Making Muscle-Computer Interfaces More Practical

T. Scott Saponas[1], Desney S. Tan[2], Dan Morris[2], Jim Turner[3], James A. Landay[1]

[1]Compuer Science and Engineering
DUB Group, University of Washington
{ssaponas, landay}@cs.washington.edu

[2]Microsoft Research
{desney, dan}@microsoft.com

[3]Microsoft Corporation
jturner@microsoft.com

ABSTRACT

Recent work in muscle sensing has demonstrated the potential of human-computer interfaces based on finger gestures sensed from electrodes on the upper forearm. While this approach holds much potential, previous work has given little attention to sensing finger gestures in the context of three important real-world requirements: sensing hardware suitable for mobile and off-desktop environments, electrodes that can be put on quickly without adhesives or gel, and gesture recognition techniques that require no new training or calibration after re-donning a muscle-sensing armband. In this note, we describe our approach to overcoming these challenges, and we demonstrate average classification accuracies as high as 86% for pinching with one of three fingers in a two-session, eight-person experiment.

Author Keywords
Electromyography (EMG), Muscle-Computer Interface.

ACM Classification Keywords
H.5.2 [User Interfaces]: Input devices and strategies

General Terms
Human Factors

INTRODUCTION

Traditional computer input devices leverage the dexterity of our fingers through physical transducers such as keys, buttons, and touch screens. While these controls make great use of our abilities in common scenarios, many everyday situations command the use of our hands for purposes other than manipulating an input device. For example, when wearing gloves in the cold, a mobile phone's interface can be difficult or impossible to use. Similarly, when someone is jogging and listening to music on their music player, their arms are typically swinging freely and their eyes are focused on what is in front of them, making it awkward to reach for the controls to skip songs or change the volume. In these situations, people need alternative input techniques for interacting with their computing devices.

One research area that offers possibilities for hands-free interfaces is computer vision. However, several issues such as occlusion and lighting limit the opportunities for this technology. Similarly, speech recognition enables hands-free control of some devices, but can be socially awkward in many settings. Another approach that has received much recent attention is hand and finger gesture recognition through forearm muscle sensing. This approach allows people to use their fingers for input without any sensing technology on or near their hands. Several efforts have demonstrated the feasibility of classifying finger gestures this way in real time [5,7]. While muscle-sensing techniques offer promise, previous work suffers from several key limitations. First, in many existing systems, users are tethered to high-end equipment employing gel-based sensors affixed to users' arms with adhesives [4,6,7]. Second, previous efforts attempting to classify an individual's finger gestures across sessions have yielded little success [3].

In this note, we extend previous muscle-computer interface research by presenting techniques for finger gesture classification using our wireless muscle-sensing armband. We also demonstrate the effectiveness of our approach through a two-session, eight-person experiment.

BACKGROUND AND RELATED WORK

When we decide to take a step, pick up a pen, or simply press our thumb to our forefinger, our brain sends messages that traverse our nervous system and eventually arrive at motor neurons. Motor neurons then stimulate muscle fibers in our skeletal muscles causing movement or force. This process generates electrical activity that can be measured as a voltage differential changing over time. While the most accurate method of measuring such electrical activity requires inserting fine needles into the muscle, a noisier signal can be obtained using electrodes on the surface of the skin. This general electrophysiological technique is known as electromyography (EMG). EMG is most commonly used in clinical applications such as assessing muscle function during rehabilitation. Many arm prosthetics also utilize EMG by mapping degrees of freedom to contractions of large remaining muscles such as those in the shoulder [2].

More recently, several efforts have explored a variety of approaches to employing muscle sensing for human-computer interaction. Costanza et al. created and evaluated a system where flexing one's bicep controlled advancing through their voicemail [1]. Bicep flexing can be performed quite subtly and detected robustly, but is limited in application because it only provides one bit of input per arm. Another scheme investigated in previous work is detecting

CHI 2010, April 10–15, 2010, Atlanta, Georgia, USA.

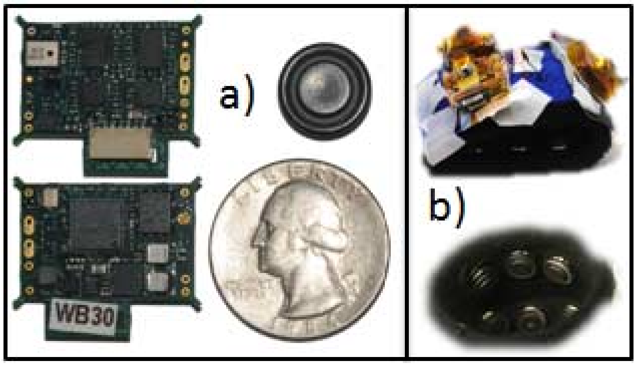

Figure 1. a) Our embedded wireless muscle-sensing device (left) and an electrode (right) pictured with a quarter for scale b) Our wireless-muscle sensing armband

wrist movements similar to those used to control a joystick for controlling games and RC cars [4,8,9]. Wrist gestures are advantageous for muscle-computer input because they offer many degrees of control and are detectable from muscles in the forearm. However, wrist-based gestures can be somewhat awkward to perform and lack subtlety in many circumstances. Conversely, many finger gestures can be performed easily and discretely even when our hands are moving, holding objects, or wearing gloves. Saponas et al. have demonstrated the feasibility of recognizing a wide variety of such finger gestures for muscle-computer input [6,7]. While their work provides classification techniques that are effective in many situations including pressing fingers on a surface and pinching fingers in air, they employ a high-end sensing device using wired sensors that require gel and adhesives. Furthermore, they require their system to be trained and calibrated each time sensors are applied to a person's arm. This is a common limitation of existing muscle-computer interfaces. In fact, in one of the few attempts to overcome this limitation, Ju et al. explored a variety of machine learning and adaptation approaches with little success at maintaining high classification accuracy without requiring new training data each time electrodes are reapplied to people's arms (their best method for classifying pinch gestures yielded a mean accuracy of 58%)[3].

We seek to extend previous work by enabling classification of finger gestures with no new training or calibration after sliding on a reusable wireless muscle-sensing armband.

OUR WIRELSS MUSCLE-SENSING SYSTEM

In this section, we describe our muscle-sensing hardware, gesture sets, and classification techniques.

Wireless Muscle-Sensing Armband

Our vision for muscle-computer interfaces is a thin armband worn on the upper forearm, capable of sensing a variety of finger gestures. To this end, we have created an embedded wireless muscle sensing device combined with electrodes and a sports sweatband (See Figure 1). Our embedded device has two analog muscle sensing channels and a Zigbee wireless radio. During the development process, we discovered that six channels were more appropriate for robust gesture recognition. As a result, our right-hand armband carries three of our two-channel devices wired together (See Figure 1b). Each sensing channel consists of one pair of silver-chloride-coated electrodes connected to a differential amplifier with very high input impedance (DC gain 20 dB). These are arranged into two sets of three channels; one set on top of the forearm the other set on the bottom. The armband streams raw data to a desktop computer where all processing takes place. In future iterations, some preprocessing might occur onboard and the rest of the processing could be carried out on a mobile device.

Gesture Sets

We explored two separate but related gesture sets for our muscle-computer interface. Our first gesture set consists of pinching one's index, middle, or ring finger together with their thumb. This can be performed in a wide variety of situations and can even be done when already holding or carrying objects. In our second gesture set, users rest their hand on a surface and press down with one of their index, middle, or ring fingers. Pressing on a surface is a gesture that can be performed when someone is resting their hand on a table, the arm of a chair, or even on one's own leg. Both of these gesture sets leverage isometric muscle contraction; that is, during the gesture, a user's muscles continue to fire without finger movement.

Similarly to Saponas et al., we combine these gestures on the right hand with a squeezing action in the left hand to create a bimanual select-plus-activate compound gesture [7]. This approach has three advantages. First, it gives the user an explicit method to indicate their intention to provide input. Second, a large squeezing action can be robustly detected with few false positives even during arm movement. Third, users can rapidly perform several of the same gesture in a row by pinching with their right hand while using their left hand to "pump" multiple successive squeezes.

Finger Gesture Training and Classification Technique

We follow a gesture classification scheme similar to that of Saponas et al [7]. They employ a machine learning approach using a support vector machine (SVM) that classifies gestures 32 times a second. Because of the limitations of sampling data with our microcontroller and transmitting them over Zigbee to a desktop computer, we only compute twelve classifications per second. Each classification result is obtained by first collecting 32 samples for each of our six sensing channels. We then extract the following types of features from the window of data: the amplitude of each channel as the root mean square (RMS) amplitude of the fully rectified signal, the RMS ratios among all six channels, spectral power in several frequency bands, the ratio of high-frequency energy to low-frequency energy within each channel, and lastly the phase coherence among each pair of channels. All of these features are combined to create a 54-element feature vector over a 30-millisecond window. The values of the feature vector are normalized based on a four-second calibration step where users sequentially pinch each of their fingers. We use these feature vectors both for training the SVM as well as for classifying gestures.

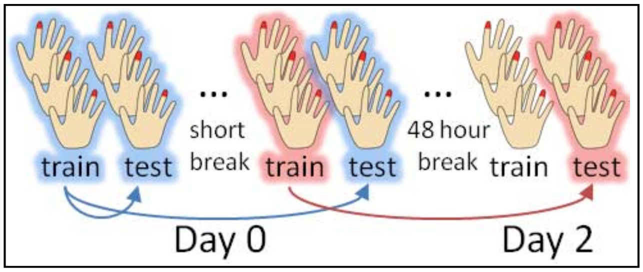

Figure 2. Experimental Setup: Eight participants engaged in three train/test sessions on two separate days.

To train our system, users perform a sequence of actions while the system computes and records feature vectors from their muscle data. To aid in this process, the computer guides the user through the training phase by presenting a finger gesture to perform for four seconds, giving the user a one-second break, then showing another gesture to perform. The system then keeps the feature vectors computed over the second half of each gesture as "good" data for training.

After training an SVM, it can be used to classify subsequent gestures. In our system, we attempt to classify every feature vector, yielding approximately twelve classifications per second. While a user is performing a gesture, the system might correctly identify their gesture most of the time, but occasionally misclassify some segments of data. We attempt to be robust to these circumstances by adding a level of smoothing where the current recognized gesture at any time is computed as the majority vote of the previous six classifications. If the system observes three consecutive identical classifications, that result will override the vote.

As mentioned in the Gesture Sets subsection, our complete finger gestures consist of a bimanual select-and-activate action where a user, for example, might pinch with their right index finger and then squeeze their left hand. The user's right-hand pinching gestures are continually classified as described above, but are only used as an input to the system at the moment they squeeze their left hand to *activate* the gesture. We infer left-hand activation using a gradient detector that watches for large changes in muscle signal amplitude. To reduce false positives due to motion artifacts caused by arm shifting and twisting, we further filter our gradient detector by requiring that activation be at least 35% of the maximum amplitude recorded during calibration as well as a ratio of at least two to one of low-frequency energy to high-frequency energy.

EXPERIMENT

We evaluated the effectiveness of our wireless armband and classification approach in a multi-session experiment. Eight participants (four female) from our research organization volunteered to participate in our experiment. They ranged in age from 23 to 31. Seven of our participants were right-handed and one participant was left-handed; however, each of our participants performed the pinching and pressing gestures with their right hand. None of our participants had any known neuromuscular diseases. We attempted to place the armbands in a similar position on every participant.

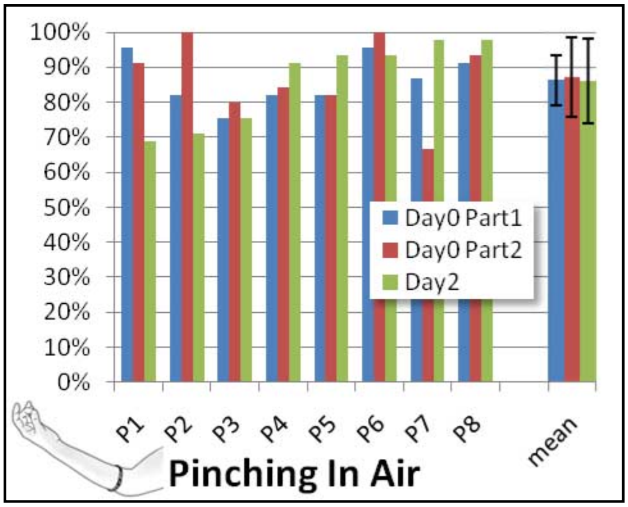

Figure 3. Classification results for pinching index, middle, and ring fingers in air from sessions on two separate days. Error bars for the mean represent standard deviation.

Each participant took part in two sessions occurring on two separate days. In the first session, participants first *trained* the system to recognize their finger gestures then *tested* the system's ability to recognize their gestures. Training entailed providing twelve examples for each finger gesture in both gesture sets. Testing included fifteen attempts of each gesture. Participants did this for both gesture sets, with the order counter-balanced across participants.

Following the first training and testing segment, we removed the armbands from participants' arms and gave them a short break to get up and walk around the room. After their break, we had them repeat the process of providing the system training data and then testing the recognition ability of the system. As illustrated in Figure 2, both testing phases during their first day used the training data they initially provided after putting on the muscle-sensing armbands. This tested the ability of the system to recognize gestures when the training and test data corresponded to identical electrode placement and when the armband was removed and re-donned between training and testing, leading to slightly different electrode placements. Approximately 48 hours later, participants came back for a second session and again engaged in a training and testing session. During testing on the second day, the system used the training data provided during the second part of the first day's session, challenging the system to classify their gestures with no new training or calibration from the placement of the armband on the second day.

Results

Our metric for performance is simply the accuracy with which our system can classify people's finger gestures. From our experiment, we can evaluate this ability based on three scenarios: when the system is trained immediately before use, when the system was trained earlier that day but the armband has been removed and reapplied, and lastly when no new training or calibration is provided that day.

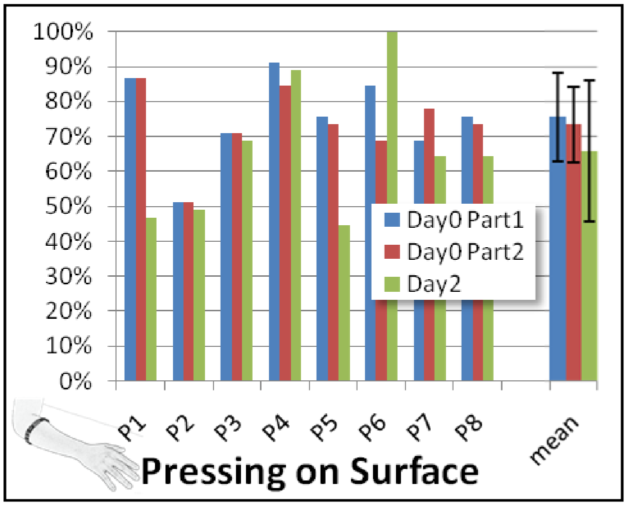

Figure 4. Classification results for pressing index, middle, and ring fingers on a surface from sessions on two separate days. Error bars for the mean represent standard deviation.

Our system performed best at recognizing the pinching gesture, with mean accuracies for same session, short break, and one-day break of 86%, 87%, and 86%, respectively; chance was 33% (See Figure 3). We found that our system classified pressing gestures slightly less well, with a mean accuracy of 76% when collecting training data immediately prior to testing, 73% after a short break, and 66% when using training from a previous day (See Figure 4).

DISCUSSION AND FUTURE WORK

Reflecting on the less robust classification of our participants' pressing-on-a-surface gestures, we think the primary source of misclassification is the variety of methods people can employ when pressing a single finger down on a surface. For example, they can relax their other fingers or they can pull them away from the surface. Similarly, they can use muscles in their hand, forearm, or shoulder to exert force. However, our system did perform well for two of our participants, suggesting that it may be possible with practice or feedback to classify these gestures.

The ability of our system to classify finger pinching gestures using only training data collected on a different day is an encouraging result toward the vision of a muscle-computer interface armband that can be quickly slipped on before starting a task in the lab or heading out the door for a run. In fact, for five of our eight participants, the system recognized their day-two pinching at an accuracy of 95%.

We believe there are two main reasons why our system performed much better for a little over half of our participants. First, as with any new input device or tool, people have different initial intuitions and abilities. In the case of our muscle-computer input technique, the more consistently a user gestures, the better the system will perform. We believe that over time, system performance would improve for any given user as they develop a more repeatable form of their gestures. Such changes over time might also warrant a method for periodically updating the system with more recent training. A second source of variance among users is the matching of their physiology to the layout of electrodes on the armband and our placement of the armband on their arm. While we did not observe an obvious systematic effect of body type in our experiment, it is a variable worthy of further investigation. We also think it may be possible to further improve on our results by reducing the impact of armband orientation by creating an armband with a dense array of electrodes where the system detects which subset of electrodes will be most effective.

CONCLUSION

In this note, we have presented a system for accurately classifying pinch gestures with no new training or calibration after re-donning a wireless armband. Our work is a step towards making muscle-computer interfaces a more practical possibility for controlling computing devices in mobile, off-the-desktop situations where traditional input devices are inconvenient or impossible to use.

ACKNOWLEDGEMENTS

We thank Jonathan Lester and Carl Ringler for assistance with the wireless device as well as John Platt and Jay Stokes for machine learning advice and tools.

REFERENCES

1. Costanza, E., Inverso, S.A., Allen, R., & Maes, P. Intimate Interfaces in Action: Assessing the Usability and Subtlety of EMG-based Motionless Gestures. *Proc ACM CHI* '07, 819-828.

2. Jacobsen, S.C. & Jerard, R.B. Computational Requirements for Control of the Utah arm. *ACM Annual Conference 1974*, 149-155.

3. Ju, P., Kaelbling, L.P., & Singer, Y. State-based Classification of Finger Gestures from Electromyographic Signals. *Proc ICML 2000,* 439-446.

4. Kim, J., Mastnik, S., and André, E. EMG-based Hand Gesture Recognition for Realtime Biosignal Interfacing. *Proc ACM IUI '08, 30-39*.

5. Peleg, D., Braiman, E., Yom-Tov, E., & Inbar G.F. Classification of Finger Activation for Use in a Robotic Prosthesis Arm. *Neural Syst Rehabil Eng*, 10(4), 2002.

6. Saponas, T. S., Tan, D. S., Morris, D. & Balakrishnan, R. Demonstrating the Feasibility of Using Forearm Electromyography for Muscle-Computer Interfaces. *Proc ACM CHI '08,* 515–524.

7. Saponas, T. S., Tan, D., Morris, D., Balakrishnan, R., Landay, J, & Turner, J. Enabling Always-available Input with Muscle-Computer Interfaces. *Proc ACM UIST* '09.

8. Wheeler, K.R, Chang M.H., & Knuth K.H. Gesture-Based Control and EMG Decomposition. *IEEE Trans on Systems, Man, and Cybernetics*, 36(4), 2006.

9. Zhang, X., Chen, X., Wang, W., Yang, J., Lantz, V., and Wang, K. Hand Gesture Recognition and Virtual Game Control Based on 3D Accelerometer and EMG Sensors. *Proc ACM IUI '09, 401-406.*