PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)

100-ft Ring Road, Bengaluru – 560 085, Karnataka, India

Report on

'Brain Controlled Interface for Controlling Robotic ARM '

Submitted by

HARSHITA R VASTRAD (PES1201701717)
V SAISRI (PES1201701763)
SHREYA V DEEXIT (PES1201701648)

Aug 2020 – May 2021

under the guidance of

Internal Guide

Dr. Niranjana Krupa
Professor
And
Prof. Shweta G
Assistant Professor
Department of ECE
PES University
Bengaluru -560085

FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGG

PROGRAM B.TECH

CERTIFICATE

This is to certify that the Report entitled

'Brain Controlled Interface for Controlling Robotic Arm'

is a bonafide work carried out by

HARSHITA R VASTRAD (PES1201701717)
V SAISRI (PES1201701763)
SHREYA V DEEXIT (PES1201701648)

In partial fulfillment for the completion of 7th semester course work in the Program of Study B.Tech in Electronics and Communication Engineering, under rules and regulations of PES University, Bengaluru during the period Aug – Dec. 2020. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The report has been approved as it satisfies the 7th semester academic requirements in respect of Capstone project work.

Signature with date & Seal
(Dr/ Prof .....)
Internal Guide

Signature with date & Seal
Dr. Anuradha M
Chairperson

Signature with date & Seal
Dr. B. K. Keshavan
Dean - Faculty of Engg. &Technology

Name and signature of the examiners:

1.

2.

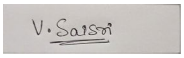
DECLARATION

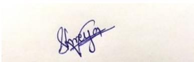
We, Shreya V Deexit, V Saisri, Harshita R Vastrad**,** hereby declare that the report entitled, ‘***Brain Controlled Interface for Controlling Robotic Arm’,*** is an original work done by us under the guidance of **Dr.Niranjana Krupa**, Professor, ECE department and **Prof.Shweta G**, Assistant Professor, ECE Department and is being submitted in partial fulfillment of the requirements for completion of 7^{th} Semester course work in the Program of Study, B.Tech in Electronics and Communication Engineering.

PLACE : BANGALORE
DATE : 08/01/2021

NAME AND SIGNATURE OF THE CANDIDATES

1.**Harshita R Vastrad -** 

2.**V Saisri -** 

3. **Shreya V Deexit -** 

ABSTRACT

Brain Computer Interface (BCI) refers to the interaction of the central nervous system or the brain with a computer where the signals generated by the brain due to external stimulus are used to control an external device. The electroencephalographic (EEG) signal obtained by the imagination of movements of hands and legs is being used in our project. The EEG signal is susceptible to external noise and hence filtering out the perturbations and from the actual EEG data classifying it and building a system is the challenge.

In this project we have preprocessed the signal to get noise-free EEG signal and extracted features from the EEG signal using time-frequency analysis. ANN model is used for the classification of Motor Imagery electroencephalography (MI-EEG) signals; to avoid overfitting of the data, we have used an early stopping method. Comparison of accuracies is done for different variations of the ANN model.

The output from the model is given to the Robotic arm. Arm can pick lightweight objects from one place and drop it to a predefined place. Robotic Arm is built using raspberry pi, DC motors, H-bridge, pi camera. Object detection is used for object identification, picking and dropping. We achieved an average accuracy of about 71.03 percent.

Keywords: Brain Computer Interface (BCI), Electroencephalography (EEG), Brain Robot Interface (BRI), Motor Imagery (MI)

Acknowledgement

With immense pleasure, presenting our project 'Brain Controlled Interface for Controlling Robotic Arm' report as a part of our curriculum in 'B.Tech in Electronics and Communication Engineering'. We wish to all people who gave us endless support throughout.

We express our profound gratitude to our guide Dr. Niranjana Krupa and Prof.Shweta G for giving us guidance and encouragement throughout the course of our project.

Shreya V Deexit

V Saisri

Harshita R Vastrad

TABLE OF CONTENTS

1. Introduction 11

1.1. Problem Statement…………………………………………………...12

1.2. Objective…………………………………………………………..12

2. Literature review 14

2.1. Brain Controlled Interface…………………………………….....14

2.2.Electroencephalogram(EEG)……………………………………......14

2.3. Motor Imagery…………………………………………………16

2.4. Wavelet Packet Decomposition……………………….............16

2.5. Related Work..……………………………………………………17

3. Methodology 19

3.1. Extracting the data……………………………………..............19

3.1.1. Dataset ……………………………………………………..19

3.1.2. Visualising the dataset…………………………………...21

3.1.3. Block diagram…………………………………………...22

3.2. Pre-processing…………………………………………………23

3.2.1. Event extraction………………………………………….23

3.2.2. Bandpass filtering………………………………………..23

3.2.3. Channel selection………………………………………...25

3.3. Wavelet packet decomposition…………………………………..26

3.4. Building the model……………………………………………28

3.5. Training ………………………………………………………….32

3.6. 10-fold cross-validation………………………………………….31

4. Design of Robotic Arm 33

4.1. Factors to take into account……………………………………..33

4.2. Hardware components……………………………………………..34

4.3.Constraints………………………………………………………….34

4.4. Computer Vision3………………………………………………….35

4.4.1. Raspberry-pi setup and configuration……………………35

4.4.2. Tensorflow setup and loading the saved model…………35

4.4.3 Object color detection……………………………………36

5. Results And Discussion 40

5.1. Analysis of result…………………………………………………..41

5.2. Conclusion………………………………………………………....42

5.3 Future scope………………………………………………………...42

References 43

List of figures

Figure 1: Brain Robot Interface Schematic……………………………………………………..13

Figure 2 : EEG electrode positions………………………………………………………..15

Figure 3: Time scheme for each trial………..……………………………………………21

Figure 4: Plot of different channels vs time along with partitioning the events……………..22

Figure 5:Process step flow diagram……………………………………………………….22

Figure 6: Frequency spectrum before bandpass filtering…………………………………....24

Figure 7: Frequency spectrum after bandpass filtering……………………………………....24

Figure 8: Left = 769, Right = 770,Forward = 771,Backward =772……………………….....25

Figure 9. DWT signal decomposition……………………………………………………..27

Figure 10. EEG Signal DWT decomposition……………………………………………..27

Figure 11: Summary of the model………………………………………………………...29

Figure 12: Illustration of early stopping…………………………………………….........30

Figure 13: Train test split………………………………………………………………….31

Figure 14: Training phase………………………………………………………………....32

Figure 15: Circuit Diagram

Figure 16:Image showing the output of detected red colored object ……………………......36

Figure 17: Setup for color detection……………………………………………………... …….36

Figure 18: Block diagram of object color detection…………………………………………37

Figure 19: Path to the robotic arm ………………………………………………………..37

Figure 20: Snapshot of Raspberry Pi terminal ……………………………………………...38

Figure 21:Robotic Arm……………………………………………………………………….. 39

List of Tables

Table 1:List of event types……………………………………………………………...20

Table 2: Decomposition level for frequency bands………………………………………..26

Table 3: Accuracy of Classification…………………………………………………………40

Table 4:Fine tuning the model by varying different parameters…………………………….41

1. INTRODUCTION

The growing number of patients suffering from paralysis is increasing throughout the globe and a lot of people must be dependent on others as they lack mobility. Any disease or injury which results in the lack of voluntary muscle movement can result in the inability of movement of the whole body. The injury of voluntary movement is the main actuator enabling people to move their body. Most patients suffer permanent paralysis while in other cases it can be temporary. BCI technology has the potential to help seriously disabled people with daily activities and human-machine interface applications. The EEG signals obtained from the brain of the user is the main attribute in a BCI system that defines how well a BCI system performs or how well a user can control a system. Robots are increasingly being used in human machine interface systems to help physically challenged people, in addition to robotics and industrial applications. Assistive robots can help disabled people perform everyday tasks in both their personal and professional lives, resulting in increased demand for them. A safe consumer can monitor robots given a variety of traditional input devices such as a mouse, a keyboard, a motion sensor or a teacher pendant in general Human Machine Interface. These machines, on the other hand, are very difficult for physically challenged or older people to use.

One of the most common neurological disorders is paralysis, which results in the loss of motion in one or more muscles of the body. Robots will help injured people who have suffered from neuromuscular injuries. Researchers designed the voice-controlled robots to assist the disabled. There are also a variety of ways to use robots that concentrate on the contact between the person and the robot without the use of a human's hand. By having the requisite facilities and preparation, certain patients' Brain Signals may be used to assist them in communicating with others as well as performing different tasks. The BCI device interprets the brain's electrical activity and produces instructions. As a result, these commands can be used to monitor external machines.

1.1. Problem Statement

- To use a brain-computer interface (BCI) to communicate between a robot and a person without the use of a human hand, and to assist patients with neurological disorders.
- The robotic arm will be able to pick up lightweight objects such as medications, water bottles, and other items from one position and drop them to a predetermined location based on the user's brain waves.

1.2. Objective

- **To use brain waves to control robotic arm movement:**
  - Pre-processing the dataset,
  - classifying and sending the classified signal output to the robot for arm movement.
- **Image processing for recognition of objects to pick and drop:**
  - This process involves object detection.
  - The robot will be able to
    - detect objects,
    - Avoid obstacles,
    - pick and drop the detected object.
- The outcome of this process is movement of the robotic arm without any motion or gesture in the patients.

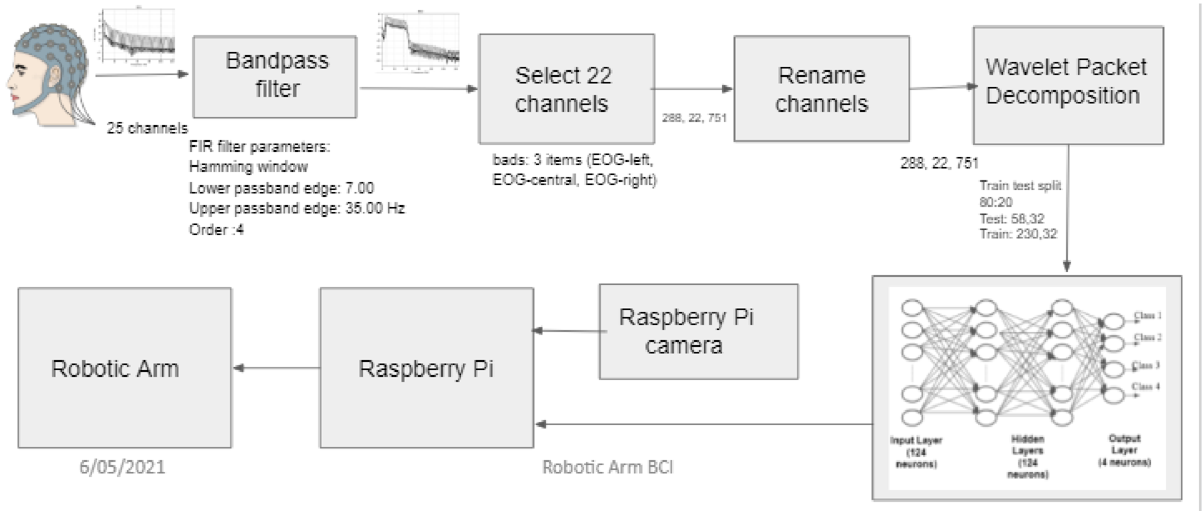


Figure 1: Brain Robot Interface Schematic

2. LITERATURE REVIEW

2.1. Brain Controlled Interface

Brain Computer Interfaces (BCIs) track brain activity by using electrodes to sense electric signals in the brain that are then transmitted to a computer. After that, the computer extracts features from the task and converts them into outputs that substitute, restore, enhance, complement, or boost human functions.

2.2. Electroencephalogram (EEG)

The electroencephalogram (EEG) is a scalp recording of the brain's electrical activity. The cortical electrical activity is reflected in the reported waveforms.

EEG signal amplitude is usually measured in microvolts (mV).

The following are the primary frequencies of human EEG waves:

1.Delta waves are the slowest. It is common as the dominant rhythm in infants up to one year old, as well as in sleep stages 3 and 4.

2.Theta- action is graded as "slow." It is common in children under the age of 13 and when they are sleeping, but it is rare in awake adults.

3.Alpha- is commonly seen in the posterior sections of the head on both sides, with the dominant side having a higher amplitude. It occurs when you close your eyes and relax, and it vanishes when you open your eyes or are alerted by some process such as thought or measuring.

4.Beta- is a form of "fast" operation. It has a symmetrical distribution on both sides and is visible from the front. It is generally thought to be a natural rhythm.When people have their eyes open, are alert, or are nervous, this is the dominant rhythm.

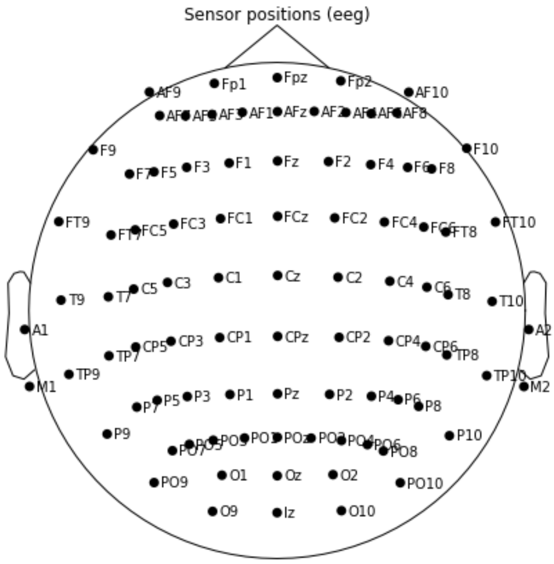


Figure 2 : EEG electrode positions

The prominent bone process normally located just behind the outer ear is referred to as the "A" (sometimes referred to as the "M" for mastoid process) which is less prominent in children and some adults. F3, F4, Fz, Cz, C3, C4, O1, O2, A1, A2 (M1, M2) are used in basic polysomnography. All EEG and EOG electrodes use Cz and Fz as 'field' or 'normal' reference points, while A1-A2 are used for contralateral referencing.

2.3. Motor Imagery

One of the standard concepts of BCI is brain computer interface based on motor imagery (MI). In MI, the user can produce induced activity from the motor cortex of the brain by imagining motor movements without any hand movement or external stimulus.

The most convenient basis for designing brain-computer interfaces is motor imagery signals recorded through electroencephalography. Since MI-based BCI allows for a high degree of independence, it enables motor-disabled people to communicate with the system by performing MI tasks in sequence.

2.4. Wavelet Packet Decomposition

The Discrete Wavelet Transform (DWT) is a multi-resolution time-frequency study of signals. DWT is preferred over Fourier Transform because it has frequency resolution as well as temporal resolution information, which is why it is called a time-frequency analysis.

Wavelet packet analysis methods are commonly used to pre-process data, resulting in a reasonably satisfactory ANN modelling performance. Typically, the decomposition level chosen is based on the length of the sequence. The maximum decomposition level (M) can theoretically be determined as $M = \log2 (N)$, where N is the length of the sequence.

The wavelet becomes smoother as the quantity of vanishing moments increases (longer wavelet filter). The wavelet filter has a length of two times that amount. The meaning of this is that if the signal behaves in an interval that is compatible with a polynomial of degree at most N and the wavelet has N vanishing moments, the wavelet coefficients in that interval will be zero. Polynomials of degree at most N are orthogonal to a wavelet with N vanishing moments. If the signal is a polynomial of degree at most 1 in the interval, a "db1" will return wavelet coefficients of zero in the interval.

2.5. Related Work

Reference paper[2] was the main reference paper used to introduce ‘Robotic Arm BCI.' We learned how to build a hybrid deep learning model using the CNN and the BiLSTM by using a bandpass filter with Hamming-windowed zero step finite impulse response (FIR). Twenty EEG channels near the primary/supplementary motor cortices were chosen after spatial filtering. The MDCBN-based multidirectional CNN-BiLSTM network (MDCBN) deep learning system was used to consider 3D multi-direction.
Using the reference paper[1], the ICA (Independent component analysis) algorithm was used to eliminate the polluted channels. The zero-phase second-order Butterworth bandpass filter with a cutoff frequency of 4 to 40 Hz, as well as the HF-CNN model of classification, were all learned from this paper. Because of the EEG uncertainty characteristics, they trained the HF-CNN according to each subject.

Reference [5] was used to investigate various classification techniques. They used three separate classification algorithms in this paper: SVM, KNN, and LDA. When all classifiers were compared, LDA had a higher accuracy of about 87.5 percent. Matlab and an Arduino board were used to interface with the robotic arm. The robotic arm is controlled wirelessly by the Bluetooth module.

The paper[3] introduced the idea of spatial filtering using typical spatial patterns. This paper also provided a brief overview of the Mutual knowledge (MI) algorithm for feature selection and the naive bayesian classifier for classification.

A comparison of different BCI classifiers was presented in a reference paper[4].

It provided an overview of the various classification strategies currently in use in the field of BCI. Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), k-Nearest Neighbor (kNN), LSTM, Bi-LSTM, CNN-LSTM, and Bi-LSTM CNN are some of the techniques used. This aided us in comprehending the variety of applications and the various models.

To focus on the interfacing and the usage of Raspberry Pi reference paper [24] was referred. ANN with 1 input layer, 1 hidden layer and 1 output layer. The object's location is fed as an input to the neural network. The neural network will process this data such that it will output a set of three angles at each joint angle of the robotic arm.The performance is measured in terms of the mean squared error (MSE) for each epoch. MSE decreased as the number of epochs increased.

In the reference paper [25] they have used 3×3 kernels in each convolutional layer, followed by a Max pooling layer. The fully-connected layer was composed of two hidden layers of 1024 hidden nodes. The output layer had 10 neurons. Then, this model was deployed on a Raspberry Pi board.They achieved a recognition rate of 89%. The one drawback of this paper is centered around high power consumption.

3. METHODOLOGY

3.1. Extracting the data

3.1.1. The Dataset

Graz University provided Dataset BCI Competition IV 2a, which we used. This dataset contains EEG data of nine healthy subjects who have performed four types of motor imagery: feet, right hand, left hand and tongue movement. For every subject, the dataset is recorded in two sessions on separate days. One of the sessions was used for instruction, while the other was used for assessment. Each training and assessment session consists of six runs separated by short breaks, with 48 trials in each run (12 trials for each class). There were 288 trials in each session (72 trials per class). Each trial consisted of instructing the participant to visualise one of four motor imagery tasks (Feet, Left hand, Right hand and Tongue) in response to a cue shown on a monitor as shown in figure 2's trial paradigm timing scheme.

The signals were captured from 22 EEG channels and three monopolar EOG channels in accordance with the international 10-20 framework. They were sampled at a rate of 250 Hz and band-pass filtered in the 0.5-100 Hz frequency range. To reduce power line noise, an additional 50 Hz notch filter was used. The EOG channels should not be used for classification and should instead be used to apply artefact removal methods afterward. The files are saved in.gdf format.

The workspace would then have two variables: signals s and a header structure h. There are 25 channels in the signal variable (the first 22 are EEG signals and the last three are EOG signals). The header structure stores event data that defines the data's structure over time. The fields below provide critical details for evaluating this data set:

h.EVENT.TYP

h.EVENT.POS

h.EVENT.DUR

h.EVENT.POS contains the location of an event in samples. The class for that event can be found in h.EVENT.TYP, and the period of that event can be found in h.EVENT.DUR.The types used in this data set are described in Table 1.

Event type	Description
276	Idling EEG (eyes open)
277	Idling EEG (eyes closed)
768	Start of a trial
769	Cue onset left (class 1)
770	Cue onset right (class 2)
771	Cue onset foot (class 3)
772	Cue onset tongue (class 4)
783	Cue unknown
1023	Rejected trial
1072	Eye movements
32766	Start of a new run

Table 1: List of event types

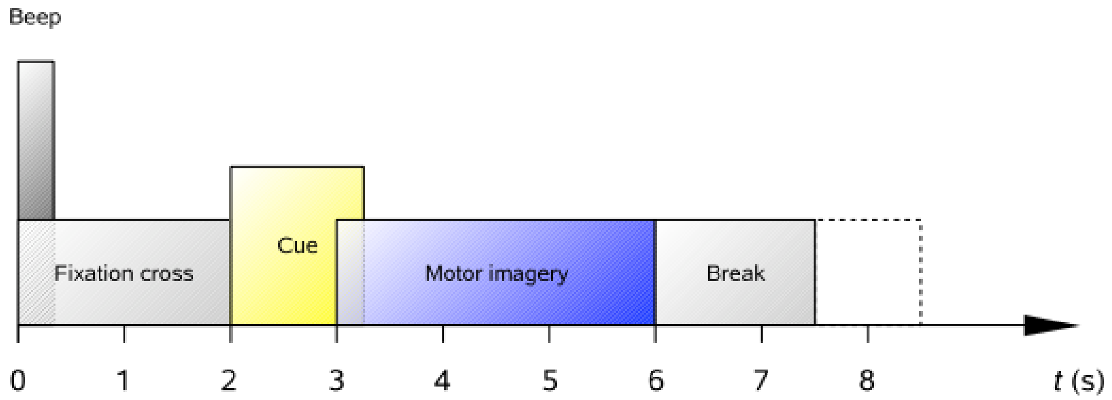


Figure 3: Time scheme for each trial

3.1.2. Visualising the Dataset

SigViewer is an application that is used to view biosignal time series such as EEG or MEG. It can build, modify, and display event information in addition to displaying raw data (such as annotations or artefact selections). Figure 4 is a plot of 25 EEG channels vs time and each event is given a separate colour to differentiate between the events as listed in Table 1.

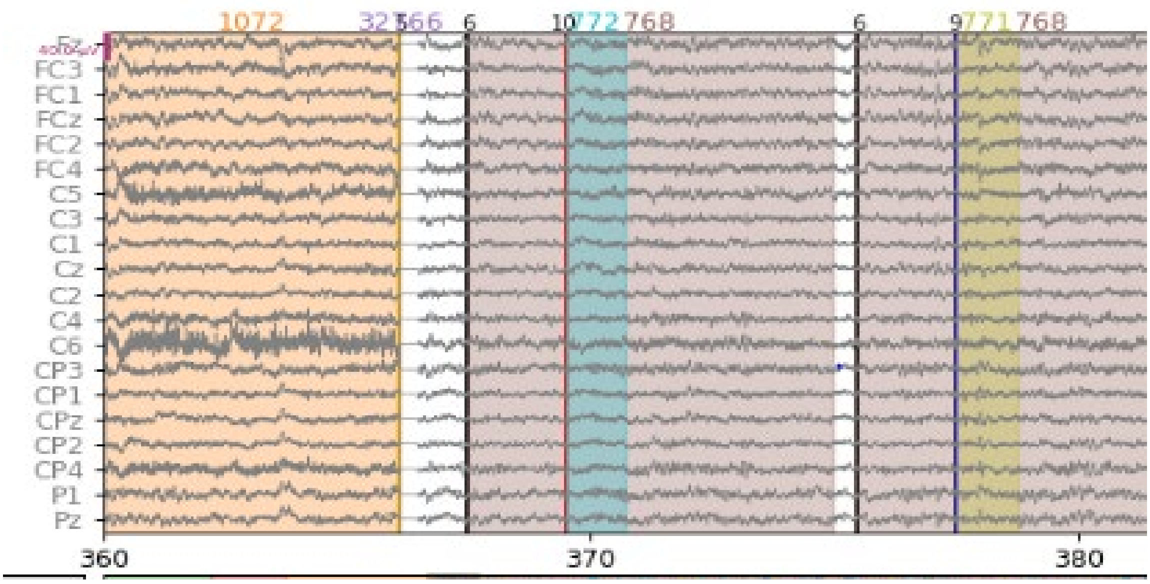


Figure 4: Plot of different channels vs time along with partitioning the events

3.1.3. Block Diagram

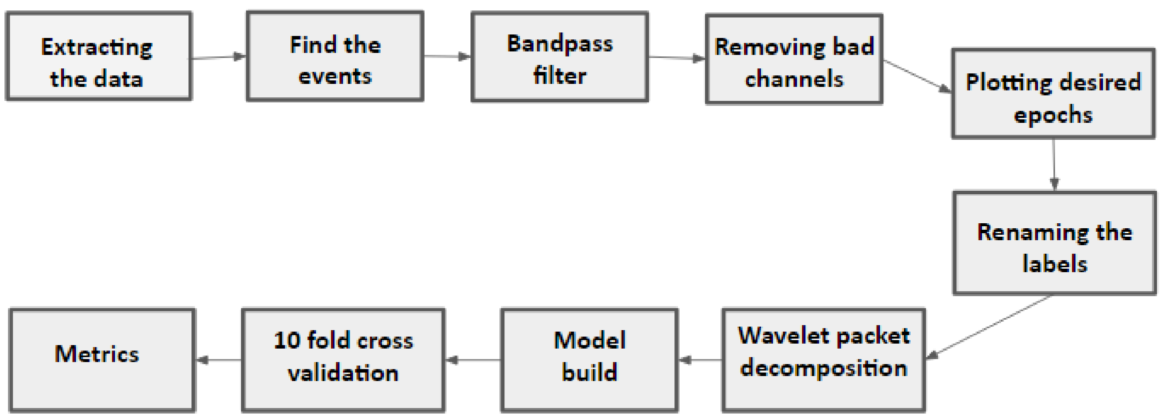


Figure 5: Process step flow diagram

3.2. Pre-processing

3.2.1. Event Extraction

On the basis of the events, voluntary and fictional movements were gathered, and these events were extracted and divided into 11 forms. Only four events out of eleven were needed, and they are as follows:

i)Cue onset left (Class 1)

ii)Cue onset right (Class 2)

iii)Cue onset foot (Class 3)

iv) Cue onset tongue (Class 4)

3.2.2. Bandpass Filter

EEG signals are non-stationary and include alot of noisy signals caused due to objects like eye movement, eye blinks, muscle movement and heart signals that are filtered out of the EEG signal in order to maximise the signal-to-noise ratio.

After extracting the necessary events, we pass frequencies in a frequency band via a one-pass, zero-phase, non-causal bandpass filter with a hamming window. The frequency range of 7Hz to 35Hz has been chosen.Figure 6 depicts the spectrum plot before bandpass. Figure 7 which is the spectrum plot that depicts the area that was used in the model.

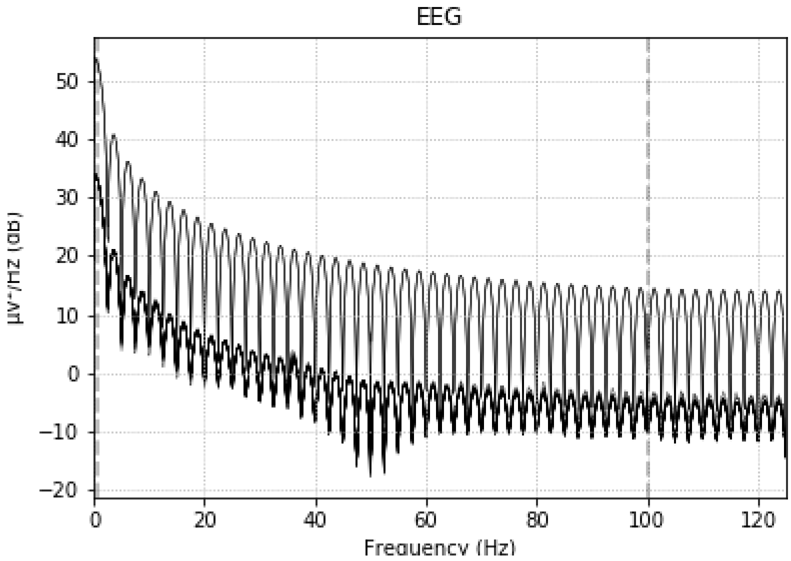


Figure 6. Frequency spectrum before bandpass filtering

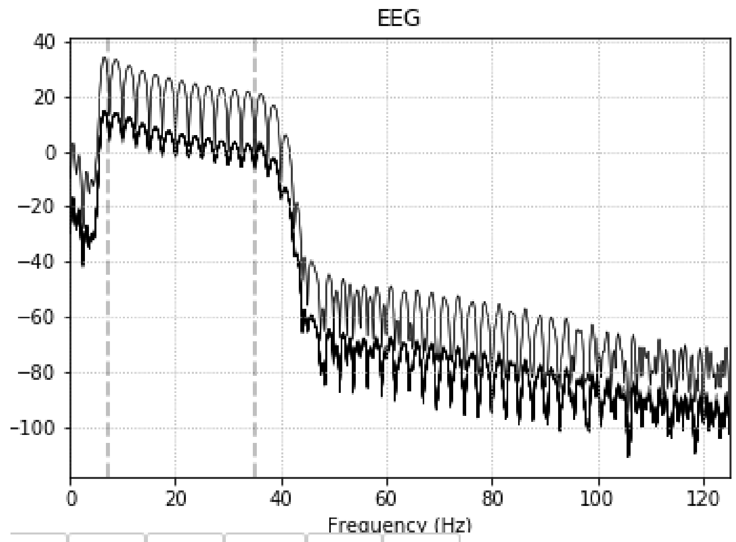


Figure 7: Frequency spectrum after bandpass filtering

3.2.3. Channel Selection

According to the literature review, the majority of EOG channel signals represent redundant information about brain function. The EEG signals are extracted using 25 electrodes, 22 of which are EEG channels and 3 of which are EOG channels. EOG channels are regarded as undesirable. The existence of EOG channels in the classification process will have an impact on the classification rate. As a result, these poor channels are omitted for improved accuracy, leaving 22 channels for classification.

Figure 8 is the plot of desired class epochs which is averaged over the 22 EEG channels. Only 4 classes (769,770,771,772 as labelled in the dataset) are extracted and averaged.

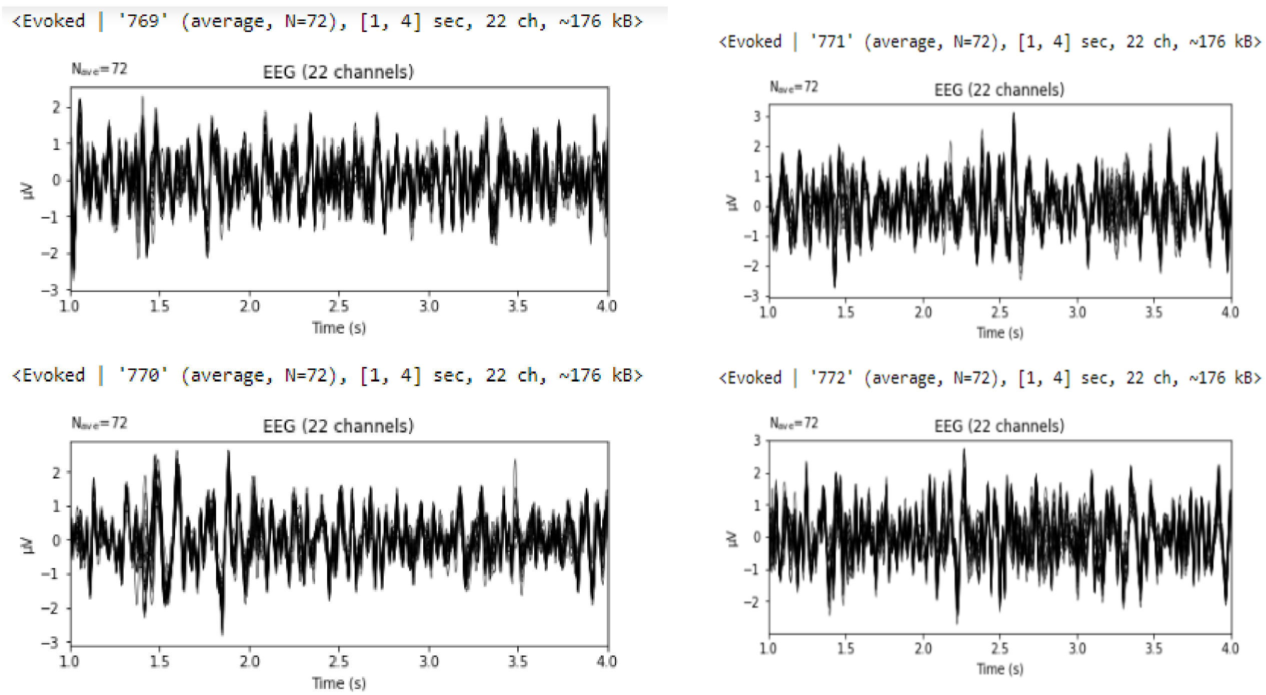


Figure 8: Left = 769, Right = 770, Forward = 771, Backward= 772

3.3. Wavelet packet decomposition

We split the EEG signal into sub-bands such as Delta (0.5 to 4 Hz), Theta (4 to 8 Hz), Alpha (8 to 16 Hz), Beta (16 to 32 Hz), and Gamma (>32 Hz) to analyse it.

On the filtered EEG signals, a two-level Discrete Wavelet Transform with 'db4' as mother wavelet was applied, yielding three groups of comprehensive coefficients d3, d4 and d5 from the signal.

Decomposition Level	Frequency Bandwidth(Hz)	Frequency Bands
D1	64-128	Noise
D2	32-64	Noise(Gamma)
D3	16-32	Beta
D4	8-16	Alpha
D5	4-8	Theta
A5	0.5-4	Delta

Table 2: Decomposition level for frequency bands

A signal's single level DWT is determined by passing it through highpass and lowpass filters, which generate informative and approximated coefficients, respectively. Using Nyquist law, by subsampling the signals with 2, half of the samples can be discarded. The estimated coefficients from every level are decomposed further for N multi-level DWT by continuously passing them through highpass and lowpass filters until the Nth level informative and approximated coefficients are obtained, as seen in figure 9. The frequency resolution of the decomposed EEG signal can be seen in figure 10.

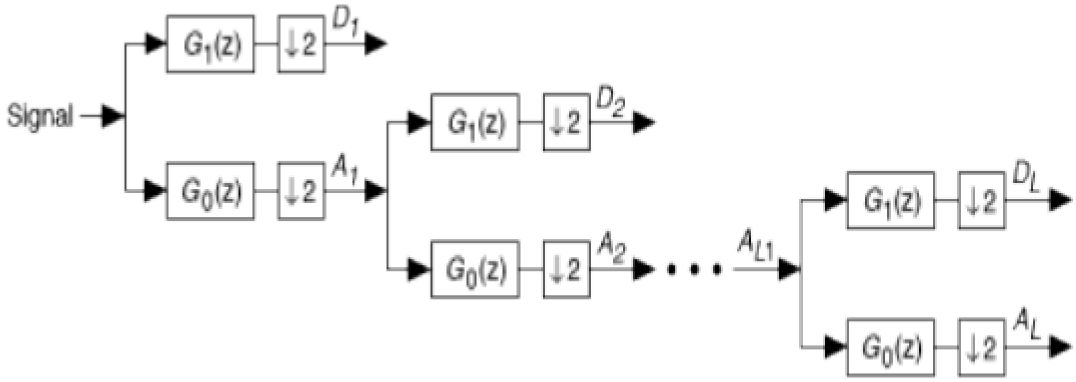


Figure 9. DWT signal decomposition

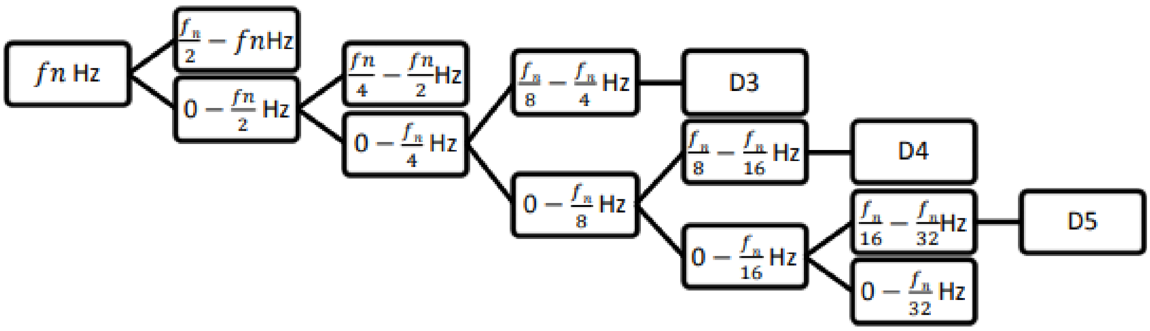


Figure 10. EEG signal DWT decomposition

3.4. Building the model

Artificial neural networks architecture is used to classify the extracted features from the EEG signals into four groups. The model is made up of four layers. The neural network's performance matrix is 4x288, with four motor imagery tasks per 288 data samples. There are 124 neurons in the input layer. A 0.5 dropout is used. Relu is the activation feature used in this layer.

The model has two hidden layers, each with 124 neurons, a Relu activation feature, and a dropout of 0.5. The L2 regularisation method was used, and it worked well.

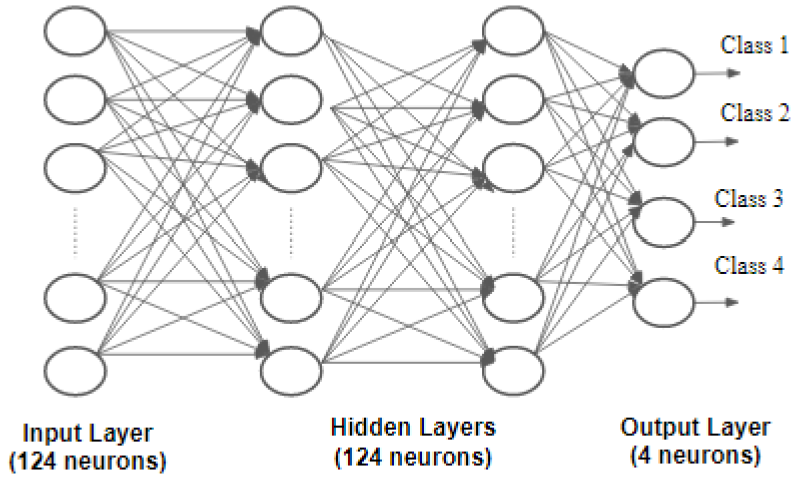
Since there are four groups to classify, the output layer has four neurons.

Each class is represented by a neuron.

Softmax is the activation function used in the output layer.

Since this is a multiclass problem, the optimizer chosen for this model is Rmsprop, and the loss is categorical cross entropy.

The description of the model we used is shown in the diagram below.



Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 124)	4092
dropout_1 (Dropout)	(None, 124)	0
dense_2 (Dense)	(None, 124)	15500
dropout_2 (Dropout)	(None, 124)	0
dense_3 (Dense)	(None, 124)	15500
dropout_3 (Dropout)	(None, 124)	0
dense_4 (Dense)	(None, 4)	500

Total params: 35,592
Trainable params: 35,592

Figure 11: Summary of the model

Early stopping

The most common issue with neural network training is deciding how many training epochs to use. Overfitting the training dataset can result in a decrease in model accuracy, whereas selecting a small number of epochs can lead to an underfit model. The Early stopping technique allows us to set out a large count of training epochs and then end training when the model's output on a hold out validation dataset stops improving.

The plot of training and validation failure against the number of epochs is shown in Figure 12.The training is stopped when the validity loss attempts to increase in relation to the training loss, as seen in the graph.

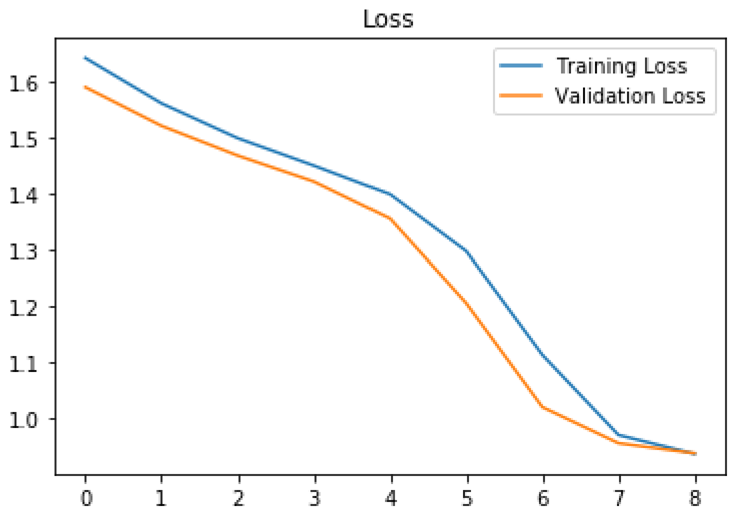


Figure 12: Illustration of early stopping

3.5. Training

Training was handled with the aid of the proposed model(algorithm). 80% of the total dataset is used for training the model. From 288 data samples, 230 data samples were used as training set and the remaining 58 data samples were used for testing. The model was initially trained with 300 epochs, after which the early stopping was implemented such that training loss never goes below validation loss.

After training the model we saved the model weight with the help of a tensorflow module called save_model. Figure 14 illustrates the training phase which displays losses and accuracy for that particular epoch of training.

```
230/230 [==============================] - 0s 226us/sample - loss: 0.8754 - acc: 0.4913
Epoch 12/300
230/230 [==============================] - 0s 187us/sample - loss: 0.8645 - acc: 0.5565
Epoch 13/300
230/230 [==============================] - 0s 222us/sample - loss: 0.8796 - acc: 0.4739
Epoch 14/300
230/230 [==============================] - 0s 209us/sample - loss: 0.8572 - acc: 0.5478
Epoch 15/300
230/230 [==============================] - 0s 209us/sample - loss: 0.8490 - acc: 0.5217
Epoch 16/300
230/230 [==============================] - 0s 213us/sample - loss: 0.8510 - acc: 0.5043
Epoch 17/300
230/230 [==============================] - 0s 209us/sample - loss: 0.8469 - acc: 0.5478
Epoch 18/300
230/230 [==============================] - 0s 204us/sample - loss: 0.8429 - acc: 0.5435
Epoch 19/300
230/230 [==============================] - 0s 209us/sample - loss: 0.8471 - acc: 0.5130
Epoch 20/300
230/230 [==============================] - 0s 204us/sample - loss: 0.8448 - acc: 0.5652
```

Figure 13: Training phase

3.6. 10-fold cross-validation

The initial sample is randomly partitioned into 10 equal-sized subsamples in 10-fold cross-validation. A single subsample from the ten is held as validation data for testing the model, while the remaining nine are used as training data. The cross-validation process is then replicated ten times (the folds), with each of the ten subsamples serving as validation data exactly once.

The ten folds results can then be summed to create a single estimate. This method has the advantage of using all data samples for both training and testing, and each observation is only used for testing once.

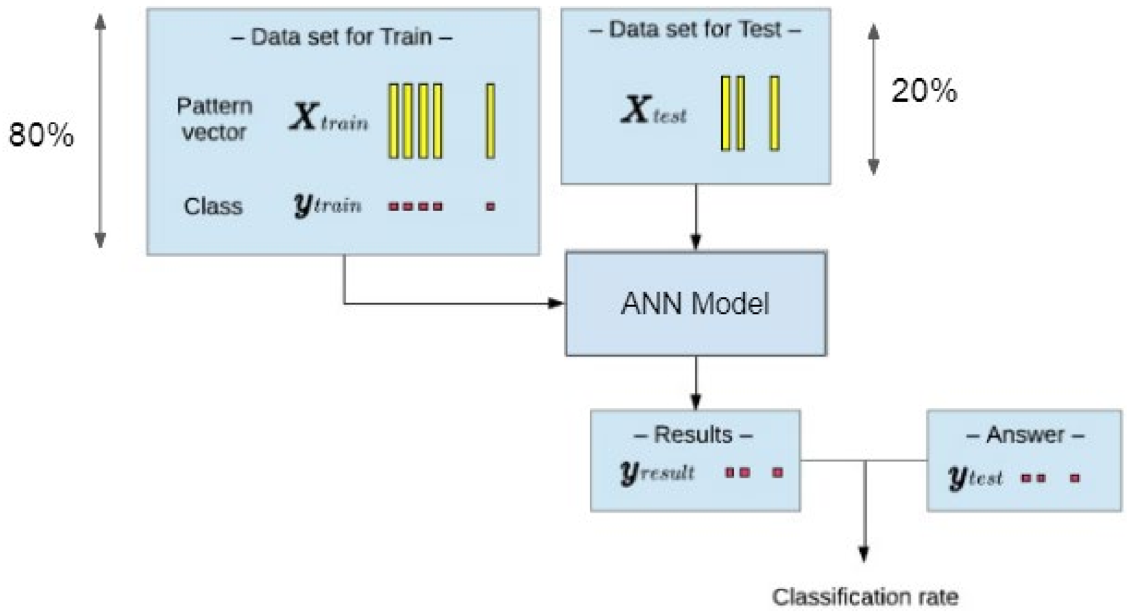


Figure 14: Train test split

4. Robotic ARM Design

4.1. Components to Take into Account

The following considerations were considered when selecting the robotic arm's shape and material:

1. Price

2.The robot's weight

3.The ease with which the pieces can be manufactured

4.Assembly is easy.

5.Parts' sturdiness and longevity

The following are the basic specifications for effective power transmission by the robotic arm:

1.Size is small.

2.Low weight and inertia moment

3.Effective stiffness is high.

4.Transmission ratio that is accurate and consistent

5.Low energy losses and friction for improved control system responsiveness

6.Backlash will be eliminated.

Both of these considerations had a major impact on the robotic arm's design decisions.

4.2. Hardware Components

- 4 DC motors
    - Two motors for controlling wheels and the other two to control robotic arm
- 2 H-bridge
    - To control DC motors
- Pi Camera
    - For object colour detection
- Raspberry Pi 3 Model A+ processor

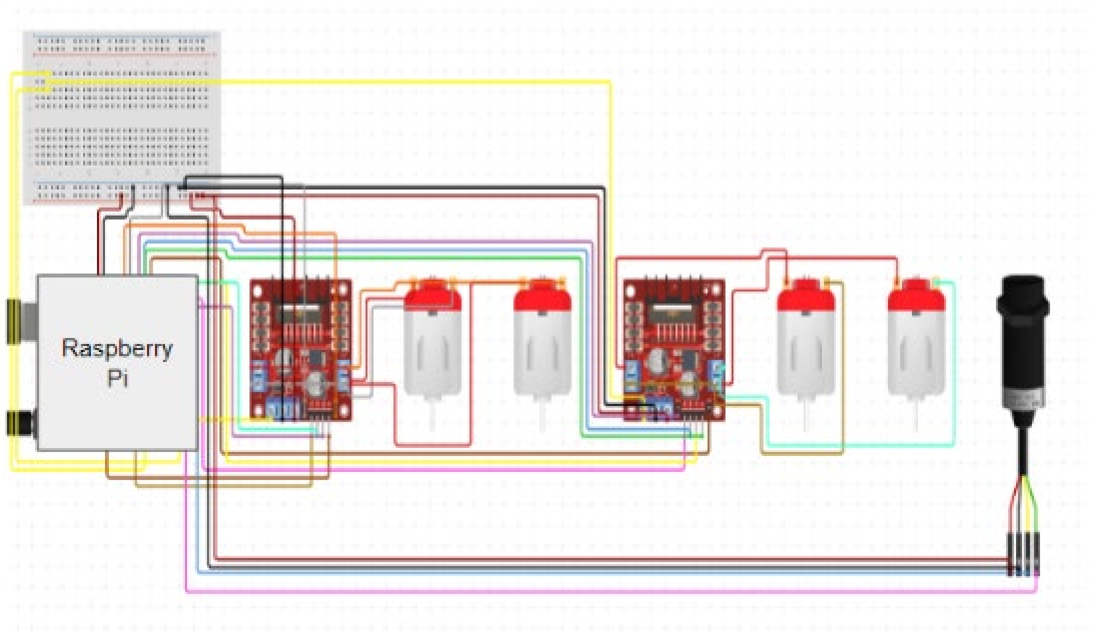


Figure 15: Circuit Diagram

4.3. Constraints

The constraints which are used in our project are as follows :

- Colored objects are considered to be Lightweight objects
- Input given to robotic arm and object color that has to be picked
    - Class 1 - Left - Orange
    - Class 2 - Right - Red
    - Class 3 - Front - Blue
    - Class 4 - Back - Green
- Based on the input Robotic arm rotates and moves in that direction,

  detecting the colored object and avoiding the obstacles encountered in the path

4.4. Computer Vision

4.4.1 RASPBERRY PI SETUP AND CONFIGURATION

- Download the raspberianos and copy the image file onto the SD card using a raspberry imager.
- Run it as CPU connecting it to a monitor keyboard and mouse and complete setup
- Enable VNC and SSH in the pi.
- Download VNC and PuTTY on the laptop to start coding.

4.4.2 TENSORFLOW SETUP AND LOADING THE SAVED MODEL

The model weights were saved and sent to the raspbian OS using SCP(Secure copy).In order to send command files over SSH we used secure copy. This is used to copy files between computers, say from your Raspberry Pi to your desktop or laptop, or vice-versa.

Command to transfer files :

```
scp saved_model pi@172.168.1.3:
```

Then using Tensorflow load module(load_model) we loaded the model on Raspberry Pi and provided the input to the model. The model accordingly classifies the input and provides us the class number.

4.4.3 Object Colour detection

Figure 16: Image showing the output of detected red coloured object

Figure 17: Setup for colour detection

Figure 18 is the process followed for color detection. First we captured the video with the help of a raspberry pi camera and read video streams in image frames. Then we converted the RGB colour space to HSV colour space.We defined the range for each colour and created a corresponding mask. Dilation followed by bitwise AND between mask and image frame is implemented to detect a specific colour. Then we created a contour for each of the four colors to view the detected colored area.

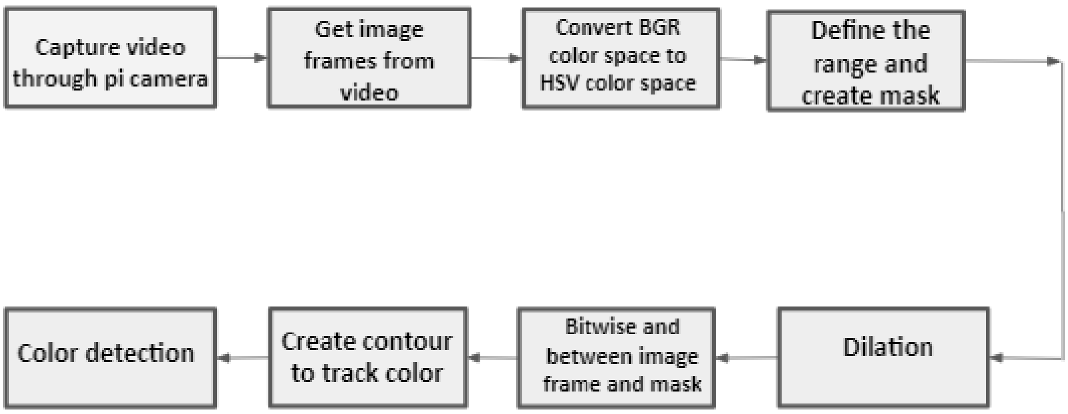


Figure 18: Block diagram for object color detection

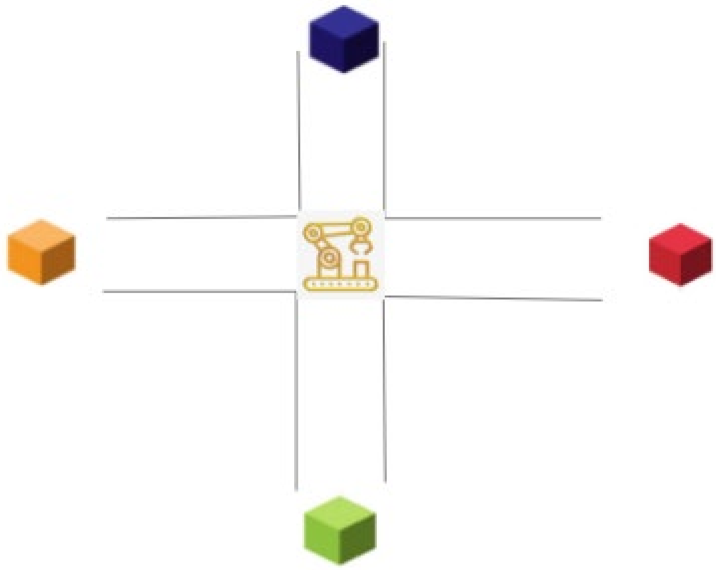


Figure 19: Path to the robotic arm

Figure 19 depicts the path which is followed by the robotic arm. According to the output received by the classified result the bot moves in a particular direction. If the classified class is

- Class 1 : turns towards left direction
- Class 2 : turns towards right direction
- Class 3 : moves forward
- Class 4 : moves backward

Figure 20 displays the classified output and the necessary actions performed according to the classified output.

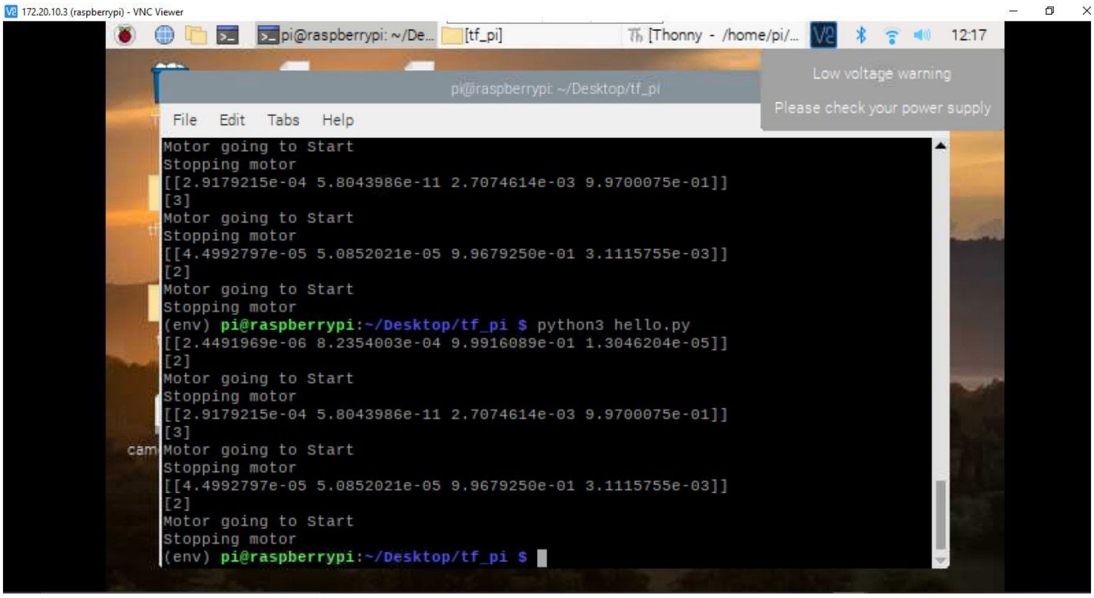


Figure 20: Snapshot of Raspberry Pi terminal

Figure 21 displays the robotic arm which is used in our project. It contains four DC motors, two motors to control forward, backward, right, left movements and the other two are to control the arm in order to facilitate the pick and drop actions. H-bridge is used to control the speed of the motors.

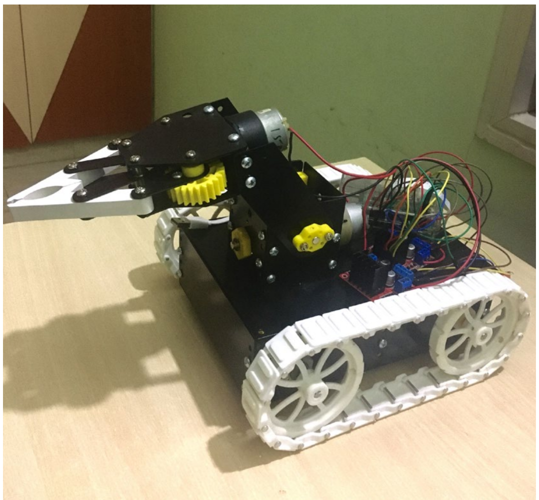


Figure 21: The Robotic Arm used in the project

5. RESULTS AND DISCUSSION

The total accuracy of all four classes (Class 1- Left, Class 2- Right, Class 3- Forward, and Class 4- Backward) is obtained. The 10 Fold Cross Validation approach is used to measure the overall average accuracy of the neural network for the subjects across various features. Table 3 shows the average classification accuracy, fold accuracy, kappa, precision, recall, and fold accuracy for different statistical features derived from 22 channels.

Fold	Accuracy	Kappa	Precision	Recall
F1	0.689655	0.585714	0.692026	0.689655
F2	0.706897	0.609040	0.719330	0.706897
F3	0.741379	0.655446	0.777157	0.741379
F4	0.689655	0.588003	0.721065	0.689655
F5	0.793103	0.724466	0.803831	0.793103
F6	0.586207	0.449802	0.696717	0.586207
F7	0.655172	0.533949	0.676194	0.655172
F8	0.741379	0.648910	0.741323	0.741379
F9	0.689655	0.577670	0.697079	0.689655
F10	0.810345	0.740439	0.819376	0.810345
Avg	0.710345	0.611344	0.734410	0.710345

Table 3: Accuracy of Classification

5.1. Analysis of Result

Initially, only one hidden layer model was used. The average level of accuracy was 67 percent. We experimented with various parameters, such as changing the optimizer used, the number of hidden layers, regularisation techniques, and the dropout value, in order to fine-tune the model. The best accuracy was achieved using a four-layer model with two hidden layers, a drop out of 0.5, rmsprop as the optimizer, and the L2 regularisation technique. We found that adding early stopping to our training process improved accuracy.

The below table depicts the different parameters varied to analyse the variation in accuracy of the model and the corresponding accuracy achieved.

PARAMETERS	ACCURACY
Changing the optimizer (adam)	68.27%
Changing drop out value(0.2)	67.06%
Number of layers	68.79% (Two hidden layers) 67.23% (Single hidden layer)
Early stopping	70.6%
L2 Regularisation	69.65%

Table 4 : Fine tuning the model by varying different parameters

5.2. Conclusion

The arm of Brainiac is unique in that it can be seen in a variety of fields. First and foremost, it can be used for people with disabilities, as well as those who do not have a physical body. They can monitor their missing body parts with the help of their brain . The benefit is that they can develop their mental and attention skills.

We developed our project for robotic arm control. By changing model parameters such as the optimizer, dropout value, and number of layers, as well as adding early stopping to minimise overfitting and L2 regularisation. By comparing the accuracies of all the variants, we can conclude that the Rmsprop optimizer, dropout of 0.5, 2 hidden Layers, early stopping, and L2 regularisation generated the best average accuracy of 71.03 percent. We integrated the classified output with the robotic arm such that it moves in a particular direction corresponding to the input it receives.

5.3. Future scope

The region where the project is to be conducted should have an excellent lighting framework, since there is an object colour recognition included. The items which should be picked by the robotic arm should be light weight. The colors are predefined to detect objects easily. The test environment should not have any obstacles, this can be solved by implementing few sensors to avoid the obstacles. Along with these, real time brain signals from the EEG headset can be used as data to the model to predict the direction of the robotic arm.

REFERENCES

[1] JI-HOON JEONG1, BYEONG-HOO LEE1, DAE-HYEOK LEE1, YONG-DEOK YUN1, AND SEONG-WHAN LEE2,**"EEG Classification of Forearm Movement Imagery Using A Hierarchical Flow Convolutional Neural Network"**, (FELLOW, IEEE)

[2] Ji-Hoon Jeong , Kyung-Hwan Shim , Dong-Joo Kim , and Seong-Whan Lee, **"Brain-Controlled Robotic Arm System Based on Multi-Directional CNN-BiLSTM Network Using EEG Signals"**,IEEE Transactions on Neural Systems and Rehabilitation Engineering (Volume: 28 , Issue: 5, May 2020)

[3] Maryam Mohammadi; Mohammad Reza Mosavi**,"Improving the efficiency of an EEG-based brain computer interface using Filter Bank Common Spatial Pattern",**2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI)

[4] Aldwin Jomar F. Castro , Justine Nicole P. Cruzit **"Development of a Deep Learning-Based Brain-Computer Interface for Visual Imagery Recognition",** 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)

[5] C P Shantala,C R Rashmi,**"Mind controlled wireless robotic arm"**,2017 IEEE International Conference on Computational intelligence and computing research.

[6] Rich Caruana, Steve Lawrence, and Lee Giles," **Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping.**"Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA

[7] Kristin P. Bennett and Emilio Parrado-Hernandez,”**The Interplay of Optimization and Machine Learning Research**”,Journal of Machine Learning Research 7 (2006)

[8] Types of Optimization Algorithms used in Neural Networks and Ways to Optimize Gradient Descent

[9] Diederik P. Kingma, and Jimmy Lei Ba, “**ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION**”,Published as a conference paper at ICLR 2015

[10] L1 and L2 Regularization Methods https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c

[11] Han Y and Bin H 2014 **Brain–Computer Interfaces Using Sensorimotor Rhythms: CurrentState and Future Perspectives**, IEEE Transactions On Biomedical Engineering .61, 1425-1435

[12] Dennis J M, William A S, and Jonathan R W 201**0 Electroencephalographic (EEG) Control Of Three dimensional Movement,** Journal of Neural Engineering .7,175-184

[13] Karl L, Kaitlin C, Alexander D, Kaleb S, Eitan R, and Bin H 2013 **Quadcopter control in three-dimensional space using a non-invasive motor imagery-based brain–computer interface Journal of Neural Engineering** .10, 1-15

[14] Ethan B, Cornelia W, Leonardo G C, Christoph B, Michael A D, Tyler A, Jurgen M, Andrea C,Surjo S, Alissa F, and Niels B 2008 **Think to Move: a Neuromagnetic Brain-Computer Interface (BCI) System for Chronic Stroke,** Journal of Stroke .39, 910–917

[15] José R M, Frédéric R, Josep M, and Wulfram G 2004 **Noninvasive Brain-Actuated Control of a Mobile Robot by Human EEG**,IEEE Transactions On Biomedical Engineering .51,1026-1033

[16] Na L, Tengfei L, Xiaodong R, and Hongyu M 2017 **A Deep Learning Scheme for Motor Imagery Classification based on Restricted Boltzmann Machines**, IEEE Transactions on Neural Systems and Rehabilitation Engineering .25,566-576

[17] Wei H, Yue Z, Haoyue T, Changyin S, and Wei F 2016 **A Wireless BCI and BMI System for Wearable Robots**, IEEE Transactions on Systems, Manand Cybernetics: Systems .46,936-946

[18] Lei Q and Bin H 2005 **A wavelet-based time–frequency analysis approach for classification of motor imagery for brain–computer interface applications**, Journal of Neural Engineering .2,65–72

[19] Caglar U and Turker T E 2017 **Analysis of Time – Frequency EEG Feature Extraction Methods for Mental Task Classification** International Journal of Computational Intelligence Systems.10,31–39

[20] Pawel H, Girijesh P, Thomas M M, and Damien C 2008 **Comparative Analysis of Spectral Approaches to Feature Extraction for EEG-Based Motor Imagery Classification**, IEEE Transactions On Neural Systems And Rehabilitation Engineering .16,317-326

[21] Kavita M, Vargantwar M R, and Sangita M R 2012 **Classification of EEG using PCA, ICA and Neural Network**, International Journal of Computer Applications . 6,1-6.

[22] Eltaf A M, Mohd Z Y, Dalia M, and Aamir M 2016 **Classification of Thoughts into Wheelchair Control Commands using Neural Network** International Journal of Sciences: Basic and Applied Research .25,119-127

[23] Mohammad H A, Emad A A, Aya S , and Khaled A 2014 **Wavelet -Based Feature Extraction for the Analysis of EEG Signals Associated with Imagined Fists and Feet Movements** ,Journal of Computer and Information Science .7,17-27

[24]Ana Riza F. Quiros,Alexander C. Abad,Elmer P. Dadios **Object locator and collector robotic arm using artificial neural networks** 2015 International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM).

[25] Son Ngoc Truong Faculty of Electrical and Electronics Engineering HCMC University of Technology and Education **A Low-cost Artificial Neural Network Model for Raspberry Pi** Engineering, Technology & Applied Science Research

‘BCI For Controlling Robotic Arm’

ORIGINALITY REPORT

8%	5%	2%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

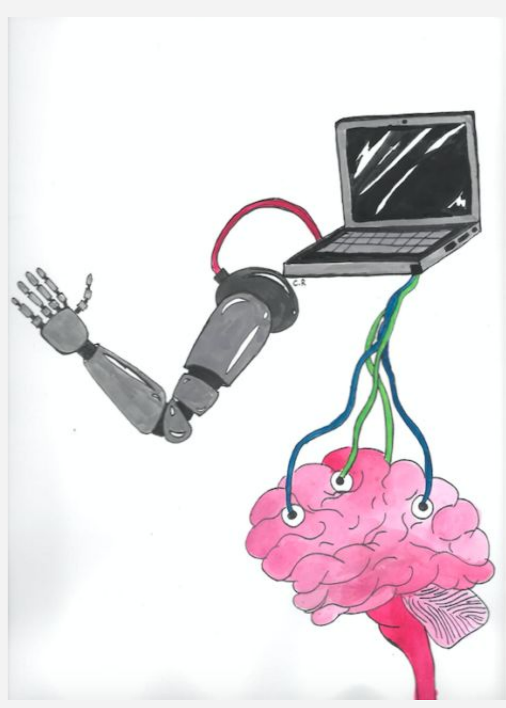
1	Submitted to University of Brighton Student Paper	2%
2	Dino Dominic Ligutan, Levin Jaeron S. Cruz, Michael Carlo D. P. Del Rosario, Jho Nathan Singh Kudhal et al. "Design and implementation of a fuzzy logic-based joint controller on a 6-DOF robot arm with machine vision feedback", 2017 Computing Conference, 2017 Publication	1%
3	www.scribd.com Internet Source	1%
4	Submitted to PES University Student Paper	1%
5	conservancy.umn.edu Internet Source	<1%
6	www.frontiersin.org Internet Source	<1%
7	www.cis.upenn.edu Internet Source	<1%

BCI FOR CONTROLLING ROBOTIC ARM

Shreya V Deexit, V Saisri, Harshita R Vastrad
Project Guide: Dr. B. Niranjana Krupa & Prof. Sweta. G

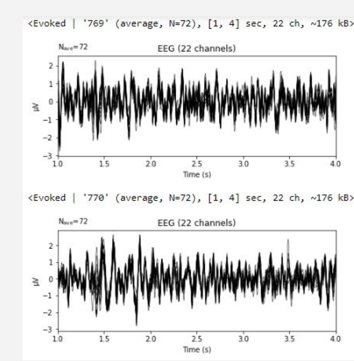
Abstract

Brain Computer Interface (BCI) refers to the interaction of the central nervous system or the brain with a computer where the signals generated by the brain due to external stimulus are used to control an external device. The electroencephalographic (EEG) signal obtained by the imagination of movements of hands and legs is being used in our project. The EEG signal is susceptible to external noise and hence filtering out the perturbations and from the actual EEG data classifying it and building a system is the challenge.



In this project we have done the analysis of the EEG signal to generate control command by getting noise free preprocessed EEG signal, feature extraction and passing it through the ANN model. The control commands are given to the robotic arm which picks and drops the objects using directions and object detection.

Theory



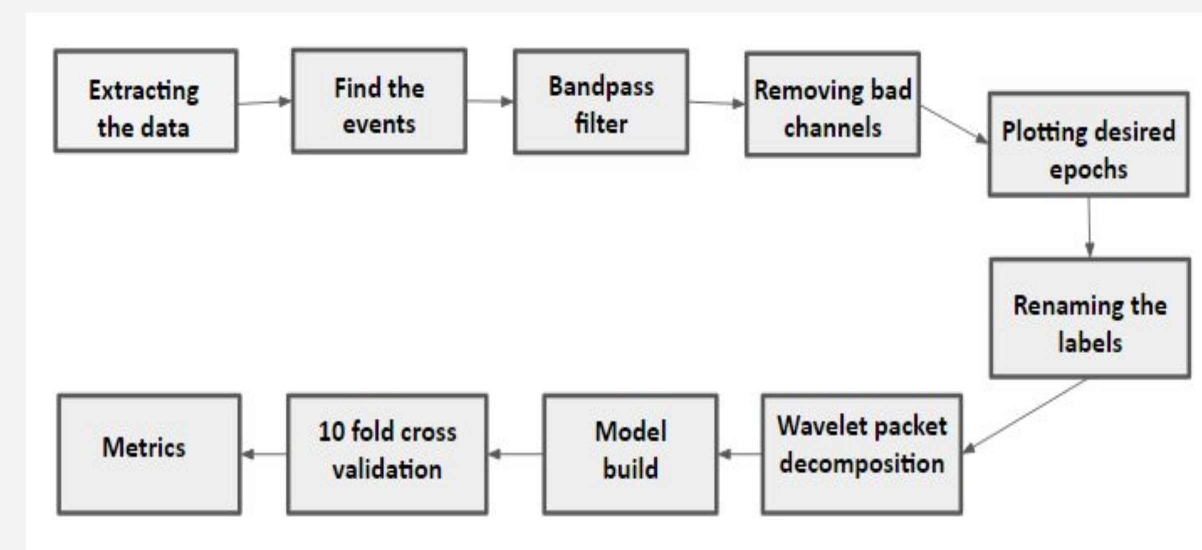
EEG-The electroencephalogram is a scalp recording of the brain's electrical activity.These signals are generally categorized as delta, theta, alpha, beta and gamma based on signal frequencies ranges from 0.1 Hz to more than 100 Hz.

Motor Imagery-One of the standard concepts of BCI is brain computer interface based on motor imagery (MI). In MI, the user can produce induced activity from the motor cortex of the brain by imagining motor movements without any hand movement or external stimulus.

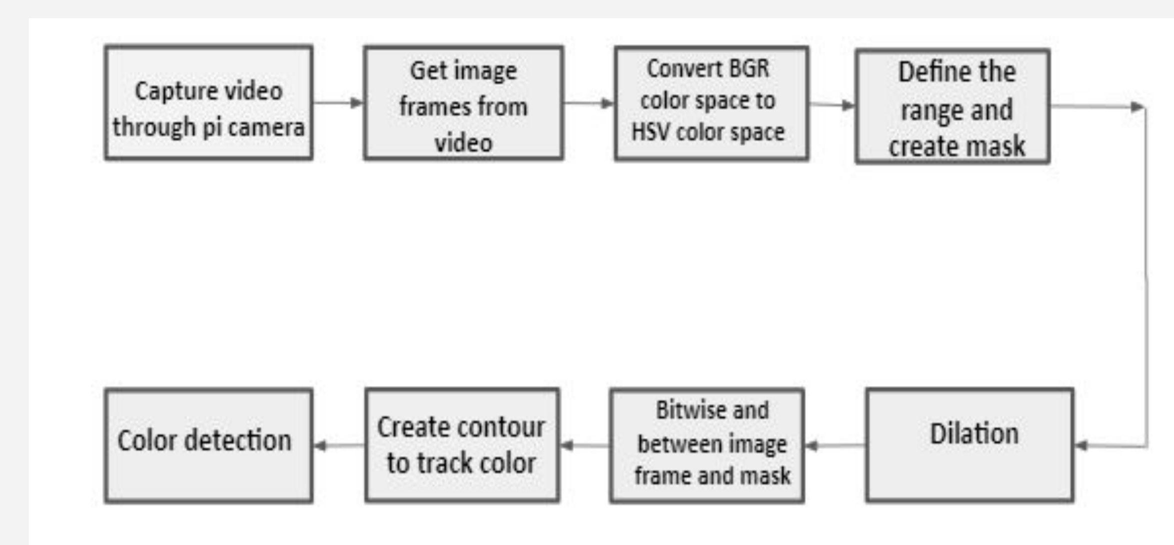
Wavelet Packet Decomposition: The Discrete Wavelet Transform (DWT) is a multi-resolution time-frequency study of signals. DWT is preferred over Fourier Transform because it has frequency resolution as well as temporal resolution information, which is why it is called a time-frequency analysis.

Workflow

1. Analysis of EEG signals to generate control command.

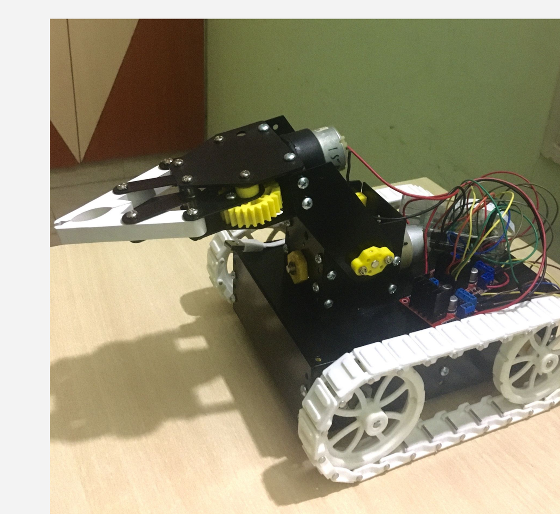


2. Object detection



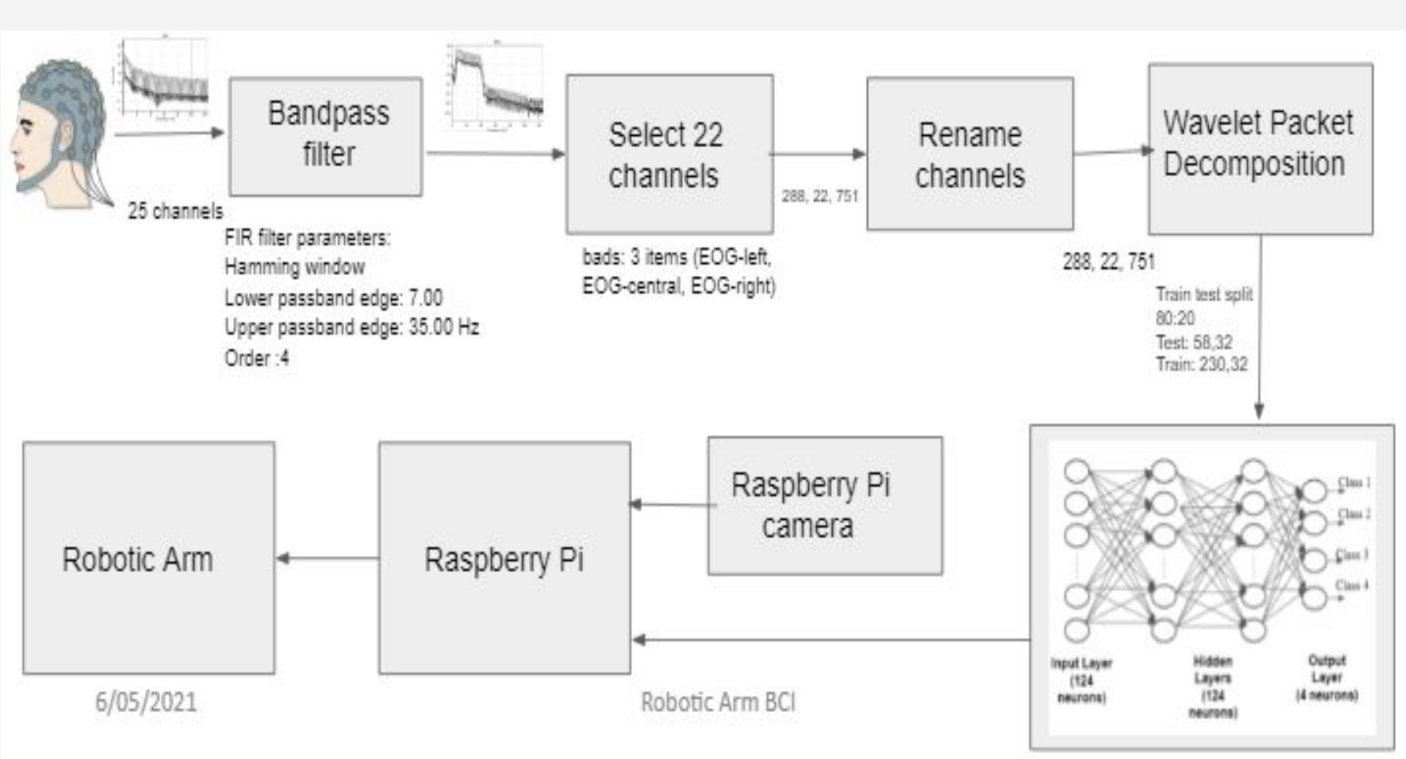
Results and Conclusion

This project can be used for people with disabilities, as well as those who do not have a physical body. They can monitor their missing body parts with the help of their brain . The benefit is that they can develop their mental and attention skills.By changing model parameters such as the optimizer, dropout value, and number of layers, as well as adding early stopping to minimise overfitting and L2 regularisation. By comparing the accuracies of all the variants, we can conclude that the Rmsprop optimizer, dropout of 0.5, 2 hidden Layers, early stopping, and L2 regularisation generated the best average accuracy of 71.03 percent.



This is the robotic Arm used in the Project. Based on th control command received it moves in that particular direction, identifies the object and picks and drops the object

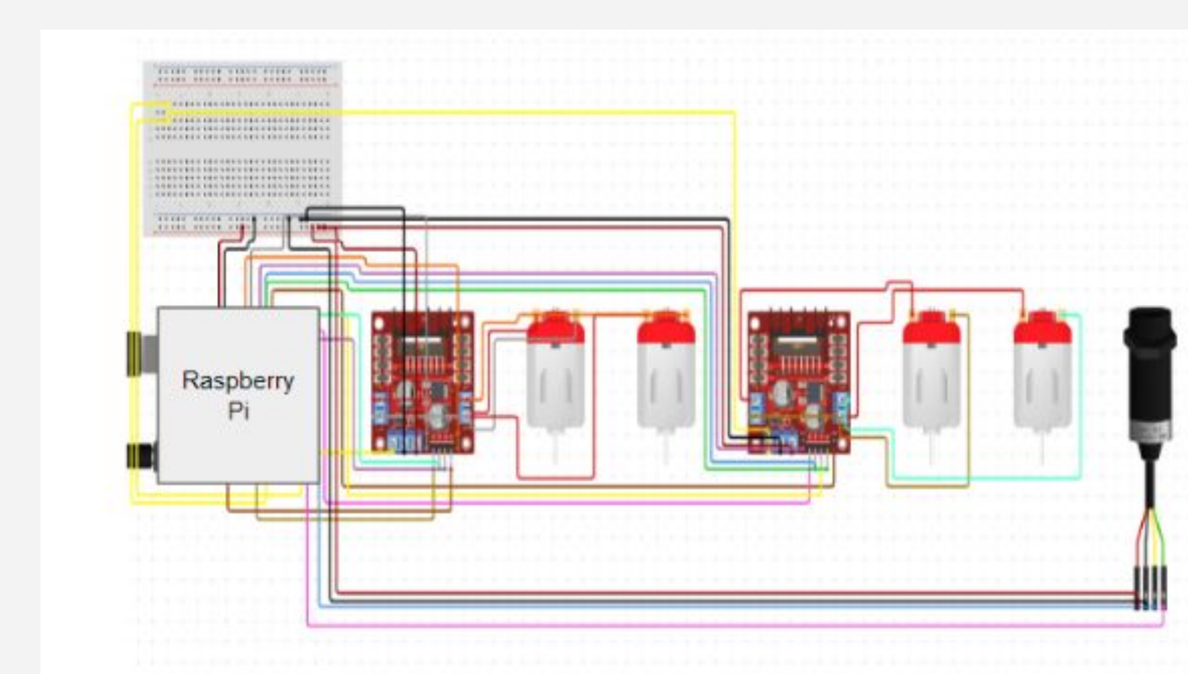
Implementation



Block Diagram

The EEG signal is extracted from the dataset, pre-processing is done by passing the signal through bandpass filter, selecting desirable channels and renaming channels. We split the EEG signal into sub-bands such as Delta (0.5 to 4 Hz), Theta (4 to 8 Hz), Alpha (8 to 16 Hz), Beta (16 to 32 Hz), and Gamma (>32 Hz) to analyse it. On the pre-processed EEG signals, a two-level Discrete Wavelet Transform with 'db4' as mother wavelet is applied, yielding three groups of comprehensive coefficients d3, d4 and d5 from the signal..8 frequency band coefficients are chosen from the range 4-32HzArtificial neural networks architecture is used to classify the extracted features from the EEG signals into four groups. The model is made up of four layers.1 input layer, 2 hidden layers and 1 output layer. Drop out of 0.5, L2 regularizer and rmsprop optimizer is used , 80% of the total data was used for training the model and the remaining 20% was used for testing.. Also the early stopping was implemented such that training loss never goes below validation loss.

The model weights are saved and sent to the raspbian os. We loaded the model on raspberry pi, the model classifies the input given to it and provides us the class number using which raspberry pi controls the direction of robotic arm. To pick and drop the object we are using object detection. Robotic Arm contains four DC motors, two motors to control forward, backward, right, left movements and the other two are to control the arm in order to facilitate the pick and drop actions. H-bridge is used to control the speed of the motors.



Circuit Diagram of the Robotic Arm