SUMMER PROJECT

Science and Technology Council, IIT Kanpur

EEG-based Servo-Motor Control

Project Report

July 2023

Contents

1 Introduction

The purpose of this project is to design a system that enables control of a servo motor using brainwave signals obtained through Electroencephalography (EEG). By detecting electrical activity in the brain, this project demonstrates a brain-computer interface (BCI) capable of translating mental patterns into actionable commands.

The process involves multiple stages — EEG signal acquisition, noise reduction through filtering, feature extraction, deep learning model development, and hardware control using Arduino. A schematic overview of the model is shown below.

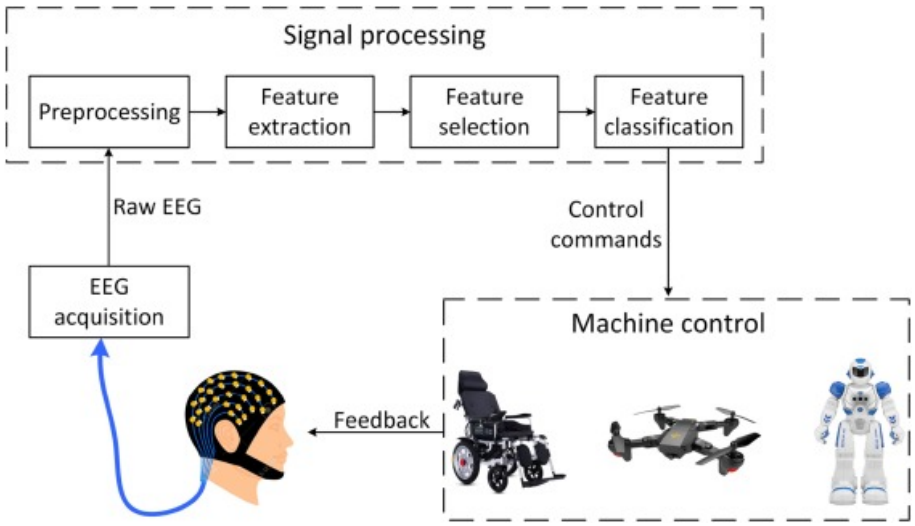

Figure 1: Overall architecture of the EEG-based control system.

2 Overview

The human brain exhibits electrical activity that can be captured as EEG signals. These signals are extremely low in amplitude (0.5–100 µV) and require amplification and noise reduction before analysis.

Brain signals are obtained using electrodes attached to the scalp and amplified using the BioAmp EXG Pill. The recorded signals are then processed in Python to generate spectrograms. A deep learning model classifies these signals into binary categories — representing "YES" or "NO". The classified output is transmitted to an Arduino board, which controls the rotation of a servo motor accordingly.

This integration of neuroscience and embedded systems forms the foundation of modern brain-controlled interfaces.

3 Theoretical Background

3.1 What is EEG?

Electroencephalography (EEG) is a non-invasive technique for recording the electrical activity of the brain. It measures voltage fluctuations resulting from ionic current flows within neurons. EEG is widely used for medical diagnostics, research on cognition, and human-computer interface systems.

3.2 EEG Signal Recording

Electrodes placed on the scalp (following the international 10–20 system) detect brain activity. The Spike Recorder software was used in this project for real-time signal acquisition and visualization. Amplified signals were then filtered to remove noise and unwanted frequencies.

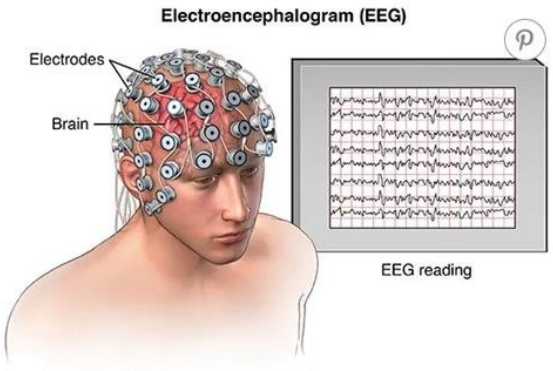

Figure 2: Data acquisition setup using EEG electrodes.

3.3 Types of EEG Waves

EEG signals are categorized by frequency bands that correspond to various mental states:

Band	Frequency Range	Associated State
Delta	0.5–4 Hz	Deep sleep, unconsciousness
Theta	4–8 Hz	Drowsiness, light sleep
Alpha	8–13 Hz	Relaxed, eyes closed
Beta	13–30 Hz	Active thinking, alertness

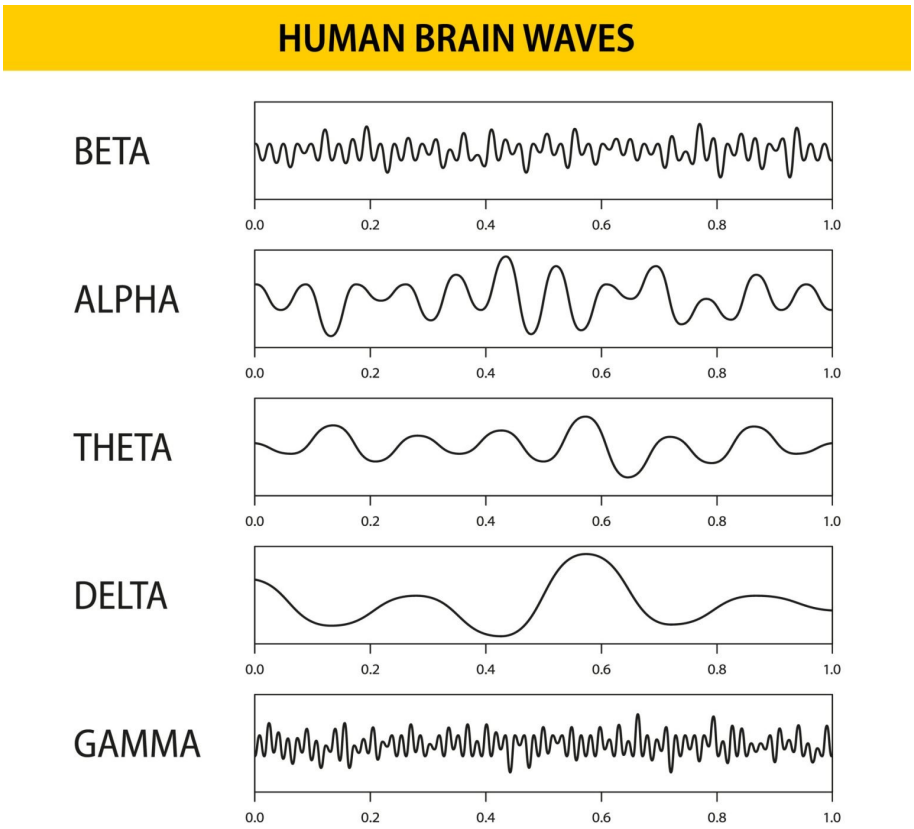

Figure 3: Typical EEG signal frequency bands.

4 Python Libraries

4.1 SciPy

The `scipy.signal` module was used to implement a Butterworth bandpass filter, isolating EEG frequencies between 0.5–30 Hz while attenuating noise from other ranges.

4.2 Librosa

Although primarily designed for audio processing, Librosa was employed to compute time-frequency spectrograms of EEG signals. These spectrograms served as CNN model inputs.

```python
def compute_spectrogram(file_path):
    signal, sample_rate = librosa.load(file_path, sr=22050)
    signal=signal[:10*22050]
    h_l = 512
    nfft = 2048
    stft = librosa.stft(y=signal, n_fft=nfft, hop_length=h_l)
    log_stft = librosa.amplitude_to_db(np.abs(stft))
    return log_stft
```

Figure 4: Python script for EEG spectrogram generation using Librosa.

4.3 TensorFlow

TensorFlow was used to design and train a Convolutional Neural Network (CNN) for binary classification of EEG spectrograms (YES/NO). Training was performed using the Adam optimizer and binary cross-entropy loss over 10 epochs.

4.4 PySerial

`PySerial` enabled serial communication between the computer and Arduino boards. One Arduino captured EEG data, while another controlled the servo motor based on classified output.

5 Model Architecture and Training

5.1 Dataset Preparation

EEG recordings were stored as `.wav` files and labeled with binary outcomes (1 for YES, 0 for NO). The dataset was organized in an Excel sheet with file paths and corresponding labels.

5.2 Neural Network Architecture

The CNN model comprises:

- Two `Conv2D` layers (32 and 64 filters) with ReLU activation
- `MaxPooling2D` layers for downsampling
- A `Flatten` layer
- Dense layers (128 and 64 neurons) with dropout (0.2)
- Output layer with `sigmoid` activation

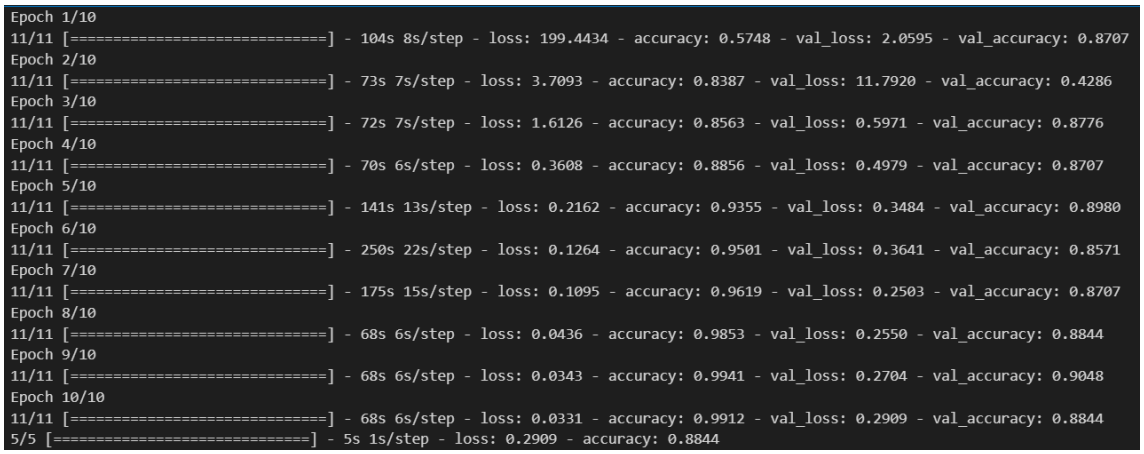

Figure 5: Convolutional Neural Network architecture used for EEG classification.

5.3 Training and Evaluation

The model was trained for 10 epochs with batch normalization and evaluated on a test set. Performance metrics are summarized below:

Metric	**Training**	**Validation/Test**
Accuracy	91.6%	88.4%
Loss	0.29	0.29

6 Results

The trained CNN effectively classified EEG spectrograms into YES and NO signals. When a YES signal was detected, the servo motor rotated by a fixed angle (90°); a NO signal returned it to its initial position. This demonstrated reliable real-time brain-controlled motor actuation.

6.1 Observations

- Signal filtering was critical in improving model accuracy.
- CNN on spectrograms performed better than raw-signal classifiers.
- The average response delay between EEG detection and motor action was under 1 second.
- Accuracy remained consistent across repeated trials.

6.2 Limitations

- The dataset was limited in size, which may cause mild overfitting.

- Only binary classification (YES/NO) was implemented.
- EEG signals are highly sensitive to noise from muscle or eye movement.

7 Conclusion

This project successfully demonstrated a real-time brain-computer interface capable of controlling a servo motor using EEG signals. By integrating neuroscience, signal processing, deep learning, and embedded control, we established a complete EEG-based actuation pipeline.

The project achieved an overall test accuracy of 88.4%, validating the feasibility of mental command-based control systems. Future improvements could include:

- Expanding datasets for better generalization,
- Implementing multi-command classification (e.g., multiple motor states),
- Real-time artifact removal for noise suppression.

This approach has promising applications in assistive robotics, neuroprosthetics, and rehabilitation systems.

8 Acknowledgement

We extend our sincere gratitude to the **Electronics Club** and the **Science and Technology Council, IIT Kanpur** for providing the platform and resources for this project. We are deeply thankful to our mentors, **Sayeedul Islam Sheikh** and **Om Shrivastava**, for their valuable guidance, mentorship, and support throughout the project.