#### A

**PROJECT REPORT ON**

**“DEPRESSION ANALYSIS USING CONVOLUTIONARY NURAL NETWORKS”**

SUBMITIED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

### DAHALE AASHISH SHISHIRKANT Exam. No.B150423021

**SHREYA THAKUR Exam. No.B150423097**

**GAUTAM THAKUR Exam. No.B150423023**

**UNDER THE GUIDANCE OF Prof. DR.R.V.BABAR**



#### DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION STE’S

**SINHGAD INSTUTUTE OF TECHNOLOGY, LONAVALA**

**2020-21**



CERTIFICATE

This is to certify that the Project report entitled

## “IOT BASED SMART CHALLAN COLLECTION AND ANTI-THIEFT SECURITY SYSTEM”

### Submitted by

**DAHALE AASHISH SHISHIRKANT Exam. No.B150423021**

**SHREYA THAKUR Exam. No.B150423097**

**GAUTAM THAKUR Exam. No.B150423023**

Is a bonafide work carried out by them under the supervision of Prof. ------ and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University for the award of the Degree of Bachelor of Engineering (Electronics and Telecommunication)

To the best of our knowledge and belief this work has not been submitted elsewhere for the award of any other degree orDiploma.

| **Prof.Dr.R.V.Babar** | **Dr. D.D.Chaudary** | **Dr.M.S.Gaikwad** |
| --- | --- | --- |
| **Guide** | **HOD (E&TC)** | **Principal** |

### Date: / /2020

**Place:Lonavala ExternalExaminer**

**ACKNOWLEDGEMENT**

I express my sense of gratitude towards my project guide **Prof. ------** for his valuable guidance at every step of study of this project, also his contribution for the solution of every problem at each stage.

I am thankful to **Dr.D.D.Chaudary** Head of Department of Electronic and Telecommunication and all the staff members who extended the preparatory steps of this project. I am very much thankful to respected Principal **Dr. M. S. Gaikwad** for his support and providing all facilities to complete thework.

Finally, I want to thank to all of my friends for their support & suggestions. Last but not the least I want to express thanks to my family for giving me support and confidence at each and every stage of this dissertation.

**CONTENTS**

|  | **LIST OF ABBREVIATIONS** | **I** |
| --- | --- | --- |
|  | **LIST OF FIGURES** | **II** |
| **SR. NO.** | **CHAPTER NAME** | **PAGE NO.** |
| **1** | **INTRODUCTION** | **1** |
|  | 1.1 Introduction | 2 |
|  | 1.2 Background | 2 |
|  | 1.3 Relevance | 5 |
|  | 1.4 Problem Definition | 5 |
|  | 1.5 Organization of Seminar Report | 6 |
|  | 1.6 literature Survey | 6 |
| **2** | **PROPOSED METHODOLOGY** | **9** |
|  | 2.1 Block Diagram | 10 |
|  | 2.2 Methodology | 10 |
|  | 2.3 Proposed System | 12 |
| **3** | **HARDWARE AND SOFTWARE DESIGN** | 13 |
|  | 3.1.1 Hardware Requirements | 14 |
|  | 3.1.2 Software Requirements | 20 |

| **5** | **ADVANTAGE** | **23** |
| --- | --- | --- |
|  | 5.1 Advantages of Projects | 24 |
| **6** | **APPLICATIONS** | **25** |
| **7** | **RESULT** | **27** |
| **8** | **CONCLUSION** | **29** |
|  | **REFRANCES** | **31** |

### LIST OF FIGURES

| **FIGURE NO.** | **DESCRIPTION** | **PAGE NO.** |
| --- | --- | --- |
| 2.1.1 | Block Diagram of Depression Analysis using convolutionary nural network | 3 |
| 2.1.3 | Block Diagram of system. | 10 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**CHAPTER 1 INTRODUCTION**

* 1. **Introduction**

Electroencephalogram (EEG) is a popular method for diagnosing various neurological diseases. Major Depressive Disorder (MDD) is a mental health disorder that can be diagnosed and treated by making use of EEG.

G. One of the main challenges in using EEG to accurately identify depression is complexity and variation that exist in the EEG of a depressed person. Manually reading EEG and diagnosing depression is very challenging. An efficient computer aided method can be used for this task. Of the many methods that exists, a deep neural network method called Convolution Neural Networks (CNN) proved to be the most efficient.

In this a multi-layer deep CNN algorithm is implemented to diagnose depression from EEG of patients. Depression is classified based on a severity index into mild, moderate and major classes. The accuracy, sensitivity and specificity were measured by varying various parameters of the proposed algorithm.

## Background

The design and implementation stage of the project, involved the raspberry pi board, PC , power supply, EEG report of patient.

Depression is a global public health problem, which has a relatively high lifetime prevalence, ranging from 2 to 15%, and is associated with significant morbidity. According to the latest data from the World Health Organization (2017)[1](https://www.frontiersin.org/articles/10.3389/fnins.2020.00192/full#footnote1), more than 300 million people are now living with depression. Presently, the most widely used methods for depression diagnosis are based on Beck’s Depression Inventory (BDI), the patient’s self-report, the doctor’s clinical experience, or some combination thereof. However, the accuracy of this diagnosis is often influenced by the doctor’s proficiency and patient’s cooperation, both of which are highly subjective. Critically, a subset of depression–mild depression–receives far less attention than does depression, despite being more common than depression and often increasing in severity over time. This lack of attention leads to missed early detection and treatment and increases the mortality risk and likelihood that mild depression will evolve into major depression. Additionally, mild depression is not only a mental illness but also often a social problem. Therefore, studies of methods that might improve the early detection and treatment of mild depression are both necessary and meaningful

## Relevance

It is difficult to achicheve result of depression level manually by doctors.

it is time consuming and require doctors attaintion for long time

## ProblemDefinition

To help doctors in getting correct diagnosis depression from EEG of patients Easily.

Reduce time of diagnosis

## Organization of ProjectReport

The organization of this report is a follow:

Chapter 1 introduces the topic and provides background survey

Chapter 2 contains the proposed methodology (block diagram) and the features incorporated in the project.

Chapter 3 includes the hardware and software requirements along with explanation of hardware and software

Chapter 4 includes the design of project. Chapter 5 includes theadvantages.

Chapter 6 includes the application, result Chapter 7 includes theconclusion.

References and Datasheets are included at the end in Appendix.

## Literature Survey

## Sudarshan et al. explored several nonlinear approaches for EEG centered detection of depression

## Detrended Fluctuation Analysis (DFA), Fractal Dimension (FD), Higher Order Spectra (HOS),Largest Lyapunov Exponent (LLE), Recurrence Quantification Analysis (RQA), Hurst’s Exponent (H) and Sample Entropy (SampEnt)

## The obtained features are graded and organized in accordance with their importance by means of its t-value and was put into 5 distinct classifiers. It is reported that Support Vector Machine (SVM) provides top diagnostic assessment with a normal accuracy of around 98%.

## The obtained features are graded and organized in accordance with their importance by means of its t-value and was put into 5 distinct classifiers.

## It was suggested to use Depression Diagnosis Index (DDI) which utilizes a mixture of the obtained non-linear features

# CHAPTER 2 PROPOSED METHODOLOGY

## BlockDiagram

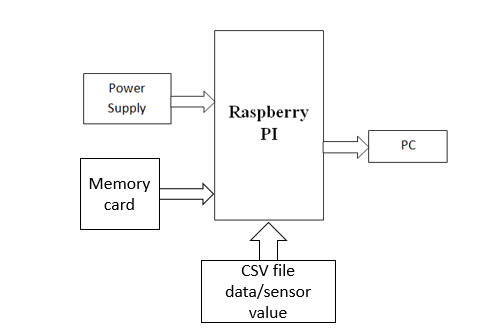
****

Fig. 2.1 Block Diagram of Depression Analysis using convolutionary nural network

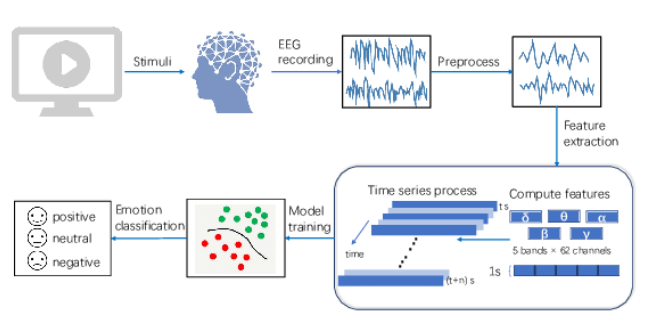
## Methodology

* Whenever Deep learning is a machine learning method that enables computers to learn by example just like how humans teach themselves.

## Most of the EEG software was found to incorporate some form of abnormality detection algorithm.

## A multi-layer deep convolution neural network (CNN) is designed and developed to classify depression based on severity into three classes: mild, moderate and major.

## PROPOSED SYSTEM Flow



# CHAPTER 3

**HARDWARE AND SOFTWARE DESIGN**

## Hardware & Software Requirements

* + 1. **Hardware Requirements:**

3.1.1(a) Raspberry pi

## Software Requirements:

3.1.2(a) Python 3.8

### 3.1.1(a) Raspberry pi

Raspberry Pi is the name of a series of single-board computers made by the Raspberry Pi Foundation.

The Raspberry Pi launched in 2012, and there have been several iterations and variations released since then. The Raspberry Pi has a single-core 700MHz CPU and just 256MB RAMA, and the latest model has a quad-core 1.4GHz CPU with 1GB RAM.

All over the world, people use Raspberry Pi to learn programming skills, build hardware projects, do home automation, and even use them in industrial applications.

The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins that allow you to control electronic components for physical computing and explore the Internet of Things (IoT).

## Software Requirements:

### 3.1.2(a) Python

Python is a powerful programming language that's easy to use (easy to read and write) and easy to connect project to the real world. Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).

Python has a simple syntax similar to the English language. Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

Python runs on an interpreter system, meaning that code can be executed as soon as it is written.

This means that prototyping can be very quick.

Python can be treated in a procedural way, an object-oriented way or a functional way.

# CHAPTER 5 ADVANTAGE

### ADVANTAGES

* 1. More reliable
  2. No need of doctors presence
  3. Qwick diagnosis
  4. Cost efficient

# CHAPTER 6 APPLICATION & RESULT

### Application

* + Increase accuracy above 90%
  + cost reduction and reliable
  + Gives output immediately saves doctors efforts

### Result C:\Users\om\Downloads\WhatsApp Image 2021-06-20 at 10.47.39 AM.jpeg

# CHAPTER 7 CONCLUSION

### Conclusion

These system will improve the road safety as well as comfortable vehicle driving. By this technology people drive will be more comfortable and easier. Any problem occurred in our vehicle then our system will be informed to others so they can be aware of it while driving. It is efficient design for anti-collision of vehicle.

# Chapter 8

# REFERENCES

### References

1 S. C. Liao, C. T. Wu, H. C. Huang, W. T. Cheng, “Major depression detection from EEG signals using kernel eigen-filter-bank common spatial patterns,” Sensors 17 (6) (2017) 1385.

1. S. Ghosh-Dastidar, H. Adeli, N. Dadmehr, “Principal component analysis – enhanced cosine radial basis function neural network for robust epilepsy and seizure detection,” IEEE Transactions on Biomedical Engineering 55 (2) (2008) 512-518.
2. L. Khedher, L. A. Illan, J. M. Gorriz, J. Ramirez, A. Brahim, A. Meyer-Baese, “Independent component analysis – support vector machine-based computer aided diagnosis system for Alzheimer’s with visual support,” International Journal of Neural Systems 27 (3) (2017) 1650050 (8 pages).
3. S. L. Oh, Y. Hagiwara, G. M. Bairy, S. D. Puthankattil, O. Faust, U. C. Niranjan, U. R. Acharya, “Automated diagnosis of depression electroencephalograph signals using linear prediction coding and higher order spectra features,” Journal of Medical Imaging and Health Informatics 7 (8) (2017) 1857-1862.

# Chapter 9

# CODE

**TEST CODE-**

import numpy

from sklearn.svm import SVC

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import accuracy\_score

from sklearn.ensemble import RandomForestClassifier

import joblib

import tkinter

from tkinter import filedialog

import os

#from pygame import mixer # Load the popular external library

import time

#mixer.init()

root = tkinter.Tk()

root.withdraw()

while True:

currdir = os.getcwd()

file = filedialog.askopenfilename()

a = numpy.genfromtxt(file, delimiter=' ')

a=a.reshape(1,-1)

clf=joblib.load('valance.pkl')

val = int(clf.predict(a))

clf=joblib.load('Arousal.pkl')

aro = int(clf.predict(a))

if val==0 and aro==0:

print('Sad')

if val==0 and aro==1:

print('Depressed')

if val==1 and aro==0:

print('Relaxed')

if val==1 and aro==1:

print('Happy')

**SVM CLASSifier-**

import numpy

from sklearn.svm import SVC

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import accuracy\_score

from sklearn.ensemble import RandomForestClassifier

import joblib

def svm\_classifier():

file\_x = 'data/features\_sampled.dat'

file\_y = 'data/label\_class\_0.dat'

X = numpy.genfromtxt(file\_x, delimiter=' ')

y = numpy.genfromtxt(file\_y, delimiter=' ')

# Split the data into training/testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.1, random\_state=100)

# Feature Scaling

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X\_train = sc.fit\_transform(X)

X\_test = sc.transform(X)

for i in range(0,len(X\_train)):

v=str(i)

fn='Test data/'+v+'.txt'

#print(X\_train[i])

c = numpy.savetxt(fn, X\_train[i], delimiter =', ')

# SVM Classifier

clf = SVC()

#clf = RandomForestClassifier(n\_estimators=10)

print(X\_train)

clf.fit(X\_train, y)

y\_predict = clf.predict(X\_train)

#print(y\_predict)

joblib.dump(clf, 'valance.pkl')

cm = confusion\_matrix(y, y\_predict)

print("Accuracy score of Valence ")

print(accuracy\_score(y, y\_predict)\*100)

###############################################################

file\_x = 'data/features\_sampled.dat'

file\_y = 'data/label\_class\_1.dat'

X = numpy.genfromtxt(file\_x, delimiter=' ')

y = numpy.genfromtxt(file\_y, delimiter=' ')

# Split the data into training/testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.1, random\_state=42)

# Feature Scaling

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X\_train = sc.fit\_transform(X)

X\_test = sc.transform(X)

# SVM Classifier

clf = SVC()

clf.fit(X\_train, y)

y\_predict = clf.predict(X\_test)

cm = confusion\_matrix(y, y\_predict)

joblib.dump(clf, 'Arousal.pkl')

print(cm)

print("Accuracy score of Arousal ")

print(accuracy\_score(y, y\_predict)\*100)

if \_\_name\_\_ == '\_\_main\_\_':

svm\_classifier()