Motor Imagery based Control of a Humanoid Robotic Arm and Hand

Chawin Ophaswongse *co2393*, Cameron Holman *ch2766*, Alex Angus *ala2197*

***Abstract*—The control of robotic prostheses through a brain computer interface (BCI) remains a challenging and relevant goal in the neuroengineering community. Here we explore the control of a virtual robotic arm and hand through decoding of electroencephalographic (EEG) data as users focus on particular mental imageries to drive prosthetic action. Utilizing one of the largest EEG datasets recorded to date, we employ the use of two mental imagery paradigms, HaLT (Hand, Leg, and Tongue) and 5F (5 finger), for the control of macro and micro properties of a virtual robotic prosthesis. The system requires a three part classification: one, binary classification between HaLT and 5F motor imageries, two, intra-macro classification, and three, intra-micro classification. Through the implementation of these three classification algorithms, we demonstrate a system through which persons with or without disabilities may control general positioning and fine finger movements in a robotic arm and hand.**

I. Introduction

BRAIN computer interface (BCI) provides numerous potential avenues of support for humans suffering from disabilities ranging from aphasia to limb amputation. To provide effective aid, direct neural recordings can be decoded in real time and translated into control signals dictating computer or machine action. Among many other possibilities, these neurally controlled actions can include word spelling, cursor movement, and manipulation of robotic prostheses [1].

While invasive methods for obtaining neural recordings can provide superior resolution when compared to non-invasive methods, non-invasive methods remain at the forefront of BCI research due to the lack of risk for the patent, increased accessibility, and lower costs. In this paper, we explore the use of neural signals recorded through electroencephalography (EEG), a non-invasive means of obtaining neural recordings first developed by Hans Berger in 1929 [2], in the control of a virtual robotic arm and hand. We explore one of the largest recorded EEG datasets to date, containing over 60 hours of EEG recordings across 13 participants as they focus on particular mental imageries [3].

Specifically, we aim to implement multiple simultaneous control paradigms at differing scales. Using this system, the user may control macro scale movements of the arm and micro scale movements of the fingers on the hand by focusing on particular mental imageries [3, 4].

Using the HaLT (Hand, Leg, and Tongue) paradigm, macro scale movements of the arm including three dimensional positioning and wrist rotation are controlled. This allows for accurate positioning of the hand in virtual space. Concurrently, the user may employ the 5F (5 Finger) paradigm for micro scale control of individual fingers [3, 4]. These finger movements can support grasping of object, button presses, and the performance of gestures that allow for more inclusive social interaction (thumbs up, pointing towards a goal, finger-counting).

II. Data Description

Our dataset is a subset of the data released by [3] that was produced in an effort to create a large, uniform EEG dataset for designing and evaluating different BCI data processing methods. It is comprised of over 60 hours of EEG recordings from 13 subjects, with a total of 75 recording sessions and 201 individual EEG BCI session segments. In this dataset there are over 60,000 motor imagery (MI) examples enacted within 4 different MI paradigms recorded during the development of a slow cortical potentials BCI system [4]. EEG signals were recorded using an EEG-1200 JE-921A system across 19 channels using the standard 10/20 international system. For each session, subjects were presented with a GUI specific to the paradigm they were recording for. Each example began with the random selection of a MI task indicated on the GUI. The patients then had a period of 1 second to perform the specified MI task which was followed by a pause of 1.5 to 2 seconds. Recording session segments lasted for approximately 15 minutes with an average of about 300 MI examples recorded for each session segment. For our application we use only two of the available MI paradigms: HaLT and 5F.

A. HaLT Paradigm

The HaLT (hands, legs, tongue) paradigm is a 6-state extension of the classical right hand, left hand, and passive motor imagery paradigm. In addition to left and right hands, it includes the motor imagery of left leg, right leg, and the tongue. During the recording of the HaLT sessions, a GUI indicated a single specific MI to perform by highlighting a cartoon picture of the body part to be imagined as moving. For left and right hands and legs, the subject was instructed to imagine moving said limb, and for the tongue the subject was instructed to imagine pronouncing a specific phoneme such as ‘en’ or ‘la’. For passive MI the subjects were instructed to think of nothing.

B. 5F Paradigm

For the 5F (five finger) paradigm, subjects were presented with a GUI similar to that used in HaLT recording sessions, except the specific finger to be moved was indicated by a number presented on the screen rather than by a highlighted

cartoon image. The numbers ranged from 1 through 5, each indicating a different finger to be imagined as moving. When presented with the action signal, subjects were instructed to imagine flexing the finger either up or down. There was no explicit passive state indicator presented by the GUI, but subjects were instructed to remain passive after executing finger MI until the next signal. The discrimination of fine motor imagery, such as individual finger movement, with coarse spatial resolution techniques such as EEG is challenging; thus, to provide as much detail as possible, the 5F paradigm sessions were recorded at 1000 Hz in addition to the standard 200 Hz sampling rate.

III. Methods

A. Data Preprocessing

Our data processing methods are largely adapted from [4]. For each session and each trial, EEG data was separated into examples by the column 'marker' which indicates the MI state to which the data belongs. The data is then represented by the three dimensional array X_{nit} where n is the example, i is the EEG channel, and t is the time since the action signal. The signals closest in time to the onset of the action signal are the most indicative of MI state. Therefore, as is done in [4], we crop each example between $t = 0$ and $t = 0.85$, where $t = 0$ is the onset of the action signal. Each example n is then composed of 19 channels each with 170 time points (0.85s $\times$ 200 Hz), yielding a total of 19 $\times$ 170 = 3230 features per example for the raw EEG signal. We found that normalizing X, such that the mean was zero and the standard deviation was 1, improved the accuracy of our classifiers. Additionally, X was digitally referenced using the common average reference, which is defined as the average of all electrode voltages.

B. Feature Selection

During the development of our classification algorithms we experimented with three types of feature vectors in addition to raw EEG signal: smoothed EEG, power spectral density, and Fourier transform amplitudes.

1) Smoothed EEG: Since we were working with prerecorded data, our smoothing of the raw EEG data can be interpreted as the digital version of a low pass analog filter. To do this we apply a Savitzky-Golay filter, across each channel, defined by:

$$\bar{X}_{ni} = \sum_{k=\frac{1-m}{2}}^{\frac{m-1}{2}} C_i X_{ni,j+k}$$

where X_{ni} a channel of the example to be smoothed, $\bar{X}_{ni}$ is the smoothed channel of the example, m is the degree of smoothing polynomial, C is a set of m convolution coefficients calculated in [5], and

$$\frac{m-1}{2} \leq j \leq n - \frac{m-1}{2}.$$

To compute the smoothed examples we used the **savgol()** function of the scipy.signal Python library. For all smoothing calculations we used polynomial fitting of order $m = 2$. This smoothing process improves the stability of our classification algorithms in terms of convergence and reduces the impact of EEG artifacts.

2) Power Spectral Density: To convey information from the data in terms of spectral power, we calculate the power spectral density (PSD) using Welch's method. Welch's method involves splitting the signal into n equal length, non-overlapping segments, computing the periodogram with a discrete Fourier transform (DFT) for each segment, squaring the results of each DFT, and taking the average across all segments. This process was computed for each EEG channel using the **welch()** function of the scipy.signal Python library. In the computation of the periodogram we sample from 1 Hz to 100 Hz for a total of 19 $\times$ 100 = 1900 features per example for the PSD feature vector.

3) Fourier Transform Amplitudes: As is done in [4], we also consider Fourier transform amplitudes (FTA) for classification of MI states. The FTA feature vector is defined as:

$$\tilde{X}_{nik} = \sum_{t=1}^{\Delta T} \bar{X}_{nit} e^{\frac{-2\pi(t-1)(k-1)}{\Delta T}}$$

where $\tilde{X}_{nik}$ is the i^{th} channel of the FTA feature vector, ΔT is the time between samples (in our case $\Delta T = 0.005$s), and $\bar{X}_{nit}$ is the i^{th} channel of the smoothed EEG vector. We perform this calculation with the Numpy **fft()** function, again with a frequency resolution of 1 Hz from 1 Hz to 100 Hz. The features of the FTA vector are complex, so in order to use it in our classification algorithms we represent the real and imaginary components as real tuples and flatten them into a 1 dimensional array such that our FTA feature vector has 19 $\times$ 200 = 3800 features per example.

C. Classification

Our classification system is comprised of three classifiers: Binary Macro Classification between HaLT and 5F MI, Intra-Micro Classification within the HaLT paradigm, and Intra-Micro Classification within the 5F paradigm. For the two Intra-Micro classifiers we experimented with the three different types of feature vectors detailed above. For the Macro Classifier we did not feel the need to use alternative features because the classification accuracies of raw EEG signals were upwards of 98% for some subjects (see Table III), which is adequate for use in a control system.

For each of our three classifiers we implemented linear SVM discriminators with a random split of 70% of the data used for training and 30% used for validation. Preliminary classification testing was performed in a Python Jupyter Notebook with the scikit-learn **linearSVC** package. With the Macro Classifier we amalgamated all HaLT and 5F examples for a given subject into a single dataset, relabeled each example as 0 (HaLT) or 1 (5F), and trained a binary classification SVM. We note that to train the Macro Classifier it was necessary for a subject to have data for both the HaLT and 5F paradigms. This was the case for only four subjects (A, B, C, and F) and was limited by the number of subjects with 5F MI sessions.

It is evident that HaLT and 5F features are easily linearly separable, as consistently high classification accuracies were achieved with raw EEG features and a low number of iterations before convergence ($<$ 1000 iterations).

With the Intra-Micro HaLT Classifier, the data was split by example, shuffled and allocated to either train or validation in the same manner as above, and labeled 1 through 6, where each label represented one of the HaLT MI states (left/right hands, left/right legs, tongue, and passive). The training processes for the Intra-Micro 5F Classifier was similar, with labels 1 through 5 each representing a finger. Each of the three classifiers were trained on only a single patient's EEG data, and, consequently, the classification accuracies vary significantly by subject. Iterations until convergence of the SVM models also varied by subject and classification paradigm with the binary Macro Classifier converging the fastest, followed by the Intra-Micro HaLT Classifier, and then the Intra-Micro 5F classifier. For some subjects the SVM model failed to converge at all in each of the Micro paradigms, with convergence failure occurring most often in 5F. This is likely due to the high degree of similarity of MI states within this paradigm. Despite non-convergence, we evaluated these classifiers at their terminal iteration with moderate success. While the performances of these classifiers were often worse than for subjects with converged models, models for all subjects in all paradigms achieved classification accuracies that were better than chance (50% for Macro, 16.7% for HaLT, and 20% for 5F), indicating that perhaps some non-linear classifier could separate these MI states more effectively. A neural network classifier was also explored, but training time was longer and preliminary results were comparable to those of the SVM classifier.

Results of each of the three classifiers for each feature vector are summarized in Tables I-III. One of these classifiers were reconstructed using the NeuroPype pipeline framework and integrated with the Robot Control GUI using TCP communication nodes.

D. Robot Control GUI with NeuroPype

The pipeline will determine only one robot motion at any given time after a marker event, namely, either a movement of the robot end-effector in the space or a finger close/open motion. The robot GUI is shown in Fig. 1. The classified MI task will be highlighted in blue when it receives a command from Neuropype. Note that the IDLE task in the center from the "passive" MI command, represented by a circle, means the robot will not perform any movement. Additionally, the "tongue" command is used to switch between two modes of macro movements A or B.

Macro movement mode A will allow the control movement of the four directions in the horizontal plane (X-Y), namely, the FORWARD, BACKWARD, RIGHT, and LEFT directions relative to the coordinate frame by using "right foot", "left foot", "right hand", and "left hand" MI commands, respectively. Mode B controls the up/down vertical movement of the robot hand end-effector and the rotation in the clockwise (CW) or counter-clockwise (CCW) direction of the last rotary joint (wrist) that connects to the hand end-effector. In each macro

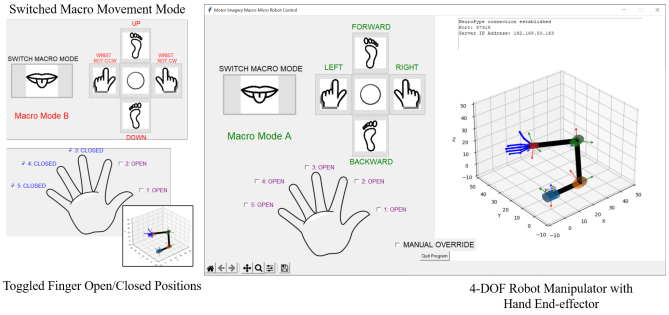


Fig. 1: Robot Control GUI. Two macro modes A and B are switched by the tongue MI. Positions of five fingers on the hand end-effector are toggled by 5F MI paradigm,

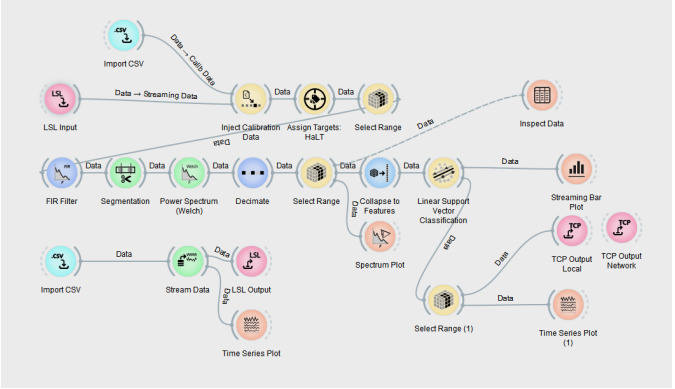


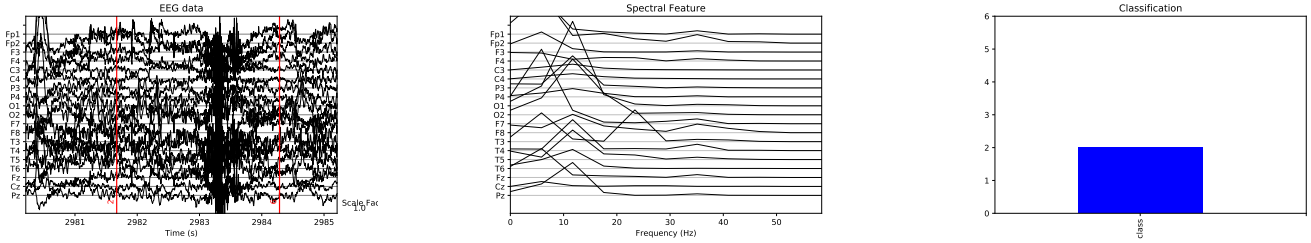
Fig. 2: NeuroPype interface of linear support vector classification (SVC) for macro MI control (HaLT paradigm)

command, the robot moves discretely by 5 cm in X (right/left), Y (forward/backward), or Z (up/down) direction or rotates by 0.25 rad around the wrist joint axis (CW/CCW).

For the micro command in the 5F MI class, each of the fingers will be toggled between its open and closed positions (checked boxes mean closed fingers). Note that because there is no data set that combines both HaLT and 5F tasks in a single data recording session, during the demonstration of the feed-forward BCI output, we will only see the robot performing only either macro or micro movements when streaming a data file (assuming that the macro vs. micro classification performance is 100%).

A demo Neuropype program with the linear support vector classification (SVC) is shown in Fig. 2. The Import CSV nodes received a time series of raw EEG data of a single subject session which was separated into training and testing portions. Three channels X3, A1, and A2 were removed.

Initially, the training data was injected to the pipeline through the Inject Calibration Data node. The signal went through an FIR filter before being segmented into 0.85 s time intervals after each of the marker events (labeled as no. 1-6 corresponding to the six HaLT MI tasks). Then the Power Spectrum (Welch) node was used to calculated the PSD of each of the 19 EEG channels. We also down-sampled the PSD data by the factor of five using the Decimate node and selected only the data within frequencies between 0 - 60 Hz.



(a) Raw EEG data with markers (b) Power Spectral Density (c) Linear SVC Output

Fig. 3: Three output visualization in NeuroPype program

Next, the PSD data of the 19 EEG channels were collapsed (by using Collapse to Features node) to a feature vector with a size of 19 $\times$ 11 = 209. This implementation resulted in an acceptable computational time required for training of the Linear SVC node in the pipeline (original size of the PSD feature vector was 1900 in the offline computation). After the SVC training has finished, the testing data will be streamed through the Stream Data node. A feature vector is extracted at a marker timestamp in the same way as the training data and enters the Linear SVC node for the online classification. The visualization nodes in the pipeline, namely, the streamed raw EEG data with markers, PSD of the 19 EEG channels, and the Linear SVC output (bar plot with a value ranging within 1-6) for each marker event, are shown in Fig. 3a, 3b, and 3c, respectively. Finally, the classification result is sent through the TCP Output node to the robot GUI.

IV. Results

A. Classification Results

Classification accuracies of the Macro and the two Intra-Micro paradigm classifiers are summarized in tables I-III. As stated previously, our classification accuracies were greater than chance in all paradigms for all subjects. This was also the case in [4], but in our exploration we considered digitally smoothed EEG signals, in addition to the PSD and FTA features, rather than raw EEG signals. In comparison with the results from [4], smoothed EEG features are more accurately classified, on average, than raw EEG features. Also in alignment with [4] is the relative classification accuracy using PSD vs FTA. Classification accuracies using FTA are higher on average than PSD. This is likely due to the more detailed encoding of information in the FTA feature vector; the FTA vector has 3800 features, whereas the PSD vector has 1900. However, the discrepancy between these two accuracies is not as large as in [4], because the researchers perform a zero-phase shift to the FTA vector to eliminate any timing delay from the onset of the action signal and the production of MI. We do not consider this phase shift in our work, but our FTA results are nevertheless comparable.

TABLE I

Intra-Micro Classification Accuracy (HaLT Paradigm)			
Subject	EEG	PSD	FTA
A	0.72	0.49	0.73
B	0.53	0.30	0.51
C	0.80	0.63	0.78
E	0.57	0.38	0.59
F	0.62	0.42	0.65
G	0.77	0.49	0.75
H	0.31	0.24	0.29
I	0.40	0.26	0.36
J	0.98	0.91	0.97
K	0.62	0.48	0.62
L	0.85	0.70	0.84
M	0.75	0.53	0.76

TABLE II

Intra-Micro Classification Accuracy (5F Paradigm)			
Subject	EEG	PSD	FTA
A	0.70	0.45	0.60
B	0.46	0.27	0.45
C	0.61	0.45	0.55
F	0.42	0.31	0.44

TABLE III

Macro Classification Accuracy	
Subject	Accuracy
A	0.98
B	0.84
C	0.95
F	0.90

V. Conclusion

In this project we created an EEG based BCI system for the control of a humanoid robotic arm and hand. With data provided by [3] we were able to successfully decode specific mental imagery states and directly translate neural activity into machine control commands. Our classification results indicate that the ability to decode neural activity is highly variable from subject to subject. For some subjects a BCI system, like the one we have set up, may work extremely well, whereas for others it may not work at all. These results are in alignment with and discussed further in [4]. Furthermore, our results confirm the quality of the EEG dataset provided by [3]. Our successful implementation of MI state classifiers and BCI control are indicative of this dataset's labelling precision and signal clarity.

References

[1] Nicolas-Alonso, L. F., Gomez-Gil, J. (2012). Brain computer interfaces, a review. Sensors (Basel, Switzerland), 12(2), 1211–1279. https://doi.org/10.3390/s120201211

[2] Berger, H. (1929). Über das elektroenkephalogramm des menschen. Archiv für psychiatrie und nervenkrankheiten, 87(1), 527-570.
[3] Kaya, M., Binli, M. K., Ozbay, E., Yanar, H., Mishchenko, Y. (2018, October 16). A large electroencephalographic motor imagery dataset for electroencephalographic brain computer interfaces. Nature News. https://www.nature.com/articles/sdata2018211.
[4] Mishchenko, Y., Kaya M., Ozbay, E., Yanar, H. (2018, August 17). Developing a Three- to Six-State EEG-Based Brain-Computer Interface for a Virtual Robotic Manipulator Control. IEEE transactions on biomedical engineering. https://pubmed.ncbi.nlm.nih.gov/30130168/.
[5] General least-squares smoothing and differentiation by the convolution (Savitzky-Golay) method. Analytical Chemistry. (n.d.). https://pubs.acs.org/doi/abs/10.1021/ac00205a007.