Analog Control with Surface Electromyography (EMG)

William Eisenhower, Scott McMichael
Research advisors: Dr. Pamela Abshire, Dr. Timothy K. Horiuchi

I. Abstract

The purpose of our project is to develop technology to translate electromyography (EMG) electrode measurements into reliable analog control signals. We designed a system that consists of a method to easily set up EMG electrodes, to perform analog filtering, to amplify EMG signals, and software algorithms to extract analog control signals from the filtered signals. Specific requirements for the system were the simultaneous use of multiple electrode channels at sufficient speed to eliminate any perceptible delay between EMG activity and the production of the analog control signal. The final form of the project consists of a demonstration using EMG readings to control a pointing device with two degrees of freedom.

II. Introduction

Electromyography (EMG), the electrical recording of muscle activity, is a relatively simple method to obtain a control signal directly from the body. While this has a number of potential applications ([1], [2]), the accuracy of the control that can be obtained is limited by the characteristics of EMG signals. Much research has gone into the best methods to accurately classify actions using an EMG signal, but these methods generally provide only ON/OFF digital control [1], [2], [3], [4]. Our project sought to explore different methods of processing EMG signals to provide full analog control. In addition to the investigation of signal processing methods, our project involved the design of software and hardware to demonstrate this analog control with EMG signals. This project was inspired by work performed at the 2004 Telluride Workshop on Neuromorphic Engineering [5].

III. Materials and Setup

In order to read in EMG signals, standard wet-cell adhesive electrodes were used in a differential pair configuration. One common ground electrode was shared by all of the differential pairs. Amplification of the EMG signals was performed by a band-pass amplifier circuit previously designed by a previous member of the laboratory [6]. The amplifier has a maximum gain of approximately 100 and cutoff frequencies of 10 Hz and 1 kHz respectively as shown in figure 2.

Figure 1: Amplifier Circuit board in Position

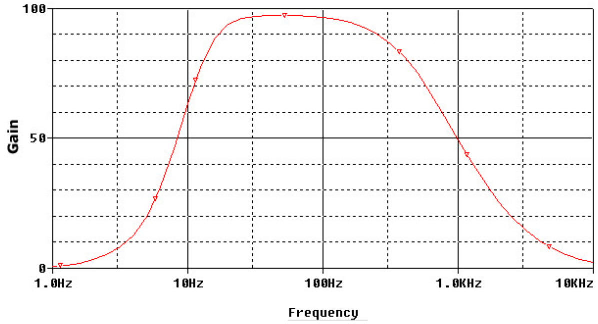


Figure 2: Amplifier Characteristics

To maximize the signal to noise ratio of EMG signals, amplification is performed as close to the electrodes as possible. Differential electrode pairs plug directly into the amplification circuit board, which is small enough to be easily supported by the electrodes' adhesive. A PMD-1208FS (Measurement Computing) USB-interfaced data acquisition tool was used to convert the amplified EMG into digital values. The analog to digital conversion has a range of ±1V and a resolution of approximately 0.5mV. Using the device in this configuration it is possible to sample from one to four separate channels at frequencies up to 12.5 kHz. For the purposes of this project each channel was sampled at a frequency of 1 kHz. This frequency was chosen as a compromise between signal information loss and computational slowdown due to a large number of samples.

Digital processing of the EMG signals was performed on a standard laptop by a Windows application written in C++. The application generates control signals that are converted into analog voltages by the PMD. This analog voltage signal was fed into our test device, which consisted of a laser pointer mounted on two servo motors. A PIC12CE674 (Microchip Inc.) microcontroller chip was also used to covert the analog control signals into absolute positions commands for the motors. The motorized test device is pictured below. The large motor in the bottom-right of the picture controls horizontal movement, while a smaller motor obscured behind the metal bracket provides up-down motion. The laser pointer is attached to the metal bracket with black electrical tape.

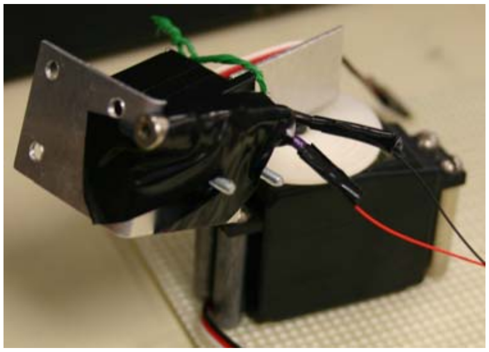


Figure 3: Motorized Test Device

Prior to beginning continuous operation of the analog control system, several sets of preliminary data must be recorded. The first set of data needed is a sample with no movement by the user; this data allows the system to determine the baseline noise present in each channel and to set noise thresholds below which no actions will be recognized. The noise threshold is based on a multiple of the standard deviation of this data after it has been rectified and filtered via a low pass filter. The multiple can be varied depending on the degree of input sensitivity desired by the user. After this, sets of training data must be recorded for each action that is to be recognized by the system. The appropriate training procedure must be run on this training data according to the action classification method that will be used (see section V). This training establishes a link between each action and the EMG signals that accompany them.

IV. Signal Processing

With the exception of the band-pass amplification circuit, all of the processing performed on the EMG signal was performed digitally using a laptop computer. Data was collected in windows of 31 samples; this number was chosen as it was the minimum window size for which the PMD could perform continuous sampling. This window size indicates that over 30 windows per second would be analyzed at a sampling rate of 1 kHz on each channel. On the 1.4 GHz Pentium M laptop used during testing, the computational work done imposed almost zero additional delay into the system for all of our processing done at a 1 kHz sampling rate with four channels. However the same configuration sampled at 2 kHz about 2.5% of the sampled data was lost as the processing could not keep up. Optimization of the processing program could likely increase the maximum lossless sampling frequency.

Processing of the EMG signal took place in six major steps. The first of these steps, (1) analog processing, consisted of the amplification and filtering of the raw EMG signal before it was sent to the PMD. The next five steps were (2) digital preprocessing, (3) low pass filtering, (4) thresholding, (5) action classification, and (6) analog calibration. The first four steps are described in detail below and the final two steps are described in sections V and VI of the paper. The first three steps improve the results of action classification [3].

Figure 4 is a representative diagram of the major changes a signal undergoes during processing. In the figure the light and dark lines represent two separate recording channels from two different electrode pairs. The thresholding step is not shown separately in this figure.

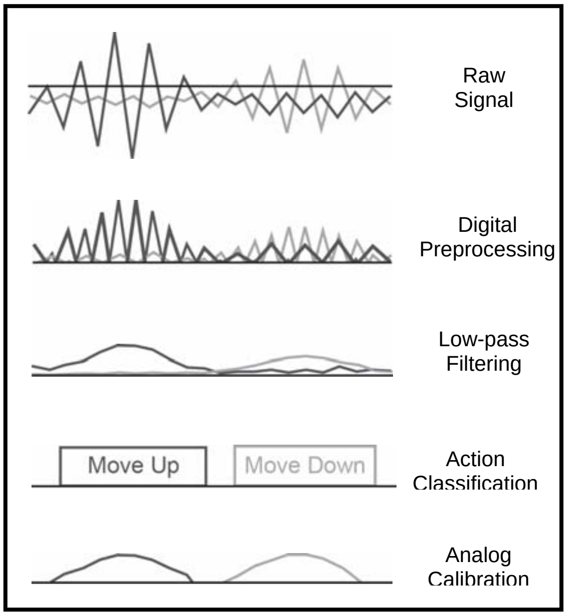


Figure 4: Simplified Signal Processing Steps

Digital preprocessing involved subtracting out the mean of the EMG signal and full-wave rectification to produce non-negative signals. Low-pass filtering was then performed using one or more passes of a moving average filter over the signal. In general, more smoothing produced a cleaner signal that better represented the physical movement that was recorded. On the other hand, more smoothing produced a time delay between physical movement and the processed signal, which made discrimination of shorter, faster movements more difficult. A default value of 30 samples was chosen in order to be compatible with the method in which the C++ application received incoming data; this value is quite low and imposes a minimum time delay or distortion of fast signals.

Thresholding is performed after smoothing of the signal. The thresholding process moves through each time instance of the signal and looks at all of the channels. If none of the channels are above their previously determined noise levels, all channel values are set to zero. If any of the channels are above their noise level, all channel values are retained. In this way the control system insures that noise and very minor muscle activity produces zero analog output. At this point, downsampling is an optional step that can be performed if computational speed and/or memory are at a premium, such as in an embedded system. Since the signal has been put through a low-pass filter, it is possible to downsample to much lower frequencies without losing significant information. Downsampling will speed up the action classification phase of the signal processing significantly. Since the laptop used for our testing was not slowed down by the additional computation, no downsampling was performed during our testing.

V. Action Classification

There are a number of different algorithms available for action classification. While the choice of algorithm will determine the accuracy of the system's classifications, the final analog calibration step is independent of the classification method used. For our work we tested a support vector machine (SVM) classification method (inspired by its use in [4]), a Gaussian distance function, and a simple Euclidean distance function.

For each of the classification methods, some form of training must be performed for each of the actions to be recognized. Training data for each action is recorded after noise data has been collected and before placing the system in continuous classification mode.

The SVM method used was the LIBSVM package [7]. Though experiments using it in MATLAB with recorded data were successful, we were not able to successfully implement the package in our C++ application. The Gaussian distance was tried as it was believed to be a compromise between complexity and accuracy. This method also worked well in MATLAB with recorded data; it was implemented in the C++ application but did not perform as well as expected. As of this time the reason for this poor performance has not been determined. The Euclidean distance function, chosen for its lack of complexity, was successfully implemented in the C++ application. Regardless of the method of classification used, only one action can be classified at each point in time.

For any classification algorithm used, each sample of data is sent to the algorithm to determine the action which best matches the data. Action classification is performed over windows of data: the action with the greatest number of classification matches in each window determines the classification of that window. The system also determines the action with the second greatest number of classification matches in the window. At this point, the system decides whether to use the classification assigned to the new window of data or to retain the classification assigned to the previous window. In order to make this decision, the system compares the number of classification matches for the first and second-most likely actions that it has recorded for that window. If the difference in their respective number of classifications does not exceed a specified separation, the classification of the previous window is retained. If the previous window was classified as an action (i.e. not "no action") then the separation needed to adopt a new classification is increased. This results in system behavior in which an action classification has inertia proportional to the separation value. A higher separation value results in more stable action classifications that are easier for the user of the system to control; a lower separation value increases the system's sensitivity to changes in actions.

VI. Analog Output

Forming a stable analog output from the EMG signal consists of using the action classification results to intelligently shape the smoothed signal produced by low-pass filtering. When training data is recorded for each of the classifiable actions prior to continuous operation of the EMG system, maximum EMG signal intensities are also recorded for each of the actions. During continuous operation, mean signal intensities are determined for each window of data. These averaged intensities are then normalized to a 0.0-1.0 scale relative to the maximum signal intensity previously recorded for that action during training. For the purposes of calculating the signal intensities, the intensity at a given point in time is the average intensity of all of the individual channel intensities at that time.

After the signal intensity has been normalized for a window of data, this intensity is then averaged with the last n window intensities. The value of n depends on the application of the EMG control system: higher values of n make the output more stable but also decrease the speed with which it responds to changes in the system user's muscle activity.

VII. Results

The EMG control system described above has been able to successfully demonstrate analog control with two degrees of freedom using three differential electrode pairs placed on the forearm. The weakest aspect of the system is its action-classification component: the highest accuracy that can be achieved is roughly two-thirds correct classifications. Furthermore, the accuracy is highly variable depending on the position of the arm and tends to degrade over time since the training data was recorded. The most obvious improvement to the system's classification ability would be to replace the system's simple Euclidean algorithm with one of the more sophisticated methods available.

Data recordings for two different user tasks are located in the Appendix. Several observations can be made from these recordings. First, it can be seen that having a shorter sampling window can lead to oscillations between action classifications in borderline situations. Increasing the size of the classification window ameliorates this effect and also slightly increases the relative smoothness of the output signal. However, increasing the window size also results in a loss of time-based precision, though the larger value used in the recordings is still small enough to successfully capture almost all of the high-speed motions. A larger output averaging length (the value *n* in section IV) increases the accuracy to which the user can maintain a fixed value from approximately $\pm$10% to $\pm$5% though it also introduces a tendency to overshoot a fixed value that the user must be conscious of. The output strength is also much slower to adapt to user changes. On the other hand, the lower adaptation speed results in less fluctuation of the output strength.

In its primary research goal, the demonstration of analog control, the system has achieved moderate success. The system is clearly capable of producing different levels of output strength that can be consciously controlled by the user of the system. The relative stability and precision of this control can be varied by adjusting system parameters as is described above.

Two secondary goals for the analog control system were to successfully operate with multiple electrode channels and to exhibit very rapid response to the user's muscle activity. Regarding the first of these goals, the control system is capable of working off of four electrode channels simultaneously. Unfortunately, the effectiveness of the system in terms of classification accuracy degrades as more electrode channels are added, making the system less reliable. Conversely, with only two electrode channels the system can be very accurate in making classifications. The four channel maximum is a limitation of the PMD used for data acquisition. Eight channels, the absolute maximum the PMD can support, could likely be used with a signal amplifier with a gain of at least 1000x (an order of magnitude larger than the current amplifier) or with a more sensitive data acquisition tool.

The speed of response of the system is generally quite good; sudden muscle actions typically produce output with no noticeable delay beyond that of the mechanical system being controlled. In the recordings of fast user motion located in the appendix, only one parameter setting resulted in any missed user movements. This case represents the most rapid user movement possible and it is unlikely that such reaction speeds would be required in any real application. The speed of transitions between actions is influenced by the degree to which the separation required to adopt a new action classification has been set to. While a higher value tends to make user control more stable, it also slightly decreases the speed at which the system can respond to user input.

One important aspect of the system that may need a revision is the circuit boards used to amplify the EMG signals. While their operation parameters are well configured to the task, there have been significant problems with their operating life. Efforts to quantify the system's performance have been made difficult by the fact that almost all of the existing boards have stopped functioning during the course of testing. The exact cause of their failures has not been determined, but it must be corrected in order for the design to be viable.

VIII. Conclusion

Though the final control system produced by this project is not as accurate or reliable as would be desired, all of the project goals have been satisfied to some extent. The project has resulted in the development of several useful tools to further future work, including methods of attaching and powering the electrodes as well as a built-from-scratch C++ application for recording and processing EMG signals. The project has also explored the effectiveness of analog EMG control relying on the amplitude intensity of EMG signals. In particular, we have arrived at the conclusion that using this method will invariably involve a tradeoff between output signal stability and the speed of response of the system. Improved accuracy of the action classification system will allow improvements in both of these areas.

IX. References

[1] Kim, Jong-Sung, Hyuk Jeong, and Wookho Son. "A New Means of HCI: EMG-MOUSE." 2004 IEEE International Conference on Systems, Man and Cybernetics. pg 100-104.

[2] Motion Control Utah Arm. 18 Feb. 2005. Motion Control, Inc. 5 Aug. 2005 <http://www.utaharm.com/index.html>.

[3] Ma, Ning, D K Kumar, and Nemuel Pah. "Classification of Hand Direction using Multi-Channel Electromyography by Neural Network." Seventh Austrailian and New Zealand Intelligent Information Systems Conference. 18-21 November 2001, Perth, Western Australia. pg 405-410.

[4] Crawford, Beau, et al. "Real-Time Classification of Electromyographic Signals for Robotic Control." Working paper, Mar. 2005.

[5] Assad, Chris, et al. "Human-machine interfacing by EMG pattern recognition: Timmer's Tail." Working paper, 2004.

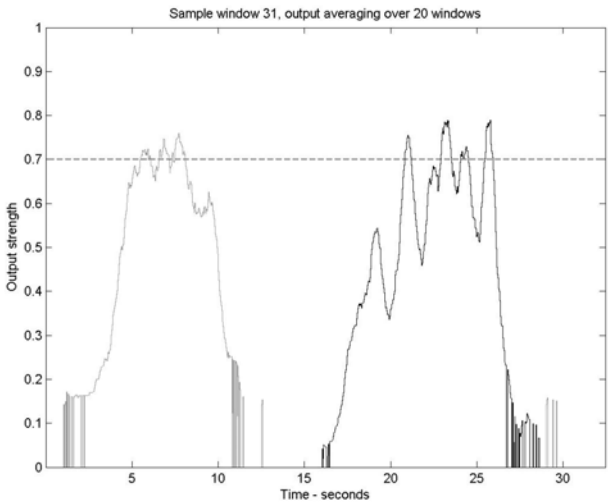
[6] Pierce, David K. "Independent Study Project Report ENEE499." University of Maryland, College Park, Working paper, 2004.

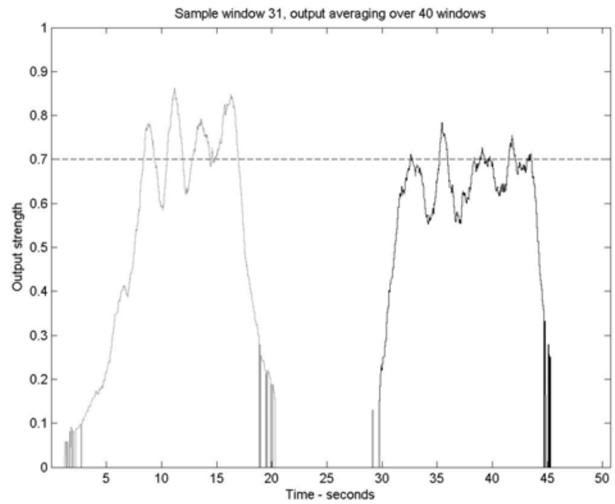
[7] Chang, C C, and C J Lin. LIBSVM. Apr. 2005. National Taiwan University. 5 Aug. 2005 <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>.

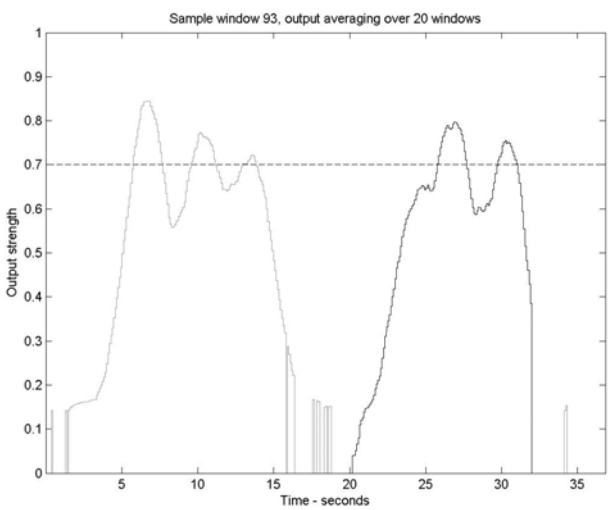
X. Appendix

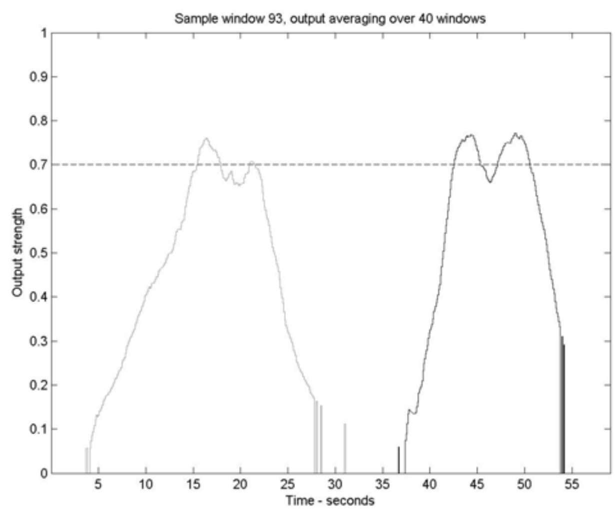
Data recordings for two different user tasks are located below. Two different system parameters were tested with two values each in order to produce four sets of recordings for each task. The first parameter varied was the sampling window size used to collect data: both the default size of 31 and a larger size or 93 were used with low pass filter sizes of 30 and 90 respectively. The second parameter was the number of sampling windows over which the output strength was averaged (*n*); this was varied between 20 and 40 windows.

Recording set 1: Slow movement control with two actions (light and dark lines)



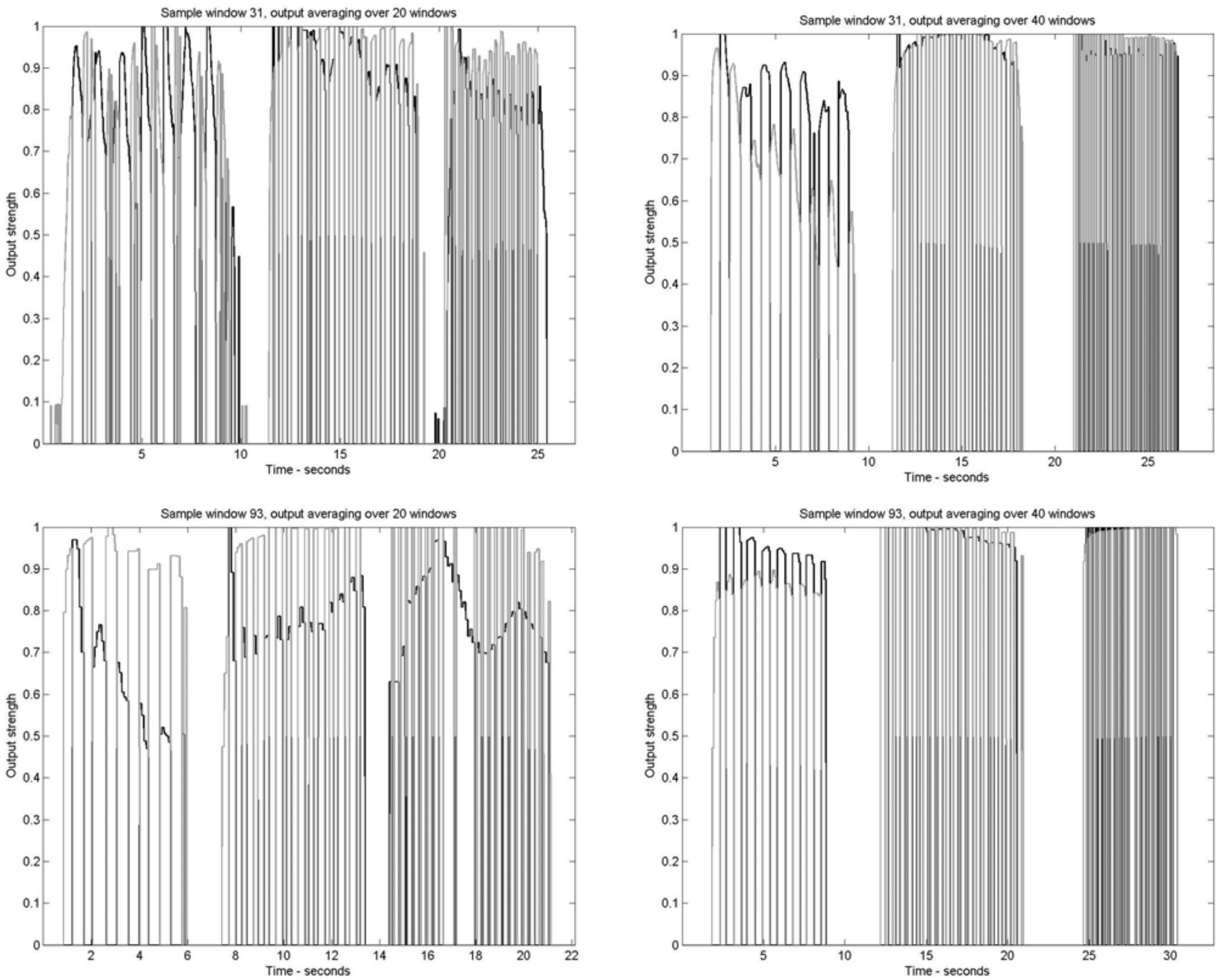






For this set of recordings, the wearer of the electrodes attempted to slowly adjust the system's output strength to 0.7 and hold it at that level, using a continuously updated display of the output as feedback. This test was undertaken using two electrodes and two directions, each recording shows this task first right and then left. Some intermittent activity due to shifting by the user between the task movements has been classified as actions.

Recording set 2: Rapid movement control with two actions (light and dark lines)



For this set of recordings, the wearer of the electrodes waved their hand back and forth for a period of several seconds. Each recording contains waving at three speeds: the first is waving at approximately 1-2 direction changes per second, the next is at approximately 4 direction changes per second, and the third is at the maximum rate the user could wave their hand (approximately 8 direction changes per second).

It should be noted that when smoothing over multiple output windows only windows with the same action classification are considered. This results in each separate action following a different smoothed curve in the figures above.