Brain Computer Interfaces: Final Project Report

Galloping penguin wave eats flying mango (GPW FM)
Amritaa Ganguly, Asmita Nijsure, Varsha Shankar

I. INTRODUCTION

This is the final project report for the Brain Computer Interfaces (BE 521). The project involved predicting finger flexion movements from intracranial EEG recordings in three subjects. The data and tasks are derivatives of the 4th International Brain Computer Interfaces Competition. [8]

In the following sections we present the our final algorithm (Section II), an explanation of our design choices and alternative methods considered (Sections II, III), a short discussion on the physiological structure of the hand and how it effects finger flexion (Section IV), and our conclusions regarding this project. Our references and code are also included.

II. METHODS

In this project we were given ECoG data and asked to predict finger flexion movements in three patients. Our final algorithm was derived from the simple linear regression method suggested in the BE 521 final project description.

The data included data glove traces, training ECoG traces and testing ECoG traces for which we needed to predict finger flexions. A flow chart of our algorithm is provided in Figure 1. First we train our prediction model using the training ECoG and data glove information. Our training prediction model is generated as follows:

- Feature Generation
- Feature Selection
- Channel Selection
- Partial Least Squares Regression Method
- Post-Processing
- Cross Validation

We then apply the trained prediction model to our testing ECoG data to generate predictions for finger flexions. We begin creating our training prediction model by computing features from the training ECoG data. Using feature selection we extracted the six features for each patient that we found to be most optimum for the application. (Section II-B). We also performed channel selection by the Recursive Channel Elimination method to derive the most optimum channels per finger per patient (Section II-C).

For Patients 1 and 3 we used a Partial Least Squares Regression (PLSR) method to derive weights for each column of input. The number of PLS components to be used to build a model was customised for each finger to give the best response. A feature matrix similar to that formed for training the classifier was formed from the test ECoG data. Using the weights derived, a prediction of finger flexion movement was then made. (Section II-D) Finger flexion movements of Patient 2 were seen to show better correlation to the data glove trace when linear regression was used instead of PLSR for prediction, and thus in our final algorithm Patient 2 continues to use the Simple Linear Prediction model. (Section II-E)

Three post-processing steps were performed on the predictions : interpolation, zero-clamping and smoothing. We used interpolation to match the length of the predicted data glove values to the length of the ECoG data. In zero-clamping negative values of the prediction trace were set to zero, such that it would resemble the rest period baseline signal in the data-glove trace. The prediction trace was also smoothed using a simple moving average filter. (Section II-F) The same post-processing steps were performed for these predictions as well. Cross validation was used to monitor the performance of the components of our algorithm. (Section II-G)

A more detailed explanation of algorithms we used is provided below.

A. Feature Generation

In this section we present the features we considered to capture information about finger flexion movements from the ECoG data.

It has been observed that the information about finger flexion movement is contained in particular frequency bands in ECoG data. As a result features that extract frequency band information have been used extensively for decoding motor movements. [10] [6] [2]

We used the band specific amplitude modulation features as described by Sanchez, et al. in the bands 1-60 Hz, 60-100 Hz, 100-200 Hz and 200-300 Hz. This feature is defined as the sum of the power of the bandpass filtered ECoG voltage signal. Elliptical filters of length 15 with 3 dB ripple in the pass band and a stopband 50 dB down from peak pass band value were used for band-pass filtering.

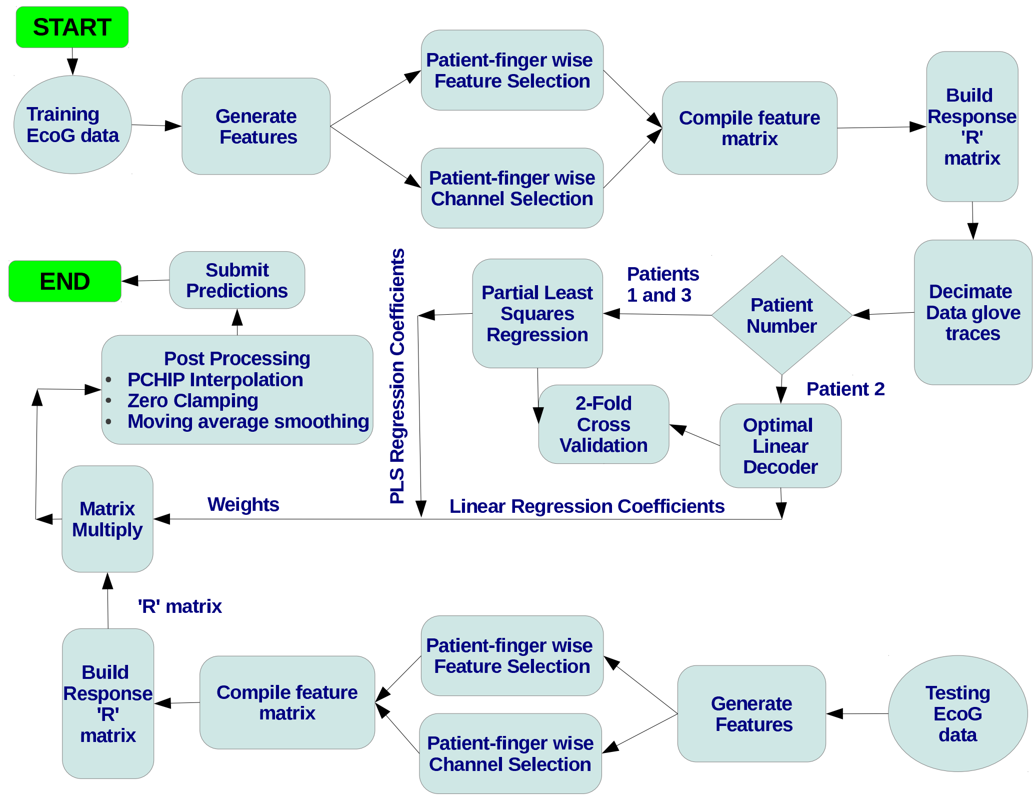

Fig. 1. **Flow Chart Summarizing General Steps of our Algorithm**

$x(t_n) = \sum_{i=0}^{100ms} v^2(t_n + i)$,
where $t_{n+1} = t_n + 50ms$.

We also used a feature employing spectral amplitudes in the 75-115 Hz (high gamma) band as has been described by Kubanek, et al. This feature has been reported by them to give a high correlation between the predictions and the actual movement when used individually.

$x(t) = \sum_{f=f_1}^{f_2} X(f)$
where X(f) is the representation of ECoG data in frequency domain.

The same feature was also calculated for other frequency bands and used, but was eventually discarded in favour of the AM features mentioned above, which resulted in better predictions of finger movements.

We also used the LMP (Locomotor Potential), which is the average time domain amplitude of the signal.

$x(t) = \sum_{t=0}^{100ms} v(t)$
The features were calculated on windows of length 100 ms with 50 ms displacement.

B. Feature Selection

A single feature may contain more information for certain fingers than it does for other fingers for the same patient. Thus we selected optimum features for each finger by running cross validation for various combinations of features for each finger. Features thus customised to each finger of each patient were then arranged in a matrix such that the data from last 150 ms is used to predict the movement at a given time. Detailed information regarding the used feature-finger-patient combination is provided in Table I.

Patient 1	LMP	AM 1	AM 2	AM 3	AM 4	AvSpAmp
Finger 1	1	1	1	1	1	1
Finger 2	1	1	1	1	1	1
Finger 3	1	0	0	1	0	1
Finger 4	1	0	1	1	1	0
Finger 5	1	1	1	1	0	0

TABLE I

THIS TABLE SUMMARIZES THE FEATURE SELECTION FOR PATIENT 1. A '0' FOR A FEATURE SIGNIFIES THAT IT WAS NOT USED FOR PREDICTION. LMP = LOCOMOTOR POTENTIAL, AM 'X'= AMPLITUDE MODULATION FEATURES IN THE BANDS X=1 :1-60 HZ, X=2: 60-100 HZ, X=3: 100-200 HZ AND X=4: 200-300 HZ RESPECTIVELY, AVSPAMP = AVERAGE SPECTRAL AMPLITUDE FEATURE IN THE BAND 75-115 HZ.

C. Channel Selection

We used the Recursive Channel Elimination (RCE) method [7] for channel selection. Instead of using the mean of the weight vector of a SVM to determine which channels are the best, we used the mean of the weight vector obtained from the linear regression algorithm. The method was as follows:

- In every iteration, a linear regressor was trained and weights were calculated.
- The mean of the weights were sorted and the channel corresponding to the weights with the lowest mean was discarded.
- Peform cross validation.
- Repeat this method was performed until the number of channels is reduced to 1.

The channels which resulted in the highest correlation values via cross validation were chosen as the optimal channels. This method was performed for each finger of each patient.

After the optimal features and channels are selected for each finger of each patient, these values are fed into an algorithm to predict finger flexion values. For Patients 1 and 3 the Partial Least Squares Regression method is used (next section) while in Patient 2, Simple Linear Regression is used (Section II-E).

D. Partial Least Squares Regression

Partial Least Squares Regression (PLSR) is used to predict finger flexions in Patients 1 and 3. PLSR is an extension of multiple linear regression and principal component analysis. It models a response variable when there are a large number of predictor variables, and those predictors are highly correlated or even collinear.

The purpose of PLSR is to build a linear model $Y = XB + E$ where Y is a vector of response variables, X is a matrix of predictor variables, B is the matrix of regression coefficients and E is a noise term. Unlike a regular linear regression model, however, a factor score matrix is derived from the predictor variables such that the variables have no correlation between them. This factor score matrix T, is computed as $T = XW$ where the W weight matrix is computed to maximise the covariance between the responses and the factor scores. An ordinary least squares procedure is then followed to find Q such that $Y = TQ + E$. Using Q and W, we then compute $B = WQ$.

Principal components regression and partial least squares regression differ in the methods used in extracting factor scores. During our trial using principal components regression, we found that the correlation values during our cross validation were significantly lower. In our implementation of PLSR, we used Matlab's plsregress().

E. Simple Linear Regression

As a part of a simple method used to cross Checkpoint 1, we implemented the optimal linear decoder as described in [12]. This simple linear regression model was also used to predict finger flexions for Patient 2.

We constructed a row vector by concatenating the feature values for each selected channel one after the other. For example, if for a certain finger, we found 50 channels and 6 features to be most effective, for each time bin (corresponding to a row vector), we would have $50 * 6 = 300$ entries in the row vector.

Feature values were thus calculated for each channel of interest using a moving window of 100 ms and a window overlap of 50 ms. Each block of time for which a feature was calculated is a time bin. Concatenated feature vectors built as described above were arranged row by row corresponding to time bins. This large feature matrix was then normalised by centering and the rescaling. The response matrix R used in [12] was built using this feature matrix. We used three previous time bins to make predictions for the following data glove movement. The weights derived from R using a least squares method of regression are given by $Wt = frac(R' * R)R' * Y$ where Y are the response variables corresponding to the data glove values for a particular finger.

F. Post Processing

PLSR and Simple Linear Regression provided us with an initial finger flexion prediction. Because the dataglove had been downsampled the predicted values need to be interpolated to match the size of the ECoG data. We used a Piece-wise Cubic Hermite Interpolating Polynomial algorithm provided by MATLAB (pchip()). These prediction traces were visually evaluated and found to contain a lot of noisy data. The flexion traces from the dataglove have a baseline value during the rest periods between movement epochs. This rest period was found to be particularly noisy in the predictions. Two methods were used for processing the predictions to extract finger flexion data. The first, we will call zero-clamping and the second, is an application of a filter.

In zero-clamping all the data points with negative values were clamped to zero. This provided the baseline values for the rest periods that agreed with the real-world constraints of finger flexions. The second method we applied involved the application of a filter that was used to extract the low frequency finger flexion values from the predictions. We used a simple moving average filter filter at an optimal filter length that was chosen for every finger per patient via cross validation. The results of post-processing is shown in Figure 2.

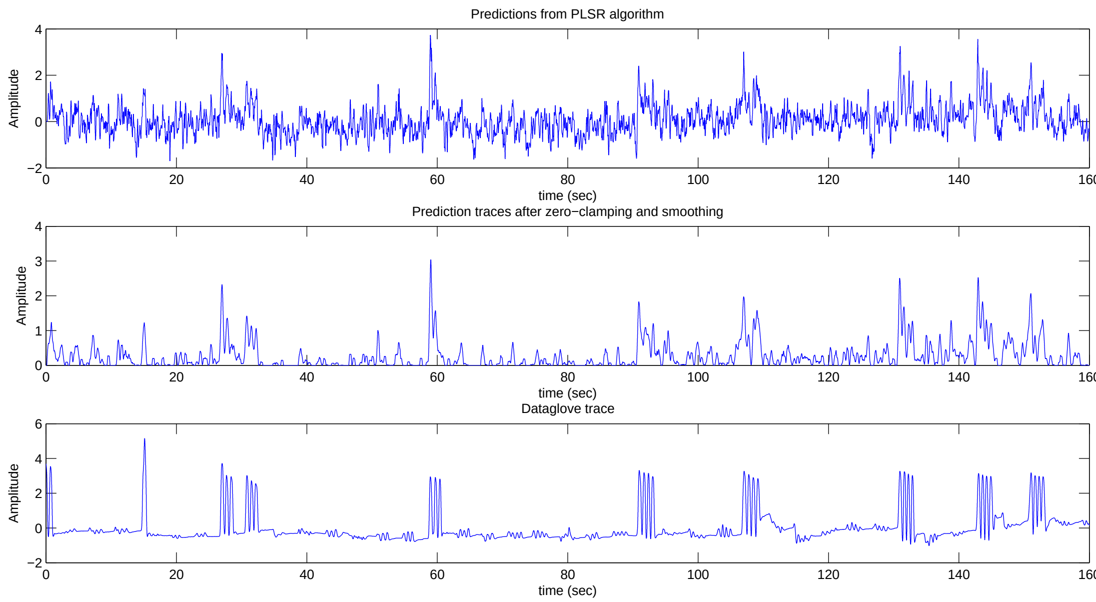

Fig. 2. **Application of Post-Processing**

G. Cross Validation Framework

A framework for easy cross validation was set up to quickly and robustly test the outcome of our numerous experiments. We used a 2-fold cross validation, the first fold trained on 3/5ths of the data and the 2nd trained on 4/5ths of the data. Testing was always performed on the remaining part of the data. The output from our cross validation always gave us a reliable measure of the comparative rise or fall in the correlation values, even though the exact values were never the same as on the testing set. An example of the predicted finger flexions is shown in Figure 3.

III. ALTERNATIVE METHODS CONSIDERED

In creating this final algorithm we had considered a number of alternative methods, including preprocessing techniques, alternative features, and prediction models.

A. Spatial Preprocessing Techniques

We were eager to try a variety of preprocessing methods to maximize the information contained in each channel of the ECoG data, similar to spatial filtering. These techniques included: Common-Averaged-Referencing, a surface Laplacian spatial filter , a multi-step preprocessing spatial filter applied in ECoG to hand-grasp tasks, and Common Spatial Pattern Algorithm.

1) Common Average Referencing: The common average reference is a method used to reduce the impact of artifacts across all channels. It is calculated by computing the global mean at a particular timepoint across all channels and subtracting this mean from each signal. For the original signal s_h, channel h, total number of channels H, the CAR-filtered signal s'_h is defined as: $s'_h = s_h - \frac{1}{H}\sum_{q=1}^{H} s_q$. While this method is mentioned as a common spatial filtering technique in numerous ECoG related movement tasks [6] our cross-validation results revealed a lower correlation between the predicted finger flexion and data glove values after applying this method and thus this method was excluded from our final algorithm.

2) Surface Laplacian Spatial Filter: The surface Laplacian filter applies the CAR method using only the channels located spatially around the channel of interest. In our project the Surface Laplacian Filter was calculated using the 8 channels located around the central electrode. [13] However, as the electrode order was scrambled in our dataset it did not lead to better results than the CAR method [8]. It would be interesting to see if this is an effective method for reducing artifacts and increasing the signal to noise ratio in ECoG data in general. Additionally, it is plausible that if the electrodes were not scrambled, and if the Surface Laplacian Spatial Filters were applied that we would achieve greater testing correlations using our algorithm.

3) Multi-step Spatial Preprocessing: A third method that was considered for spatial preprocessing included a multi-step spatial processing method employed by Pistohl et. al. in decoding hand grasp movements from ECoG data. The three steps of this method included: 1) Apply Common Average Referencing. 2) Centering

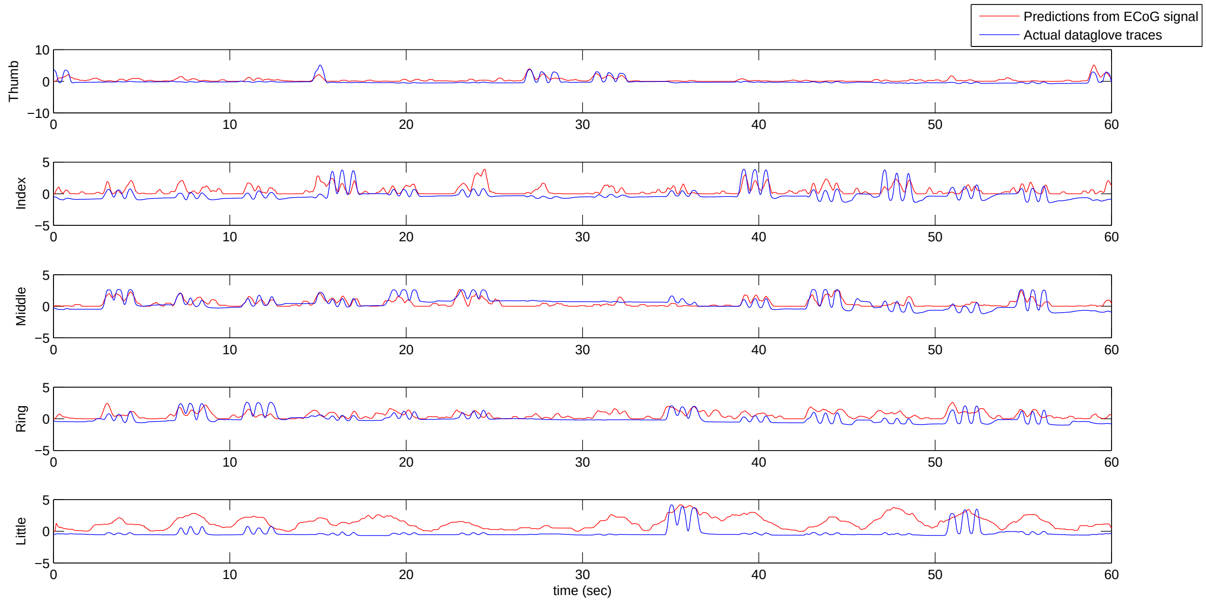

Fig. 3. **Comparison of Finger Flexion Predictions During Cross Validation**

the Data by subtracting the mean of the signal. 3) Normalizing the Data by the standard deviation of the signal. [9]

4) Common Spatial Pattern Algorithm: The Common Spatial Pattern Algorithm (CSP) was presented by Huang, et. al. as a method for increasing the spatial filter for EEG recordings used for Brain Computer Interfaces. We applied the technique to the ECoG data we were given to highlight the independent components of each electrode's signal. The CSP works by first solving for the eigenvectors of the signal, and filtering out the correlating values between electrodes. [5]

B. Features

In addition to the features presented in Section II-A we additionally considered the Katz Fractal Dimension.

1) Katz Fractal Dimension: We hypothesized that a finger flexion may correspond to a transient change in the ECoG data, and thus considered the fractal dimension of the waveform as a possible feature for our project. The Katz Fractal Dimension is defined as: $FractalDimension = \frac{\log(N-1)}{(\log(\frac{d}{L})+\log{(N-1)})}$ where N is the total number of data points in a data section to be analyzed, L is the sum of the Euclidean distances between data points (total length of the section), and d is the Euclidean distance between the first point in the window and with the farthest point (diameter of the section). Although the Katz Fractal Dimension was fast to calculate, it provided us with little additional information in classifying and predicting finger flexions and was excluded using our feature selection algorithm. [1]

C. Prediction Models

In addition to linear regression and PLSR we also considered a combination of Support Vector Machines (SVM) and Linear Regression.

1) SVM and Linear Regression: This procedure involved training a support vector machine for each finger to classify movement versus non-movement. The idea stemmed from [11] where the possibility of using generalised features from ECoG data to learn classifications is explored. Similarly, we attempted to classify movement versus non movement using learned features and an SVM trained on these features. The training labels were created by setting an empirical threshold on the amplitude of the data glove signal.

In [3], six states of the data glove movement are decoded. Each of five states correspond to a single finger's movement with the sixth state being one of rest. Post the identification of the state, a linear regressor is trained for each finger to determine amplitude.

We followed a similar procedure, but chose not to include this as part of our final algorithm because we found that the predictions were very sensitive to the quality of the labeling received by the SVM. When a labelled window of movement was shifted by as little as 1000 sample points, we found our correlation values to drop significantly.

IV. DISCUSSION

From our observations of finger flexions from the data glove recordings we notice that the th fourth (ring) finger's flexion was highly correlated with the third (middle) and fifth (little) finger's movements. We believe that this high correlation is due to the physiological organization of the muscles in the hand. Indeed, a brief glance at Differing Perspectives in Motor Learning, Memory, and Control by Goodman confirms that the ring finger is indeed innervated by the same nerves as the fifth (little) finger. Additionally the tendon muscles around the knuckles of the hand couple the movement of the ring finger with the third (middle) finger. Because these three fingers share common nerves and muscles, we expect their ECoG signals to also be coupled, and thus we observe the high correlation for our predicted values between the three fingers, and also observe the high correlation between fourth (ring) finger movement with the surrounding fingers in the data glove information. [4]

V. CONCLUSIONS

In this project, we devised a method to predict finger flexion movements of a human hand from raw Electrocorticographic signals. Various features and algorithms used in current literature were examined for analysis, data reduction and improvement in prediction. In the end, we came up with a method in which various parameters of the algorithm are tuned for each finger of a particular patient to make the best prediction of its flexion movements.

It was seen that simple methods such as filtering and post-processing resulted in larger improvements in prediction as compared to more involved methods such as Surface Laplacian Spatial Filter, Common Spatial Pattern Analysis and Katz Fractal Dimension. We also realised that it was most optimal to tune all the parameters of a method to make the best predictions possible before trying a new method, something that we realised towards the end of the project.

The competition format of the project resulted in a more intense and productive approach to the problem, along with successful maintenance of code sanity and modularity.

REFERENCES

[1] Elnaz Banan Sadeghian and Mohammad Hassan Moradi. Fractal Dimension for Detection of ERD/ERS Patterns in Asynchronous Brain Computer Interface. *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*, pages 560–563, May 2008.

[2] Laurent Bougrain, Nanying Liang, Cortex Inria, and Nancy University-loria. Band-specific features improve Finger Flexion Prediction from ECoG. *Evolution*, 2009:1–4, 2009.

[3] Rémi Flamary and Alain Rakotomamonjy. Decoding Finger Movements from ECoG Signals Using Switching Linear Models. *Frontiers in neuroscience*, 6(March):29, January 2012.

[4] David Goodman, Robert B Wilberg, and Ian M Franks. Differing Perspectives in Motor Learning, Memory, and Control. 1985.

[5] Gan Huang, Guangquan Liu, Jianjun Meng, Dingguo Zhang, and Xiangyang Zhu. Model based generalization analysis of common spatial pattern in brain computer interfaces. *Cognitive Neurodynamics*, 4(3):217–223, June 2010.

[6] J Kubánek, K J Miller, J G Ojemann, J R Wolpaw, and G Schalk. Decoding flexion of individual fingers using electrocorticographic signals in humans. *Journal of neural engineering*, 6(6):066001, December 2009.

[7] Thomas Navin Lal, Thilo Hinterberger, Guido Widman, Michael Schr, Jeremy Hill, Wolfgang Rosenstiel, Christian E Elger, and Bernhard Sch. Methods Towards Invasive Human Brain Computer Interfaces. *Computer*.

[8] Kai J Miller and Gerwin Schalk. Prediction of Finger Flexion 4 th Brain-Computer Interface Data Competition. *New York*, pages 5–6, 2008.

[9] Tobias Pistohl, Andreas Schulze-Bonhage, Ad Aertsen, Carsten Mehring, and Tonio Ball. Decoding natural grasp types from human ECoG. *NeuroImage*, 59(1):248–60, January 2012.

[10] Justin C. Sanchez, Aysegul Gunduz, Paul R. Carney, and Jose C. Principe. Extraction and localization of mesoscopic motor control signals for human ECoG neuroprosthetics. *Journal of Neuroscience Methods*, 167(1):63–81, January 2008.

[11] Pradeep Shenoy, Kai J Miller, Jeffrey G Ojemann, and Rajesh P N Rao. Generalized Features for Electrocorticographic BCIs. *Journal of Neuroscience*, pages 1–8.

[12] David K Warland, Pamela Reinagel, and Markus Meister. Decoding Visual Information From a Population of Retinal Ganglion Cells. *Molecular and Cellular Biology*, pages 2336–2350, 2009.

[13] Jonathan R Wolpaw and Elizabeth Winter Wolpaw. *Brain-Computer Interfaces: Principles and Practice*. 2012.