Motor Imagery based Brain Computer Interface for Windows Operating System

Joel Jackson Gomez[1], Sushmitha N[2]

[1]Student, Dept. of Information Science & Engineering, RV College of Engineering, Bangalore, India
[2]Assistant Professor, Dept. of Information Science & Engineering, RV College of Engineering, Bangalore, India

---

Abstract - *This paper proposes a motor imagery based brain computer interface (MI-BCI) that enables physically challenged individuals to use a computer system that is running Windows operating system. Using the MI-BCI, the user can perform basic operations on Windows such as movement of mouse and execution of mouse clicks. The MI-BCI uses Electroencephalogram (EEG) waves, captured from the user in real time, as input to a convolution neural network (CNN), which classifies the input into one of seven events: mouse movement in left, right up or down direction, left and right mouse click and finally the idle state in which no action is taken. The CNN has five layers including a max pooling layer and a fully connected (FC) layer. The hardware used to capture EEG data is the 8 channel Enobio EEG device with dry electrodes. The training data consists of beta waves (a type of EEG wave) recorded while the subject imagined moving the right arm in one of the four directions; right, left up and down and also executed left and right eye winks which is mapped to left and right mouse clicks in Windows. EEG waves are also captured while the subject is in a idle state (That is, when the subject does not want to move the mouse). The control of the mouse in Windows operating system is achieved using Python libraries. The proposed MI-BCI system is very responsive, and achieved an average accuracy of 92.85% and a highest accuracy of 97.14%.*

***Key Words*: Electroencephalogram, Motor Imagery, Brain Computer Interface, Convolution Neural Network, Beta waves**

1. INTRODUCTION

A Brain Computer Interface (BCI) is a system that maps some form of brain signal to commands on a computer system. The main goal of BCI is to replace or restore useful function to people disabled by neuromuscular disorders such as amyotrophic lateral sclerosis, cerebral palsy, stroke, or spinal cord injury. The BCI described in this paper was developed to assist physically challenged individuals in performing basic tasks on a personal computer system. It enables the user to use motor imagery to control the mouse on the Windows operating system. Motor imagery (MI) is a mental process by which an individual imagines or simulates a physical action. MI activates similar brain areas as the corresponding executed action and retains the same temporal characteristic [12]. A motor imagery-based brain-computer interface (MI-BCI) enables the brain to interact with a computer by recording and processing electroencephalograph (EEG) signals made by imagining the movement of a particular limb [1]. EEG waves can be used as input to control a BCI system even when the user has severe physical and neurological impairments [2]. These waves are chosen to be the input to the BCI system because they are easy to capture and the hardware involved is quick to setup. The amplitude of the EEG waves are measured in microvolts (mV). There are various types of EEG waves that occur at different frequency bands. Gamma waves have a frequency between 25 and 140 hertz, Beta waves have a frequency 14 to 30 hertz, Alpha waves have a frequency between 7 to 13 hertz, Theta waves have a frequency between 4 to 7 hertz, Delta waves have a frequency up to 4 hertz. The beta band is associated with motor tasks including motor imagery. An event-related synchronization (ERS) during motor planning and an event-related desynchronization (ERD) after task execution are reported in the beta band [3]. Hence beta waves are used as the input to the BCI. To collect the data, the Enobio 8 EEG device was used. It contains 8 channels which can be placed on the scalp using dry or wet electrodes. The EEG waves were recorded with dry electrodes that are placed directly on the scalp and are held in position by a cap. Dry electrodes can capture EEG waves just by coming into contact with the scalp. The electrodes are arranged on the subject's scalp so that they cover the frontal lobe which is involved in voluntary, controlled behavior such as motor functions [4]. The beta waves were collected while the subject performed motor imagery tasks such as imagining movement of right hand in right, left, up and down direction. These raw beta waves were extracted from the Enobio software using its Lab Stream Layer (LSL) feature. The Lab Streaming Layer is an opensource middleware which allows real time streaming and receiving of data between applications over a Transmission Control Protocol (TCP) network. The collected data is analyzed using MATLAB to find patterns among the brainwaves recorded during different tasks. Convolutional Neural Networks is chosen to act as the classifier for the BCI input because it can automatically extract features and its quick performance makes it well suited for real-time predictions. Reference [5] utilizes a 5 layered CNN with a max pooling layer and a fully connected (FC) layer to accurately classify EEG waves of motor imagination tasks. when compared to Recurrent Neural Networks (RNN) such as Long Short Term Memory (LSTM) and a hybrid of LSTM and CNN, the CNN model's performance was found to be of similar accuracy, but the latency of the model was much less. Hence

CNN is chosen as the classifier in this project. The data is then preprocessed and used to train the Convolutional Neural Network (CNN). Python libraries are utilized to achieve the control of the mouse in Windows operating system. The user can control the mouse movement by imagining movement of right hand and left and right mouse clicks is controlled by left and right eye winks of the user.

1.1 Related Work

Cuntai Guan, Neethu Robinson, Vikram Shenoy Handiru and V. Prasad, [6] discuss designing a BCI based upper-limb rehabilitation system for stroke patients. The authors propose a regularized Wavelet-Common Spatial Pattern (W-CSP) method for the classification of the movement directions, together with a mutual information based feature selection. They were able to achieve an average accuracy of 80.24% ± 9.41 of single trial classification in discriminating the four different directions. In their work, they also investigate methods to classify movement speeds and reconstructing movement trajectory from EEG.

Jeong-Hun Kim, Felix Bieβmann, and Seong-Whan Lee, [7] investigate two methods for decoding hand movement velocities from complex trajectories. The first method is using Multiple Linear Regression (MLR) and the second method is Kernel Ridge Regression (KRR). The results show that both MLR and KRR can reliably decode 3D hand velocity from EEG, but in the majority of subjects KRR outperformed the MLR decoding. KRR can predict hand velocities at high accuracy with fewer training data compared to MLR. How ever KRR also has a high computational cost.

D. AlQattan and F. Sepulveda, [8] propose a method to decode EEG waves into one of six American Sign Language (ASL) signs using Linear Discriminant Analysis (LDA) and nonlinear Support Vector Machines (SVM) for classification. both the algorithms showed the an average accuracy of 75% when the Entropy feature type was examined.

Rami Alazrai, Hisham Alwanni and Mohammad I. Daoud, [9] present a new EEG-based BCI system for decoding the movements of each finger within the same hand so that subjects who are using assistive devices can better perform various dexterous tasks. The BCI uses quadratic time-frequency distribution (QTFD), namely the Choi–William distribution (CWD) for feature extraction. The extracted features are passed to a two-layer classification framework that has an SVM classifier in the first layer and five different SVM classifiers corresponding to five fingers in the second layer. The classifier in the first layer identifies which one of the five fingers is moving and the corresponding classifier in the second layer identifies the type of movement for that finger. The average F1-scores obtained for the identification of the moving fingers are above 79%, where the lowest F1-score of 79.0% was computed for the ring finger.

Jzau-Sheng Lin and Ray Shihb, [10] investigate the performances of two deep learning models named Long-short term memory (LSTM) and gated recurrent neural networks (GRNN) to classify EEG signals of motor imagery (MI-EEG). The authors use the deep learning models to control a wheelchair. The experimental findings show that the GRNN always achieves better performance than the LSTM in the application to control an electric wheelchair.

Martin Spuler ,[11] presents a BCI that can interact with Windows applications by controlling the mouse and keyboard through code modulated visual evoked potentials (cVEPs). The user is presented with black and white visual stimuli that allows the user to move the mouse, click the mouse and type characters with a speed of more than 20 characters per minute. The drawback of the system was that since the BCI runs in synchronous mode a target is selected every 1.9s, which means even when the user does not want to perform any action it will still select an action to be performed. The author has proposed the addition of an asynchronous mode and a control state to improve the usability of this BCI.

2. METHODS AND MATERIALS

This section provides details on the methodology used to build and test the motor imagery based BCI. There are four stages in the methodology which are data acquisition, data analysis and pre-processing, training and testing the CNN model and finally using Python to connect Enobio, CNN model and Windows.

2.1 Data Acquisition

The Enobio 8 channel EEG device is used to for EEG signal acquisition. The 8 electrodes numbered 1 to 8 are placed in the international 10-20 system positions "AF3", "FC1", "C3", "Fz", "Cz", "C4", "FC2" and "AF4" respectively, so they can capture beta waves from the frontal lobe. Figure 1 shows the positions of the eight electrodes on the scalp. The data is collected while the subject executes motor imagery tasks of imagining movement in the right arm in four directions in center out motions as well as individually blinking the left and right eye. The idle state of the user is also captured so that the CNN model can differentiate between idle and non idle states. This is done to ensure that the model does not execute an action in Windows when the user is idle (for example; when the user is reading, watching a video or image.) The Lab Streaming Layer (LSL) feature of the Enobio software is used to stream the raw EEG data to a python program, which writes it to a comma separated values (CSV) formatted file. This is done for each of the six user actions as well as a resting state, consequently generating 7 csv files. The sampling rate used is 500 samples per second. The duration of each EEG recording is 2 seconds. Each csv file contains a 100 such recordings. Each recording contains 1000 rows and 8 columns corresponding to the 8 channels.

The EEG dataset is collected from a single subject so as to build a customized BCI.

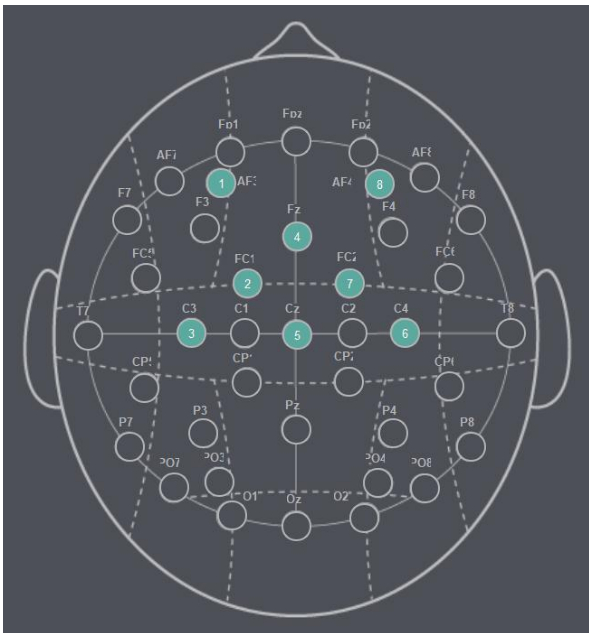

Fig -1: Positions of the eight electrodes

2.2 Data Analysis and Pre-processing

The data is analysed in MATLAB to study the different patterns of each task. The significant difference in the patterns implies that the beta waves corresponding to an action can be classified. In the pre-processing stage, the dataset is scaled using the standard scaler which is available in Python's Sklearn library. The standard score z of a sample x is computed as

$$z = (x - u) / s$$

where u and s are the mean and standard deviation of the training set. The training labels are converted from string format to machine readable format using the label binarizer function from the Sklearn library.

2.3 Training and Testing the CNN Model

The dataset is shuffled and split into training and testing sets with a split ratio of 90/10 to allow maximum number of records for training. The model is trained for one epoch with a batch size of one. The model achieves an average training accuracy of 79.09% with a standard deviation of 1.56 and an average testing accuracy of 92.85% with a standard deviation of 4.12. The highest accuracy achieved by the model is 97.14%. The functional components of the BCI such as the LSL client, CNN model and the Mouse Control Module are tested individually using the Unittest testing framework in Python. The evaluation metric used to evaluate the performance of the model is the confusion matrix shown in figure 2. The average latencies of the LSL client, CNN model, Mouse Control Module was found to be 0.275, 0.319, 0.102 seconds. The average latency of the fully integrated system was found to be 3.835 seconds. These results are tabulated in table 1.

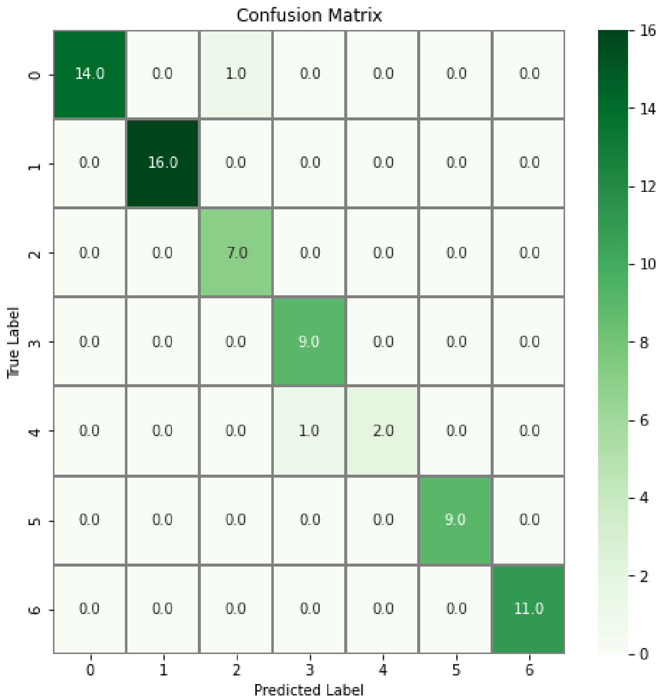

Fig -2: Confusion matrix for CNN model

Figure 2 shows the confusion matrix used to evaluate the model accuracy. It was computed for the model with accuracy of 97.14% The columns represent the predicted values and rows represent the actual values. There is some misclassification for some of the records but the overall accuracy is good.

Table -1: Performance analysis of BCI components.

Performance/ Module	Average Latency (Seconds)	Number of function calls
LSL client	0.275s	181372
CNN Model	0.319s	1285882
MCM Module	0.102s	44751
BCI (all components integrated)	3.835s	1562503

2.4 Using Python to connect Enobio, CNN model and Windows

Python contains libraries for data manipulation, pre-processing, model building and training, which makes it suitable for the BCI. The EEG data stream is captured from the Enobio software using a LSL client setup in python using the Pylsl. This LSL client provides the EEG sample. This sample is input into the CNN model to classify the user action. The CNN model contains a one-dimensional convolution layer that consists of 100 filters with a kernel size of 3 which is used to extract features from the input sample. The next layer is a one-dimensional max-pooling layer of size 2 which is used to reduce the dimensionality of the previous layer. The flattened output of the max-pooling layer is then sent to the hidden layer which consists of 20 neurons. The final layer is the output layer which contains 7 neurons. The output of the CNN model is one of 7 classes, which are "Right", "Left", "Up", "Down", "Right-Wink", "Left-Wink" and "Nothing". If the output belongs to any one of the 4 directional classes (Right, Left, Up and Down), the mouse pointer is moved in that direction with commands executed in Python. If the output is classified as either "Left-Wink" or "Right-Wink", the command to click the mouse is executed in Python. If the output is classified as "Nothing", no action is taken. The control of the mouse in Windows is achieved by using the "pyautogui" and "mouse" libraries in Python.

3. IMPLEMENTED SYSTEM

This section describes the architecture of the implemented BCI and gives a detailed explanation of its components. It also provides details on the architecture of the CNN model that was built for the BCI.

3.1 System Architecture

The project has a three-tiered architecture which consists of an LSL client, CNN model, Mouse Control Module. The LSL client is responsible for capturing a portion of the live data stream from the Enobio software. The CNN model will classify the EEG input as one of the seven output classes. The Mouse Control Module receives the classification from the CNN model. It then matches the classification to the appropriate output command that is executed in in Windows. The data only flows in one direction as shown in figure 3.

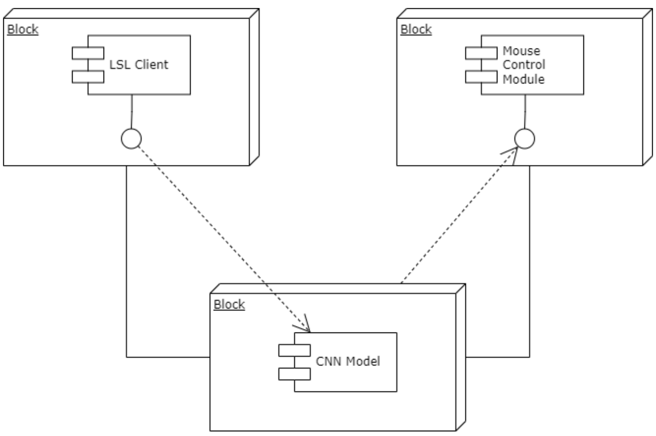

Fig -3: Three tier system architecture of MI-BCI

3.2 Architecture of CNN Model

The CNN model contains five layers as shown in figure 4. The first layer is a one-dimensional convolution layer containing 100 filters and a kernel size of 3. The second layer is a one-dimensional max-pooling layer of size 500. The third and fourth layers are fully connected hidden layers containing 500 and 100 neurons with leaky ReLU activation. ReLU stands for rectified linear unit.

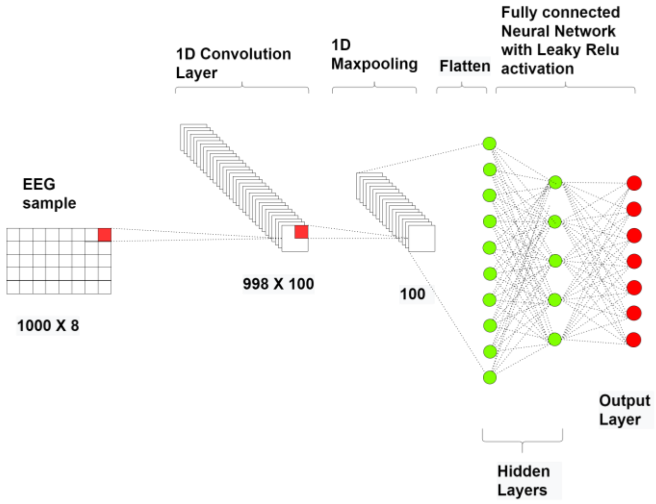

Fig -4: Architecture of CNN model

Leaky ReLU was chosen over ReLU as the activation function for neurons in the hidden layer as it prevents neurons from dying due to negative inputs. The leaky ReLU function for a given input x is defined as,

$$\text{LeakyReLU}(x) = \max(0, x) + \text{negative\_slope} * \min(0, x)$$

or

$$\text{LeakyRELU}(x) = \begin{cases} x, & \text{if } x \geq 0 \\ \text{negative\_slope} \times x, & \text{otherwise} \end{cases}$$

Where negative slope controls the angle of the negative slope. Its default value is 1e-2. The fifth layer is the output layer containing 7 neurons with "softmax" activation. The softmax function takes an input vector z of K real numbers and normalizes it into a probability distribution.

It applies the standard exponential to every element in the input vector and divides each exponential by the sum of all the exponentials to normalize them. The sum of the elements of the output vector is 1. The softmax function for an input vector z of K real numbers is defined as,

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{for } i = 1, \dots, K \text{ and } \mathbf{z} = (z_1, \dots, z_K) \in \mathbb{R}^K.$$

It is commonly used in the output layer of a multiclass neural network. A multiclass classification is the problem of classifying instances into one of more than three classes. The first two layers of the CNN model are used for feature extraction and the next three layers are used for classification. Before the data is sent to the hidden layers it is flattened to make it one dimensional. To regularize the model and avoid overfitting, during training random nodes are dropped. A dropout factor of 0.5 is used during training to avoid overfitting the model.

4. DISCUSSION

The user is able to use the MI-BCI system to control the mouse on Windows. The actions that can be performed include mouse movement as well as clicking. However, after the execution of an action, the user must take a moment to relax and allow their brain waves to return to their normal state before taking the next action. Due to this drawback, the user cannot continuously control the computer. This also means that the MI-BCI does not select an action when the user does not want to execute an action, which is an improvement on [11]. Another drawback is that the user cannot control the keyboard with this BCI. To address this problem, the addition of a c-VEP method to control the keyboard such as the one described in [11] will be the next step.

5. CONCLUSIONS

This paper has presented a motor imagery based BCI system which uses convolution neural networks to classify an individual's EEG waves into computer commands. Although drawbacks of the current system are outlined, next steps in improving the system are discussed. The five-layer CNN model achieved the an average accuracy of 92.85% with a standard deviation of 4.12 and a highest accuracy of 97.14%. The BCI is very responsive with an average latency of 3.85 seconds for signal acquisition, classification and execution of the appropriate command, which is very good for a user interface system.

ACKNOWLEDGEMENT

I would like to thank my guide, Sushmitha N, Assistant Professor, Department of Information Science & Engineering, RV College of Engineering for introducing me to the subject of electroencephalography and brain computer interfaces and for her continual support, guidance and patience in completing this project.

I would also like to express my sincere gratitude to Dr. B. M. Sagar, Head of Department, and Dr. G. S. Mamatha, Professor and Associate Dean, Department of Information Science &

Engineering, RV College of Engineering for providing the necessary resources such as the Enobio device to complete this project.

REFERENCES

[1] Arefeh Nouri, Zahra Ghanbari, Mohammad Reza Aslani, Mohammad Hassan Moradi, "A new approach to feature extraction in MI-based BCI systems", Artificial Intelligence-Based Brain-Computer Interface, Jan. 2022, pp. 75-98, doi:10.1016/B978-0-323-91197-9.00002-3.

[2] Lazarou Ioulietta, Nikolopoulos Spiros, Petrantonakis Panagiotis C., Kompatsiaris Ioannis, Tsolaki Magda, "EEG-Based Brain–Computer Interfaces for Communication and Rehabilitation of People with Motor Impairment: A Novel Approach of the 21st Century". Frontiers in Human Neuroscience, vol. 12, 2018, doi: 10.3389/fnhum.2018.00014.

[3] Jerrin Thomas Panachakel, Kanishka Sharma , Anusha A S , and Ramakrishnan A G, "Can we identify the category of imagined phoneme from EEG?", 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Oct. 2021, doi: 10.1109/EMBC46164.2021.9630604.

[4] Firat Rengin B., "Opening the "Black Box": Functions of the Frontal Lobes and Their Implications for Sociology", Frontiers in Sociology, vol. 4, Feb. 2019, p. 3, doi: 10.3389/fsoc.2019.00003.

[5] Xiangmin Lun, Zhenglin Yu, Tao Chen, Fang Wang and Yimin Hou , "A Simplified CNN Classification Method for MI-EEG via the Electrode Pairs Signals", Frontiers in Human Neuroscience, vol. 14, Sept. 2020, p. 338, doi: 10.3389/fnhum.2020.00338.

[6] C. Guan, N. Robinson, V. S. Handiru and V. A. Prasad, "Detecting and tracking multiple directional movements in EEG based BCI," 2017 5th International Winter Conference on Brain-Computer Interface (BCI), 2017, pp. 44-45, doi:10.1109/IWW-BCI.2017.7858154.

[7] J. -H. Kim, F. Bießmann and S. -W. Lee, "Reconstruction of hand movements from EEG signals based on non-linear regression," 2014 International Winter Workshop on Brain-Computer Interface (BCI), Feb. 2014, pp. 1-3, doi: 10.1109/iww-BCI.2014.6782572.

[8] D. AlQattan and F. Sepulveda, "Towards sign language recognition using EEG-based motor imagery brain computer interface", 2017 5th International Winter Conference on Brain-Computer Interface (BCI), Jan. 2017, pp. 5-8, doi: 10.1109/IWW-BCI.2017.7858143.

[9] Rami Alazrai, Hisham Alwanni, Mohammad I. Daoud, "EEG-based BCI system for decoding finger movements within the same hand", Neuroscience Letters, vol. 698, Apr. 2019, pp. 113-120, doi: 10.1016/j.neulet.2018.12.045.

[10] J. Lin, and R. Shihb, "A Motor-Imagery BCI System Based on Deep Learning Networks and Its Applications", in Evolving BCI Therapy - Engaging Brain State Dynamics, Oct. 2018, doi: 10.5772/intechopen.75009.

[11] Spuler M., "A Brain-Computer Interface (BCI) system to use arbitrary Windows applications by directly controlling mouse and keyboard.", Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society, Aug. 2015, doi: 10.1109/EMBC.2015.7318554.

[12] Reinhold Scherer, Carmen Vidaurre, Chapter 8 - Motor imagery based brain–computer interfaces, Smart Wheelchairs and Brain-Computer Interfaces, 2018, pp. 171-195, doi:10.1016/B978-0-12-812892-3.00008-X.