

# A 3D-Printed, Adjustable Armband for Electromyography-Based Finger Movement Classification With Haptic Feedback

Michelle Wang<sup>\*,†</sup>, Miasya Bulger<sup>\*</sup>, Yue Dai<sup>\*</sup>, Kira Noël<sup>\*</sup>,

Christopher Axon<sup>\*</sup>, Anna Brandenberger<sup>\*</sup>, Stephen Fay<sup>\*</sup>, Zenghao Gao<sup>\*</sup>, Saskia Gilmer<sup>\*</sup>, Jad Hamdan<sup>\*</sup>, Prateek Humane<sup>\*</sup>, Jennifer Jiang<sup>\*</sup>, Cole Killian<sup>\*</sup>, Ian Langleben<sup>\*</sup>, Bonnie Li<sup>\*</sup>, Alejandra Martinez Zamora<sup>\*</sup>, Stylianos Mavromatis<sup>\*</sup>, Sasha Njini<sup>\*</sup>, Roland Riachi<sup>\*</sup>, Carrie Rong<sup>\*</sup>, Andy Zhen<sup>\*</sup>, Marley Xiong<sup>\*</sup>

<sup>\*</sup>*McGill NeuroTech, McGill University, Montreal, QC, Canada*

<sup>†</sup>Corresponding author. *michelle.wang6@mail.mcgill.com*

**Abstract**—Recent work in prosthetic devices suggests that forearm surface electromyography (sEMG) is a promising technology for human-computer interactions. Specifically, a system able to detect individual finger movement can have many clinical and non-clinical applications. Popular consumer-grade sEMG armbands are limited by their fixed electrode arrangement, which can negatively affect the classification of subtle finger gestures. We propose a low-cost, 3D-printed armband with fully adjustable electrode placement for the detection of single-finger tapping motions. We trained machine learning classifiers on features extracted from eight-channel sEMG signals to detect movement from nine fingers. We obtained a classification accuracy of 70.6% for a K-Nearest Neighbours classifier using features extracted from 1000 ms windows of data, with a peak accuracy of 89.7% when only trained on a subset of particularly clean data. We also introduce a novel haptic feedback mechanism to improve user experience when using the armband, and propose an augmented reality typing interface as a potential application of our armband.

**Index Terms**—augmented reality, electromyography, haptic interfaces, human computer interaction, machine learning

## I. INTRODUCTION

Surface electromyography (sEMG) is a technology that involves placing electrodes on a person’s skin to record electrical activity associated with the activation of certain muscles [1]. sEMG is non-invasive and relatively easy to use [1], and has been found to yield a signal quality comparable to that of signals obtained through invasive electromyography [2]. As a result, much research has been conducted in the field of sEMG-based control interfaces. In particular, forearm sEMG signal classification has many clinical applications such as upper limb prosthetic devices [3]–[6] and exoskeletons [7], [8], as well as potential commercial applications.

Historically, forearm sEMG has been studied as a means to detect gross hand and arm motions [9]–[11]. Recently, as technology became more developed and accessible, the emphasis has shifted to more subtle gestures such as single finger movements [12]–[14], offering the possibility of more refined

This work was supported in part by wrnch, the McGill Faculty of Engineering, and Microsoft. Thank you for your generosity.

control in human-computer interactions, both in clinical and non-clinical settings [15].

Recent work in individual finger movement classification on sEMG data has shown promising results. One group used a seven-electrode configuration to obtain an accuracy of 93% across twelve participants for a four-finger problem [16], and another one obtained 98% accuracy for a twelve-movement set (which included ten individual finger movements) for a single subject using 32 channels [17]. Both of these studies used professional-grade equipment for sEMG data acquisition [16], [17]. Another study used an eight-channel consumer-grade sEMG armband and obtained an accuracy of 72% for a five-finger classification problem [18]. To our knowledge, no group has attempted to classify individual finger movement across both hands using consumer-grade hardware.

Commercially available armbands allow users to quickly and conveniently establish reliable skin contact for multiple electrodes. However, one of the difficulties involved with using these armbands is the lack of adjustability in electrode positioning. For example, the (now discontinued) Myo armband by North (formerly Thalmic Labs) consists of eight sensors arranged in a ring. This design does not allow users to place the sensors on specific muscles [19], and thus is not well-suited for classification problems involving more dexterous movements. Given these known limitations, we propose an sEMG armband design that allows users to place electrodes anywhere within the area of the armband. This feature makes the armband better suited for individual finger movement classification problems. Additionally, it allows each user to have a unique electrode configuration that takes into account their physiological differences. This armband is 3D-printed and can be used with consumer-grade sEMG equipment.

We developed this low-cost sEMG armband specifically for a control interface based on finger movement classification. Hence, we also integrated a haptic feedback mechanism to improve general user experience: upon detection of finger movement, the armband produces vibrations on the user’s hand near the finger whose movement was detected.

Finally, a major potential application of low-cost sEMG

armbands is that they present a wearable gesture control mechanism to increase ease-of-use in augmented reality (AR) or virtual reality (VR) interactions. Traditional AR/VR devices require the user to be equipped with specialized hand remotes in order to interact with the interface. Computer vision solutions to this limitation present new challenges with a limited field of vision resulting in a limited area for motion. With sEMG armbands, the fingers and hands can move freely, and the user can easily switch between tasks in the real world that require physical motion and tasks in the AR/VR interface. Improving convenience and lowering cost may also help bring AR/VR technologies one step closer to mainstream adoption. As a proof-of-concept, we used our armbands to implement a prototype for an augmented reality typing interface, where finger movements can be used to navigate between different productivity applications and to type text.

## II. METHODS

### A. Armband Design

Most existing sEMG armbands are expensive, bulky rings of sensors placed on the arm. They are not 3D-printable and prioritize a smaller form factor over signal quality. In our design, we wanted to prioritize profile, signal quality and cost. Our design needed to be low-profile so as not to interfere with the user's daily activities. Having that in mind, we created very thin designs that could wrap around the user's arm and hide the electronics components. Through our signal testing, we discovered that the electrode placement with the best signal varied widely between people. As such, we needed the electrode placement to be fully adjustable. The surface of our design is a grid of holes that allows the sEMG electrodes to be attached to any part of the armband. That also meant that our design needed to be able to hold any part of the armband tight enough against the user's skin for a clear sEMG signal. To do so, the armband wraps around the user's arm and is tightened with adjustable belts, and we made the armband in three parts: hand, wrist and forearm pieces (see Fig. 1). Lastly, we wanted to make an inexpensive armband to make the device more accessible. Our design is fully 3D-printable and the cost of the filament for printing two full armbands is less than CAD\$10. We chose flexible TPU filament to allow our design to wrap flexibly around the user's arm and be tight enough for a clear sEMG signal. We used AutoDesk Inventor for the design process.

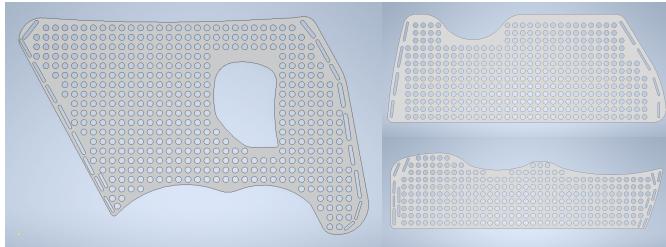


Fig. 1. Armband Computer-Aided Design (CAD). Hand (left), wrist (top right), and forearm (bottom right) pieces of the armband.

### B. Haptic Feedback Mechanism

We explored several methods for providing haptic feedback. The intention was to improve user experience by indicating to the user which finger our model predicted was moved. We compared the different feedback methods on cost, complexity and invasiveness; ideally, the method would not interfere with the user's activities, either by physically preventing motion or by distracting them. Ultimately, we chose to use short vibration pulses since this method was low-cost, simple and, from early testing with one finger at a time, we could place the mechanism low on the hand. It had the additional advantage that the feedback method would be familiar to users due to its similarity to mobile phone keypad haptics.

To implement the haptic feedback mechanism, we placed five cell phone vibration motors on the back of each of the user's hands, one per digit. When our model predicted a finger had typed, the vibration motor associated with that digit would pulse. The mechanism was controlled with an Arduino Trinket Pro 3V. Additional supporting electronics hardware included an FTDI-serial adaptor to connect the development board to our model, as well as various wires and header pins.

When we initially chose this mechanism, we had only tested with one motor and believed we could place the motors on the back of the hand near the wrist. However, when we tested with all five motors, we found it was very difficult to identify which motor had pulsed. We solved this problem with two different approaches: changing the placement of the motors and shortening the pulse length. First, we moved the motors further from each other. This placement (shown in Fig. 2A) was much more invasive than the original one we had planned, but it still only interfered with daily life in limited ways so we decided to proceed. Second, we shortened the length of the pulse to 80 ms. In general, users were better able to localize shorter pulses than long ones. We found 80 ms to be the optimal pulse length for a user to be able to identify while still receiving fairly strong feedback. We were initially concerned that the vibration motors could produce noise in the sEMG data, but only found significant noise when the vibration motor was directly on top of an electrode and running constantly.

The exact positioning of the motors varied slightly between individuals. To allow for this, we created a 3D-printed adaptor (see Fig. 2B) for the motors so that they could be connected to the same holes in the armband as the sEMG electrodes, meaning they could be placed anywhere on the back or side of the hand. This variable placement also meant that pulses from some of the motors were felt more strongly than others. For instance, pulses from motors on a bone were felt more strongly than those placed on areas with more fatty tissue like the side of the hand. Further, some of the vibration motors provided more force than others with the same pulse length and positioning on the hand. To account for these two factors, we built the controlling software to allow for custom multipliers for each motor. For example, if motor A is comparatively weak, we could set the multiplier to 1.2 and its pulse length would be 1.2 times 80 ms.

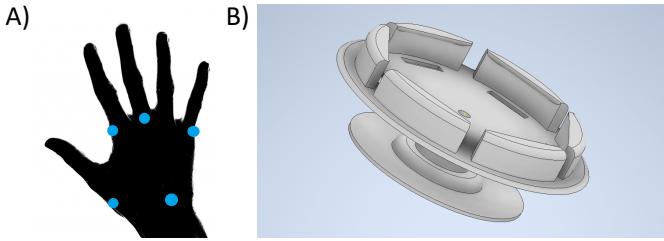


Fig. 2. **Haptic feedback system details.** A: Positioning of vibration motors on one hand. B: Motor-armband adaptor.

### C. Data Acquisition

We recorded sEMG data from thirteen healthy participants (mean age  $19.8 \pm 1.2$  years; five males, eight females; twelve right-handed, one left-handed) with normal upper limb function. Twelve of them were members of our development group. The participants received no monetary compensation.

All data collection was performed using an eight-channel OpenBCI Cyton biosensing board with a sampling rate of 250 Hz. All eight channels were used, with two electrodes per channel (placed in a bipolar configuration). We also used a ground electrode, for a total of 17 electrodes. The open-source OpenBCI Graphical User Interface (GUI) was used to visualize the sEMG timeseries in real time and to save the raw sEMG data.

We used dry flat Ag/AgCl (silver/silver chloride) snap electrodes to collect sEMG data from our participants. The electrodes could be easily attached to any hole on the 3D-printed armband by being snapped into electrode cables (which were connected to the biosensing board) through a hole in the armband (see Fig. 3A). This allowed us to reduce the set-up time and to achieve more consistent electrode placement across data collection sessions for the same participant.

We based our electrode configuration on the one used in [20], with electrodes spaced approximately 4 cm apart. Since our biosensing board had eight channels, we used four channels per hand. For the left hand, we were only interested in the index, middle, ring and pinkie fingers, and so we dedicated one channel for each finger. For the right hand, we were interested in classifying all five fingers, therefore we decided to forego having a dedicated channel for the right middle finger, with the hope that neighbouring electrodes might still capture middle finger activity. Putting the thumb electrodes in the locations suggested in [20] did not produce a good thumb signal for us. After some experimentation, we decided to put one thumb electrode on the wrist and the other one higher up on the hand itself. This produced a clear signal whenever the thumb was flexed and extended. The ground electrode was put on the extremity of the ulna bone, near the wrist. Fig. 3B shows our set-up using dry electrode and the wrist pieces of the armbands.

We collected a total of 46 trials, each one using one of three different data collection paradigms, or “modes.” We used these different modes because we wanted to collect the participants’ data when they were typing on a keyboard and when they

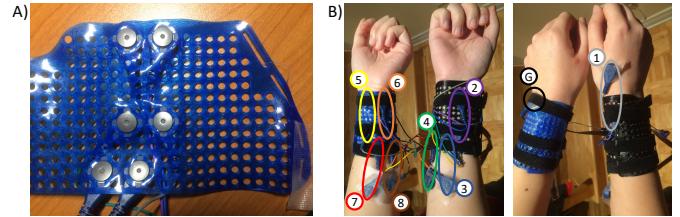


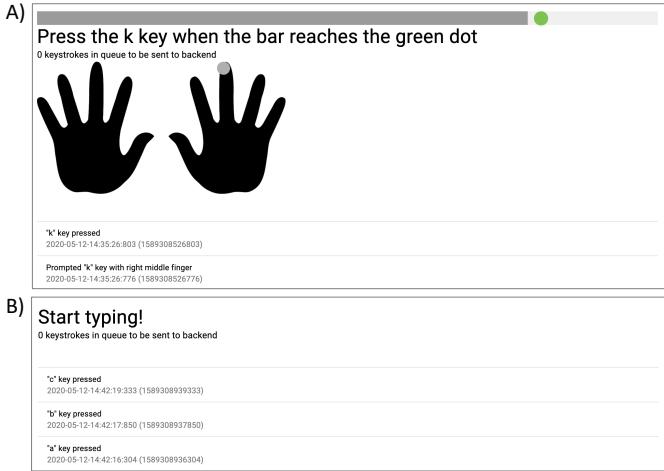
Fig. 3. **Electrode configuration on the armband.** A: Dry flat snap electrodes secured to an unrolled armband piece. B: Complete set-up with a ground electrode (G) and channels dedicated to the right thumb (1), right index finger (2), right ring finger (3), right pinkie finger (4), left index finger (6), left middle finger (5), left ring finger (7) and left pinkie finger (8). Only the wrist piece of the armband is used here; electrodes that are supposed to be on the other pieces of the armband are directly taped to the participant’s skin.

were executing typing motions in the air, without a keyboard. We were also interested in collecting trials where the time interval between consecutive finger movements was fixed, as well as trials where participants executed movements at their own pace. The three modes were the following:

- **Mode 1:** participants were prompted on-screen to type a letter using a keyboard when instructed. They received prompts in the form of a sentence and a visual cue (see Fig. 4A). The prompted keys corresponded to the left pinkie finger, left ring finger, left middle finger, left index finger, right thumb, right index finger, right middle finger, right ring finger and right pinkie finger. The prompts were randomized and spaced approximately 2 s apart. Each keystroke was logged with a timestamp; this indicated the approximate time the movement was executed. 10 trials were collected using this mode.
- **Mode 2:** similar to the previous mode, except participants did not type on a keyboard, instead performing a tapping movement in the air. Participants held their forearms either parallel to the ground or alongside their body. For each prompt, participants were presented with a progress bar and asked to execute the movement when the progress bar reached a certain point, at which point a timestamp was also logged by the recording dashboard (see Fig. 4A). We prioritized this mode because it was the one that is most similar to our end-use case, where no keyboard would be used. 22 trials were collected using this mode.
- **Mode 3:** participants were asked to type on the keyboard at their own pace. No prompts were given, and the timestamp associated with each keystroke was logged (see Fig. 4B). To ensure that all participants typed the same letters with the same fingers, we asked them to follow the standard touch-typing finger mapping. Similarly to the first mode, a timestamp was logged with each keystroke. 14 trials were collected using this mode.

### D. Signal Processing

The eight-channel sEMG trace was divided into sliding windows with a 100 ms overlap. We used the timestamps logged by the recording dashboard to label the windows. Previous literature recommends that an sEMG control system



**Fig. 4. Components of the data collection dashboard.** **A:** Progress bar, instruction and visual cue used in Mode 1 and Mode 2. In Mode 1, timestamps for both prompts and keystrokes are logged, while in Mode 2 only the prompts are logged. **B:** Timestamps of keystrokes being logged when recording in Mode 3.

have a maximum response time of 300 ms for a better user experience [21]. However, having too short a window size could potentially negatively affect the labelling of our data, especially for trials where participants performed finger movements in mid-air: here the accuracy of the labelling process depended on whether or not participants timed their movement well with respect to the given cues. Therefore, we tried window sizes of 1000 ms, 500 ms, and 200 ms, keeping in mind that larger window sizes would potentially result in better labelling, at the cost of the system response time and general user experience.

Signal processing was done in Python, using functions from the SciPy library [22]. We first applied a notch filter at 60 Hz to eliminate interference from the power lines, then used a fourth-order Butterworth filter to eliminate activity outside the 5-50 Hz frequency band, as recommended for sEMG analysis [23]. For each trial, we also labelled a subset of the previously unlabelled windows as “no movement.” The number of “no movement” trials was chosen so that it was comparable to the number of samples in the other finger classes.

We were conscious that some of our trials contained noisy data that could negatively affect our classification. We observed that noisy data could result for example from inadequate contact between the electrodes and a participant’s skin or interference between neighbouring electrodes. We visually inspected the filtered data for all trials by plotting a subset of windows for each finger. 18 trials were deemed unacceptably noisy, so they were labelled as “bad” and omitted from all subsequent analysis. 12 trials had signals that were relatively clean and seemed differentiable between most fingers; they were labelled as “good.” The 16 trials that were neither “bad” nor “good” were also kept in our analysis. The 28 trials we ended up using contained data from twelve of our thirteen participants, hence models trained on data from these trials

would have to generalize well across different participants. The “good” trials only contain data from six participants. Many of the “bad” trials were collected using Mode 3. This is possibly due to some interference between the keyboard and our sEMG recording system. Mode 1 also required participants to use a keyboard, but they typed slower and used a limited range of keys, so there was overall less hand movement, which could explain why the trials were generally less noisy.

### E. Finger Movement Classification

We developed a ten-class classification problem. The classes were the following: “no movement”, and each of the ten fingers except the left thumb, which was omitted because it was redundant in a typing interface (only one thumb was required for the spacebar).

**1) Feature Extraction:** We used a set of eleven time-domain features and four frequency-domain features (listed in Table I) that have been widely used in sEMG signal classification. These features were extracted separately for every window and for every channel in our dataset.

A detailed description of most of our time domain features can be found in [24]. Where appropriate, we used a threshold of 20 mV. MMAV is implemented in the same way as MAV2 in [24]. RMS 3 computed by first splitting a window into thirds (i.e. three non-overlapping subwindows of equal length) then calculating the RMS as usual.

For the frequency domain features, we first computed the power spectral density (PSD) using Welch’s method with a Hanning window. We then divided the PSD values into nine equally-spaced frequency bands from 5 Hz to 50 Hz, and computed the features for each frequency band.

**2) Machine Learning Models:** We trained and tested four different machine learning models using implementations provided by the Scikit-learn Python library [25], all with default hyperparameters. The models were the following: K-Nearest Neighbours (KNN), Decision Tree Classifier (CART) and Linear Discriminant Analysis (LDA). The features datasets were split into 80% training and 20% testing sets. Classification accuracies and F1 metrics were determined using ten-fold cross-validation.

TABLE I  
FEATURES USED IN CLASSIFICATION

Time domain	Frequency domain
• Integrated EMG (IEMG)	• Mean
• Mean absolute value (MAV)	• Minimum
• Modified mean absolute value (MMAV)	• Maximum
• Variance (VAR)	• Variance
• Variance of absolute signal amplitude (VAR ABS)	
• Root mean square (RMS)	
• Root mean square with three sub-windows (RMS 3)	
• Slope sign change (SSC)	
• Waveform length (WL)	
• Willison amplitude (WAMP)	
• Zero crossing (ZC)	

## F. Augmented Reality Interface

We implemented a real-time prediction system for finger movement classification. The OpenBCI GUI's networking feature was used to stream the sEMG data to a Python server, which performs real-time filtering and feature extraction. The server then emits a prediction using a previously trained model. This prediction is sent to the haptic feedback system and to a Unity application which serves as our AR typing interface. Fig. 5 shows a view of a “dock” environment where the user can open an application by moving a finger that is mapped to a specific key (“A,” “F,” “J” or “Enter”) or press the spacebar to search for an application by its name.

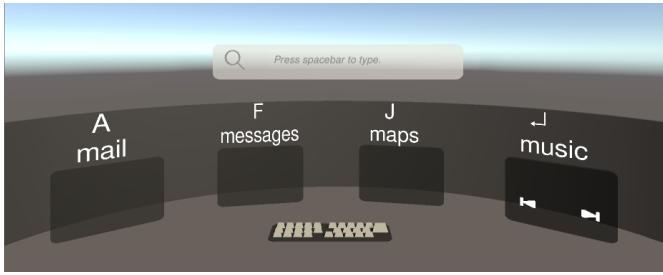


Fig. 5. Prototype of AR typing interface.

## III. RESULTS

Table II shows classification results for the three models we used. We first trained on all of our 28 trials, using window sizes of 200 ms, 500 ms and 1000 ms. These datasets had 16 801, 39 724 and 70 899 samples respectively. The 1000 ms windows dataset gave us a 70.6% accuracy using a KNN classifier, our best result at that stage. We then selected two subsets of the 1000 ms dataset: one containing only the “good” trials (39 030 samples), and one containing only the Mode 2 trials (47 200 samples). KNN models trained these two subsets obtained accuracies of 89.7% and 85.3% respectively.

TABLE II  
CLASSIFICATION RESULTS

Dataset	Model	Accuracy	F1 Micro	F1 Macro	F1 Weighed
200 ms All trials	KNN	60.6%	0.606	0.579	0.602
	CART	52.3%	0.521	0.488	0.520
	LDA	53.2%	0.532	0.503	0.536
500 ms All trials	KNN	66.7%	0.667	0.648	0.667
	CART	60.0%	0.600	0.575	0.600
	LDA	60.0%	0.591	0.554	0.591
1000 ms All trials	KNN	70.6%	0.706	0.687	0.704
	CART	65.6%	0.656	0.630	0.656
	LDA	61.2%	0.612	0.573	0.613
1000 ms “Good” trials	KNN	<b>89.7%</b>	<b>0.897</b>	<b>0.870</b>	<b>0.897</b>
	CART	84.9%	0.848	0.814	0.852
	LDA	83.7%	0.837	0.801	0.837
1000 ms Mode 2 trials	KNN	85.3%	0.853	0.829	0.851
	CART	80.5%	0.807	0.777	0.807
	LDA	76.5%	0.765	0.729	0.769

Fig. 6A shows normalized confusion matrices for the KNN models that were trained on all trials (1000 ms windows) and on “good” trials only (1000 ms windows). Our models also produced a vector of predicted probabilities for the ten classes for every sliding window, and we visualized these predictions on continuous sEMG timeseries data (see Fig. 6B) in order to have a better idea of how the model will behave when applied to real-time data. From both the confusion matrices and the heatmaps, we observed that many misclassification errors seemed to occur between the index and pinkie finger classes. During data collection, we noticed that, for some participants, the sEMG traces associated with index finger movement and pinkie finger movement were very similar: for some subjects, movements from either finger would be detectable through the same channels. Moreover, for some participants, it was difficult to find good electrode positions for the pinkie fingers, and sometimes movements from the pinkie fingers did not produce big changes in the sEMG signals. These insights reveal that electrode placements can greatly affect the outcome of the machine learning models.

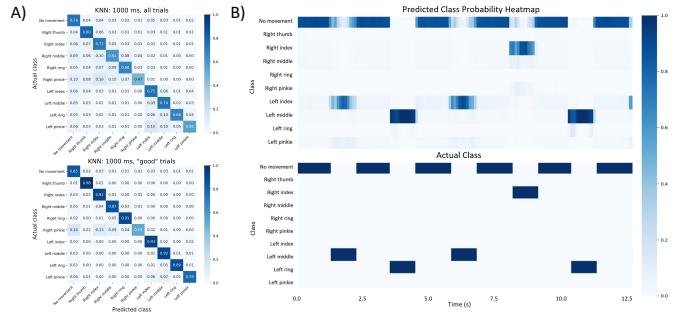


Fig. 6. Confusion matrices and model prediction heatmap. A: Confusion matrices of KNN models trained on 1000 ms windows of data from all the trials (top) and from only the “good” trials (bottom). B: Comparison between probability heatmap produced by the KNN model trained on the *1000 ms, all trials* dataset (top) and actual finger classes (bottom).

## IV. DISCUSSION

In this work, we address an important limitation found in commercial sEMG armbands such as the popular Myo armband, namely that they do not allow users to place electrodes at specific locations, and as such are not convenient to use in control systems involving dexterous and precise movements. Our design for a 3D-printed sEMG armband makes it easy for users to place electrodes in virtually any position on the surface of the armband, thus allowing for user-specific electrode configurations. Using our armband, we were able to record finger movement sEMG data from thirteen participants with different forearm anatomies. Moreover, we developed a simple and fully-adjustable haptic feedback system using small, inexpensive vibration motors and 3D-printed adaptors that fit into the holes of the armband. This haptic feedback feature not only offers users a richer, multimodal sensory experience, but also has potential in improving the system’s overall performance: we suggest that the feedback can be used

to train users to move their fingers in a way that can easily be classified by the model.

Using a set of fifteen time-domain and frequency-domain features, we achieved a 70.6% classification accuracy in a ten-class problem using a KNN model trained on all of our data. Furthermore, another model trained on a subset of particularly good trials obtained an accuracy of 89.7%. These results are promising and show that our armband seems to be well-suited for the detection of subtle movements such as single finger motions.

However, the large window size required to get these results (1000 ms) is a significant limiting factor in our control interface, as it increases the system's response time. Moreover, rapid finger movements that fall within a single window might not be classified accurately, and so users would have to wait an appropriate amount of time between each movement, which can be frustrating. Using smaller window sizes produced lower classification accuracies, but this may be due to our labelling process, therefore more work needs to be done with respect to achieving high classification performance on smaller windows. It is also worth noting that, when collecting data, we did not use a fixed number of prompts for each finger, and we did not record the same number of trials for all participants. This means that our datasets could have been biased towards certain finger classes or certain participants. Therefore, future experiments using a more rigorous data collection protocol are required to validate our results.

Finally, our AR typing interface prototype demonstrates how our armbands can be used in a real-time prediction system. We suggest that the use of an sEMG armband instead of the traditional hand-held remotes required for AR/VR interactions would allow users to seamlessly switch from tasks in AR/VR applications and tasks in their physical environment, thus increasing overall productivity. This would also be an improvement over the integration of computer vision technology to the AR/VR interface, which is limited by the camera's field of view. In the future, more refined control of AR interfaces can be achieved if high classification accuracy is obtained on even larger movement sets (for example, movements involving multiple fingers or other parts of the arm), and this can perhaps encourage more widespread integration of AR technology-based devices into daily life.

#### ACKNOWLEDGMENT

We would like to thank Building 21 for providing a workspace for our team.

#### REFERENCES

- [1] C. Castellini and P. van der Smagt, "Surface EMG in advanced hand prosthetics," *Biol. Cybern.*, vol. 100, no. 1, pp. 35–47, Jan. 2009.
- [2] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 5, pp. 847–853, 2007.
- [3] M. Tavakoli, C. Benussi, and J. L. Lourenco, "Single channel surface EMG control of advanced prosthetic hands: A simple, low cost and efficient approach," *Expert Syst. with Appl.*, vol. 79, pp. 322 – 332, 2017.
- [4] R. Boostani and M. H. Moradi, "Evaluation of the forearm EMG signal features for the control of a prosthetic hand," *Physiological Meas.*, vol. 24, no. 2, pp. 309–319, Mar. 2003.
- [5] R. N. Khushaba, S. Kodagoda, M. Takruri, and G. Dissanayake, "Toward improved control of prosthetic fingers using surface electromyogram (EMG) signals," *Expert Syst. with Appl.*, vol. 39, no. 12, pp. 10731 – 10738, 2012.
- [6] J. Rafiee, M. Rafiee, F. Yavari, and M. Schoen, "Feature extraction of forearm EMG signals for prosthetics," *Expert Syst. with Appl.*, vol. 38, no. 4, pp. 4058 – 4067, 2011.
- [7] Z. O. Khokhar, Z. G. Xiao, and C. Menon, "Surface EMG pattern recognition for real-time control of a wrist exoskeleton," *BioMed. Eng. OnLine*, vol. 9, no. 1, p. 41, Aug. 2010.
- [8] T. Lenzi, S. M. M. De Rossi, N. Vitiello, and M. C. Carrozza, "Proportional EMG control for upper-limb powered exoskeletons," in *2011 Annu. Int. Conf. IEEE Eng. in Medicine and Biol. Soc.*, 2011, pp. 628–631.
- [9] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 40, no. 1, pp. 82–94, 1993.
- [10] K. Nazarpour, A. R. Sharafat, and S. M. P. Firoozabadi, "Application of higher order statistics to surface electromyogram signal classification," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 10, pp. 1762–1769, 2007.
- [11] M. Zardoshti-Kermani, B. C. Wheeler, K. Badie, and R. M. Hashemi, "EMG feature evaluation for movement control of upper extremity prostheses," *IEEE Trans. Rehabil. Eng.*, vol. 3, no. 4, pp. 324–333, 1995.
- [12] M. V. Arteaga, J. C. Castiblanco, I. F. Mondragon, J. D. Colorado, and C. Alvarado-Rojas, "EMG-driven hand model based on the classification of individual finger movements," *Biomed. Signal Process. and Control*, vol. 58, p. 101834, Apr. 2020.
- [13] G. R. Naik, D. K. Kumar, and Jayadeva, "Twin SVM for gesture classification using the surface electromyogram," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 301–308, 2010.
- [14] N. M. Esa, A. M. Zain, and M. Bahari, "Electromyography (EMG) based classification of finger movements using SVM," *Int. J. of Innovative Comput.*, vol. 8, no. 3, pp. 9–16, 2018.
- [15] K.-J. You, K.-W. Rhee, and H.-C. Shin, "Finger motion decoding using EMG signals corresponding various arm postures," *Exp. Neurobiol.*, vol. 19, no. 1, pp. 54–61, Jun. 2010.
- [16] A. Andrews, E. Morin, and L. McLean, "Optimal electrode configurations for finger movement classification using EMG," in *2009 Ann. Int. Conf. IEEE Eng. Medicine and Biol. Soc.*, 2009, pp. 2987–2990.
- [17] F. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. V. Thakor, "Towards the control of individual fingers of a prosthetic hand using surface EMG signals," in *2007 Annu. Int. Conf. IEEE Eng. in Medicine and Biol. Soc.*, 2007, pp. 6145–6148.
- [18] W. Caesarendra, T. Tjahjowidodo, Y. Nico, S. Wahyudati, and L. Nurhasanah, "EMG finger movement classification based on ANFIS," *J. Phys.: Conf. Series*, vol. 1007, p. 012005, Apr. 2018.
- [19] Z. Arief, I. A. Sulistijono, and R. A. Ardiansyah, "Comparison of five time series EMG features extractions using Myo armband," in *2015 Int. Electronics Symp. (IES)*, 2015, pp. 11–14.
- [20] J. Ahmad, A. Butt, M. T. Riaz, M. Bhutta, M. Z. Khan, and I. Ul-Haq, "Multiclass myoelectric identification of five fingers motion using artificial neural network and support vector machine," *Advances in Sci., Technol. and Eng. Syst.*, vol. 2, pp. 1026–1033, 07 2017.
- [21] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, 2003.
- [22] P. Virtanen *et al.*, "Scipy 1.0: Fundamental algorithms for scientific computing in Python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, Mar. 2020.
- [23] C. J. De Luca, L. Donald Gilmore, M. Kuznetsov, and S. H. Roy, "Filtering the surface EMG signal: Movement artifact and baseline noise contamination," *J. Biomechanics*, vol. 43, no. 8, pp. 1573 – 1579, 2010.
- [24] A. Phinyomark, S. Hirunviriya, A. Nuidod, P. Phukpattaranont, and C. Limsakul, "Evaluation of EMG feature extraction for movement control of upper limb prostheses based on class separation index," *5th Kuala Lumpur Int. Conf. Biomed. Eng.*, vol. 35, pp. 750–754, 2011.
- [25] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.