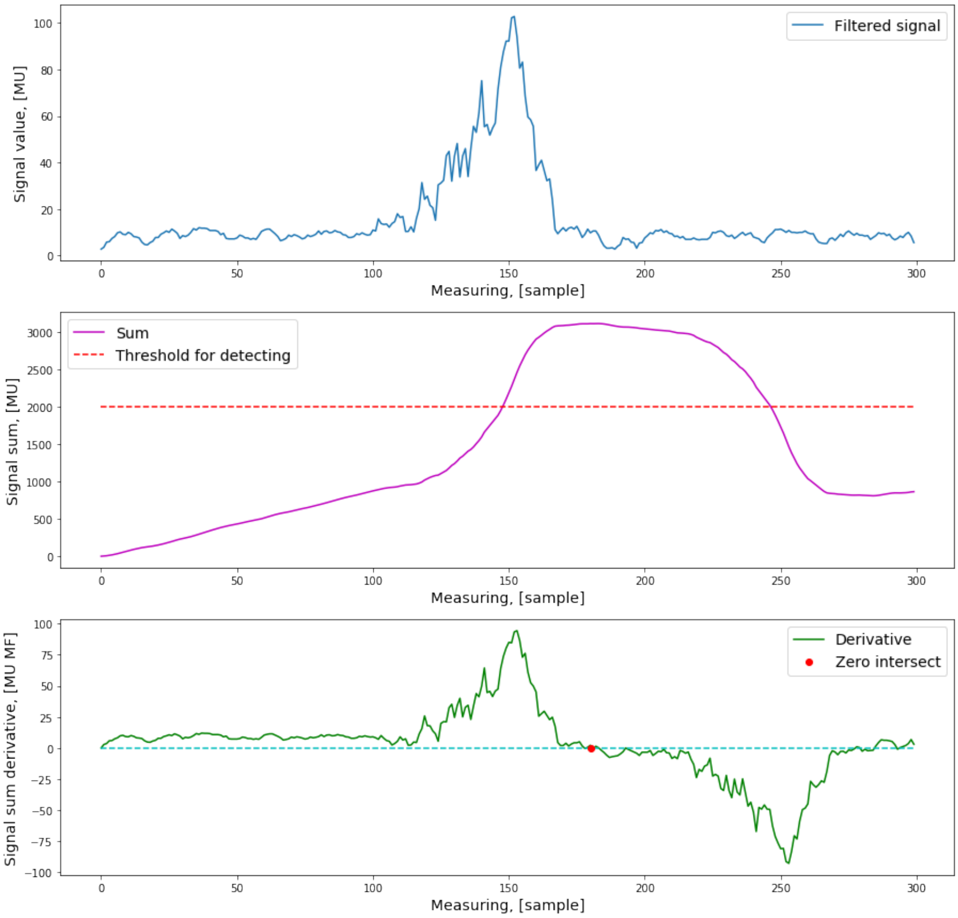


Figure 2.11: Signal, sum function and sum derivative are used to recognize gesture position.

2.7 Models architecture

First, for PC software, the deep recurrent neural network model was used, including two internal LSTM layers of 50 units and one internal normal layer of 64 neurons (Figure 2.12). All 140 points for 8 arrays were input. To avoid overfitting, a dropout of 0.2 was used on the LSTM layers. This means that every epoch, 0.2 of randomly selected layer neurons are disabled.

Next, a much simpler model was tested. At the input, it was given 8 values (the average for each sensor), and the inner layer was only one and contained 16 neurons (Figure 2.13). On the first two layers, a dropout of 0.3 was set. The weight of the model is much less (55 kilobytes versus 13 megabytes in the first one).

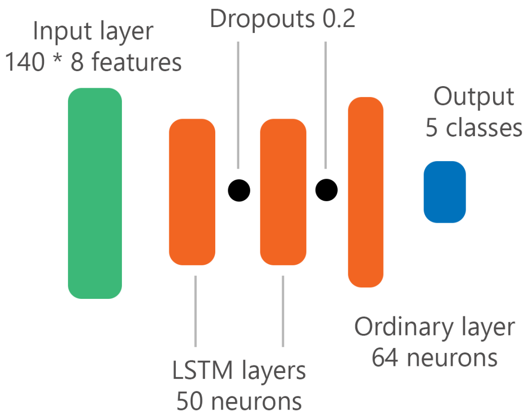


Figure 2.12: Scheme of LSTM-based neural network for PC.

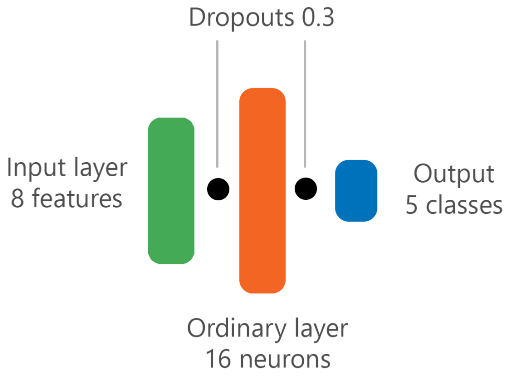


Figure 2.13: Scheme of simple neural network for PC.

For the microcontroller, I tested models without LSTM layers. It was a model with two inner layers (16 and 32 neurons). Both layers have 0.2 dropout (Figure 2.14). The input layer has 16 inputs, and in total the model has 1047 parameters. It is small enough to run the network on tiny devices like the WeMos D1 mini.

This NN is used to recognize 7 gestures. For a simpler task (recognition of 4 gestures for controlling the robot) an even simpler model was used (Figure 2.15). This is a model with two inner layers: each has 16 neurons. Both layers have a value of 0.2 in dropout too.

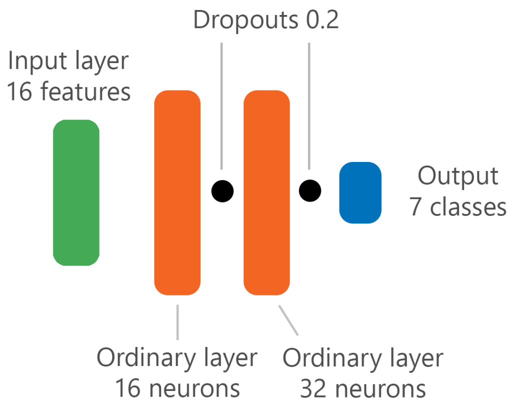


Figure 2.14: Scheme of neural network for Arduino with 7 gestures.

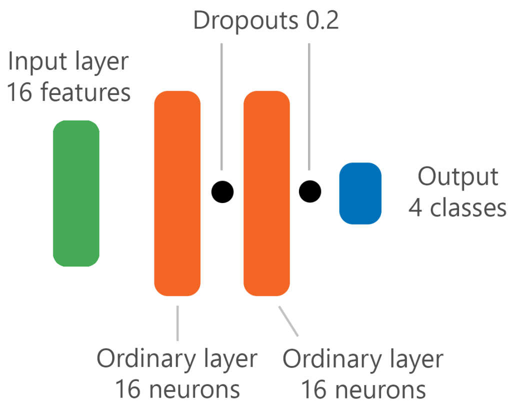


Figure 2.15: Scheme of neural network for Arduino with 4 gestures.

In addition, it is necessary to compare obtained results with the usual machine learning methods. It is SVM (Support Vector Classification) with the following parameters:

- Regularization parameter: 1.0
- Kernel: Radial basis function

and k-NN with the following parameters:

- Number of neighbours: 3
- Metric: Minkowski distance
- Power parameter for the Minkowski metric: 2

3 Technical part

3.1 Models fitting

The LSTM-based model used a dataset with 2012 samples divided into 5 classes. The validation part was 0.2 of all samples. The model corresponded to 30 train epochs. It showed an accuracy of about 0.98 on test samples. Fitting process is shown in Figure 3.1.

Then I increase the number of epochs for lighter models. The PC model without LSTM shows an accuracy of about 0.975 in the same dataset after 150 epochs (Figure 3.2). But the second model lighter very much (189 parameters versus versus 111000 in LSTM one).

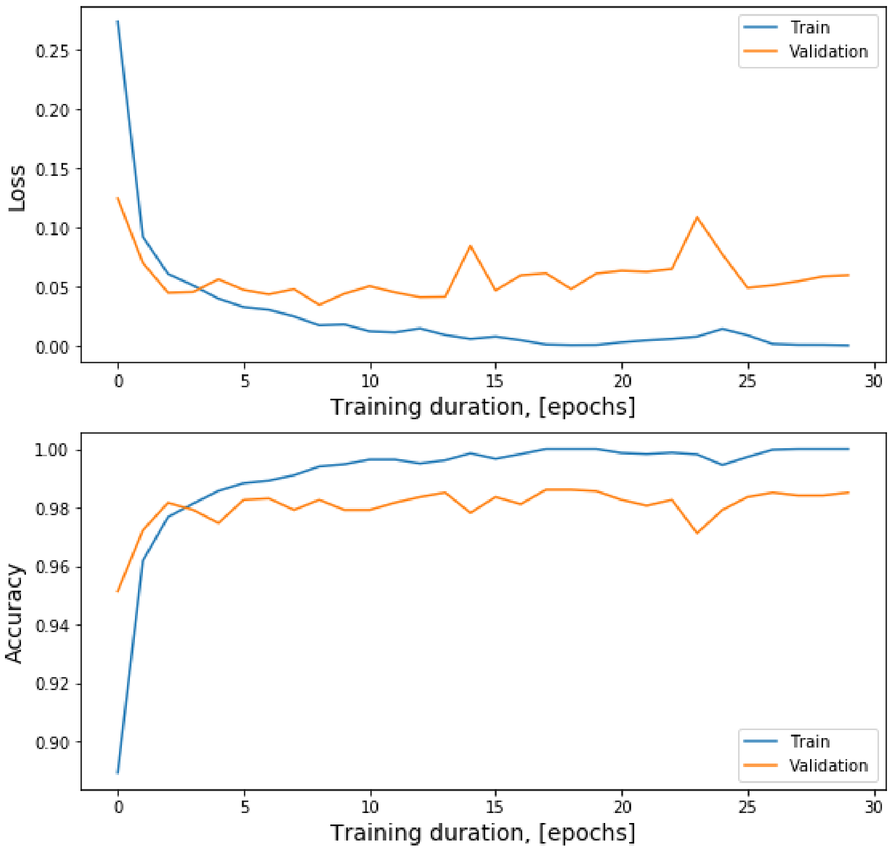


Figure 3.1: Fitting history of LSTM-based model.

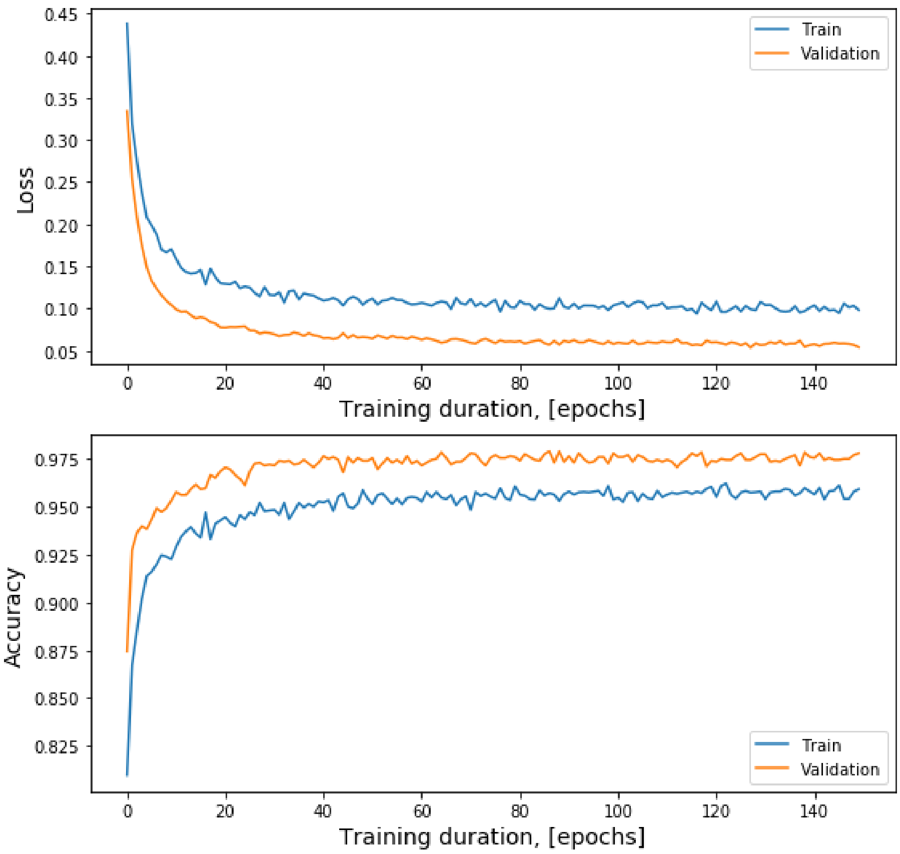


Figure 3.2: Fitting history of simple PC model.

With 5 gestures, the light model is more profitable, but with much more such a simple architecture will not provide the necessary feature extraction. The model based on recurrent neural networks, on the contrary, has great potential, and with increasing gestures it will not lose accuracy.

Then the models for the microcontroller was fitted. The model for 7 gestures recognition used 3713 samples for training. After 50 epochs it takes 0.9265 accuracy. The model for robotic control, which could recognize 4 gestures, had accuracy of about 0.96 after 50 epochs. The validation part was 0.2 for the all experiments. In addition, our hypothesis about gesture selection using PCA was tested. The same model using the Horns gesture instead of the Pointer gave a worse result (0.95).

Table 3.1 shows comparing different models.

Table 3.1: ML models comparison

Model	Val. accuracy	Val. loss	Dataset size [samples]
PC LSTM	0.9831	0.0926	2012
PC not-LSTM	0.9752	0.0607	2012
Arduino (7 gestures)	0.9265	0.1791	3713
Arduino (with pointer)	0.9594	0.1249	2241
Arduino (with horns)	0.9501	0.1411	2155
SVC	0.911	0.1935	2155
K-NN	0.909	0.2044	2155

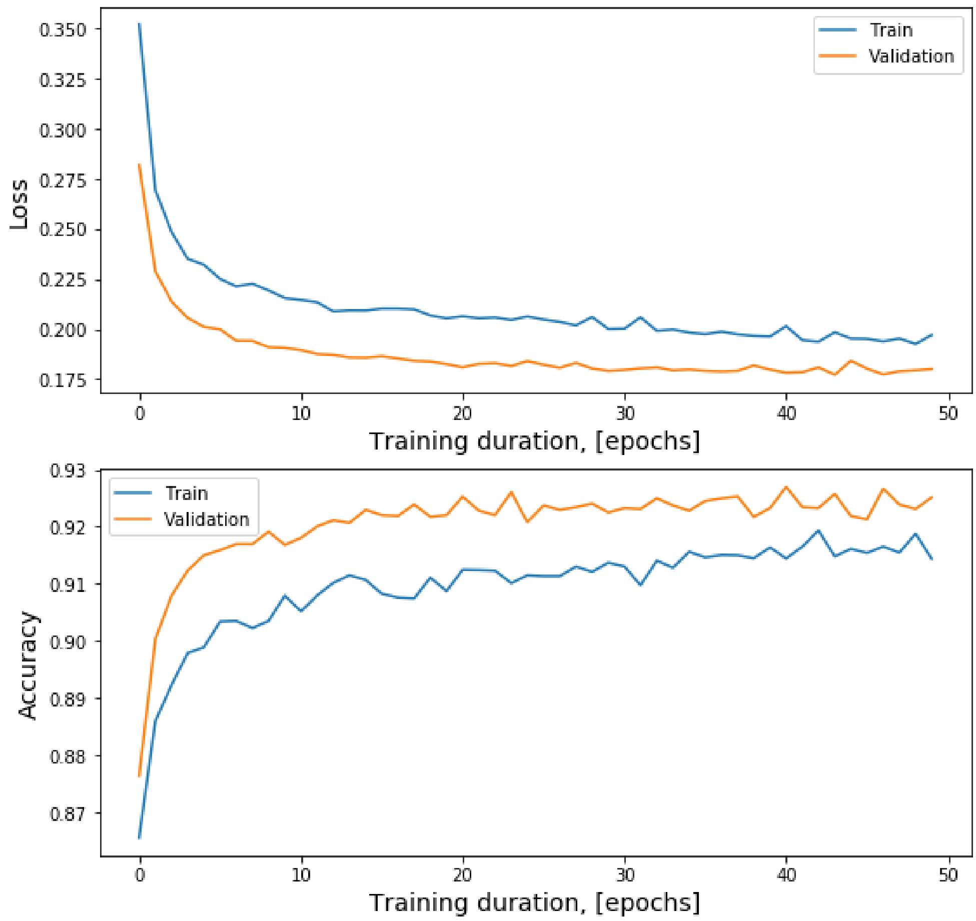


Figure 3.3: Fitting history of model for arduino with 7 gestures.

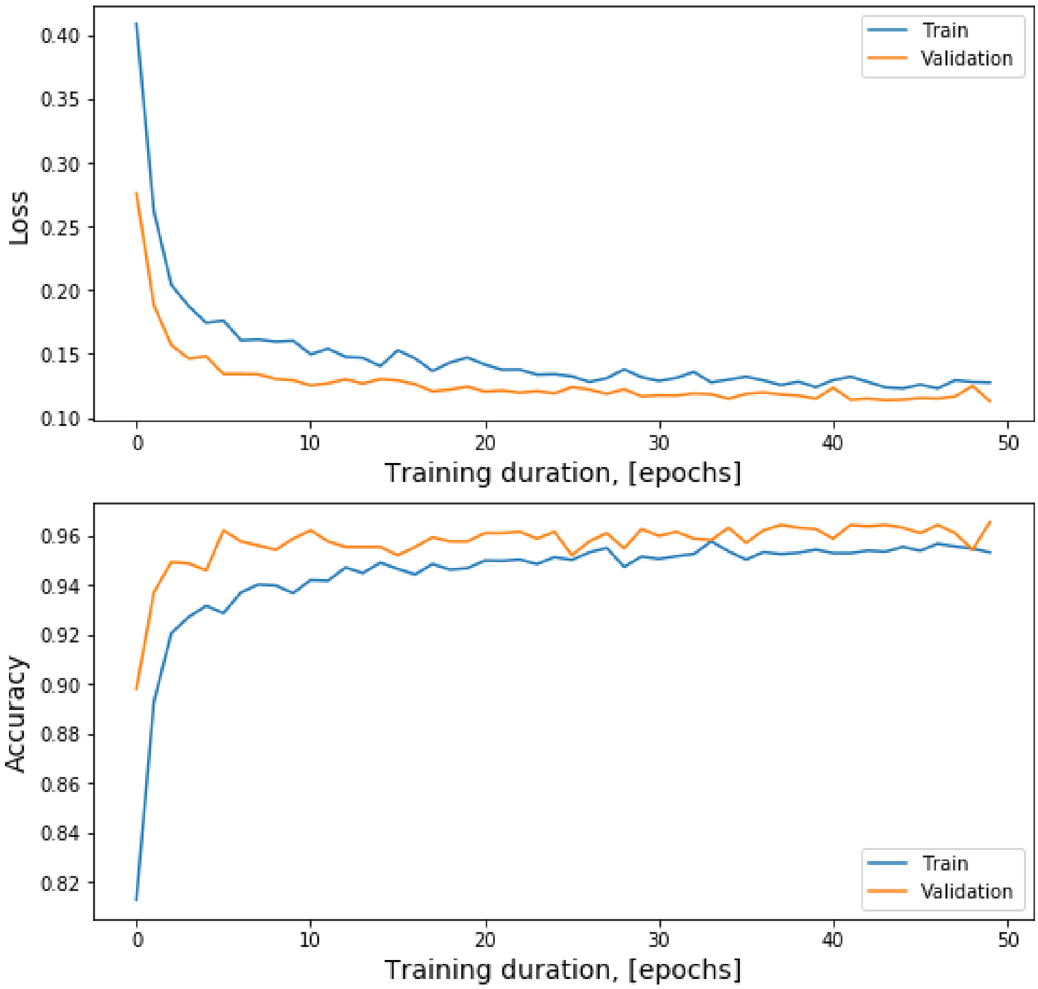


Figure 3.4: Fitting history of model for arduino with 4 gestures.

3.2 Creating PC software

Training a neural network requires a fairly large amount of data. Therefore, there was a need to automatically detect the presence of a gesture and write its data to a file. The main advantage of such system is that during subsequent recognition of gestures the same criteria for the presence of a gesture will be used as in the preparation of data. It means recognition accuracy will not be degraded due to time shifts.

I decided that simply using a threshold is not enough, because depending on the strength of the hand, the data structure for one gesture can vary greatly. It means that it is necessary to detect the end of the gesture and record what came before it. For this, it is necessary to determine the function of the sum of signal values for a certain time window and determine its maximum, more precisely, the intersection of its derivative with zero.

The threshold condition is added to this to exclude accidental movements. Since the data is normalized, different thresholds can be defined for recording and detection, depending on the person. This will not affect accuracy. In the program I use sum of all 8 sensors.

As a result, the program introduced the function of automatic data recording when a gesture is detected to prepare the dataset. The interface allows to select a label that will mark the data line, threshold, enable filtering and normalization. The screenshot is presented in Figure 3.5.

Figure 3.5: Main screen of PC software.

Figure 3.6: Calibration screen of PC software.

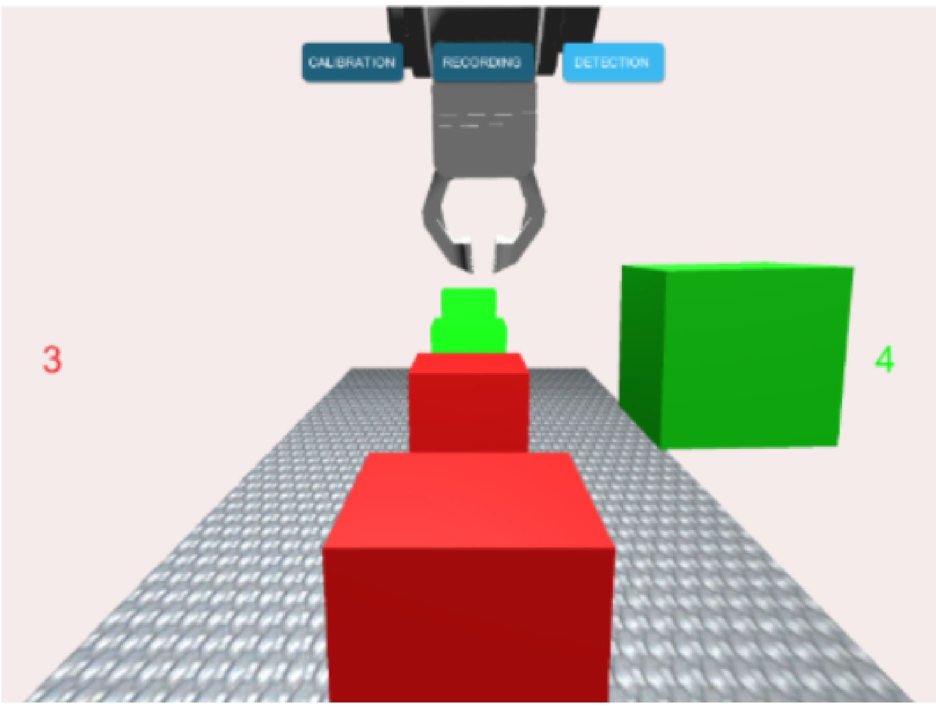


Figure 3.7: PC test game.

3.3 Plant scheme

In addition to the Myo Armband, there are two Micro Servo GG SG90 servo motors and the WeMos D1 Mini board in the plant. The one servomotor stands on the second one (see in Figure 3.8). In this way two freedom degrees and four possible movement kinds for control were obtained. The connection diagram of all parts is shown in Figure 3.9.

Figure 3.8: Stand view

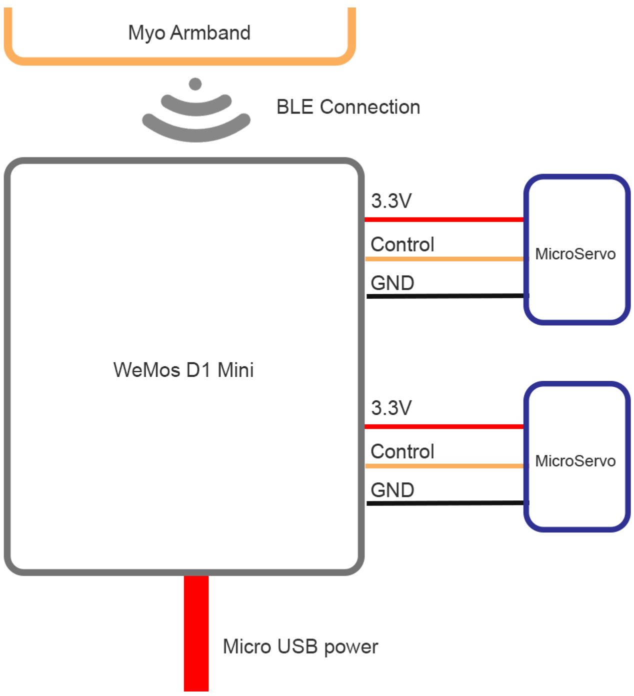


Figure 3.9: Plant scheme

3.4 Creating Arduino software

Created software for Arduino-like boards can work in two modes.

The first mode collects and writes EMG data to a log file. There is a threshold signal sum for detecting gestures. In the latest version, the script writes the last two lines of EMG (16 values) for each detected gesture. The recording format can easily be converted to a CSV file and used to train ML models in Python.

The second mode is gesture recognition and control of the robotic system. The TensorFlow model is ported from Python to C code through the Tinymlgen library [21]. Thus, the same threshold and detect gestures for recognition were used. There is a delay of 30 ms between processing each gesture. This provides the next signal from the Myo Armband Bluetooth module (with a frequency of 25 Hz) and time for calculating the result of the ML model [22].

3.5 Robot control

To avoid collisions and random errors, there is a stack with a detection history. The stack has five previous recognize results. Finally, the control system receives a gesture value that is pushed onto the stack at least three times. Theoretically, in accordance with the theory of probability, it increases the recognition accuracy from 0.96 to almost 1:

$$P_G = 1 - (1-P)^5 - (1-P)^4 P$$

where P_G is general probability of true gesture detection with using stack and $P = 0.96$ is probability of unity true gesture detection. Table 3.2 shows example of result with different gestures sets in the stack.

In addition, servo motor control is based on a series of continuously repeating gestures. The number of degrees by which the servomotor rotates depends on the number of identical gestures (each of which is obtained as a result of stack analysis) in the current series. If the series is interrupted (another gesture or not gesture), the counter is reset, and the next movement starts at a low speed. This dependence can be seen in Table 3.3. Also, in case of prolonged inactivity (about 500 ms), the stack is reset. This prevents the movement in the wrong direction, which may occur due to the accumulated history of gestures.

Table 3.2: Example of result gesture selection from the stack

Example	Stack (1)	Stack (2)	Stack (3)	Stack (4)	Stack (5)	Result Gesture
Example 1	Wave In	Wave In	Palm	Wave In	Palm	Wave In
Example 2	Fist	Pointer	Palm	Fist	Fist	Fist
Example 3	Pointer	Pointer	Palm	Palm	Fist	No gesture

Table 3.3: The dependence of the angle of rotation on the number of gestures in the series

Number of gestures	Angle of rotation (degrees)
0	0
1–4	1
5–7	2
8–14	3
>14	4

3.6 Wireless data transmission

As mentioned earlier, Myo Armband is equipped with a Bluetooth module. To be precise, this is the Bluetooth Low Energy (BLE) module. BLE is a specification of the Bluetooth protocol that is characterized by low peak power consumption and is widely used in IoT devices [23].

Myo Armband has a specific ID and sends data to everyone who is subscribed to it at certain intervals. The data set depends on the mode, which can be changed by a response request via Bluetooth. In our case, the device sends information about EMG.

Since the frequency of EMG sensors is 8 times greater than the frequency of sending data (200 Hz and 25 Hz, respectively), in one packet data comes from 8 previous EMG signals [22].

4 Results

4.1 Results and further research

The results of the gesture recognition models were satisfactory. The model that was used to control the robot is further improved by internal post-processing of the results on the microcontroller. The robot visualizes recognition well and is a prototype of a real mechanical system controlled by EMG signals

More powerful models that were created for use on a computer may come in handy for a more detailed analysis of human EMG signals. LSTM neural networks will extract features that are very difficult to process using a standard mathematical model.

There are several suggestions for further research. Since Myo Armband transfers data in batches from 8 previous measurements, it would be logical to use all of them at each recognition (a total of 64 values).

If the data is normalized separately for each sensor, the resulting matrix can be used as an input for a small LSTM neural network. This will be a simplified version of our PC model. And in theory, it should give similar results.

If the data is normalized separately for each moment in time, the resulting matrix can be visualized as a single-color image 8×8 (Example for Vawe In and Fist classes on the Figure 4.1). Then the images of the same class will have some similar elements located in different places. Convolutional Neural Network can be used, reducing the task to the classification of images [24, 25]. This task has many high-precision solutions based on CNN and theoretically, results will improve.

4.2 Possible applications

Possible applications can be divided into medical and technical. The medical applications primarily include the development of mechanized hand prostheses. This is a non-trivial task. To train a person to manage such a device is quite difficult. It is necessary to perform research and choose the most intuitive combination of muscle activity. At the same time, the gestures must be as different as possible. Other medical uses include rehabilitation of paralyzed patients, as well as the diagnosis and treatment of diseases associated with muscle activity.

Technical applications include the control of small robotic systems: quadrocopters (for example, for delivering products), cameras with remote access (for example, during a surgical operation). The technology can also be used for control in augmented and virtual reality. This is a situation where having some kind of mechanical controller is very inconvenient (human can't see own hands). But gesture control can be a good alternative.

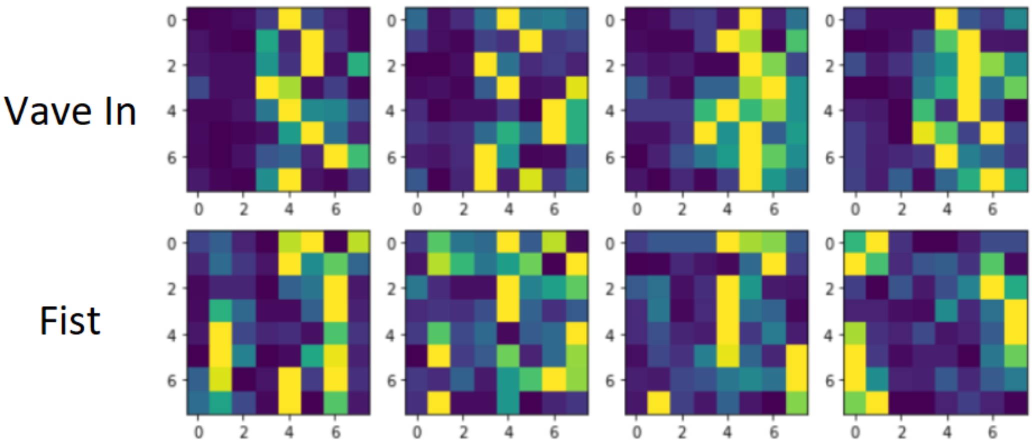


Figure 4.1: Examples of gestures image view

5 Conclusions

As a result of this work, the Myo Armband system was analyzed. Software was written to extract pure EMG data and write it to a file. The data on the correlation and distribution of EMG values are obtained. The corresponding graphs are drawn. The Myo Armband system is well suited for the tasks assigned to this work. The disadvantages include the termination of release and support of the device, as well as the presentation of data in relative terms, instead of volts. In general, the device is great for educational and research purposes.

Possible applications of the Myo Armband system were analyzed. A further direction in the development of this particular work may be the creation of a mechanical prosthesis of a hand or a robot that works under specific conditions and requires a reaction from a person with already occupied hands. In other case an EMG-based device can serve as an input device, both for ordinary computers and smartphones, and for augmented and virtual reality systems.

During the development of the control model, it was decided to replace the Matlab environment with a combination of other programming environments. Machine learning models for gesture recognition based on the Python language have been created. They were divided into PC models and models for Arduino depending on the number of parameters and recognition speed. The maximum number of recognizable gestures is 7, with an accuracy of 0.926. The best recognition result is 0.983 in a PC model with 5 gestures. The accuracy of the best model in the test sample exceeded some results with similar experiments described in articles over the past two years.

A robot was created, driven by Myo Armband. To do this, a model that recognizes 4 different gestures is used. Accordingly, the robot has 4 possible types of movement and 2 degrees of freedom. The robot is based on the WeMos D1 Mini board and is programmed in the Arduino environment. Myo Armband data is transmitted wirelessly via Bluetooth.

All the main goals of the work are fulfilled. Further possible directions of research and improvement of the current development are also indicated.

References

[1] KARABULUT, Derya; ORTES, Faruk; ARSLAN, Yunus Ziya; ADLI, Mehmet Arif. Comparative evaluation of EMG signal features for myoelectric controlled human arm prosthetics. *Biocybernetics and Biomedical Engineering*. 2017, vol. 37, no. 2, pp. 326–335. ISSN 02085216. Available from DOI: 10.1016/j.bbe.2017.03.001.

[2] KRAUSZ, Nili; LAMOTTE, Denys; BATZIANOULIS, Iason; HARGROVE, Levi; MICERA, Silvestro; BILLARD, Aude. Intent Prediction Based on Biomechanical Coordination of EMG and Vision-Filtered Gaze for End-Point Control of an Arm Prosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2020, pp. 1–1. ISSN 1534-4320. Available from DOI: 10.1109/TNSRE.2020.2992885.

[3] *Johns Hopkins University Applied Physics Laboratory* [online] [visited on 2020-05-31]. Available from: https://www.jhuapl.edu/.

[4] *Adora* [online] [visited on 2020-05-30]. Available from: https://adora-med.com/.

[5] ARTAMEEYANANT, Patcharin; SULTORNSANEE, Sivarit; CHAMNONGTHAI, Kosin. An EMG-based feature extraction method using a normalized weight vertical visibility algorithm for myopathy and neuropathy detection. *SpringerPlus*. 2016, vol. 5, no. 1. ISSN 2193-1801. Available from DOI: 10.1186/s40064-016-3772-2.

[6] *Researchgate* [online] [visited on 2020-05-30]. Available from: https://www.researchgate.net/publication/316538856_A_wearable_general-purpose_solution_for_Human-Swarm_Interaction/figures?lo=1.

[7] *GForce-Pro EMG Armband* [online] [visited on 2020-05-31]. Available from: http://www.oymotion.com/en/product32/149.

[8] *MyoWare Muscle Sensor* [online] [visited on 2020-05-31]. Available from: https://www.sparkfun.com/products/13723.

[9] *Compacttool* [online] [visited on 2020-05-30]. Available from: https://compacttool.ru/viewtovar.php?id=1334.

[10] SPIEWAK, Christopher. A Comprehensive Study on EMG Feature Extraction and Classifiers. *Open Access Journal of Biomedical Engineering and Biosciences*. 2018, vol. 1, no. 1. ISSN 26374579. Available from DOI: 10.32474/OAJBEB.2018.01.000104.

[11] NAZMI, Nurhazimah; RAHMAN, Mohd Abdul; YAMAMOTO, Shin-Ichiroh; AHMAD, Siti; ZAMZURI, Hairi; MAZLAN, Saiful. A Review of Classification Techniques of EMG Signals during Isotonic and Isometric Contractions. *Sensors*. 2016, vol. 16, no. 8. ISSN 1424-8220. Available from DOI: 10.3390/s16081304.

[12] XIA, Peng; HU, Jie; PENG, Yinghong. EMG-Based Estimation of Limb Movement Using Deep Learning With Recurrent Convolutional Neural Networks. *Artificial Organs*. 2018, vol. 42, no. 5, pp. E67–E77. ISSN 0160564X. Available from DOI: 10.1111/aor.13004.

[13] COTE-ALLARD, Ulysse; FALL, Cheikh Latyr; DROUIN, Alexandre; CAMPEAU-LECOURS, Alexandre; GOSSELIN, Clement; GLETTE, Kyrre; LAVIOLETTE, Francois; GOSSELIN, Benoit. Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2019, vol. 27, no. 4, pp. 760–771. ISSN 1534-4320. Available from DOI: 10.1109/TNSRE.2019.2896269.

[14] HE, Yunan; FUKUDA, Osamu; BU, Nan; OKUMURA, Hiroshi; YAMAGUCHI, Nobuhiko. Surface EMG Pattern Recognition Using Long Short-Term Memory Combined with Multilayer Perceptron. *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2018, pp. 5636–5639. ISBN 978-1-5386-3646-6. Available from DOI: 10.1109/EMBC.2018.8513595.

[15] SAMADANI, Ali. Gated Recurrent Neural Networks for EMG-Based Hand Gesture Classification. A Comparative Study. *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2018, pp. 1–4. ISBN 978-1-5386-3646-6. Available from DOI: 10.1109/EMBC.2018.8512531.

[16] *Long short-term memory* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/Long_short-term_memory.

[17] *Support vector machine* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/Support_vector_machine.

[18] *K-nearest neighbors algorithm* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm.

[19] *Principal component analysis* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/Principal_component_analysis.

[20] *Savitzky–Golay filter* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/Savitzky%5C%E2%5C%80%5C%93Golay_filter.

[21] *Tinymlgen library* [online] [visited on 2020-05-31]. Available from: https://pypi.org/project/tinymlgen/.

[22] *Myo Bluetooth Protocol Released* [online] [visited on 2020-05-31]. Available from: https://developerblog.myo.com/myo-bluetooth-spec-released/.

[23] *Bluetooth Low Energy* [online]. San Francisco (CA): Wikimedia Foundation [visited on 2020-05-31]. Available from: https://en.wikipedia.org/wiki/Bluetooth_Low_Energy.

[24] CHEN, Hongfeng; TONG, Runze; CHEN, Minjie; FANG, Yinfeng; LIU, Honghai. A Hybrid Cnn-Svm Classifier For Hand Gesture Recognition With Surface Emg Signals. *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*. 2018, pp. 619–624. ISBN 978-1-5386-5214-5. Available from DOI: 10.1109/ICMLC.2018.8526976.

[25] CHEN, Lin; FU, Jianting; WU, Yuheng; LI, Haochen; ZHENG, Bin. Hand Gesture Recognition Using Compact CNN via Surface Electromyography Signals. *Sensors*. 2020, vol. 20, no. 3. ISSN 1424-8220. Available from DOI: 10.3390/s20030672.

[26] PHINYOMARK, A.; HIRUNVIRIYA, S.; NUIDOD, A.; PHUKPATTARANONT, P.; LIMSAKUL, C. Evaluation of EMG Feature Extraction for Movement Control of Upper Limb Prostheses Based on Class Separation Index. *5th Kuala Lumpur International Conference on Biomedical Engineering 2011*. 2011, pp. 750–754. ISBN 978-3-642-21728-9. Available from DOI: 10.1007/978-3-642-21729-6_183.

[27] MORAIS, Gabriel Doretto; NEVES, Leonardo C.; MASIERO, Andrey A.; CASTRO, Maria Claudia F. Application of Myo Armband System to Control a Robot Interface. *Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies*. 2016-2-21, pp. 227–231. ISBN 978-989-758-170-0. Available from DOI: 10.5220/0005706302270231.

[28] GERYES, Maroun; CHARARA, Jamal; SKAIKY, Ahmad; MCHEICK, Ali; GIRAULT, Jean-Marc. A novel biomedical application for the Myo gesture control armband. *2017 29th International Conference on Microelectronics (ICM)*. 2017, pp. 1–4. ISBN 978-1-5386-4049-4. Available from DOI: 10.1109/ICM.2017.8268823.

[29] VISCONTI, P.; GAETANI, F.; ZAPPATORE, G.A.; PRIMICERI, P. Technical Features and Functionalities of Myo Armband: An Overview on Related Literature and Advanced Applications of Myoelectric Armbands Mainly Focused on Arm Prostheses. *International Journal on Smart Sensing and Intelligent Systems*. 2018, vol. 11, no. 1, pp. 1–25. ISSN 1178-5608. Available from DOI: 10.21307/ijssis-2018-005.

[30] RAHIM, Md Abdur; SHIN, Jungpil. Hand Movement Activity-Based Character Input System on a Virtual Keyboard. *Electronics*. 2020, vol. 9, no. 5. ISSN 2079-9292. Available from DOI: 10.3390/electronics9050774.

[31] LI, Chengzhang; MA, Zheren; YAO, Lin; ZHANG, Dingguo. Improvements on EMG-based handwriting recognition with DTW algorithm. *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2013, pp. 2144–2147. ISBN 978-1-4577-0216-7. Available from DOI: 10.1109/EMBC.2013.6609958.

[32] LUO, Jing; LIU, Chao; FENG, Ying; YANG, Chenguang. A method of motion recognition based on electromyographic signals. *Advanced Robotics*. 2020, pp. 1–9. ISSN 0169-1864. Available from DOI: 10.1080/01691864.2020.1750480.

[33] HASSAN, Hussein F.; ABOU-LOUKH, Sadiq J.; IBRAHEEM, Ibraheem Kasim. Teleoperated robotic arm movement using electromyography signal with wearable Myo armband. *Journal of King Saud University - Engineering Sciences*. 2019. ISSN 10183639. Available from DOI: 10.1016/j.jksues.2019.05.001.

[34] SIMÃO, Miguel; NETO, Pedro; GIBARU, Olivier. EMG-based online classification of gestures with recurrent neural networks. *Pattern Recognition Letters*. 2019, vol. 128, pp. 45–51. ISSN 01678655. Available from DOI: 10.1016/j.patrec.2019.07.021.