Development of an electromyography sensor wristband for controlling a prosthetic hand in virtual reality

Peter Rott , B. Eng.

***Abstract*—Surface electromyography (sEMG) is a non-invasive method of measuring electrical activity caused by muscle contraction. sEMG signals have been progressively utilized in a variety of applications in recent years, including rehabilitation, pattern identification and control of prosthetic devices.**

The present work describes the development process of a flexible EMG wristband which is capable of acquiring six EMG sensor signals to control a virtual prosthetic hand in real time. The development process includes the design of the wristband, data acquisition, data processing and the implementation of two machine learning algorithms to classify four defined hand gestures. Furthermore, this study investigates whether the classifiers achieve the same classification accuracy with an averaged value of the EMG sensor signals compared to six EMG sensor signals as input.

A diverse group of five participants took part in the study to gain a train dataset, performing a series of predefined hand gestures while wearing the wristband. The implemented classifiers, an artificial neural network (ANN) and a support vector machine (SVM) can classify defined hand gestures, where the average classification accuracy of the ANN when using individual sensor signals is 0.97, while it decreases to 0.76 when using the average signal of all sensors. The accuracy results for the SVM are 0.97 for the individual signals and 0.77 for the average signal. Ultimately, an ANN is selected for the real time implementation because it performs better with random test data by an average of 0.02. With the real time implementation, the EMG wristband achieved an average accuracy of 0.90 and an average delay time of 6.29 ms. In the end, a hand prosthesis can be controlled in a virtual environment with the predicted data from the EMG wristband.

In summary, the results show that in the context of the comparison, the average value of all EMG sensors leads to poorer performance. Ultimately, the developed system can classify the defined hand gestures in real time and provides a basis for further studies with larger samples or for statistical analyses to explore optimal sensor configurations and to improve the accuracy of hand gesture classification systems.

***Index Terms*—Surface electromyography (sEMG), Hand gesture recognition, Machine learning, Wearable EMG armband**

I. INTRODUCTION

THE development of wearable devices has made significant progress in recent years, with global use of wearable devices expected to increase at a compound annual growth rate of 38% between 2017 and 2025 [1]. In general, the design of wearable systems is based on the incorporation of smart sensors, artificial intelligence (AI), the Internet of Things (IoT), and Big Data, making it possible to obtain information of interest from the human body [2].

Hand Gesture Recognition (HGR) is a key aspect of Human-Computer Interaction (HCI), which is an investigation of computer technology developed to understand human commands. HGR models are human-computer systems which can identify which gestures were executed and when they were carried out [3]. This not only offers advantages for amputees in controlling smart prostheses [4], but also finds application in other areas such as language recognition [5], rehabilitation devices [6], and device control [7]. An important measure for these HGR models are the action potentials produced by the muscle contraction for the hand movement. These muscle signals can be recorded using non-inverse surface electromyography (sEMG) sensors. sEMG sensors detect the electrical signals representing the sum of subcutaneous motor action potentials generated through muscular contraction and are non-stationary [8]. A further advantage of these sensors is that they do not have to be placed directly on the hand to record the purpose of the hand movement. This suggests that these sensors can also be used with amputees who are unable to perform hand movements but would like to [9]. To detect these intended hand gestures several Machine Learning (ML) algorithms are used, such as support vector machines, linear discriminant analysis, and neural networks [3].

This thesis aims to develop a sEMG sensor wristband to control a prosthetic hand in virtual reality. The wristband is designed to capture signals from the forearm muscles of the user with six sEMG sensors. Afterwards, the EMG data is transmitted to the computer via a microcontroller. On the computer it is further processed to extract features that are relevant for gesture classification. In the end, the virtual hand prosthesis is controlled by classifying the EMG data received from the wristband. A second thesis deals with the development of a hand model in a virtual environment and the classification of hand gestures using a camera tracking algorithm. The overall goal is to design a training tool to implement new hand gestures into the wristband. The camera tracking algorithm provides the right labels for various hand gestures. These labels are then synchronized with wristband data, allowing the user to practice new hand movements for the prosthesis. The second goal is to provide the user with a virtual setup in which they may learn to control the hand prosthesis without wearing it.

First, already developed EMG wristbands are listed and described in their characteristics. Afterwards, the used ma-

Peter Rott is with the Department of Medical and Health Technologies, MCI, Innsbruck, Austria, e-mail: p.rott@mci4me.at

terials and implemented methods are explained, starting with the EMG signal acquisition, which is influenced by muscle contraction, signal noise and placement of the sensors [10], [11]. Furthermore, signal processing, feature extraction, the two common ML approaches, an artificial neural network (ANN) and a support vector machine (SVM), and the modeling of the wristband's 3D printed parts are explained in more detail. In the end, the EMG wristband is validated and tested. Therefore the EMG data of five healthy participants are recorded to train and compare the performance of the ANN and the SVM. In the study, participants are asked to perform four defined hand gestures: fist, pinch, thumb up and spread. The aim of the study is on the one hand to determine whether the precise positioning of the wristband in relation to the classification is irrelevant by averaging the sensor values. On the other hand, recorded datasets are used to test the robustness of the ML algorithms and to identify which approach is better suited for a real time implementation. Before the system is implemented in real time, it is first trained and tested with datasets that are not classified in real time. To evaluate the system, various confusion matrices and classification reports are created. Finally, the wristband is tested in real time with a HoloLens interface to control a virtual prosthesis and to provide better labeling for new data.

The results of this work should help in prosthetic hands adaptation and allow users to train user-specific gestures into the hand prosthesis. The development of an EMG sensing wristband provides a user-friendly and non-invasive approach to prosthetic hand control, allowing for more intuitive and natural movements in virtual environments and in real life.

All used codes in this paper can be accessed through the git repository https://github.com/PeterR96/EMG_wristband_to-_control_virtual_hand_prosthesis

A. State of the art

In 2015, Thalmic Labs introduced the Myo wristband, an innovative sEMG capture method designed for consumers. Retail prices for this eight-channel Bluetooth connected wristband were significantly lower than medical recording systems (˜$200 USD). The wristband does not require a recording site to be set up and is non-intrusive. However, to achieve this, compromises were made in both data quality and signal bandwidth. For example, the wristband has only eight channels and a limited sampling rate of 200 Hz. It can distinguish between four various hand motions (fist, spread hand, folding the hand to the left and right), and it can also evaluate the wristband's position in space. Unfortunately, it is currently discontinued [12].

The recently announced gForcePro+ EMG armband has an elastic wristband, Bluetooth BLE4.2, an eight-channel high-sensitive EMG, and 9-Axis motion sensors. Users can obtain unprocessed EMG data. Up to 16 user-defined gestures and the mobile gForceAPP gesture training platform are supported. There are two modes available: 8-bit mode up to a maximum frequency of 1000 Hz and 12-bit mode up to 500 Hz. As an additional feature the armband has a feedback vibration function integrated [13]. However, these named features made the gForce-Pro six times more expensive (˜$1200 USD) than the Myo wristband for the same number of channels and recording resolution [8].

Mahmoud Tavakoli presented in [14] a simple band with a sampling frequency of 1000 Hz that only utilized two sensors to recognize five movements. So far, the amount of gestures considered reduces the applicability of the wristband. Also, the best location for the sensors differs depending on the subject, leading to significant variations in the results. Even though the system only has two channels, it is difficult to apply in real-world situations since the design is too bulky and does not function wirelessly [8].

Another sEMG armband [8], called the 3DC Armband was developed by the Biomedical Microsystems Laboratory at Laval University, which has 10 channels and a 9-axis inertial unit. In addition, it has a sampling frequency of 1000 Hz with a 10-bit Analog Digital Converter (ADC) and a weight of 63 g. Furthermore, each sensor module of the device has a height of 3.7 cm and a thickness of 1.6 cm. The total costs are approximately $150 USD. Due to the suggested connection parts for the sensor boxes and the complexity of the sensors employed, this device has a very complicated system and a non-standardized production process. However, they offer good capturing attributes, but the reproducibility is restricted by the complexity of the system [15].

Recently, the Medical Robotics and Bio signal Processing Laboratory [15] produced the WyoFlex band. It consists of four sEMG sensors with sampling frequencies of up to 1600 Hz. Using a graphical user interface created in Node-RED, the gathered sEMG data may be displayed and saved for further post-processing steps. The armband can recognize four hand motions and costs roughly $250 USD.

Table I presents the most relevant works in the literate as well as the two commercial sEMG armbands and summarize their characteristics. Furthermore several studies have been performed to analyse and classify EMG data. The signal is typically band pass filtered, with studies using a range of cutoff frequencies, including 10-500 Hz [15], 15-200 Hz [16] or 20-500 Hz [8], [17], [18].

ML is a method that can be applied to address the problem of EMG-based hand gesture classification. The most important classifiers for recognizing hand gestures are Support Vector Machines (SVM) [19], Artificial Neural Networks (ANN) [20], Convolutional Neural Networks (CNN) [21], k-Nearest Neighbors (kNN) [22] Random Forest (RF) [23] and Dynamic Time Warping (DTW) [24].

The following domains describe the typical characteristics used for hand gesture recognition: time, frequency and time-frequency [20]. Hand gesture identification is generally determined by features collected from these domains. Most frequently used features in the literature are waveform length (WL) [19], EMG spectrograms [22], root mean square (RMS) [23], mean absolute value (MAV) , slope sign change (SSC) [8], short-time energy and the zero-crossing rate [25]. Data windows usually are applied to the data and used to extract the features. However, different lengths have been used for the window, including 100 ms [25], 150ms [19], 200 ms [26] and 250 ms [8], [16]. According to studies which compared

TABLE I: Characteristics of the presented EMG armbands

	Myo wristband	**gForcePro+**	**Double channel EMG armband**	**3DC armband**	**WyoFlex armband**
EMG sensors	8	8	2	10	4
IMUs	9	9	x	9	x
Sample rate	200 Hz	500/1000 Hz	1000 Hz	1000 Hz	1600 Hz
BLE/WIFI	BLE 4.0	BLE 4.1	x	2.4 GHz low power custom protocol (similar to BLE)	WIFI
Gesture recognition	4	16	5	11	6
Price	was ∼\$200	∼\$1200	NI	∼\$150	∼\$250

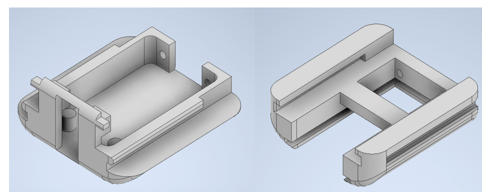
several time windows for real time classification, the latency should be kept within 100-250 ms, but the classifier's performance should take priority above speed [27], [28]. To further improve classification performance, several publications use strategies such as normalized EMG data [23] and offering a transfer learning scheme [29]. Most studies undertake real-time classification of EMG signals [30], [19], [20], [24], [29]. There are, however, self-developed EMG armbands that use non-real-time classification [8], [14], [25].

II. Methods

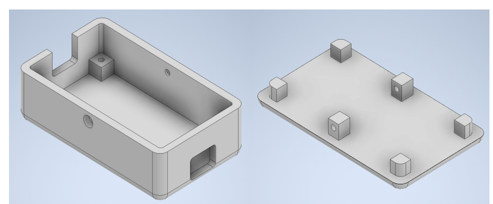
A. Materials

Six Gravity Analog EMG sensors [31] are used to capture muscle activity on the forearm. The sensors consist of two main parts, the dry electrode board (sensor) and signal transmitter board (controller). This setup allows EMG signals to be obtained non-invasively. The analog output voltage varies between 0 and 3 V, with a reference voltage of 1.5 V. The intensity of the signal is determined by muscular activity. The output signal waveform represents muscle activity and aids in the analysis and investigation of EMG data. As data transmitting unit a NodeMCU ESP32 microcontroller is used. The microcontroller is a Joy-it development and allows communication by both WiFi or Bluetooth with a frequency of 2.4 GHz. It requires minimal energy and is compatible with Arduino IDE, the operating voltage is at 3.3 V (or operable via 5 V-microUSB). The ESP32 is linked to a computer through a USB cable to deliver the required 3.3 V to the sensors and establishes a serial connection to record data. Furthermore, the ESP32 includes a built-in 12bit ADC converter for digitally displaying the recorded data. All black cables of the EMG sensors are soldered together and connected to the ground pin of the ESP32. The same process is done with the red cables, these are attached to the 3.3 V pin. The EMG signals are transmitted via the blue cables of the sensors, which are connected to pins D32, D33, D34, D35, D36 and D39.

The manufacturing method used for the wristband is 3D printing. Here, the Creality Ender 5 Pro printer is used for all parts of the wristband; a case for the EMG sensor and controller, a cover for the ESP32, and two parts to attach a rubber band to the sensor case. All parts are printed out of PLA with the following printer settings: printer speed: 80 mm/s, filling density of 20%, printer nozzle temperature of 210° and bed temperature of 60°. The design of the the cases for the sensor and the controller unit as well as the case for the ESP32 is given in Figure 1.



(a)



(b)

Fig. 1: (a) 3D printed case for the EMG sensor and the controller (b) Case for the ESP32

B. Data acquisition

To record relevant data for hand gesture recognition with the EMG wristband, the user has to wear it at the forearm as shown in Figure 2.

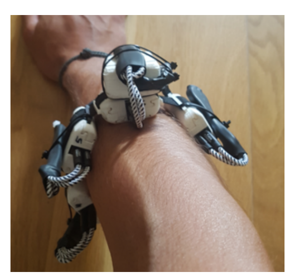


Fig. 2: Position of the EMG wristband

It is important that the ESP32 housing always faces upwards to ensure consistent recording conditions. Then a serial connection with the ESP32 to a computer must be established.

The ESP32 is printing all six EMG sensor values to the serial monitor at a sample rate of 1000 Hz. To save all the data into a .txt file a program called CoolTerm is used.

To train and validate the ML algorithms, it is necessary to record data from different individuals. Ultimately, the train dataset consists of data recordings of five healthy participants (three females and two males) between the ages of 20 - 26. Before the data collection is started, all participants signed an informed consent form. All participants are asked to perform the hand gestures fist, fingers spread, pinch and thumbs up as well as the the rest position. They are asked to perform each gesture 10 times in sequence with a hold duration of 5 sec followed by a rest position for 5 sec. In total, the dataset consists of around 200 hand gestures plus the same number of rest positions. During the execution of the hand movements, the elbow is resting on a surface.

C. Signal Filtering

Figure 3 presents a flow chart of the data processing steps. First step after data acquisition is the offset removal. Afterwards, the EMG signals are filtered with a band pass filter with cut off frequencies of 20 Hz and 450 Hz to reduce unwanted signal noise. Furthermore a notch filter is applied to eliminate the noise of the surrounding 50 Hz grid.

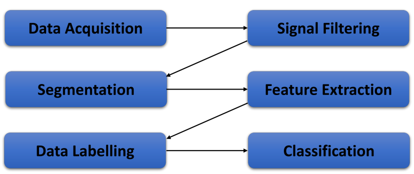


Fig. 3: Flowchart of the EMG data

Then the signals are rectified to obtain only positive signals. Afterwards the signal is enveloped and additionally filtered again with a low pass filter with a cut off frequency of 10 Hz to create a graphed signal waveform. The result of the filtering can be seen in Figure 4.

One objective of the work is to test whether classification accuracy is affected when only an averaged value of all six EMG sensors is used as input. Therefore, both the filtered values of the individual EMG sensors and the averaged value of all sensors are stored separately for each participant. The following two steps, feature extraction and data labeling, are also performed for both scenarios.

D. Feature Extraction

To extract the features, the filtered EMG signal is divided into time segments. As presented in [3], [27], [28], the time window should not exceed 300 ms to increase the classification performance. For this reason the approach of a sliding time window of 250 ms with an overlap of 50 ms is selected for the feature extraction in non-real time. For the real-time implementation a fixed time window of 250ms is defined. Features in the time domain are most commonly used for EMG pattern recognition, because they are usually easy to implemented and don't require high computational resources

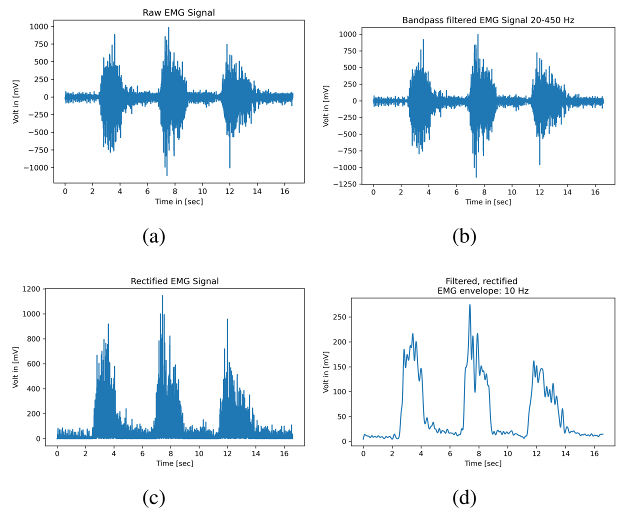


Fig. 4: (a) Raw EMG Signal (b) Band pass filtered EMG Signal (20-450 Hz) (c) Rectified EMG signal (d) Enveloped EMG signal low pass filtered with 10 Hz

[32]. The extracted features are mean absolute value, zero crossing, slope sign changes, wavelength, variance, integrated EMG and the root mean square.

E. Data labeling

Since the used ML algorithms are based on the supervised learning approach, data labelling is an essential step to train them. For this purpose, it is important to distinguish the areas of the signal where gesture execution has occurred from those where the user does not execute a gesture. All gestures used must be assigned with a specific label. These assignments are 0, 1, 2, 3, 4 for rest, fist, pinch, thumb up and spared. To determine the active gesture regions, two thresholds are calculated. The average wavelength feature data serves as the basis for calculation in both scenarios, individual sensor and averaged sensor signals. Here, the average value Avg_{WL} is computed from the first eight time windows. The thresholds are defined using Equation 1 and 2.

$$Thr1 = Avg_{WL} \cdot 1.85 \tag{1}$$

$$Thr2 = Avg_{WL} \cdot 1.5 \tag{2}$$

The first threshold (Eq.1) marks the start of the gesture and the second (Eq.2) defines the end of a gesture. These values were tested across all data sets and were found to be appropriate to identify the execution region for all gestures. However, in some cases, the thresholds still had to be improved manually. Figure 5 shows the results of a random data set of a thumb gesture. Detected gesture executions are marked by the perpendicular lines. Threshold one is presented as a red line and threshold two as a green one. To increase the accuracy of the correct label assignment, two additional functions are integrated to test whether there are deviations in the labeling (e.g. Figure 5 at around window 200). Thereby it is checked if there is a 0 label between two gesture labels, if this is the case the 0 label is replaced by the gesture label. In addition, it is determined whether a gesture label exists between two recognized 0 labels before and after the gesture label. In this case the gesture label is replaced by a 0 label. After the labels have been determined, the features with assigned labels are automatically stored in a csv file.

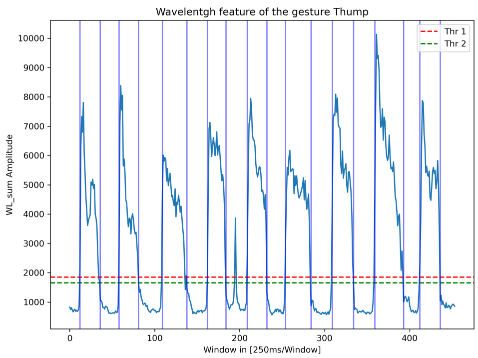


Fig. 5: Estimation of the start and end points from a hand gesture with the WL feature

F. Non-real time classification

For the classification of the EMG data, an ANN and a SVM are implemented in Python first of all in non real. To obtain the best possible performance of both ML methods, a parameter study is conducted. The model parameters used for the comparison are listed in Table II. This study is based on the whole dataset of the five participants. The best parameters are determined using the stratified five fold cross-validation grid search process based on the work presented in [18]. This means that the training dataset is randomly split into five equal-sized parts. The classifier is trained using four subsets, while the remaining subset is utilized for validation and accuracy evaluation. This method is performed five instances, with each of the five subgroups utilised for validation exactly once. Following that, the mean accuracy scores for five convolution results are compared.

TABLE II: Parameters of the classifier study

ANN	**SVM**
Hidden layers: 2, 3, 4	Kernel: rbf, linear
Neurons in each layer: 300, 600, 1000	C: 1,10,100,1000
Dropout rate: 0.2, 0.3	Gamma: 1, 0.1, 0.01, 0.001, 0.0001

After the best parameters are determined for each model, both classifiers can be trained with them. For this purpose, the dataset is split into 75% training data and 25% test data. In the end, four models are saved and compared with each other in respect to the classification accuracy.

G. Real time implementation

For the real time implementation an interface between the ESP32 and Python must be established. The ESP32 calibrates the EMG sensors once in the setup function, afterwards the EMG data of all sensors are continuously sent to the python script when its running. Here the data is stored and processed in arrays of 250 values per sensor, this corresponds to a time span of 250 ms at a sample rate of 1000 Hz. The data is filtered with a band pass and a notch filter as described before. Furthermore, the features are extracted and then normalized to be classified by the ANN model loaded in the beginning. The ANN model predicts the label for the given time window and prints it to the console. Since it was not possible to implement the sliding time window approach, the ANN model is retrained with a time window segmentation of 250 ms. To evaluate the real time system, ten sequences of ten randomly selected hand movements are executed. Afterwards the results of the ANN are compared with the desired outputs.

H. Integration in virtual reality (HoloLens)

The approach of data labeling by HoloLens is based on a ML algorithm for hand gesture detection on the HoloLens. In this algorithm, a label for the corresponding hand gesture is generated every 250 ms. In the end, these labels are assigned to the dataset which is recorded with the EMG wristband. The approach should help to determine the definition of a movement execution more precisely than the threshold approach. To ensure synchronization of the predicted labels with the EMG dataset, it is important to initiate the data recording of both devices at the same time. Therefore it is necessary to send a command from the HoloLens to the ESP32 when the EMG data acquisition should start. To accomplish this, a server and a local WiFi network is created on the ESP32. Before building the network, the wristband is calibrated once in the setup function. Then the ESP32 waits until the HoloLens logs into the network and the data recording scene on the HoloLens is started. With the start of the data recording a time stamp is sent once to the ESP32 to be able to synchronize the datasets in the later stage of processing. Finally, the accurately defined start and end points of a hand movement by the HoloLens can be assigned to the feature matrix. With the created data set, a new ML algorithm may be trained. To control a virtual prosthesis on the HoloLens, the predicted labels of the ANN model are sent to the prosthesis control scene on the HoloLens via a server in real time.

III. Results

A. Non-real time implementation

The parameter study revealed that the parameters four hidden layers, 300 neurons and a dropout rate of 0.2 lead to the best accuracy of 0.97 for the ANN with individual sensor signals. For the averaged EMG signals, only the dropout rate changes to 0.3, but the model achieves only an accuracy of 0.76. For the SVM, a rbf kernel, C = 100 and gamma of 0.01 are the best parameters for the individual EMG data. With these settings, the model achieves an accuracy of 0.97. For the averaged EMG signals, C increased to 1000 and gamma to 0.1. As with the ANN, the accuracy of the SVM drops to 0.77 in this case. As a result, the difference in classification accuracy for the train dataset between the two approaches are 0.21 for the ANN and 0.20 for the SVM. This demonstrates that the averaged values are less suitable for the classification of hand gestures. For this reason, only the individual EMG sensor data are considered for further implementation. The average EMG sensor data is only used to label the train datasets.

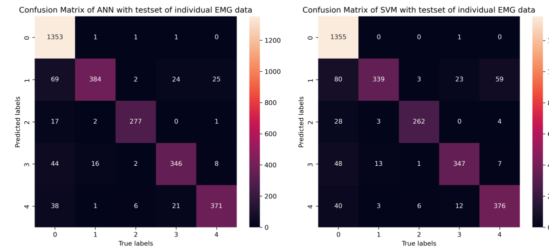
To define which classifier performs better with unlabeled data and is in the end better suited for real time implementation, the ANN and the SVM are tested with ten random hand gesture sequences from one participant. The accuracies of the models are calculated for each test sequence and are presented in Table III. There is also the average accuracy rate determined. The ANN shows accuracies between 0.82 and 0.95 with an average accuracy of 0.88. The SVM, on the other hand, provides accuracies between 0.77 and 0.95 with an average accuracy of 0.86.

TABLE III: Classifier accuracies (acc) for each test sequence

	1	2	3	4	5	6	7	8	9	10	Avg
ANN acc.	0.89	0.84	0.91	0.95	0.92	0.86	0.87	0.89	0.82	0.84	0.88
SVM acc.	0.86	0.77	0.90	0.95	0.90	0.88	0.88	0.89	0.84	0.78	0.86

Furthermore, the whole unlabeled dataset is classified by each model. The overall results of the whole test sequence dataset are shown in Figure 6 as confusion matrices and as classification report in Table IV. The ANN model gains a high precision of 0.89 for label 0, suggesting that 89% of the cases predicted as label 0 are correct. It also has a perfect recall of 1.00, which means that it correctly detects all cases labelled as 0. Label 0 has an F1-score of 0.94, which is an indicator of an overall good performance. For label 1, the ANN model achieves a precision of 0.97, indicating a high level of accuracy in identifying instances of this class. However, the recall is 0.79, suggesting that some instances labeled as 1 are not correctly identified. The F1-score for label 1 is 0.87, reflecting the balance between precision and recall. The ANN model performs well for label 2, achieving high precision (0.96) and recall (0.92). The F1-score for label 2 is 0.94, indicating good overall performance. Label 3 also shows relatively good performance with a precision of 0.88, recall of 0.86 and F1-score of 0.87. For label 4, the ANN model achieves a precision of 0.95 and recall of 0.84, resulting in an F1-score of 0.89.

The SVM model performs slightly worse than the ANN model, with precision, recall and F1-scores between 0.67 and 1.00 for different labels. Overall, the ANN model obtains an accuracy of 0.91, meaning that 91% of instances in the test dataset are properly identified. The accuracy of the SVM model is 0.89. The macro average scores for both models represent the average performance over all labels, with the ANN model outperforming the SVM model (precision: 0.93, recall: 0.88, F1-score: 0.90). When the weighted average scores are taken into account, the ANN model beats the SVM model in all terms of accuracy (0.92 vs. 0.89), recall (0.91 vs. 0.89), and F1-score (0.91 vs. 0.89).



(a) (b)

Fig. 6: (a) Confusion matrix for ANN with whole test dataset (b) Confusion matrix for SVM with whole test dataset

TABLE IV: Classification report for ANN and SVM tested with test dataset

	ANN			SVM			
Label	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.89	1.00	0.94	0.87	1.00	0.93	1356
1	0.97	0.79	0.87	0.95	0.67	0.79	504
2	0.96	0.92	0.94	0.96	0.88	0.92	297
3	0.88	0.86	0.87	0.91	0.83	0.87	416
4	0.95	0.84	0.89	0.84	0.86	0.85	437
Accuracy			0.91			0.89	3010
Macro avg	0.93	0.88	0.90	0.91	0.85	0.87	3010
Weighted avg	0.92	0.91	0.91	0.89	0.89	0.89	3010

All in all the results show, that the ANN has the better average classification accuracy of 0.91 and 0.88 in both the combined and the individually considered test datasets compared to the SVM with 0.89 and 0.86, respectively. However, it is feasible to conclude that both the ANN and SVM models performed well in classifying the test dataset. Based on the difference of 2% accuracy, the ANN model is used for the real time implementation in the end. Other works such as [18], [20], [24], [30], in addition showed that an ANN is the best performing approach in real time classification.

B. *Real time implementation*

For the real time implementation the ANN model is trained again with the participant data, the previous test sequence data, and another dataset from the same participant of the test sequences. Here, each gesture is performed 30 times to gain a bigger and a more personalized dataset for the real-time classification. After that, the ANN model is evaluated with ten random hand gesture recording sequences in which a real time classification is carried out. Figure 5.6 shows an example of the tenth data recording sequence. The filtered EMG signal and the assigned gesture labels as well as the predicted labels of the ANN are displayed. Obviously, the most frequent misclassifications occur at the end of a gesture.

The results of the training are given in the classification report in Table V. It shows that the ANN model achieves high precision, recall and F1 score across all labels, all values ranging from 0.94 to 0.99. The overall weighted average score of 0.97 also reflects the very good performance of the model on the given training and test set.

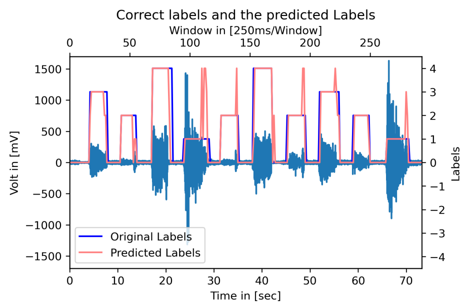


Fig. 7: Filtered EMG data (light blue), gesture labels (dark blue) and the predicted labels (pink)

TABLE V: Classification report of ANN training for real time classification

Real time performance off the ANN model				
Label	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.97	0.98	0.98	1567
1	0.97	0.94	0.95	399
2	0.94	0.98	0.96	416
3	0.97	0.95	0.96	388
4	0.99	0.95	0.97	404
Accuracy			0.97	3174
Macro avg	0.97	0.96	0.96	3174
Weighted avg	0.97	0.97	0.97	3174

TABLE VI: ANN accuracies and delay times for real time classification

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg**
ANN acc.	0.92	0.93	0.88	0.89	0.93	0.89	0.90	0.95	0.85	0.87	0.90
Time delay [ms]	0.24	0.28	0.25	45.50	0.25	8.49	0.25	0.28	7.07	0.24	6.29

To determine the performance using real time data, ten random motion sequences are again recorded, each consisting of ten gestures. Additionally, the average delay time required to process and classify the 6x250 sensor values is calculated. Hence, it is the time taken by the program after acquiring the 250 EMG sensor signals until it can start acquiring again. The results are listed in Table VI with the individual classification accuracies and delay times. Furthermore, an average value of the accuracies and the delay times is calculated at the end. The performance accuracy of the ANN model ranges from 0.85 to 0.95 with a final average of 0.90, which is 7% lower than the accuracy score of the training dataset. The delay times vary from 0.24 ms to 45.50 ms with an average of 6.29 ms.

During the recordings it became obvious that the performance of the model significantly depends on the correct positioning, skin preparation and calibration of the EMG wristband before the recording. When looking at the results of test sequence ten in Figure 7, it becomes clear that the inaccuracy of the classifier occurs mainly during a change of motion. This could be caused by the fact that the training data is not perfectly labeled by the threshold method.

C. Combination with the HoloLens

On the one hand, the connection with the HoloLens is intended to improve and automate the EMG signal labeling and on the other hand, to provide the possibility of controlling a virtual hand prosthesis via the predicted labels of the ANN. The result of the labeling approach by the HoloLens is shown in Figure 8 (red line). It is obvious that the synchronization of the data works, but the labeling is not yet very accurate.

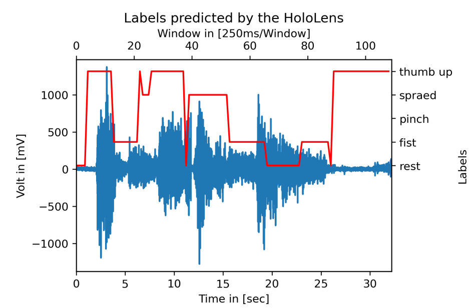


Fig. 8: Filtered EMG data from a test sequence with assigned labels from the HoloLens

The transfer of the predicted labels from the EMG wristband to the HoloLens has been successfully implemented. The hand prosthesis can now be controlled by the labels in an interval of 250 ms in addition to the delay time of the program.

IV. Conclusion

In summary, the implemented system has achieved the overall goal of real time classification. Data from six sEMG sensors are sent via an ESP32 with a USB cable to a Python program on the computer, where these are processed and classified by an ANN in real time. The average accuracy of ten test classifications is 90% with an average delay time of 6.29 ms for four defined hand gestures. For this purpose, the data are first segmented into time windows of 250 ms and filtered with a band pass filter (20-450 Hz) and a notch filter (50 Hz). Then, the features MAV, ZC, SSC, WL, VAR, IEMG, and RMS are extracted and normalized for each EMG sensor to serve as input signal for the ANN. The ANN is selected based on a performance comparison with a SVM in non-real time. Furthermore, it has been demonstrated that an average value of all sensor data is not sufficient to obtain comparable results to the individual sensor values. In the end, a combination with a HoloLens application is implemented and the EMG wristband can be used to control a virtual hand prosthesis. Moreover, it is possible to label datasets of the EMG wristband with the HoloLens. However, this feature needs to be improved since the results are not yet accurate enough to reliably train an ML algorithm. The developed system provides a basis for further investigations with larger samples or statistical analyses to explore optimal sensor configurations and to improve the accuracy of hand gesture classification. Potential future work packages could be the wireless connection of the EMG wristband and a reduction of features. The second one could be accompanied by a reduction of EMG sensors and related costs. Additionally, an interface could be developed to better control the connection and applications of the EMG armband and the HoloLens.

REFERENCES

[1] K. Guk *et al.*, "Evolution of wearable devices with real-time disease monitoring for personalized healthcare," *Nanomaterials (Basel, Switzerland)*, vol. 9, no. 6, 2019.

[2] S. M. Lee and D. Lee, "Healthcare wearable devices: an analysis of key factors for continuous use intention," *Service Business*, vol. 14, no. 4, pp. 503–531, 2020.

[3] A. Jaramillo-Yánez, M. E. Benalcázar, and E. Mena-Maldonado, "Real-time hand gesture recognition using surface electromyography and machine learning: A systematic literature review," *Sensors (Basel, Switzerland)*, vol. 20, no. 9, 2020.

[4] W.-T. Shi, Z.-J. Lyu, S.-T. Tang, T.-L. Chia, and C.-Y. Yang, "A bionic hand controlled by hand gesture recognition based on surface emg signals: A preliminary study," *Biocybernetics and Biomedical Engineering*, vol. 38, no. 1, pp. 126–135, 2018.

[5] J. G. Abreu, J. M. Teixeira, L. S. Figueiredo, and V. Teichrieb, "Evaluating sign language recognition using the myo armband," in *18th Symposium on Virtual and Augmented Reality - SVR 2016*. Piscataway, NJ: IEEE, 2016, pp. 64–70.

[6] A. Nelson, S. McCombe Waller, R. Robucci, C. Patel, and N. Banerjee, "Evaluating touchless capacitive gesture recognition as an assistive device for upper extremity mobility impairment," *Journal of rehabilitation and assistive technologies engineering*, vol. 5, p. 2055668318762063, 2018.

[7] D. V. Redrovan and D. Kim, "Hand gestures recognition using machine learning for control of multiple quadrotors." IEEE, 2018.

[8] U. Côté-Allard, G. Gagnon-Turcotte, F. Laviolette, and B. Gosselin, "A low-cost, wireless, 3-d-printed custom armband for semg hand gesture recognition," *Sensors (Basel, Switzerland)*, vol. 19, no. 12, 2019.

[9] J. M. Weiss, *Easy EMG - E-Book: A Guide to Performing Nerve Conduction Studies and Electromyography*, 3rd ed. Philadelphia: Elsevier, 2023. [Online]. Available: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=6827084

[10] A. F. T. Ibrahim, V. R. Gannapathy, L. W. Chong, and I. S. M. Isa, "Analysis of electromyography (emg) signal for human arm muscle: A review," in *Advanced Computer and Communication Engineering Technology*, ser. Lecture Notes in Electrical Engineering, H. A. Sulaiman, M. A. Othman, M. F. I. Othman, Y. A. Rahim, and N. C. Pee, Eds. Cham: Springer International Publishing, 2016, vol. 362, pp. 567–575.

[11] R. H. Chowdhury, M. B. I. Reaz, M. A. B. M. Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors (Basel, Switzerland)*, vol. 13, no. 9, pp. 12 431–12 466, 2013.

[12] Kinemic GmbH, "Geräte steuern mittels gesten und bewegungen — kinemic gmbh," 28.06.2021.

[13] "gforcepro+ emg armband_ oymotion bio-sensor," 22.02.2023. [Online]. Available: http://www.oymotion.com/en/product32/149

[14] M. Tavakoli, C. Benussi, P. Alhais Lopes, L. B. Osorio, and A. T. de Almeida, "Robust hand gesture recognition with a double channel surface emg wearable armband and svm classifier," *Biomedical Signal Processing and Control*, vol. 46, pp. 121–130, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809418301794

[15] M. Gomez-Correa and D. Cruz-Ortiz, "Low-cost wearable band sensors of surface electromyography for detecting hand movements," *Sensors (Basel, Switzerland)*, vol. 22, no. 16, 2022.

[16] S. Kanoga, A. Kanemura, and H. Asoh, "Are armband semg devices dense enough for long-term use?—sensor placement shifts cause significant reduction in recognition accuracy," *Biomedical Signal Processing and Control*, vol. 60, p. 101981, 2020.

[17] M. Ergeneci, K. Gokcesu, E. Ertan, and P. Kosmas, "An embedded, eight channel, noise canceling, wireless, wearable semg data acquisition system with adaptive muscle contraction detection," *IEEE transactions on biomedical circuits and systems*, vol. 12, no. 1, pp. 68–79, 2018.

[18] K. H. Lee, J. Y. Min, and S. Byun, "Electromyogram-based classification of hand and finger gestures using artificial neural networks," *Sensors (Basel, Switzerland)*, vol. 22, no. 1, 2021.

[19] C. Tepe and M. C. Demir, "Real-time classification of emg myo armband data using support vector machine," *IRBM*, vol. 43, no. 4, pp. 300–308, 2022.

[20] C. Motoche and M. E. Benalcázar, "Real-time hand gesture recognition based on electromyographic signals and artificial neural networks," in *Artificial Neural Networks and Machine Learning – ICANN 2018*, ser. Lecture Notes in Computer Science, V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, and I. Maglogiannis, Eds. Cham: Springer International Publishing, 2018, vol. 11139, pp. 352–361.

[21] U. Cote Allard *et al.*, "A convolutional neural network for robotic arm guidance using semg based frequency-features," pp. 2464–2470.

[22] W. Chen and Z. Zhang, "Hand gesture recognition using semg signals based on support vector machine," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. IEEE, 2019, pp. 230–234.

[23] M. F. Wahid, R. Tafreshi, M. Al-Sowaidi, and R. Langari, "Subject-independent hand gesture recognition using normalization and machine learning algorithms," *Journal of Computational Science*, vol. 27, pp. 69–76, 2018.

[24] M. E. Benalcazar *et al.*, "Real-time hand gesture recognition using the myo armband and muscle activity detection," in *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*. IEEE, 2017, pp. 1–6.

[25] J. Yang, J. Pan, and J. Li, "semg-based continuous hand gesture recognition using gmm-hmm and threshold model," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 1509–1514.

[26] M. A. Oskoei and H. Hu, "Support vector machine-based classification scheme for myoelectric control applied to upper limb," *IEEE transactions on bio-medical engineering*, vol. 55, no. 8, pp. 1956–1965, 2008.

[27] T. R. Farrell and R. F. Weir, "The optimal controller delay for myoelectric prostheses," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 15, no. 1, pp. 111–118, 2007.

[28] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the optimal window length for pattern recognition-based myoelectric control: balancing the competing effects of classification error and controller delay," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 19, no. 2, pp. 186–192, 2011.

[29] U. Cote-Allard *et al.*, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 27, no. 4, pp. 760–771, 2019.

[30] J. O. Pinzon-Arenas, R. Jimenez-Moreno, and J. E. Herrera-Benavides, "Convolutional neural network for hand gesture recognition using 8 different emg signals," in *2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*. Piscataway, NJ: IEEE, 2019, pp. 1–5.

[31] "Analog_emg_sensor_by_oymotion_sku_sen0240-dfrobot," 30.03.2023. [Online]. Available: https://wiki.dfrobot.com/Analog_EMG_Sensor_by_OYMotion_SKU_SEN0240

[32] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for emg signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

Peter Rott is master student in the Department of Medical and Health Technologies, MCI, Innsbruck, Austria and completed his Bachelor of Engineering in Mechanical Engineering, with the focus on Mechatronics at the Hochschule-Bonn-Rhein-Sieg, Germany.