Training an EMG-Based Machine Learning Model to Classify Hand Gestures in a Spatial Virtual Reality Environment

Tóki Lava Olsen
Aalborg University
Aalborg, Denmark
tolsen18@student.aau.dk

Charlotte Johansen
Aalborg University
Aalborg, Denmark
cjohan19@student.aau.dk

Kristian Hedegaard Sørensen
Aalborg University
Aalborg, Denmark
ksaren20@student.aau.dk

Abstract

This study explored the use of an electromyography (EMG) based machine learning model to classify hand gestures in a spatial virtual reality (VR) environment. A total of 18 participants participated in the evaluation. The EMG signals were recorded from the forearm of each participant, and they were used to train intra-subject classification models to predict the movement of a hand prosthetic. The results showed a significant gap between the offline and online performance of the trained model, with macro F1 scores averaging 0.86, while a notable average performance drop was observed during the online tests, where the macro F1 score fell to 0.52. Individual participants achieved more satisfactory scores, suggesting the presence of individual differences that may be influenced by various contributing factors. The results show promising steps towards training and testing a machine learning algorithm to control a hand model with EMG signals in VR. With further development, this approach has the potential to support amputees in training a prosthetic hand within a spatial VR environment prior to receiving the physical device.

Keywords Electromyography; Virtual; Reality; Myoelectric; Hand prosthetic; Training; Machine; Learning

ACM Reference format:
Tóki Lava Olsen, Charlotte Johansen, and Kristian Hedegaard Sørensen. 2025. Training an EMG-Based Machine Learning Model to Classify Hand Gestures in a Spatial Virtual Reality Environment.

1 Background

Human hands play a crucial role in everyday activities, and the sudden loss of them can be deeply traumatic and leave a person with disabilities that greatly impact a person's quality of life [6]. Amputee Coalition of America estimates that over 5.6 million people live with limb loss or difference of limbs in the US alone. Upper limp amputations only make up 17% of all amputations [3], with the average amputee in America being a 35-year-old male, according to Stearns et al [31]. Prostheses have been used for a long time, but there is still much to be desired regarding functionality and comfort. Issues such as: users being just as or more functional without it, difficult controls or discomfort such as increased weight or temperature can lead to abandonment of the prosthetic [4].
Fortunately, people suffering from upper-limb amputations can utilize myoelectric prosthetics, which pick up signals from remaining stump muscles to substitute for the original limb functions. However, the variation in the size of the residual stump, mobility, and muscle contraction, make it difficult to find one solution

tailored to all amputees [24]. Most commercially available prosthetic systems utilize low degrees of freedom (DOF), such as single DOF grippers, which are typically controlled using either binary (on/off) or continuous (proportional) signals, depending on the system. More advanced prostheses apply more DOFs and allow for more complex hand gestures such as pinches, grasps and individual finger movements, through the use of machine learning (ML) which can be trained to recognize a broader range of patterns (Pattern recognition(PR)) from electromyographic (EMG) signals [6]. ML algorithms can suffer in some areas such as a lack of robustness, therefore a certain amount of training by both the user and ML algorithm is required. The user has to get familiar with the prosthetic to generate the most distinct EMG patterns for each posture. The idea is that each movement produces a unique signal which the ML model can classify and differentiate between [18]. Woodward and Hargrove outlined two data collection methods: passive and active. In the passive method, the subject maintains a fixed arm position, with the elbow at a 90° angle and the upper arm parallel to the body. In contrast, the active method allows the subject to move their arm freely within the workspace during each movement, with an emphasis on elbow flexion and extension [33]. Generally, PR systems collect data passively. This passive data collection results in high classification rates for amputees in controlled lab conditions, but does not always translate well to real-life scenarios, where the arm is typically moving while performing gestures[33]. To account for this, active data collection can be used, studies have found that classification is strongly dependent on arm position [12, 14, 30].

The structure of the data collection seems to vary with each study, however, some patterns can be found. A window for when the gesture is supposed to be performed was found in previous research; from what we gathered, this can range from 2.5 seconds [33] to 5 seconds [6, 18]. Some specify a "relax" state where the user is not performing a specified gesture or moving to/from the gesture area. For example, a relaxed state may involve resting the arm on a chair's armrest [18].

Some of the most common gestures trained for prediction are flexion, extension, supination, and pronation of the wrist, fists, grasp of an object, and different pinches [6, 18, 23, 33, 34]. Models used in the field of predicting gestures based on EMG signals include neural networks [2, 8, 13, 18, 23, 32, 34], support vector machines [5, 8, 9], linear discriminant analysis (classification method) [6, 33], and linear regression [15, 22]. Depending on the task, for continuous control (e.g., measuring the movement along with the gesture and stopping in the middle of it to do something else), regression models are used [22], and for discrete movements e.g. open/close hand, classification models are used [5].

A range of different approaches have been used to predict the movement of a hand prosthetic based on the EMG signals from the residual muscles. Research has been made on intra-subject classification, where the algorithm is trained and tested on each subject [17, 23], and on cross-subject classification, where the algorithm is trained on one group of individuals and tested on a different group [9]. Accotli et al. compared both intra-subject and cross-subject classification, where the performed gestures were two wrist movements (flexion/extension), two wrist rotations (pronation/supination), and four hand grasps (lateral, power, bi-digital, and open), resulting in 8 total, with their arm fixed on a table. The results were more satisfactory with intra-subject classification, with a median accuracy of up to 89% for amputees, while the cross-subject median accuracy was only up to 35%. Both intra-subject and cross-subject models performed better on non-amputees [9]. This highlights the importance of calibration to account for individual differences, as done by e.g., Gandolla et al. [13]. According to the co-supervisor from the Health, Science and Technology Institute, it is not uncommon that an amputee will have to recalibrate their prosthetic every day when donning it [10].

In the context of EMG pattern, it can be divided into two kinds, steady-state (continuous muscular contractions) and transient EMG (EMG associated with the onset of the muscle contraction) [9]. Hudgins et al. discovered that the transient EMG is more descriptive of an intended movement than the steady-state EMG is [16]. A newer study by D'Accolti et al. proposed a pattern recognition controller based on the transient EMG signal, and achieved results comparable to state-of-the-art steady state EMG controllers [9]. This suggests that the transient phase of the EMG signal contains valuable information for training a model to recognize hand gestures. Both steady-state and transient EMG can be used to provide insight.

The success of the classification system largely depends on the choice of representation of the continuous EMG signal. Trying to predict solely based on the raw EMG signal will not give good results for control purposes[16]. Time domain features (derived from the raw signal over time) are often quick to implement because they are calculated directly from the raw EMG signal [25]. This type of feature is well-suited for myoelectric-controlled devices, as it supports fast calculation and prediction, helping to minimize delay perceived by the end user. A disadvantage of time domain features is that they assume the EMG signal is stationary. However, EMG signals are usually non-stationary, meaning that features can vary a lot during different actions [25].

Providing visual feedback in Virtual Reality (VR) reduces the mental effort required by amputees during the training stage of using a hand prosthesis, making it easier for them to adapt, control it reliably, and become accustomed to the device. Therefore, the virtual environment can help a patient adapt to a myoelectric prosthesis [23]. VR provides benefits in terms of embodiment and visualization, along with the ability to train the use of a prosthesis before receiving it. Specifically, virtual hand representations can enhance embodiment by inducing phantom sensations in the amputees' missing limb during VR training. When the virtual limb moves, it can increase the feeling of prosthesis ownership and make it more likely for the amputee to continue usage of the prosthesis [27].

Training and rehabilitation activities that are engaging, such as VR, can be more effective compared to conventional rehabilitation. If the activity is enjoyable, it can enhance long-term use as well [11]. Additionally, more and repeated training further promotes neuroplasticity [13].

While virtual-reality training offers clear benefits, e.g, increasing embodiment, reducing cognitive load, promoting repeatable and engaging rehabilitation sessions, successful gesture recognition still hinges on the underlying signal fidelity. EMG recordings are inherently sensitive to a multitude of factors, e.g., muscle mass, fatigue, placement and pressure of electrodes, perspiration, and arm posture [19, 20, 26].

In this report, we focus on classifying EMG signals from hand gestures performed in different arm positions. This serves as a step toward enabling amputee patients to train myoelectric control of a prosthetic hand before receiving the physical device. To promote a sense of embodiment during training, the study is conducted in a

virtual reality (VR) environment. This setup is intended to better simulate naturalistic scenarios involving active arm movement, in contrast to strictly controlled laboratory conditions where the arm remains stationary.
In the broader context, the aim is to better prepare users for real-world prosthetic use and decrease the risk of abandoning the prosthetic.

2 Implementation

The VR environment was made in Unity. It consisted of a grid with 9 cubes, 23 cm on each side, placed inside a 1.2 x 1.2 meter area with 15 cm of empty space between the cubes, and approximately 40 cm distance between the participant and the grid. There were different states the user could have in the environment. (1) resting, (2) moving to box, (3) in box, (4) moving to the rest position. Logging of data was separated into two CSV files. One for the raw EMG data and the other for unity data, such as HTC tracker x,y,z positions (see the worksheets for detailed logging information). The system was designed to capture both offline and online performance metrics of the model. Offline performance refers to the classification results obtained from a held-out test set, providing a static evaluation of the model's accuracy. In contrast, online performance refers to the model's real-time predictions, where it processes live data and delivers real-time feedback to the user.

2.1 Unity Data Collection Flow

The beginning of the training started with the user performing a Maximum Voluntary Contraction (MVC) for a calibration period of five seconds. The 30% and 60% thresholds were calculated as the percentage of the MVC and displayed on a slider next to the cube where they had to place their hand inside. The participant's muscle activation was calculated as the exponential moving average (EMA) smoothing during training and shown on the slider for the participant so they could collect data at 30% and 60% of their MVC. The slider can be seen in figure 1. Next, the participant placed their hand in a comfortable resting position, and this was set as their "rest position" throughout the data collection. A random cube in the grid would turn blue, indicating that the participant should move their arm into that cube. A "ghost hand" signifier shows the gesture the user should perform in the cube, see figure 1.

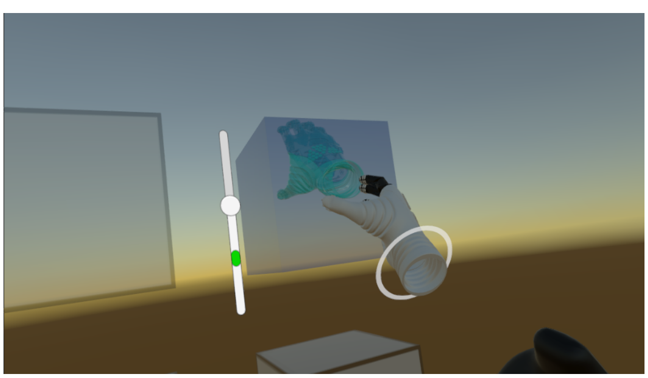

Figure 1. Screenshot from VR recording, showing the ghost hand, and the slider showing the contraction force

When they were in the correct position, the cube would turn green and the "ghost hand" would disappear. The given gesture was held for 5 seconds as seen in [6, 18]. After the 5 seconds, the user would be prompted to go back to their rest position and rest for 3 seconds. This approach ensured a consistent flow in the system, which was maintained until the data collection was completed. Figure 2 shows the structure of the Unity data collection flow for the user, which is obtained to later test the model and provide offline performance metrics.

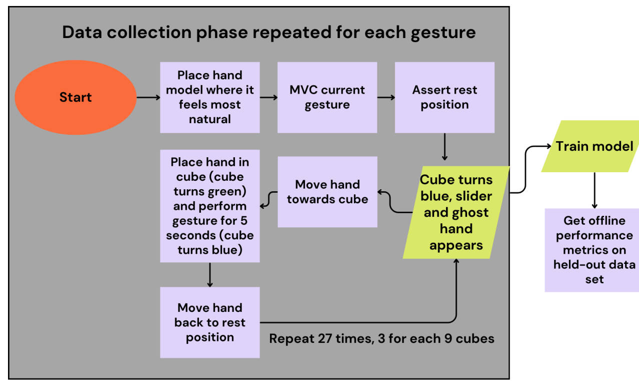

Figure 2. Figure showing the structure of the Unity data collection flow for the user. The first section is to be repeated until all repetitions are done, then the model will provide the offline results from the performance metrics.

2.2 Unity Online Test Flow

The online test flow was slightly different than the data collection flow. The core of the system was the same, with the same grid in the same position. There was no slider to indicate 30% and 60% and no MVC. The beginning of the data collection started with a calibration of the model, where the user would perform each gesture for 3 seconds, with a 0.5-second rest between them. This allowed us to record the mean probability for each gesture. Next, they would, as before, place their hand in a comfortable resting position, and the resting position was set. A random cube would turn blue, and a text would appear with the gesture they were supposed to do. The participant then inserted their hand and performed the mentioned gesture. Each gesture was performed nine times in random cubes, resulting in a total of 36 cube interactions. After completing this segment, a single cube appeared on the screen. When the participant touched it, the cube would move. It traces a square path, right, up, left, and down, with each side roughly spanning the arm's reach. Participants were instructed to perform a gesture and follow the moving cube using the virtual hand. This was repeated for all four gestures.

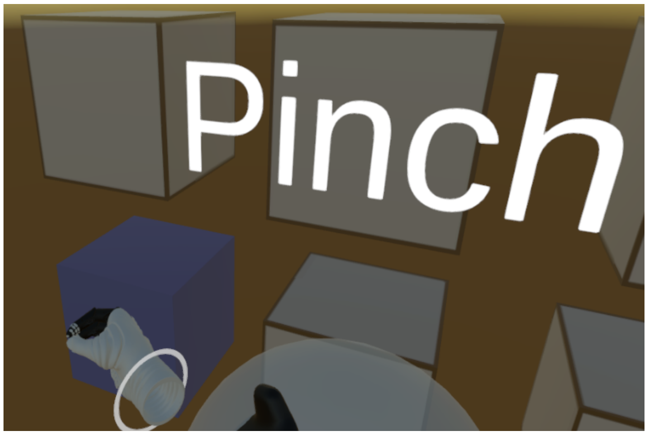

Figure 3. Image of the test scenario. The image shows the hand model, a portion of the cube grid, a highlighted blue cube indicating the target for the participant to move into, and an instruction prompting the participant to perform a pinch gesture once inside the cube.

Figure 2.2 shows the structure of the online test of the model, which is used to create a classification report for the model while running in real-time and provide live feedback for the user.

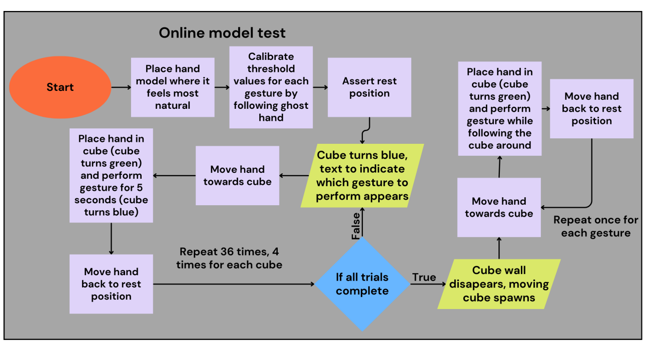

Figure 4. Figure showing the structure of the Unity data collection flow for the user during the online test.

3 Data Preprocessing

The data stream from the raw EMG signals is noisy, which can lead to wrong predictions in a model. Figure 5 shows an example of the raw EMG data for the MVC, three resting phases, and three gesture phases (denoted "in box").

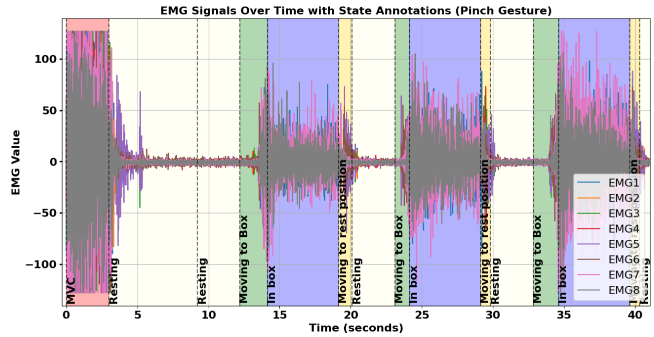

Figure 5. EMG signals over time for three performances of the pinch gesture.
Red: MVC, White: Resting, Green: Moving to cube, Blue: In cube, Yellow: Moving to rest position

Before performing any further analysis, we started by merging and cleaning the data. The data gathered was, as mentioned, collected in two CSV files; these needed to be merged to get the full range of the data.
The EMG data was collected at 200 Hz, while the Unity data was sampled at 90 Hz. When merging the two datasets, Unity values are forward-filled, repeating the last recorded value until a new value is available. This ensured there were no empty rows in the merged dataset.
The data cleaning was done in 5 steps.

- Trim all data before the MVC entry. Everything before that is unnecessary.
- Create an index column counting rows.
- Cut off data after the last gesture.
- Remove extra spaces from text columns, except ID.
- Save the cleaned version, while also keeping the originals untouched.

The data was split into sliding windows of 200 ms, with 100 ms overlap, since the nature of the data is continuous. The feature extraction happened per window and is explained in section 3.1. Since the goal gesture label was a string, we encoded it as an integer using a label encoder to ensure compatibility with the model input. Next, the data was standardized using Z-score normalization. Lastly, the data was collected in the same file again with: Labels, Tracker position, Cube position, and the extracted features.

3.1 Feature Extraction

As mentioned, the data was windowed in 200 ms with 100 ms overlap, and the feature extraction calculations were done on each EMG channel in each window. Thus, from each window, the features were extracted.
The time domain features used in this project were mean absolute value, mean absolute value slope, zero crossings, slope sign changes, and waveform length.

3.1.1 Mean Absolute Value

The Mean Absolute Value (MAV) is a popular feature used in EMG signal analysis (e.g., used by Hudgins et al. [16], Martinez et al. [22], D'Accolti et al. [9], and Mattioli et al. [23]). MAV is an estimate of the average absolute value of the signal, X_i within segment i, which contains N samples. It can be calculated as:

$$\bar{X}_i = \frac{1}{N}\sum_{k=1}^{N} |x_k| \quad \text{for } i = 1, \ldots, I \tag{1}$$

where X_k is the kth sample in the segment i and I is the total number of segments in the sampled signal.
As defined by Hudgins et al. [16].

3.1.2 Mean Absolute Value Slope

Mean Absolute Value Slope is the difference between the sums of adjacent segments, i and $i + 1$, and is defined by:

$$\Delta\bar{X}_i = \bar{X}_{i+1} - \bar{X}_i, \quad \text{for } i = 1, \ldots, I - 1. \tag{2}$$

As defined by Hudgins et al. [16].

3.1.3 Zero Crossings

Zero crossings (ZC) is used to capture the frequency information of the EMG signals, defined as the number of times the signal's amplitude crosses zero. A threshold is necessary to mitigate the impact of noise in the data, as it will help avoid low-voltage fluctuations or background noise [16, 25]. Given two consecutive samples x_k and x_{k+1}, incrementing the zero crossing count if:

$$\begin{gathered} x_k > 0 \text{ and } x_{k+1} < 0, \text{ or} \\ x_k < 0 \text{ and } x_{k+1} > 0, \text{ and} \\ |x_k - x_{k+1}| \geq 0.01\,\text{V}. \end{gathered} \tag{3}$$

As defined by Hudgins et al. [16].

3.1.4 Slope Sign Changes

Slope Sign Change (SSC) is another way to represent frequency information of the EMG signal. It counts how many times the slope of the signal changes direction—from increasing to decreasing or vice versa [16, 25]. Like with zero crossings, a threshold is used again. Given three consecutive samples $x_{k\text{-}1}$, x_k, and x_{k+1}, the slope sign change count is incremented if:

$$\begin{aligned} & x_k > x_{k-1} \text{ and } x_k > x_{k+1}, \text{ or } x_k < x_{k-1} \\ & \text{and } x_k < x_{k+1}, \text{ and } |x_k - x_{k+1}| \geq 0.01\,\text{V} \\ & \text{or } |x_k - x_{k-1}| \geq 0.01\,\text{V}. \end{aligned} \tag{4}$$

As defined by Hudgins et al. [16].

3.1.5 Waveform length

Waveform length (WFL) gives a sense of how complex the signal is over a time window. Adds up all the small changes between each point in the EMG signal. The total length reflects how much the signal is changing, which give insight into amplitude, frequency, and duration of the EMG signal [16, 25].

$$l_0 = \sum_{k=1}^{N} |\Delta x_k| \tag{5}$$

where $\Delta x_k = x_k - x_{k\text{-}1}$ (the difference in consecutive samples voltage values). As defined by Hudgins et al. [16].

3.1.6 Delta Values

According to the background research, the transient EMG signal contains information about the gesture. This is why we were also interested in the gesture onset, which we identified using Delta values.

The features mentioned in section 3.1.1 through 3.1.5 were used as they were, but the window-to-window delta values for each feature were also calculated. These were also used as features, but their most important use case was to give an indication of gesture start and end. For this, three variables were used: an *Onset* flag, *GestureStart* flag, and *GestureEnd* flag.
If the delta values changed a certain amount across all the EMG channels, it would trigger the *Onset* flag. *GestureStart* was set to true when the onset was true after being false, and *GestureEnd* was set to true when the onset was false after being true. The output was a gesture window, which is the window the model will be trained on, which is from gesture start to gesture end.

3.2 Feature Visualization

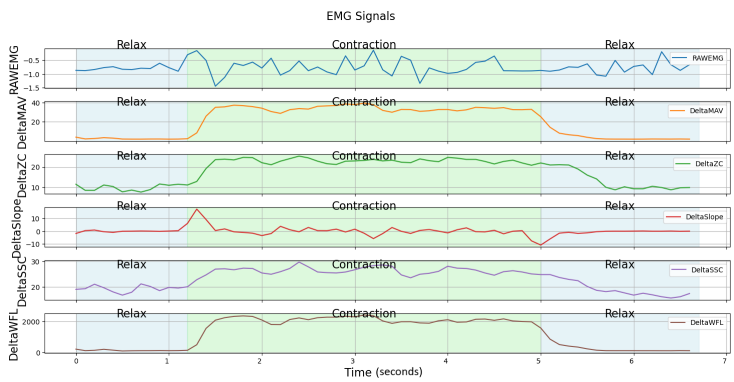

Figure 6. Time-series visualization of extracted EMG features from top to bottom during a fist gesture: Raw EMG, Mean Absolute Value (MAV), Zero Crossings, Slope, Slope Sign Change (SSC), and Waveform Length (WFL) averaged across all 8 EMG channels for visual purposes. Each subplot shows the feature response over time during gesture execution. There is a noticeable increase in the features during muscle contraction.

Figure 6 shows the raw EMG signal along with the extracted features. As expected, the raw EMG signal is noisy and difficult to interpret directly. Each subplot illustrates the effect of a specific transformation on the raw signal. E.g., the Mean Absolute Value (MAV) feature captures the overall intensity of the muscle activity and shows a clear rise and fall that aligns with muscle contractions.

4 Training of Model

We chose the Random Forest model, based on the results presented in the worksheets, where Random Forest (RF) was compared with Stochastic Gradient Descent (SGD) Classifier, Neural Network (NN), Support vector machine (SVM) classifier, and Linear discriminant analysis (LDA). These models were chosen for comparison with Random Forest as they represent some of the most commonly used approaches for this type of classification, as outlined in the background section.

Model	A.	P.	R.	F1
RF	0.94	0.94	0.93	0.94
SGD	0.86	0.89	0.87	0.87
NN	0.90	0.92	0.89	0.90
SVM	0.86	0.86	0.86	0.86
LDA	0.80	0.82	0.80	0.80

Table 1. Comparison of classification performance across different models, in offline test. All scores are macro values.
A = Accuracy, P = precision, R = Recall.

For each participant the model was trained on two hand gestures (1) three finger pinch (thumb, index and middle finger) (2) closed fist and two wrist motions: (3) extension and (4) flexion, see figure 7 for visualization. All four will be referred to as "gestures" going forward for simplicity.

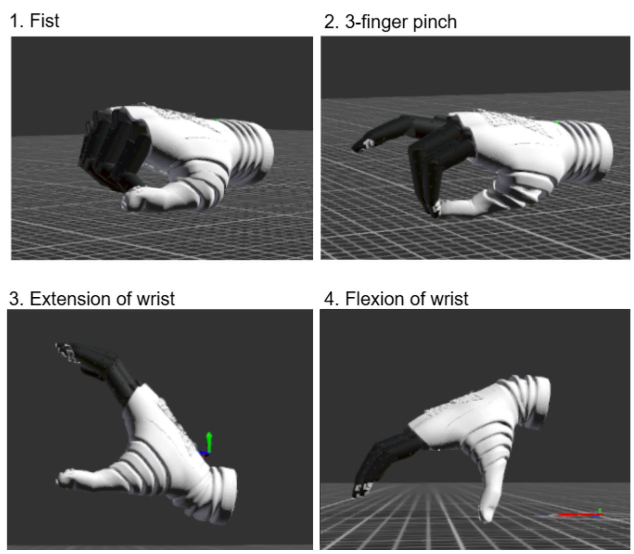

Figure 7. Figure showing the four gestures

According to Accotli et al.'s results, intra-subject models perform more satisfactorily [9]. This was also evident in our own findings during early testing, which led to the decision for all users to train their own model to achieve better accuracy. Users trained the model one gesture at a time to simplify both the user experience and the data sorting process. Figure 8 shows a diagram of the flow of the training, to provide an overview of the system from data collection to a model ready for testing. The feature extraction points towards the features explained in section 3.1

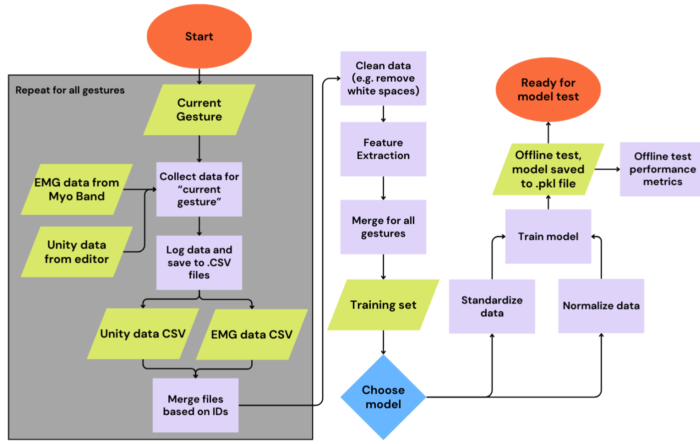

Figure 8. This diagram illustrates the data acquisition flow during Unity-based data collection. It represents the system-level architecture corresponding to the process described in Section 2.1.

4.1 Features used for training

The features used for training of the model are the ones extracted in section 3.1 amongst other features, they are as follows:

- MAV and ΔMAV for 8 channels,
- MAV slope and ΔMAV Slope for 8 channels,
- Zero Crossings and ΔZC for 8 channels,
- Slope Sign Changes and ΔSSC for 8 channels,
- Waveform length and ΔWFL for 8 channels,
- HTC vive tracker x,y, and z,

resulting in 83 features.

5 Random Forest online prediction process

EMG data were sampled at 200 Hz and processed in sliding windows of 40 samples with a 20-sample overlap. Each window contained 11 columns of data (8 EMG channels and 3 HTC Vive tracker positions), resulting in a feature vector of 440 elements, which was transmitted to a Python server for calculating features and classifying based on the features. Python then returned a predicted gesture and a probability for that prediction. Before the trial with the cubes started, we wanted the participants to perform each gesture for 3 seconds with half a second rest between each gesture, one by one. This allowed us to record the mean probability for each gesture. We used this mean probability as a threshold for valid predictions made by the model. If the model returned a prediction above the mean probability threshold, that prediction was pushed into an array of size three. If all elements in the array contained the same prediction values, e.g., [3, 3, 3] (3 is the pinch gesture), this satisfied a uniformity check, meaning all predictions agreed, and the pinch animation was then played. The array was checked each time a new prediction was returned by Python to see if all elements were the same. This ensured that a new animation did not play each time a prediction was made, since there were roughly 2-3 predictions each second. This was slower than the window size transferal, due to multiple factors, e.g., the calculations, the transferring of the data, and instances where the uniformity check was not satisfied across multiple predictions, resulting in a longer time between final predictions. If the MAV across channels was below a predetermined threshold of low muscle activity, we pushed a 4 into the array(4 is the key for the "rest" animation).

6 Evaluation

According to Lock et al. [21], differences can be found between offline and online testing of a model, where completely new data is presented to the model. The goal of the test was to assess online performance alongside the user's experience of embodiment of the virtual hand representation.

6.1 Participants

18 people participated in the evaluation (6 female, 12 male), and they were aged 22-30. The participants were all able-bodied.

6.2 Materials and setup

The materials for the evaluation were a computer for running the VR environment, an HTC Vive VR headset, one controller, one HTC Vive tracker, and a MyoBand for measuring EMG signals. The setup can be seen in figure 9

Figure 9. Picture of the setup while wearing the VR headset and performing a gesture.

6.3 Method

The evaluation took place in a room big enough for outside-in tracking for a VR setup with a chair for the participant. The evaluation had two main phases: data collection for the model and online testing of the model. A visual overview of the evaluation process showing the different phases can be seen on figure 10.

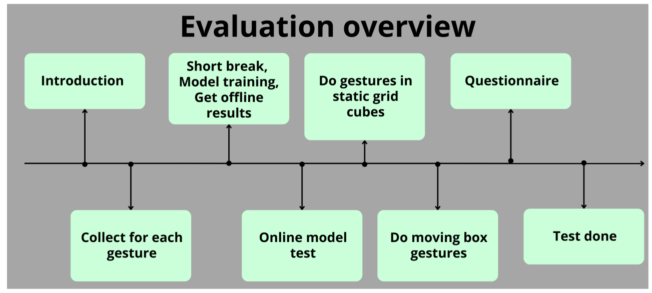

Figure 10. Overview of the evaluation process

Prior to participation, the subjects were informed about the project, its goals, and how their data would be used. All participants provided written informed consent.
Participants were introduced to the VR environment and the gestures they would be performing. This was done through images of the gestures and the VR environment, while outside of VR, on a piece of paper, and with verbal explanation from the facilitator.

6.3.1 Data Collection Phase

The participant collected data for one gesture at a time; they performed MVC of the current gesture, and then performed the gesture three times in all cubes in random order. The flow of the system during training is explained in section 2.1. EMG data were collected throughout this phase for model training.

- Participants were guided through each gesture to ensure they understood how to perform it.
- Assistance was provided as needed to ensure proper understanding and execution of gestures.

When they had completed each gesture three times in each cube, the data collection phase ended.

6.3.2 Post data collection

Following the data collection, participants were given a short break outside of VR, while the classification model was trained on their individual data they just created.

6.3.3 Online Model Performance Test Phase

Before the testing phase, participants were briefed on the updated procedure. They were informed that they would control a virtual hand using their EMG signals.

- Participants performed the prediction probability calibration.
- Participants placed their hand in a highlighted cube and performed the given gesture for five seconds, after which they went back to the rest position for three seconds, and a new cube appeared. Each gesture was performed nine times in random cubes. Unlike the training phase, the gesture order was counterbalanced using a Latin Square design, rather than focusing on one gesture per block.
- After completing this segment, the single moving cube appeared as mentioned in section 2.2

6.3.4 Post Online Testing

Participants completed a final short questionnaire (see section 6.4) regarding their sense of embodiment with the virtual hand and their experience performing the movements.
Upon completion, participants were thanked for their time.

6.4 Questionnaire

The questionnaire that the participants were asked to fill out was the Virtual Embodiment Questionnaire [28]. It addresses three factors: Virtual body acceptance refers to the perception of the virtual hand being their own. Agency, which refers to the perception of control over the virtual hand, and through the virtual hand, control events in the environment. Change refers to the change in the perceived body schema due to the stimulation [28].
The purpose of this questionnaire is to explore whether our VR hand representation can enhance the sense of embodiment of the hand, which was found to be important in the background section. We aim to determine if this implementation has the potential to produce such effects. The questionnaire is scaled from 1-7 (1 = strongly disagree, 7 = strongly agree).

7 Results

This section covers the results from the online test, comparing them with the offline test and the questionnaire.
The online performance metrics were calculated by comparing all the predictions the model made to the "goal gesture", from when the hand was in the cube.

7.1 Comparison of Online and Offline Performance Metrics

The average classification report is shown in table 2, to compare the average online performance metrics to the average classification report from the offline evaluations using a held-out (test) dataset. A clear disparity is seen between the two. A Wilcoxon test that compared gesture-wise macro performance showed statistically significant differences between the train and test scores for all metrics and all gestures, with no p-values higher than $p < .001$.

	Off. P	On. P	Off. R	On. R	Off. F1	On. F1
Extension	0.94	0.62	0.87	0.50	0.90	0.50
Fist	0.88	0.60	0.87	0.43	0.88	0.48
Flexion	0.94	0.70	0.83	0.60	0.88	0.63
Pinch	0.77	0.47	0.88	0.56	0.82	0.45
Macro	0.88	0.60	0.86	0.53	0.86	0.52

Table 2. All participants' offline and online average performance metrics.
Off. = Offline, On. = Online, P = Precision, R = Recall.

Figure 11 shows a violin plot of the macro F1 scores for each gesture across all participants.

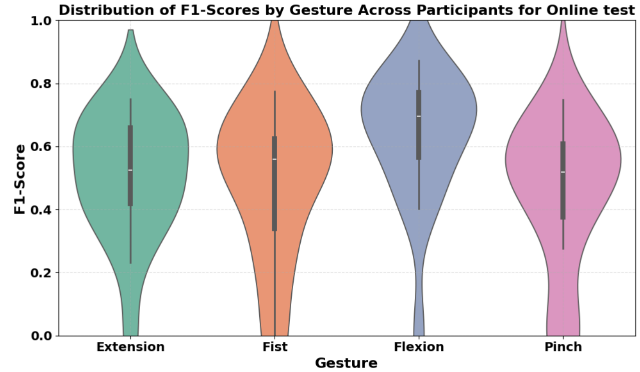

Figure 11. Violin plot showing macro F1 scores for each gesture across all participants

A confusion matrix of the online test results across all participants can be seen in figure 12. The online confusion matrix shows that the gestures that were most often mistaken are pinch and extension, followed by pinch and fist. Compared to the offline confusion matrix (see figure 13), where the most common wrong prediction was flexion predicted as pinch. It can be seen from the confusion matrices that the online test generally made more wrong predictions than the offline performance metrics.

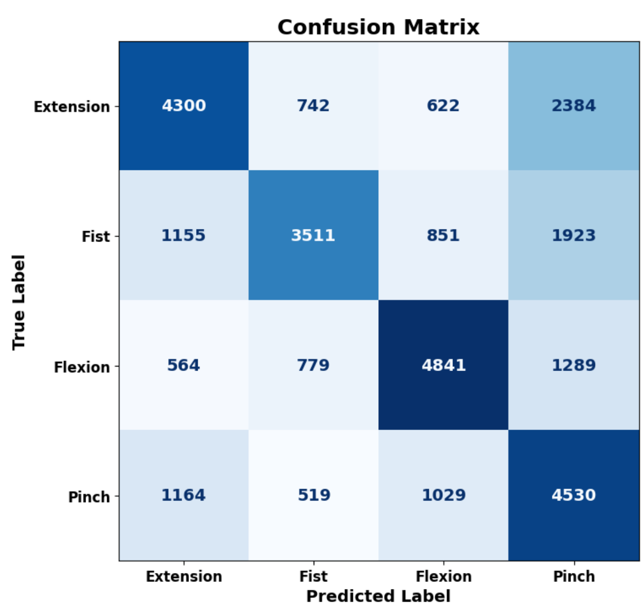

Figure 12. Confusion matrix of the results from the online tests across all participants

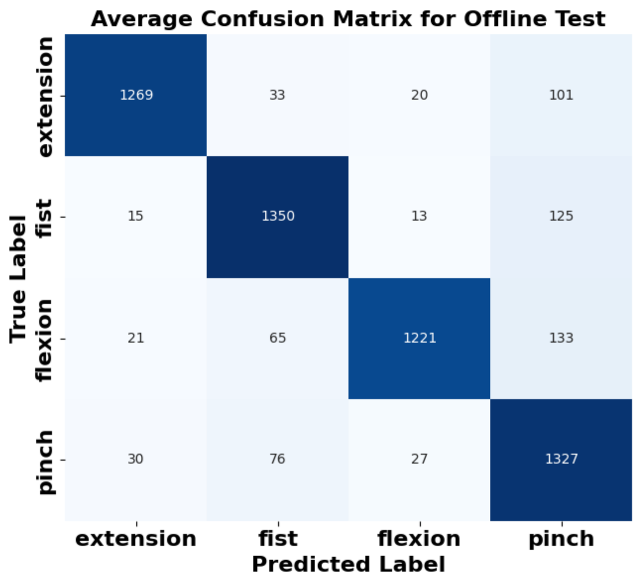

Figure 13. Confusion matrix of the results from the offline tests across all participants

Figure 14 shows the average learning curve for the model, plotting training and validation accuracy against the training set size. The training set size was a data sample across all participants. The accuracy for training and validation plateaued at approximately 3000 samples; therefore, only results up to 4000 samples are shown. The average training set size for all participants was approximately 6200 samples.
The shaded area around each curve represents one standard deviation, illustrating the variability in the data. Wider bands indicate greater variability between participants. Training accuracy remains high across all training sizes, with a slight decrease as the training set size increases. Validation accuracy improves steadily with increasing amount of training data until it reaches around 3000 in training set size.

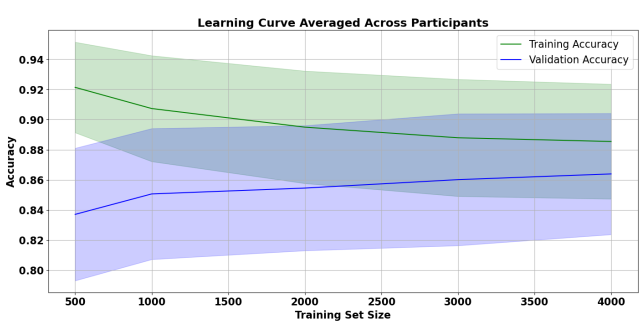

Figure 14. Learning curve showing training and validation accuracy as a function of the training set size, with the highlighted area representing the variability in the data.

For the individual results, figure 15 shows the macro F1 score for each participant.

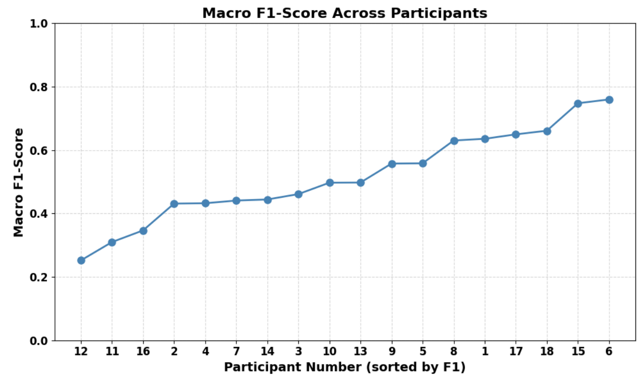

Figure 15. Point graph of macro F1 scores for each participant

As can be seen in figure 15, there is a big gap between the individual F1 scores for the participants. The participant who has the highest F1 score is participant 6, where the online test and offline test metrics gap was not as big as the average gap. The performance metrics for participant 6 can be seen in table 3, with a comparative figure on figure 16

	Off. P	On. P	Off. R	On. R	Off. F1	On. F1
Extension	0.97	0.73	0.84	0.76	0.90	0.75
Fist	0.84	0.87	0.90	0.62	0.87	0.73
Flexion	0.96	0.96	0.90	0.70	0.93	0.81
Pinch	0.80	0.69	0.91	0.82	0.86	0.75
Macro	0.89	0.81	0.88	0.73	0.88	0.76

Table 3. Participant 6 performance metrics.
Off. = Offline, On. = Online, P = Precision, R = Recall.

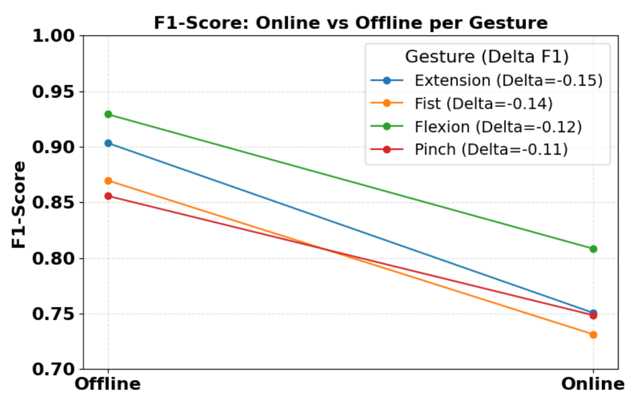

Figure 16. F1 scores for each gesture for participant 6.

To see if there was a significant difference between online and offline performance for participant 6, who was the highest scoring participant on F1 score, a Wilcoxon signed rank test was performed. This was done because the average results for all participants showed significant differences, and we wanted to examine whether this was the case for this individual as well. We did not check for normality or equal variances because the assumption of

normality is unreliable when the sample size is small, which is why the Wilcoxon signed-rank test was conducted to compare offline and online performance across the gesture specific values for recall, precision, and F1-score. Due to the small sample size (n = 4, one data point per gesture), the statistical power of the test is limited. Each data point represents the performance of a single gesture in the two conditions (offline vs online).
The test revealed no significant difference between the dependent variables: precision, $z = -0.80$, $p = .50$ (median difference = 0.055), recall, $z = -1.64$, $p = .13$ (median difference = 0.145), or F1 score, $z = -1.64$, $p = .13$ (median difference = 0.1300).
These results suggest no significant difference between the offline and online performance metrics across the evaluated metrics for participant 6.

7.2 Individual cube results

In addition to the overall results of the online testing, we looked at the classification results for each gesture at each cube's position, (see figure 17 for individual cube positions) to see if there were any significant differences in how well the model performed at various arm positions. We chose to focus on the cubes' position instead of the specific X/Y coordinates for the cube, as the cube represents an area in Unity space. It is important to note that a participant's hand could have been at multiple coordinates inside the cube. The layout of the cubes with their given numbers in Unity can be seen in figure 17. After determining that the data was parametric, a one-way ANOVA was conducted to examine the effect of cube position on macro F1 scores across participants. The effect of cube position was not statistically significant, $F(8, 153) = 1.17$, $p = .319$.

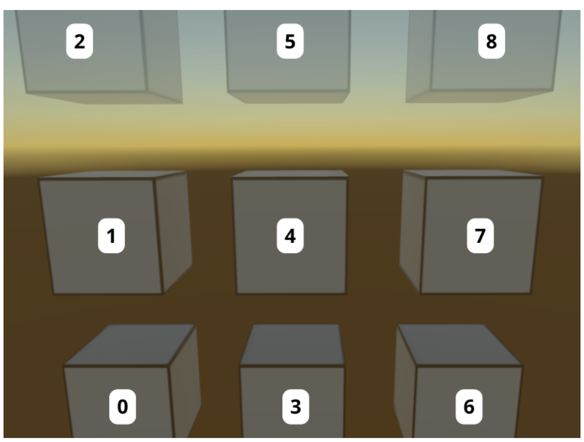

Figure 17. Position of the cubes in the Unity application

7.3 Moving Cube

At the end of the online test, a moving cube was implemented to test how well the model predicts while the arm is in motion.
The online precision, recall, and F1 scores for the moving cube are presented in table 4. There is no offline test for the moving cube, as we did not have the participants train while moving their arm.

Gesture	Precision	Recall	F1-score
Extension	0.17	0.16	0.15
Fist	0.08	0.12	0.09
Flexion	0.15	0.13	0.1
Pinch	0.07	0.12	0.09
Macro Avg	0.12	0.13	0.11

Table 4. Online performance metrics for moving cube

Shapiro–Wilk tests indicated that the F1-score distributions per gesture, per participant, significantly deviated from normality ($p < .001$ for each gesture). To assess the equality of variances, a Levene's test was conducted, which indicated no statistically significant difference in variances across the groups ($p = .75$). We checked if there were any indications of significant differences between the F1-score based on which direction the cube moved, with a Kruskal–Wallis test. This was to determine if the model performed differently depending on which direction the arm was moving. Each gesture was only performed once per moving cube, per participant, leading to a low sample size for the Kruskal-Wallis test, which limits the power of the test.
The test showed no significant difference between the movement directions, $H = 0.360$, $p = .948$. This indicates no significant difference between how the model performed, no matter which direction the participant's arm moved. This suggests that the model is consistent in performance and does not depend on the direction the arm moves.

7.4 Virtual Embodiment Questionnaire (VEQ)

Acceptance (4.75 out of 7) and agency (5.1 out of 7) scores were relatively high, indicating that participants experienced a mild sense of body acceptance and agency during the test. The Change (3.4 out of 7) score, which refers to the change in the perceived body schema due to the stimulation, is closer to the midpoint of the scale. This suggests that participants experienced a moderate or neutral perception of changes in their own body schema, indicating that the virtual hand did not lead to strong alterations in how participants perceived their own hand. The results can be seen in Table 5

Acceptance (Ownership)	Control (Agency)	Change
4.75	5.1	3.4

Table 5. Average scores for all participants on the VEQ. The scores range from 1 to 7.

8 Discussion

As shown in the test results in section 7.1, there is a significant gap between the online and the offline performance metrics, and we believe that it is possible to decrease this gap through additional practice or training of the movement by the participants.
Scheme et al. states that in general, there can be a disparity between classification accuracy and real-time usability [29]. Furthermore, Lock et al. found that systems with good accuracy did not always help users perform better in virtual/clinical online tests, because accuracy can be misleading [21]. Abbaspour et al. presented findings similar to ours, such as online performance being worse than offline performance. They tested several models, with their best

performing model in the online condition being a multilayer perceptron (feedforward NN) with the offline average accuracy being 91.1% and online average accuracy being 69.8% [1], making the difference in accuracy for offline and online 21.3%. We did not use accuracy since it can be a misleading metric compared to F1-score, with the difference between our F1-scores in offline and online averages being 0.34. This demonstrated the same pattern with a lower prediction score when testing online compared to offline.

Abbaspour et al. also demonstrated differences subject- and gesture-wise (10 gestures) across nine machine learning models and found the gap across all models [1]. They reported a difference in individual participants' performance [1], which is similar to what we observed as well. We observed instances of participants, such as Participant 6, where the model worked well. On the other hand, we also observed cases at the opposite extreme, where the model would never predict a certain gesture (e.g., extension), which makes the average value fall considerably.

The relatively high ecological validity in this test came at the cost of some accuracy, which was also clear in the results. To increase performance further, it might be necessary to explore richer feature sets, such as using time-frequency domain features, or further participant-specific fine-tuning to improve classification even more.

The online test confusion matrix in figure 12 revealed that the model often confused the pinch gesture with extension and fist. One possible explanation for the pinch/extension confusion is that a slight wrist extension often occurs naturally during a pinch, which may cause EMG data for these gestures to appear more similar. The confusion between pinch/fist could stem from participants either using more fingers than instructed during a pinch or not fully clenching their fist. This results in gesture signals being less distinct and more difficult to differentiate.

The moment a participant entered the cube, we calculated whether the predicted gesture was the same as the goal gesture. This could have led to many prediction errors, given that the participant was instructed to perform the gesture once they were in the cube. Another issue with this approach was that the model had a delay for when it was able to predict, as explained in section 5, meaning there would inevitably be incorrect predictions during the first second, which led to lower performance scores.

During online testing of the model, the participants were prompted on the screen as to which gesture they should perform when they put their hand in the cube. As participants were less guided through the online test, and the variability had increased, stemming from all the gestures being performed in the same session, some mistakes were observed. Some participants mistook e.g., flexion for extension, which would have yielded wrong predictions. Participants might have felt less confident in their gesture performance, which potentially resulted in less accurate performance of the gestures or performance with more variability.

Something worth noting when reading and interpreting the results from this study is that it has not been tested or evaluated in any way by amputees. All participants were able-bodied and healthy individuals. It is also worth noting that controlling EMG signals can be challenging; effective control typically requires training [18]. Results from a single trial may not accurately reflect the participants' and models' potential performance.

8.1 Embodiment

The questionnaire results showed a mild sense of body acceptance and agency, which is important for the argument of providing good visual feedback that can reduce the mental effort during the training stage for their prosthetic [23]. The results can also partly stem from the fact that VR is known to provide benefits in terms of embodiment[27], and not so much from the hand and animations specific to this application. One factor that may have influenced the VEQ results is the delay associated with triggering the animation for the predicted gesture, which was governed by a uniformity check. Specifically, the participant was required to insert their hand into the interaction cube and perform a gesture, after which the system would wait for three consecutive identical predictions exceeding a certain confidence threshold before initiating the animation. During the testing of the model, we occasionally encountered issues where the virtual hand failed to visually animate the predicted gesture, despite Unity successfully receiving the gesture classification. As a result, participants received no visual feedback corresponding to their physical hand movement. This loss of feedback may have disrupted the sense of agency and acceptance of the virtual hand, which might affect the participant's response in the VEQ, particularly in relation to perceived control and embodiment of the hand. When this issue was encountered, the participants were told to continue testing and ignore the static hand, since we still got the gesture classification through Unity.

We observed that when the model produced accurate predictions, the animation was typically triggered within approximately one second. In contrast, when the model's predictions were less consistent, a longer delay, spanning several seconds, was often observed before the animation was activated. The standard maximum undetectable delay for users of myoelectric control has been accepted to be delays less than 300 ms [29]. Our method exceeds this threshold by a considerable margin, even for the most favorable prediction conditions, which could have affected the embodiment scores from the questionnaire.

8.2 Thresholds and Calibrations

The rest gesture for the hand was inconsistent, as the rest threshold was determined by calculating the MAV across all channels. If the calculated MAV is below the threshold set for low muscle activity, it plays the rest animation. However, the threshold for low muscle activity was predetermined rather than individually calculated. As a result, the model frequently misclassified gestures during rest, because the participant's muscle activity often exceeded the fixed threshold. As stated by Carvalho et al., using a single threshold to find "rest" can lead to false positives in noisy signals [7], which seemed to be the case from our results. Going forward, we suggest calculating low muscle activity thresholds for each participant if the threshold method is to be used. This led to confusion between the rest animation and other animations as the model continued making predictions even when the participant was actually resting, which might have led to a lower sense of agency.

A limitation in our calibration process was that gesture thresholds were based on the model's predicted labels rather than the ground truth. For instance, if a participant performed a "Fist" gesture but the model predicted "Pinch," that incorrect prediction was recorded under "Pinch." This led to misclassified data skewing the average probabilities for some gestures, making the thresholds inaccurate, especially for gestures prone to misclassification.

8.3 Moving Cube

The results for the moving cube were not promising. Our initial suspicion lies in the fact that no training data was collected while the participants actively moved their arms, which in turn meant that the model had no reference for the patterns of a moving arm. Additionally, the model generated continuous predictions, resulting in approximately 100 prediction entries per gesture during each moving cube phase. Consequently, a single misclassification spanning across several seconds could disproportionately impact the overall results for that entire moving cube segment. With the low sample size, wrong predictions would have a significant impact on the performance metrics.

When testing the model, we implemented a moving cube to see how the model would perform while the arm was moving. We did not train the model on any gestures in movement, which potentially could have helped make the model more robust. The way the model works now breaks the gesture down into multiple parts, moving the arm to a desired location and then performing the gesture once there, whereas in a real scenario, the gesture is typically done more smoothly in one continuous movement. Training of the gestures, both being performed and held in movement, could potentially contribute to a more robust model.

8.4 Individual differences and variables

There are several variables to take into account for the variation in F1-scores observed across participants. The quality of EMG signals can be influenced by numerous factors, including muscle fatigue, perspiration, muscle configuration, muscle mass, shifts in electrode placement, and changes in arm posture. Based on our tests, we speculate that the build of the arm might have had an impact on the EMG signals, which, according to the co-supervisor [10], may be because of the muscles being more separated and the patterns being more discriminable.

While there were clear differences between some of our participants, the overall sample was still homogeneous, with all participants being healthy adults between the ages of 22 and 30. As stated by Stearns et al, the average amputee is slightly older than our sample at 35, but our gender breakdown is closer to their findings, with two-thirds of our participants being male [31].

To minimize variability related to electrode displacement, we ensured that participants did not remove the Myo armband at any point during training and testing. However, this precaution may have introduced another source of signal degradation, namely, perspiration. It is well documented that perspiration can affect EMG signal quality, and given that participants wore the armband continuously throughout the experiment, this factor may have contributed to reduced signal clarity in some cases, which could partially explain the variability in model performance observed across participants.

Several participants reported experiencing fatigue in their shoulder or arm during the training and testing of the model. As mentioned in the background, fatigue, among other factors, can affect the EMG signal, which may have contributed to the difficulty in accurately predicting some gestures. It might have been more appropriate to have the participant relax for longer, or have them back another day to test the model to reduce fatigue factors. This would pose other problems, such as making it harder to place the Myo band in the exact same place again, and the participant having time on another day.

8.5 Training and Data collection

During the model's training phase, we noticed that after some participants performed their MVC, the calculated 30% and 60% effort levels were sometimes too low to capture the full range of motion involved in the gesture without exceeding the 30% mark. For instance, when a participant performed a fist, at 30%, they might only partially close their hand, which would lead to not providing a complete fist gesture. This results in the model being fed with partially imprecise data on what the pattern for a fist was. The opposite problem was noticed a few times for pinch at the 60% mark, where the participant had to pinch down too hard, such that they lost the correct posture for the fingers and almost clenched their whole hand. The improper performance of a gesture (either too hard or not complete) could significantly impact the data and how well the model can differentiate between gestures, such as fist and pinch. In particular, if participants perform a gesture at a "comfortable" effort level during testing, it may not align well with either the 30% or 60% effort levels data recorded during training, leading to misclassification.

The workaround for this problem could have been to also provide the 30% and 60% slider during the online testing, to provide visual feedback as to how hard their contraction was. In addition to effort level mismatches, physical factors related to the experimental setup may also have affected prediction accuracy. The fixed distance between the participant and the 3x3 cube grid was chosen to allow comfortable reach to all cubes. However, during testing, it was observed that participants with shorter arm lengths experienced difficulty reaching certain positions in the grid, particularly the upper-left cube, as all participants performed the gestures with their right arm. This physical strain or awkward positioning could lead to inconsistent or incomplete gesture execution, which may negatively impact the model's prediction accuracy. Gestures performed under physical discomfort or at the edge of a participant's reach may result in noisier input data, reducing the model's ability to correctly classify the gesture.

The learning curve presented in figure 14 suggests that the model is overfitting on smaller training set sizes, which is indicated by the large gap between training and validation accuracy. The gap does narrow as the training set size increases, which indicates improved generalization and reduced overfitting. This is to be expected with the variability and noise present in EMG signals. The validation accuracy plateaus slightly beyond 2500 samples (windows in this case), which corresponds to 4 minutes and 10 seconds of active training time. Participants were actively training each gesture for 2 minutes and 15 seconds, suggesting the model might approach its highest potential under the current feature set.

9 Future Work

A key direction for future work is to improve the model's overall performance before proceeding, this can be investigated through several approaches. One could do more feature engineering or selection. Another approach could be to use the hierarchical classification method, where an initial classifier distinguishes between grasp and wrist movements, and subsequent classifiers identify the specific type within each category.

It might be valuable to implement an iterative testing approach for the model, where the participant would be back to test the model again multiple times to minimize participant fatigue and help participants adapt and improve their control of the myoelectric interface. This method would better simulate real-world usage scenarios we aim to emulate.

Future projects should also consider training and testing on amputees. There are a lot of considerations to take into account when dealing with amputees, compared to participants with intact limbs. Muscle anatomy, electrode placement, the size and shape of the residual limb, and gesture execution could significantly impact the EMG signals. Future studies should consider custom or adaptive sensor configurations to accommodate these differences.

10 Conclusion

This study explored the use of an EMG-based machine learning model to classify hand gestures in a spatial VR environment, to simulate real-world scenarios that include active arm movement and gestures in space. While the offline performance of the trained model was high, with macro F1 scores averaging 0.86, a notable performance drop was observed during the online tests, where the average macro F1 score fell to 0.52. This significant gap highlights the challenges of translating controlled offline model accuracy into real-time applications. However, individual results revealed variability, with some participants achieving relatively stable performance between online and offline testing. This indicates that factors such as gesture consistency, muscle signal quality, and physical comfort likely influence performance. Furthermore, discrepancies between the training conditions (static, controlled gestures) and the test environment (dynamic, moving gestures) may have contributed to reduced online accuracy. The embodiment questionnaire indicated that participants experienced a moderate sense of control and acceptance, suggesting that the VR environment and system design effectively created a sense of interaction, despite the technical and model-related limitations.

The results highlight a significant gap in offline and online results, while also revealing considerable variation across participants, with both low and high individual performances. Despite these variations, the findings provide a strong foundation for training and testing a machine learning algorithm to control a hand model with EMG signals in VR. With further development, this approach has the potential to support amputees in training the use of a prosthesis within a spatial VR environment prior to receiving a myoelectric prosthetic.

Acknowledgments

The authors would like to thank all participants in this study, as well as Bastian Ilsø Hougaard for help with the setup of Unity files, MYO data logging, and pictures and Eleonora Vendrame for further insights into the setup of EMG hardware and other questions related to the work with EMG.

A large language model has been used to reorder and structure certain sentences

References

[1] Sara Abbaspour, Autumn Naber, Max Ortiz-Catalan, Hamid GholamHosseini, and Maria Lindén. 2021. Real-Time and Offline Evaluation of Myoelectric Pattern Recognition for the Decoding of Hand Movements. *Sensors* 21, 16 (2021), 5677. https://www.proquest.com/scholarly-journals/real-time-offline-evaluation-myoelectric-pattern/docview/2565707256/se-2 Copyright - © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; Last updated - 2025-04-21.

[2] Adenike A. Adewuyi, Levi J. Hargrove, and Todd A. Kuiken. 2016. Evaluating eMg Feature and classifier selection for application to Partial-hand Prosthesis control. *Frontiers in neurorobotics* 10 (2016), 15–15.

[3] Amputee Coalition. 2024. *Limb Loss Statistics*. Retrieved february 11, 2025 from url={http://ccrma.https://amputeecoalition.org/resources/limb-loss-statistics/.edu/~jos/bayes/bayes.html},

[4] Elaine Biddiss and Tom Chau. 2007. Upper-limb prosthetics: Critical factors in device abandonment. *American journal of physical medicine rehabilitation* 86, 12 (2007), 977–987.

[5] Sebastian Bitzer and Patrick Van Der Smagt. 2006. Learning EMG control of a robotic hand: towards active prostheses. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2819–2823.

[6] Alexander Boschmann, Dorothee Neuhaus, Sarah Vogt, Christian Kaltschmidt, Marco Platzner, and Strahinja Dosen. 2021. Immersive augmented reality system for the training of pattern classification control with a myoelectric prosthesis. *Journal of neuroengineering and rehabilitation* 18 (2021), 1–15.

[7] Camila R. Carvalho, J. Marvin Fernández, Antonio J. del Ama, Filipe Oliveira Barroso, and Juan C. Moreno. 2023. Review of electromyography onset detection methods for real-time control of robotic exoskeletons. *Journal of neuroengineering and rehabilitation* 20, 1 (2023), 1–141.

[8] Claudio Castellini and Patrick van der Smagt. 2009. Surface EMG in advanced hand prosthetics. *Biological cybernetics* 100, 1 (2009), 35–47.

[9] Daniele D'Accolti, Katarina Dejanovic, Leonardo Cappello, Enzo Mastinu, Max Ortiz-Catalan, and Christian Cipriani. 2023. Decoding of multiple wrist and hand movements using a transient EMG classifier. *IEEE transactions on neural systems and rehabilitation engineering* 31 (2023), 1–1.

[10] Strahinja Dosen. 2025. Personal Communication.

[11] Mattias Erhardsson, Margit Alt Murphy, and Katharina S. Sunnerhagen. 2020. Commercial head-mounted display virtual reality for upper extremity rehabilitation in chronic stroke: a single-case design study. *Journal of neuroengineering and rehabilitation* 17, 1 (2020), 154–154.

[12] Anders Fougner, Erik Scheme, Adrian D. C. Chan, Kevin Englehart, and Øyvind Stavdahl. 2011. Resolving the Limb Position Effect in Myoelectric Pattern Recognition. *IEEE transactions on neural systems and rehabilitation engineering* 19, 6 (2011), 644–651.

[13] Marta Gandolla, Simona Ferrante, Giancarlo Ferrigno, Davide Baldassini, Franco Molteni, Eleonora Guanziroli, Michele Cotti Cottini, Carlo Seneci, and Alessandra Pedrocchi. 2017. Artificial neural network EMG classifier for functional hand grasp movements prediction. *Journal of international medical research* 45, 6 (2017), 1831–1847.

[14] Yanjuan Geng, Ping Zhou, and Guanglin Li. 2012. Toward attenuating the impact of arm positions on electromyography pattern-recognition based motion classification in transradial amputees. *Journal of neuroengineering and rehabilitation* 9, 1 (2012), 74–74.

[15] Janne M. Hahne, Meike A. Wilke, Mario Koppe, Dario Farina, and Arndt F. Schilling. 2020. Longitudinal Case Study of Regression-Based Hand Prosthesis Control in Daily Life. *Frontiers in Neuroscience* 14 (2020). https://doi.org/10.3389/fnins.2020.00600

[16] B. Hudgins, P. Parker, and R.N. Scott. 1993. A new strategy for multifunction myoelectric control. *IEEE transactions on biomedical engineering* 40, 1 (1993), 82–94.

[17] Gunter Kanitz, Christian Cipriani, and Benoni B. Edin. 2018. Classification of Transient Myoelectric Signals for the Control of Multi-Grasp Hand Prostheses. *IEEE transactions on neural systems and rehabilitation engineering* 26, 9 (2018), 1756–1764.

[18] Morten B. Kristoffersen, Andreas W. Franzke, Raoul M. Bongers, Michael Wand, Alessio Murgia, and Corry K. van der Sluis. 2021. User training for machine learning controlled upper limb prostheses: a serious game approach. *Journal of neuroengineering and rehabilitation* 18, 1 (2021), 32–32.

[19] Iris Kyranou, Katarzyna Szymaniak, and Kianoush Nazarpour. 2025. EMG dataset for gesture recognition with arm translation. *Scientific Data* 12, 1 (2025), 100.

[20] Iris Kyranou, Sethu Vijayakumar, and Mustafa Suphi Erden. 2018. Causes of performance degradation in non-invasive electromyographic pattern recognition in upper limb prostheses. *Frontiers in neurorobotics* 12 (2018), 58.

[21] Blair Lock, Kevin Englehart, and Bernard Hudgins. 2005. Real-time myoelectric control in a virtual environment to relate usability vs. accuracy. *MyoElectric Controls/Powered Prosthetics Symposium, Fredericton* (01 2005).

[22] Itzel Jared Rodriguez Martinez, Andrea Mannini, Francesco Clemente, Angelo Maria Sabatini, and Christian Cipriani. 2020. Grasp force estimation from the transient EMG using high-density surface recordings. *Journal of Neural Engineering* 17, 1 (2020), 016052–016052.

[23] Fernando E. R. Mattioli, Edgard A. Lamounier, Alexandre Cardoso, Alcimar B. Soares, and Adriano O. Andrade. 2011. Classification of EMG signals using artificial neural networks for virtual hand prosthesis control. (2011), 7254–7257. https://doi.org/10.1109/IEMBS.2011.6091833

[24] Nawadita Parajuli, Neethu Sreenivasan, Paolo Bifulco, Mario Cesarelli, Sergio Savino, Vincenzo Niola, Daniele Esposito, Tara J Hamilton, Ganesh R Naik, Upul Gunawardana, et al. 2019. Real-time EMG based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation. *Sensors* 19, 20 (2019), 4596.

[25] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. 2012. Feature reduction and selection for EMG signal classification. *Expert systems with applications* 39, 8 (2012), 7420–7431.

[26] Cosima Prahm, Alexander Schulz, Benjamin Paaßen, Johannes Schoisswohl, Eugenijus Kaniusas, Georg Dorffner, Barbara Hammer, and Oskar Aszmann. 2019. Counteracting electrode shifts in upper-limb prosthesis control via transfer learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27, 5 (2019), 956–962.

[27] Linda Resnik, Katherine Etter, Shana Lieberman Klinger, and Charles Kambe. 2011. Using virtual reality environment to facilitate training with advanced upper-limb prosthesis. *Journal of Rehabilitation Research & Development* 48, 6 (2011).

[28] Daniel Roth and Marc Erich Latoschik. 2020. Construction of the Virtual Embodiment Questionnaire (VEQ). *IEEE transactions on visualization and computer graphics* 26, 12 (2020), 3546–3556.

[29] Erik Scheme and Kevin Englehart. 2011. Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use. *Journal of rehabilitation research and development* 48, 6 (2011), 643–660.

[30] E Scheme, A Fougner, Ø Stavdahl, A D C Chan, and K Englehart. 2010. Examining the adverse effects of limb position on pattern recognition based myoelectric control. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE, 6337–6340.

[31] Stephen A Stearns, Allan A Weidman, Toni F Engmann, Lauren Valentine, Angelica Hernandez Alvarez, Jose Foppiani, Samuel J Lin, and Sammy Dowlatshahi. 2024. Hand and wrist amputation: a demographic analysis using the National Inpatient Sample. *HAND* 19, 7 (2024), 1166–1170.

[32] Michael Wand, Morten B. Kristoffersen, Andreas W. Franzke, and Jurgen Schmidhuber. 2022. Analysis of Neural Network Based Proportional Myoelectric Hand Prosthesis Control. *IEEE transactions on biomedical engineering* 69, 7 (2022), 2283–2293.

[33] Richard B. Woodward and Levi J. Hargrove. 2019. Adapting myoelectric control in real-time using a virtual environment. *Journal of neuroengineering and rehabilitation* 16, 1 (2019), 11–11.

[34] Yu Wu, Dai Jiang, Xiao Liu, Richard Bayford, and Andreas Demosthenous. 2018. A Human-Machine Interface Using Electrical Impedance Tomography for Hand Prosthesis Control. *IEEE transactions on biomedical circuits and systems* 12, 6 (2018), 1322–1333.

Received on 27 May 2025