Accurate Analysis of Mood Detection using Eye-Images rather than Facial Expression Recognition (FER)

by

Arafat Hossain

18101023

Akash Chakraborty

18101019

Syeda Rifa Syara

18101162

Saadman Rahman

18101605

Fahad Muntasir Tanmoy

18101325

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Supervisor: Tanvir Rahman
Co-Supervisor: Arif Shakil
Department of Computer Science and Engineering
Brac University
June 2022

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Arafat

Arafat Hossain
18101023

Aborty

Akash Chakraborty
18101019

Rifa

Syeda Rifa Syara
18101162

Saadman Rahman
18101605

Fahad

Fahad Muntasir Tanmoy
18101325

Approval

The thesis/project titled “Accurate Analysis of Mood Detection using Eye-Images rather than Facial Expression Recognition” submitted by

1. Arafat Hossain (18101023)
2. Akash Chakraborty (18101019)
3. Syeda Rifa Syara (18101162)
4. Saadman Rahman (18101605)
5. Fahad Muntasir Tanmoy (18101325)

Of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May,2022.

Examining Committee:
Supervisor:
(Member)

Tanvir Rahman

Tanvir Rahman
Lecturer
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD

Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

1) This material is our own unique work, which has not been recently published somewhere else.
2) The paper is not lately being considered for publication somewhere else.
3) The paper reflects our own exploration and examination in an honest and complete way.

Abstract

There are several works on mood detection by machine learning from physical and neuro- physical data of people, along with works on emotion recognition using eye-tracking. We want to show that a person's mood can be detected using their eye images only. The mood is reflected through one's eyes. The goal is to establish a connection between an individual's mood and one's eye images. The machine learning algorithm that we are going to use is Convolutional Neural Network (CNN) because it does not require external feature extraction. They system learns to extract the features by itself. In this paper, we developed two CNN models and used FER-2013 as our dataset from which we used only the eye images for each of the six emotions: happy, fear, sad, angry, neutral and surprise to create our own dataset. We trained and tested our models with both FER-2013 dataset as well as our own dataset and compared the results. For FER-2013 dataset, our final accuracy score for model 1 was 83.78% with a validation accuracy score of 65.35%. It was seen that our model 1 showed the final accuracy score of 69.19% with a validation accuracy of 72.08% whereas for model 2, the final accuracy was 66.55% with a validation accuracy of 72.36% when trained and tested with our own dataset. The low accuracy for our dataset is due to the limitations that we faced for insufficient training and testing images. The accuracy can be improved with a better dataset for training our models.

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.

Secondly, to our supervisor Tanvir Rahman sir for his kind support and advice in our work. He helped us whenever we needed help.

And finally to our parents without their throughout support it may have not been possible. With their kind support and prayer we are now on the verge of our graduation.

Table of Contents

Chapter 1

Introduction

Facial expressions (FR) can be used to determine a person's mood. This innate inclination is an important quality that humans can benefit from because while making any decisions in the actual world, a person's mood is kept in mind. According to scientists, 55% of communications are communicated solely through facial expressions [7] , while the rest can be delivered through language and voice. Recognizing and considering moods of humans while making decisions will make human-computer interactions more realistic [12].

Eye-tracking can be applied in a variety of fields, including psychology, engineering, advertising, and computer science [4] . Eye-tracking techniques, for example, have been used to identify drowsiness of drivers [6] . Furthermore, eye-tracking and pupil monitoring could be used to determine a learner's cognitive load [1] . Furthermore, some research has been done in eye responses to emotional stimuli [3].

The identification and categorization of distinct moods or states of mind is part of the analysis of moods from human eye expression. For example, the behavior of an offender or criminal can be forecasted by studying photographs of their faces from video captured through surveillance cameras. Mood analysis can be used in a range of applications, including video surveillance and also HCI systems [11]. Moreover, understanding the mood of a student can lead to more effective presentation style and better learning in the educational profession [5]. This type of automatic emotion identification could potentially be valuable in psychological investigations [9] . Such research, for example, might provide a baseline for healthy peoples' emotional reactions which could then be contrasted and utilized to diagnose mental diseases like autism [2] or depression [8].

1.1 Research Problem

In today's world where AIs are ever growing smarter and are able to deal with more complex tasks. We have from self-driving cars to AIs that can accurately predict the weather, do floor cleaning and perform calculations faster than any human can ever think of. AI companions are a thing now that have been gaining more and more attention and services like Amazon's Alexa are very common nowadays. AI companions like Alexa or Apple's Siri function as virtual assistants. While Alexa and Siri perform tasks that most people would want, there are smarter AI companions that not only can search things up on the internet and change the music, but also react to certain actions performed by the user. One thing they are lacking is the ability to tell what a person is feeling just by looking at their faces. Facial expression recognition is such a basic human characteristic that even a baby can grasp it just by looking at someone, they do not need to be taught. Regardless of that, Facial Recognition technology has come a long way since then. While they certainly have improved over the years, there is still room for improvement left. As of now most of the FR (Facial Recognition) technology needs to see the motion of the full face to determine the person's mood. This may give inaccurate results in some cases where the person maybe trying to hide their true emotions. Most of us try not to appear weak and vulnerable in front of others, in which case one might let out a sad but generous smile, or when one hiding something, they might appear to be quite cheerful and it may be obvious to us humans what the other person is really, but a machine might see a smile and predict the person is happy, since they are smiling. To tackle this sort of inaccuracy we propose a model which uses eye - images to determine the mood of a person. A person may smile while telling a lie, the most common thing to do in a situation like this is looking away from the person while lying. While it may be obvious for a human being to tell, a machine cannot do that. In order to overcome this exact problem, our model tries to predict a person's mood based on their eye movement. This also has effects on security like CCTV footages, if we can predict a person's action accurately beforehand, crimes such as theft and robbery can be prevented and especially at times like these where wearing masks are mandatory and people are meant to hide their faces, we believe our Mood Detection model using eye images will prove to be most useful. While we are aware there are countless Facial Recognition models out there, we aim to predict a person's mood more accurately by taking the images of their eyes.

1.2 Research objective

FER technology has been around for a very long time. But they rely heavily on analyzing the whole face and facial features. However, in today's world where we are recommended to use mask for our own safety, the problem might arise where the traditional FER technology might be unable to accurately detect faces.
Hence our project focuses only on eye tracking to detect a person's mood. We focus on:

1. Establishing connection between mood detection and eye images.
2. Constructing a model that provides the highest accuracy possible.
3. TAccurately detect a person's mood by using images of their eyes.
4. Making mood detection easier in today's scenario where everyone wears mask.
5. Providing reliable results in detecting moods.

Chapter 2

Literature Review

In the paper [17], the researchers have constructed a “model for emotional analysis” of the people in study through gathering various key characteristics from sensors. Sensors on mobile devices and user keyboard habits have been used for collecting information, grouped as physical and neuro-physical. The data collection procedure took a lengthy time based on the fact that the users were taught prior to the data collection stage. The system has been given “sleep quality, energy, mobility/movement and heart pulse” as “physical parameters” whereas “keystroke pattern-based movements” are given as “neuro-physical parameters” [17] . Traditional machine learning techniques and deep learning approaches were used to classify the users’ emotion/mood. Deep Learning approaches do not require any “pre-processing steps”, which are required by traditional machine learning algorithms. They have used “Feedforward Neural Network (FFNN), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Long Short-term Memory neural network (LSTM)” as “Deep Learning methodologies” and “Multinomial Naïve Bayes (MNB), Support Vector Regression (SVR), Decision Tree (DT), Random Forest (RF) and Decision Integration Strategy (DIS)” as “conventional Machine Learning algorithms” for predicting the emotions of users [17] . Their findings of the experiments show that using Deep Learning techniques and combining physical and neuro-physical factors improve the system’s classification success in predicting the mood or emotional state of users. CNN, in particular, has a high level of categorization success. When users’ emotions change, their bodies react to the changing situation, which can be measured using physical/neuro-physical parameters. The link between physical and neuro-physical parameters is also confirmed by experimental results [17] .

This study [12] aims to provide a simple and novel way for using pictures of faces for assessing a psychological patient’s mood. The authors have created a system that implements a number of machine learning approaches to identify the mood of the psychologically or mentally disturbed patients under observation. The main issue can be broken down into three sections. The first phase includes classification of a person’s mood intro five categories which are sad, angry, happy, normal and surprised. For training the K nearest neighbor (KNN) classifier, they have used a collection of input photos for each class. They applied Speeded-up robust features (SURF) for detecting local features and descriptors from the input sets of images which are then used for training KNN classifier. The second phase includes extraction of SURF features from the test image. The descriptors and features retrieved from the test image are provided to the trained KNN classifier in the third phase, which classifies the image into one of the five classes. After conducting 250 experiments for each of the five classes, their classifier showed 77.4% of overall accuracy.

The researchers intended to test if “eye movements”, along with the size of the pupil and “invisibility to the eye-tracker” might assist them discern emotional states in their article [9] . In order to create a framework for bringing out positive and negative emotions, the researchers showed video clips to the people at study for bringing out emotions and also conducting interviews for spontaneous emotions [9] . Gender heterogeneity were kept limited when the eye movement data were gathered from the participants. Several low-features such as “distance between eye gaze points position”, “distance between left eye and eye tracker”, “distance between right eye and eye tracker” and normalized pupil size for each eye were retrieved and statistical functions were applied to them. The authors analyzed statistical functional features such as “mean, standard deviation, variance, maximum, minimum and range for the low-level features, fast and slow changes of eye gaze for each eye, left and right head rotation, large and small pupil size and the absence of left, right and both eyes” using both statistical and artificial approach classifications [9] . They generated a “Gaussian Mixture Model (GMM)” consisting of 8 mixtures for each segment of each participant for the low-level features. The GMM models were trained by HMM that was applied through the use of Hidden Markov Model Toolkit (HTK). The statistical features of the 8 mixtures for each subject, used as supervector, were given to Support Vector Machine (SVM) classifier. SVM was utilized to identify the categories for all subjects in a “binary subject-independent situation” because of its high generalization properties. To limit the influence of a small quantity of data, the authors used a “leave-one-subject-out cross-validation” method with no overlapping across testing data and training data [9] . Appropriate classification of the subjects as “positive” or “negative” depending on the patterns of “eye movements” was their major goal. Overall, their experiment resulted in an average of 66%, demonstrating that eye activities such as “eye movements”, “pupil dilation” and “invisibility to the eye tracker” contain effective clues for distinguishing and also recognizing emotional states. In contrast to provoked emotions, sudden emotions do have greater ocular activity patterns, according to their findings.

The authors in this paper [11], have focused on recognizing emotions with Hidden Markov Model (HMM) using the “distance calculation approach”, which involves calculating distance between the sclera and the iris. Algorithms such as “face detection”, “feature extraction”, “distance calculation” and “emotion classification” have been built for recognizing emotion using eye tracking. They have used non-intrusive webcam for capturing images and an image segmentation method to segment the eye components for emotion analysis. HMM, with the distance calculation approach, has been used to classify emotions into six different categories including anger, happy, fear, disgust, sleep and lateral movement of thinking, resulting with high correlation coefficient and 77% accuracy. Factors such as camera calibration and head orientation were not taken into consideration, which would otherwise improve the accuracy of the results.

Here [10], the authors demonstrated a method for recognizing emotions based on pupil information such as “pupil size” and “gaze position” which were observed while browsing images. The researchers have used visual stimuli in the form of colored images to elicit emotions and have investigated two types of Neural Network-based learning tools (NNs). The emotions are assessed based on the images used as stimulus and are categorized into three different forms: negative, neutral or positive. The authors have chosen Neural Networks for training their model into learning the pattern’s geometric qualities, alongside their serial order. The most typical strategy when working with NNs is to use input neurons to indicate the sequential order of the input patterns. When the image is negative or in other situations, they constructed a mechanism for making binary decisions. They then trained a new learning tool to exclusively classify positive and neutral images. Therefore, they constructed a decision tree (DT) where decisions are taken by NN machines. Their model achieved an accuracy of 71.7% when the whole dataset was considered. Their paper displayed findings that a learning machine can estimate the subject’s moods (with a good degree of accuracy) when visual colors are shown to them. Their method also has the advantage of requiring just two measurements (“gaze position” and “pupil size”), which are cheap, easy and less intrusive to record and interpret than other signals like “MEG” or “EEG”, which are commonly used in investigations on emotions [10].

In this study [20], the goal of the authors was to show that emotions can be discerned using “eye tracking”. For collecting data there were 30 people whose emotions were recorded during various types of videos presentation along with soundtrack. The researchers calculated “pupil diameter” along with specific characteristics of eye movement such as “fixations” and “saccades”. They classified the emotions using SVM, LDA, and KNN. The oculograhic data was recorded using an “EyeTribe visual eye-tracker”. EyeTribe determines the user’s gaze point’s (X, Y) coordinates in relation to the screen they are looking at. Pixels are used to express the coordinates. Three pairs of feelings were classified, as well as all three classes of emotions together [20]. The highest accuracies were 80%. That has been obtained by SVM classifier. One of the most important things was the lighting condition throughout the experiment. Using LDA, they were able to recognize the emotions generated by movies with comparable dynamics with a classification accuracy of 78 percent [20]. This research implies the method of emotion recognition through eye-tracking can be successfully practiced.

When the system is trained with one dataset and tested with another dataset, the classification accuracy plummets, according to this paper [13].However, a broad and dependable network can still be achieved through a way. In all DNN-based facial expression research, the recommended network is trained and customized for a given dataset, and from that very dataset, the test dataset is also obtained. Due to network biasedness for a particular data type, the outcome of the dataset’s test data will have shockingly low error rate. However, if the system is trained with one particular dataset and tested with a different one, the results will be opposite. This ends up in the classifiers being useless in natural setting. For databases, the researchers have used “Radboud Faces Database (RaFD)”, “Cohn-Kanade AU-Coded Facial Expression Database Version 2 (CK+)” and the “Japanese Female Facial Expression (JAFFE) Database” [13]. There are three networks from which network 1 is just trained with RaFD database, network 2 is trained with CK+ database and a combination of the three datasets is used to train network 3. It is critical to develop algorithms that are resistant to changes in the environment and conditions when implementing a solution in consumer devices. One of the techniques presented in this paper for obtaining a more general and reliable DNN is to mix as much feasible data. Incorporating a diverse set of samples from various situations and attributes into DNN training sets would result in a more reliable system for unpredictable scenarios, proved to be the most significant factor in consumer electronics. The test error for each database in network 3 is low in comparison with the test error from the system or network trained by that database, is a finding that proves their statement.

This study [18] puts forth a survey conducted on “detecting emotions through tracking eyes”, focusing on some specific features of the “eye-tracking” data which are related to emotions. Several features of the emotion categorization task are resented, including the emotional response utilized in this research, the number of participants, the emotions identified and categorized, the features and classifiers used, and the prediction rates. Despite the fact that ‘eye-tracking’ is gradually developing into a common sensor method for human-computer interaction, it is still considered a unique method for recognizing emotions [18]. Emotion detection has been used in a variety of applications, including safe driving, mental health monitoring, and social security. Human-Computer Interaction (HCI) has grown in importance as a subject of study in computer science. Understanding, identifying, analyzing and reacting to a human’s mood: all require HCI. The study of Fischer et al. and Cowie et al. in HCI focuses on “user modeling and emotion recognition” [18]. Affective “computer systems” are those that are capable of detecting human emotion. Affective computing is a branch of research that brings together “computer science”, “psychology”, “cognitive science”, and “artificial intelligence” to develop systems that can recognize, read, store, and respond to human emotions. An emotion identification system will aid in the detection of human emotions using data from various sensors such as “eye-tracking” and “electroencephalography (EEG)” data, among others. Physiological signals such as “EEG brainwave signals”, “pupil responses”, “electrooculography (EOG)”, “electrocardiogram (ECG)”, “electromyogram (EMG)”, and “galvanic skin” reaction have recently been reported in a number of articles (GSR). Shu et al. evaluated the impact of studies on physiological signals and emotion perception. In studies that directly employed “eye-tracking’ techniques for the purpose of identifying emotions, Artificial Neural Network was shown to be 90% effective as a classifier utilizing “pupil size”, “pupil position” and speed of the movement of the eyes as parameters. Only pupil diameter was used in the least successful approaches, resulting in relatively identical and poor precisions of 58.9% and 59.0%, respectively [18]. The most commonly used criterion was pupil diameter followed by fixation length and finally pupil position and EOG.

In the paper “Multimodal Emotion Recognition Using Deep Neural Networks” researchers [14] tried to prove that it is possible to recognize “Multimodal emotion” by using “Deep Neural Networks”. They proposed a novel model called Bimodal-LSTM, which uses temporary features to recognize emotions in multimodal inputs. SEED is a public dataset that uses “EEG” and “eye movement” parameters as inputs. Other “state-of-the-art” approaches are surpassed by the novel Bimodal-

LSTM model. They claim that various modalities, including “facial expression”, “speech”, “electroencephalography (EEG)”, “electrocardiogram (ECG)”, “pupillary diameter (PD)” and others, convey emotional information. For emotion recognition, they implemented SEED and DEAP datasets and achieved exemplary results. We mostly noticed two models for predicting emotions in this research: first, the BD-DAE model which is a noise-removal auto encoder extension; second, the Bimodal-LSTM model [14]. The latter model achieved excellent accuracy of 93.97% on SEED dataset and accuracy of 83.23% and 83.83% respectively using DEAP dataset.

Chapter 3

Working Plan

The purpose of this proposed paper is to establish a connection between mood detection and eye images. We want to show that people's moods can be detected by only using the images of their eyes. We have created a dataset that contain around 3000 images of eyes that we will utilize to train our system to recognize moods. We will apply the datasets as input and follow the steps outlined in the figure in order to conduct accurate research.

1. We separate the datasets into categories based on a variety of factors. The pictures we gathered will be classified into different moods such as; angry, fear, happy, neutral, sad, and surprise.

2.We will use CNN algorithm for training the system in classifying eye images. This is shown in Figure 3.1.

3. After completing the training stage of the classifier, we will provide test images to it.

4. The CNN classifier will then classify the image into one of the six categories. This is shown in Figure 3.2

We chose Convolutional Neural Network (CNN) because it provides more accurate results.

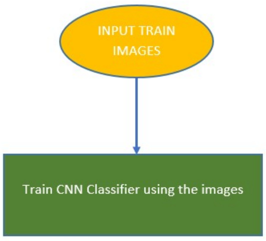


Figure 3.1: Training classifier using training images

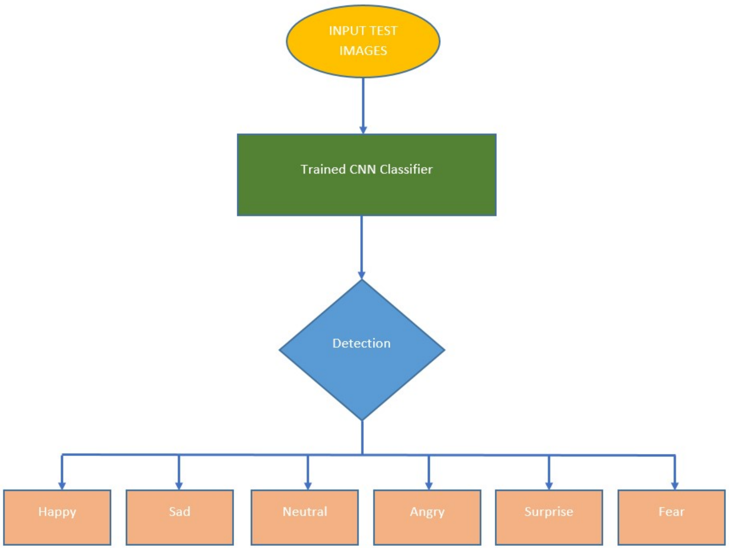
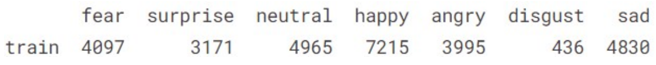


Figure 3.2: Complete algorithm flowchart

3.1 Dataset description

Our research paper has used two different datasets: FER-2013 and a dataset that has been created by us. FER-2013 is a standard dataset used to train and test facial emotion recognition models. It has a total of 28,709 training images and 7,178 testing images. The dataset contains pictures of seven different emotions: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise. Each emotion has around 5000 samples other than "Disgust", which has around 500 samples. The exact statistics of the training and testing images are given below along with bar plots:



```
       fear  surprise  neutral  happy  angry  disgust   sad
train  4097      3171     4965   7215   3995      436  4830
```

Figure 3.3: FER-2013 training dataset statistics

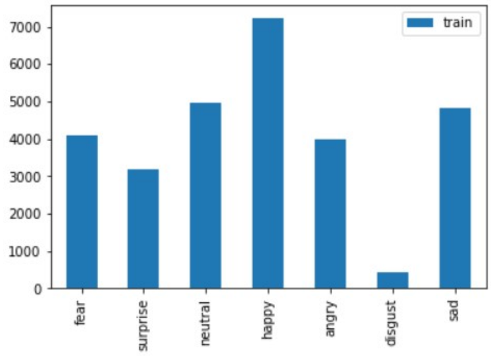
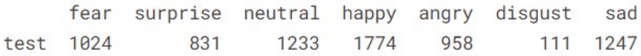


Figure 3.4: FER-2013 training images bar plot



```
      fear  surprise  neutral  happy  angry  disgust   sad
test  1024       831     1233   1774    958      111  1247
```

Figure 3.5: FER-2013 testing dataset statistics

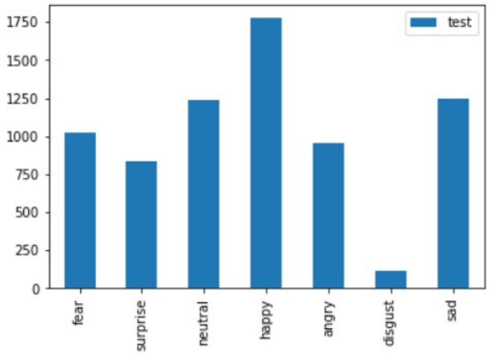
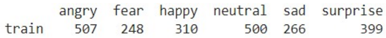


Figure 3.6: FER-2013 testing images bar plot

The other dataset that we have created, contains the images of individuals' eyes for each emotion. This dataset is insufficient due to absence of quality eye images for each category. However, we have gathered a total of around 3000 training images and around 500 testing images. We have dropped out "Disgust" emotion from both of our datasets and trained our model using six emotions only. The selected emotions are categorized as follows: 0: Angry, 1: Fear, 2: Happy, 3: Neutral, 4: Sad, 5: Surprise. The exact statistics of our dataset is given below along with its respected bar plots:



```
       angry  fear  happy  neutral  sad  surprise
train    507   248    310      500  266       399
```

Figure 3.7: Our training dataset statistics

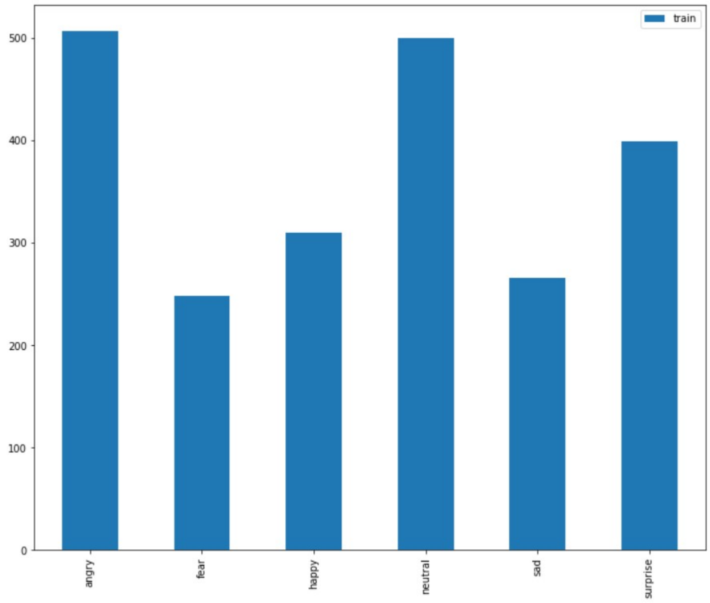
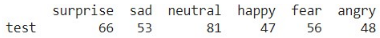


Figure 3.8: Our dataset's training images bar plot



```
      surprise  sad  neutral  happy  fear  angry
test        66   53       81     47    56     48
```

Figure 3.9: Our testing dataset statistics

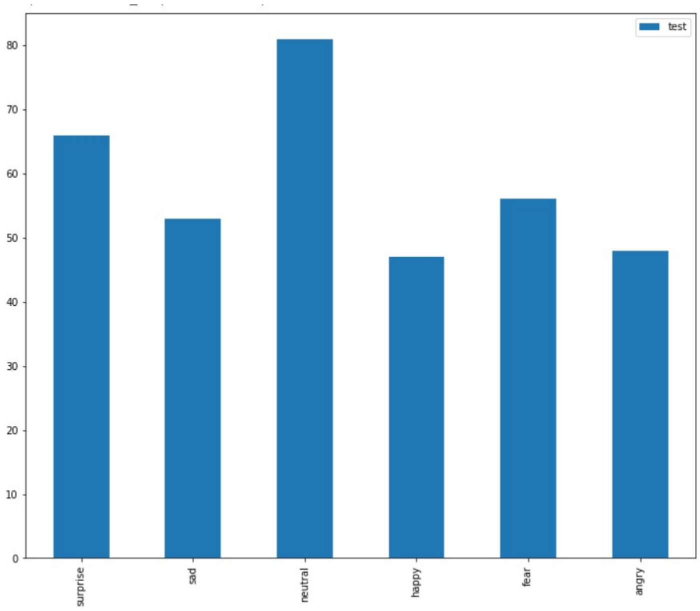


Figure 3.10: Our dataset's testing images bar plot

3.2 Dataset Preprocessing

FER-2013 datasets were already preprocessed, that is, the images were all gray scaled and images' size were 48 by 48. The dataset that we created was raw and the images of eyes from each picture were cropped horizontally in 16:9 aspect ratio and kept in different folders as per their category. These images, before feeding them to our model for training purposes, were gray scaled and their sizes were rescaled and restricted to 48 by 48.

The input sample from FER2013 dataset are shown below :

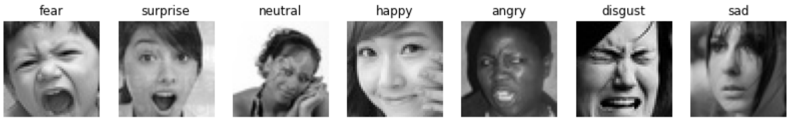


Figure 3.11: FER2013 dataset

The input sample from our own dataset are shown below :

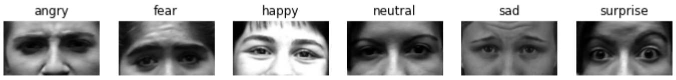


Figure 3.12: FER2013 dataset

3.3 Machine Learning Model

For our machine learning model, we have chosen Convolutional Neural Network (CNN). CNN gives excellent accuracy when it is utilized for classifying and recognizing images. It uses a hierarchical model that builds a network in the shape of a funnel and then outputs a fully-connected layer in which all neurons are connected to each other and the output is processed [16]. CNN does not require external feature extraction. The system learns to extract features, and then the key notion of CNN is that it generates invariant features by convolutioning images and filters, which are then passed on to the next layer. The features in the following layer are twisted with various filters to yield more invariant and abstract features, and the process is repeated until the final feature [15].

CNN stands for Convolutional Neural Network which is a class of deep neural networks mostly used in analyzing visual imagery [21] . A digital image is a representation of visual data shown in binary values [19] . An RGB image is a matrix of pixel values where a grayscale image is the same but in a single panel [21] . Each of the pixels in an image contain a pixel value which is used to show how bright an individual pixel is going to be. When a large amount of data is passed through a human brain, we can process it instantly. Each neuron has its own receptive field which is connected to other neurons in a way that they cover the entire visual field [19] . Convolutional neural networks are made up of multiple neural networks. So just like the human brain, each of the neurons in a convolutional neural network processes data only on its receptive field. The layers are arranged in such a fashion that the network detects simpler patterns first and then moves on to more complex patterns [19] . Artificial neurons of a CNN are mathematical functions that calculate the weighted sum of multiple inputs and outputs as activation value. Each of the layers generate several activation functions when an image is inputted in a CNN [21] .

A CNN is divided into three layers: a convolution layer, a pooling layer and a fully connected layer.

3.3.1 Convolution Layer

The CNN's fundamental component is its convolution layer. The image's essential characteristics, such as the diagonal and horizontal edges, are extracted by the layers. This layer operates a dot product between two matrices, one of which is the kernel, or collection of learnable parameters [XXI] . The limited region of the receptive field is the other matrix. The kernel slides over the picture's height and breadth during the forward pass, providing an image representation of that receptive region. This creates an activation map, a two-dimensional representation of the image that shows the kernel's reaction at each spatial place in the image [XXI] . The output volume can be calculated using the formula:

$$W_{out} = \frac{W - F + 2P}{S} + 1$$

Figure 3.13: Formula of convolution layer

This will result in a Wout x Wout x Dout output volume [19] .

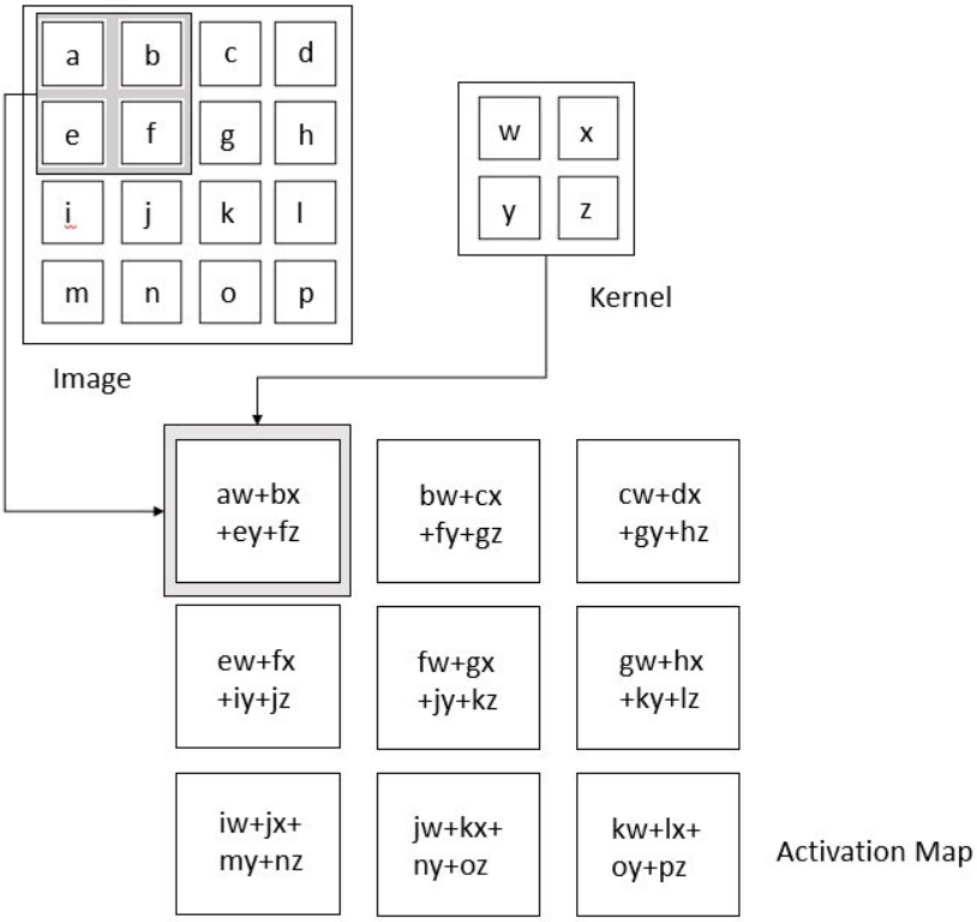


Figure 3.14: Convolution Operation **cnnexplain**

3.3.2 Motivation behind Convolution

Convolution takes advantage of three key concepts in computer vision: sparse interaction, parameter sharing, and equivariant representation [19] . Let's describe each one of them in detail.

Matrix multiplication by a matrix of parameters characterizing the interaction between the input and output unit is used in trivial neural network layers. This implies that each output unit communicates with each input unit. Convolution neural networks, on the other hand, exhibit sparse interaction. This is accomplished by making the kernel smaller than the input; for example, an image may include millions or thousands of pixels, but by processing it with the kernel, we may discover relevant information in tens or hundreds of pixels. This means we need to keep fewer parameters, which decreases the model's memory requirements while also improving the efficiency of the model [19] .

If computing one feature at a spatial position (x1, y1) is beneficial, it should be useful at other spatial points as well (x2, y2). It means that neurons are forced to employ the same set of weights for each two-dimensional slice, i.e., for each activation map. In a standard neural network, each element of the weight matrix is utilized once and then never again, however in a convolutional network, the weights applied to one input are the same as the weights applied elsewhere. The layers of a convolution neural network will have equivariance to translation due to parameter sharing. It states that if we modify the input in a way, the output will also change based on the modification of the input [19] .

3.3.3 Pooling Layer

The pooling layer replaces the system's output at certain points using a summary statistic of surrounding outputs. This minimizes the spatial dimension of the representation, reducing the required number of weights and calculations. It is through this procedure that each of the pieces are treated separately [XXI] . Pooling functions include the "rectangular neighborhood average", the "rectangle neighborhood L2 norm", and a "weighted average" depending on the "distance from the central pixel". The most common approach, however, is "max pooling", which returns the maximum output of the neighborhood [19] .

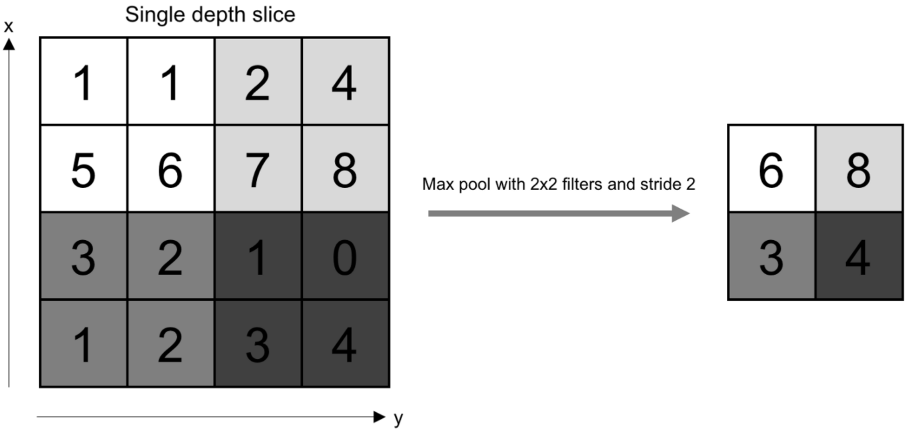


Figure 3.15: Pooling Operation (Source: O'Reilly Media)

We may use the following method to compute the size of the output volume if we have the following: "activation map of size W*W*D"; "pooling kernel of spatial size F"; "stride S" [19] .

$$W_{out} = \frac{W - F}{S} + 1$$

Figure 3.16: Pooling Operation (Source: O'Reilly Media)

The result will be an output volume of the size Wout*Wout*D. Pooling provides partial translation invariance under all conditions, which means that an item may be identified regardless of where it appears on the frame [19] .

3.3.4 Fully Connected Layer

As in ordinary FCNN, neurons in this layer have complete connection with all neurons in the previous and following layers. As a result, it may be calculated using a "matrix multiplication" followed by a "bias effect". The FC layer aids in mapping the representation between input and output [19] .

3.3.5 Non-Linearity Layers

Convolution is a linear operation whereas images are non-linear. This is why non-linearity layers are inserted right after convolution layers to provide non-linearity to the activation map [19] . Some popular types of non-linear operations include:

1. Sigmoid

The mathematical formula for sigmoid non-linearity puts a real-valued number in between the range of 0 and 1 [19] .

2. Tanh

Tanh puts a real-valued number in between the range of -1 and 1 inclusive. Even though the activation saturates, the output is 0 centered [19] .

3. ReLU

Rectified Linear Unit calculates the function f(k)=max (0, k), keeping the activity threshold at zero. It is more dependable and converges 6x faster than sigmoid and tanh. However, it requires a proper learning rate because it can be weak during training [19] .

3.3.6 Designing a Convolutional Network

We have seen what a convolution network is and what are the various components that are needed to make a successful model. We can now build a CNN model using the following tools and have built the following two models: model 1 and model 2.

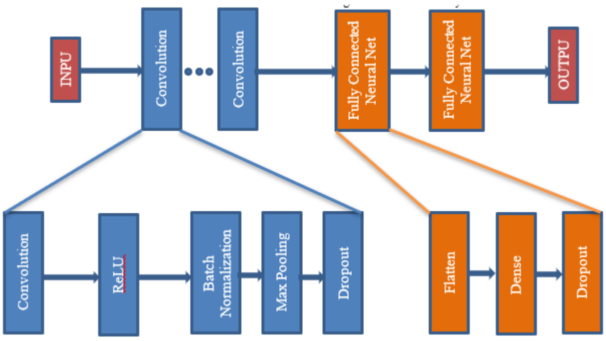


Figure 3.17: Model 1 Architecture

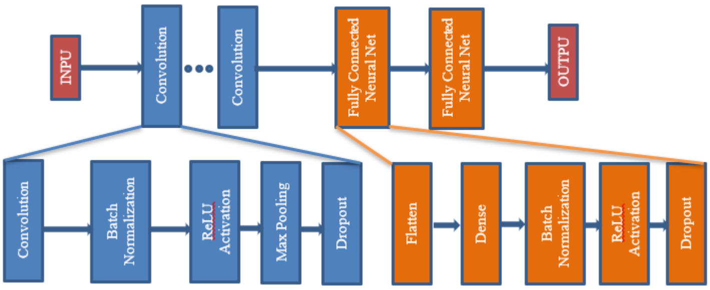


Figure 3.18: Model 2 Architecture

3.3.7 Applications of Convolutional Neural Network

Some of the applications of CNN are given below:

1. Detecting Objects: The object detection models used in autonomous vehicles, facial detection and other applications are build using the help of upgraded CNN infrastructures such as R-CNN, Fast R-CNN, etc. [19] .

2.Semantic segmentation: CNN is used to include information into an image segmentation model by a group of Hong Kong academics in 2015. Moreover, a fully functional CNN model developed by UC Berkley surpassed “state-of-the-art” semantic segmentation in performance [19] .

3. Image captioning: CNNs and recurrent neural networks are used to provide subtitles for photos and videos. This may be used for a variety of purposes, including activity recognition and visually impaired video and picture descriptions. YouTube has made extensive use of it in order to make sense of the massive number of videos that are regularly posted to the network [19] .

Chapter 4

Findings and Discussion

Our dataset was later classified into different classes by having them all kept in different folders. The images are collected from different images of different expressions. We have taken only the eyes of the images for our dataset.

4.1 Implementation

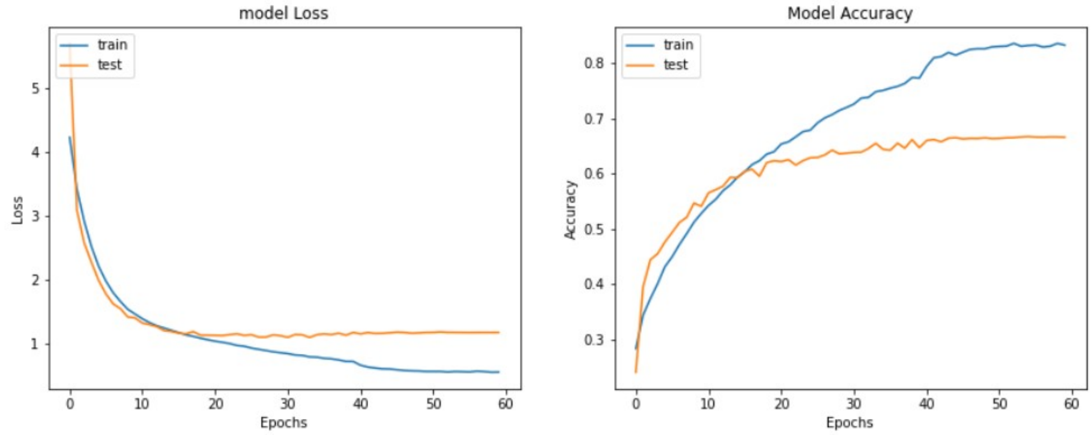
We have designed two models based on CNN. We used the FER-2013 dataset to first train and test our models and then used our own dataset and compared the results for each of the models. The FER-2013 dataset is more sophisticated and has more test and train images than our dataset and contains thousands of images for a single emotion while our own contains a couple hundred for the same emotion. The difference in the size of the dataset has shown significant differences in the results for both of our models for both cases. However, our concise dataset has given sufficient results in comparison to the vast and proper FER-2013 dataset.

4.2 Testing our model

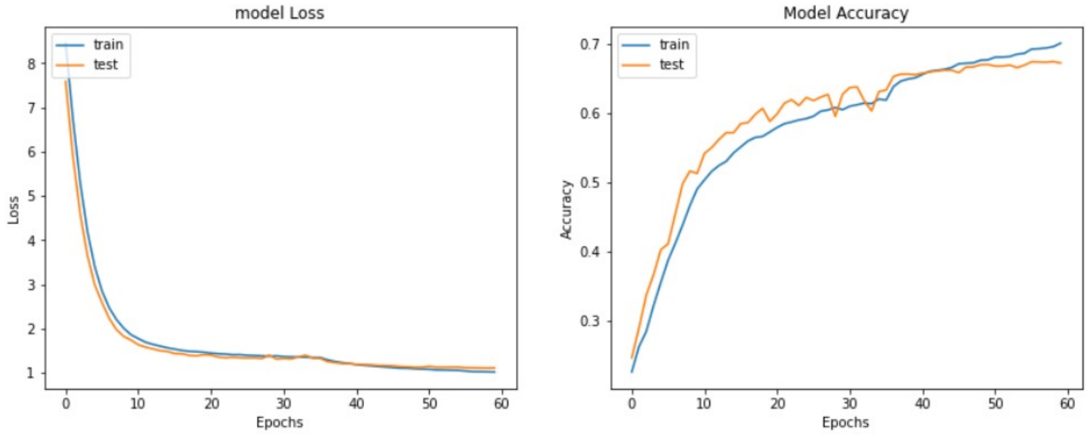
We first used FER-2013 dataset to train both of our models and then used our own dataset to find and predict emotions using eye images. We used tensorflow and keras to build our two CNN models. In both cases, the same two models were used for running different numbers of epochs to better understand our models and compare the results.

4.3 Training and Testing with FER-2013

We selected FER-2013 dataset as a standard to train and then test our two models to compare the results with our own dataset. We ran 60 epochs with a batch size of 64 to get better results. Our final accuracy score for model 1 was 83.78% with a validation accuracy score of 65.35%. A huge gap can be seen between the training curve and the testing curve in the accuracy vs epochs graph. This shows that even though the model can classify the images well during the training phase, the performance is not the same during the testing phase.



For model 2, final accuracy was 76.56% with a validation accuracy of 67.21%: Model 2 shows a far better result in comparison to model 1. The difference between training accuracy and validation accuracy is much less than the case in model 1. This shows that the model's performance in sorting images into their respective classes is quite similar in both training and testing phase.



4.4 Training and Testing with our own dataset

Finally, we used our own dataset to train both of our models. For each of the models, we ran using a different number of epochs to observe the increasing accuracy rate of our model. We first ran our model using 10 epochs which gave us a final accuracy of 36.59% with a validation accuracy of 30.20% for model 1. We can see unexpected rise and fall of both training and testing curves in both Loss vs epochs graph and Accuracy vs epochs graph. The curves are also far away from each other in each points.

Final accuracy of 22.74% with a validation accuracy of 13.68% for model 2 as shown in Figure. The validation accuracy is much low compared to the training accuracy in low epochs. Although we can witness a steady rise in training accuracy, validation accuracy almost remains the same throughout. The drop in loss is steady in training phase as well. However, the drop in loss is trivial in testing phase.

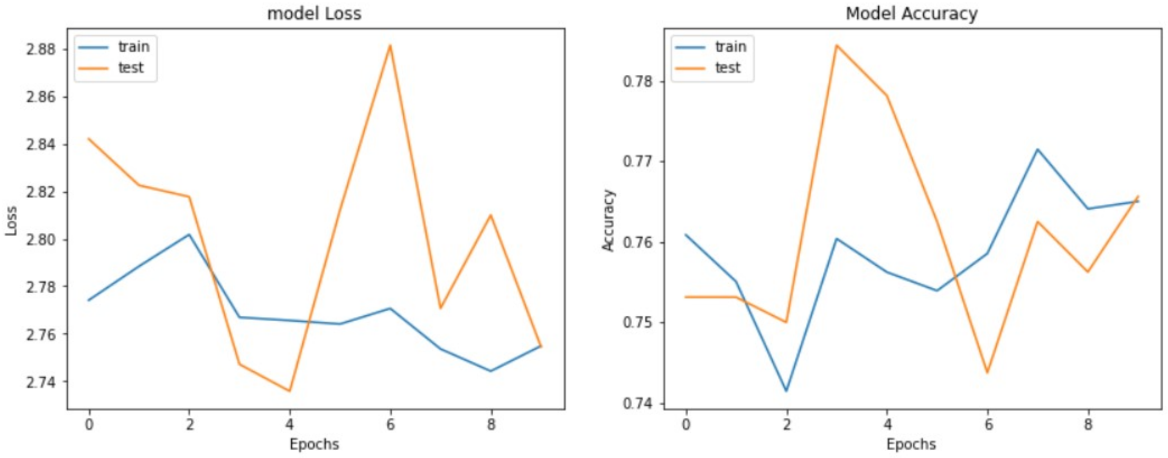


Figure 4.1: 10 Epochs with Model 1

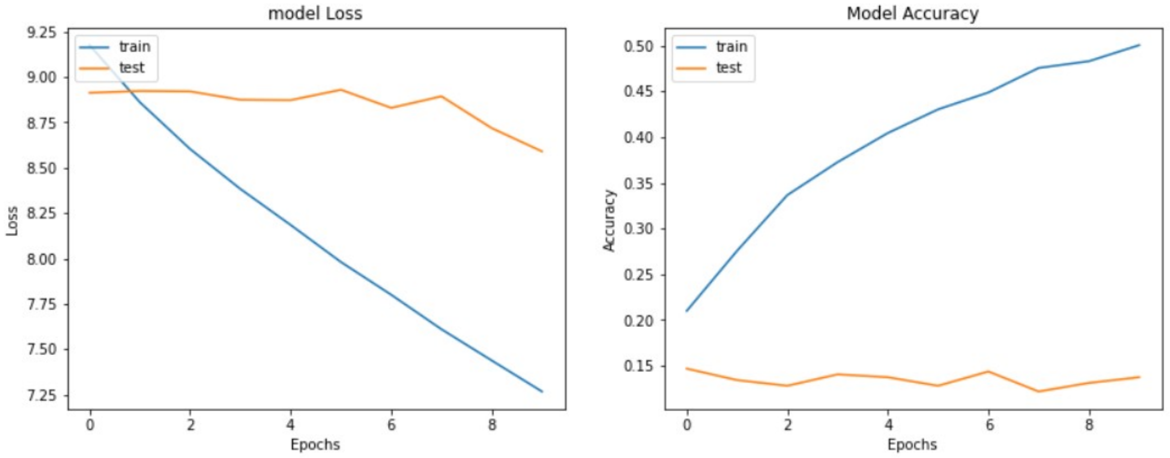


Figure 4.2: 10 Epochs with Model 2

We later increased the number of epochs to 30 and got a better final accuracy score of 77.09% and a validation accuracy of 75.50% for model 1. In the Loss vs epochs graph, the training curve started from a lower loss level and it decreased further, although not much. However, the testing curve started with a higher loss value which decreased considerably during the first 10 epochs and it dropped further to its lowest value during 24th epoch as it reached closer to the training curve. In the Accuracy vs epochs graph, the training curve started with an accuracy of around 0.69 and increased up to 0.8 and above. The testing curve began with a lower accuracy and showed a considerable rise during the first 10 epochs. The validation accuracy remained almost close to the training accuracy for the next 20 epochs.

Model 2 showed a final accuracy of 64.04% and validation accuracy of 68.38%. The drop in loss is higher in model 2, compared to model 1, for both testing and training curves. Although, the final loss level in model 2 is still higher in comparison to the final loss level in model 1. The training accuracy increased throughout the 30 epochs. However, the testing curve showed a considerable rise during the first 15 epochs as it reached closer to the training accuracy. The testing accuracy fluctuated in the remaining 15 epochs as the curve rose and fell. Nonetheless, both showed better results compared to 10 epochs.

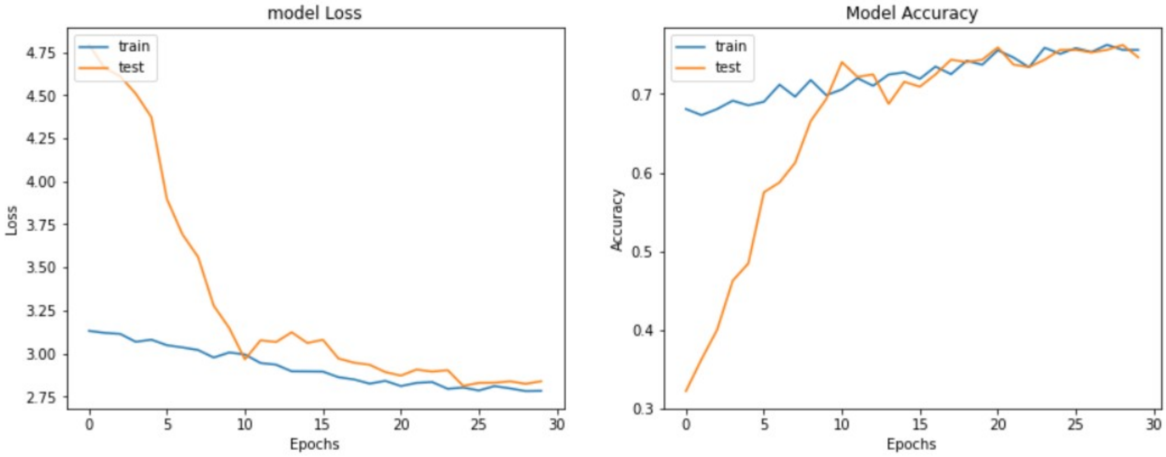


Figure 4.3: 30 Epochs with Model 1

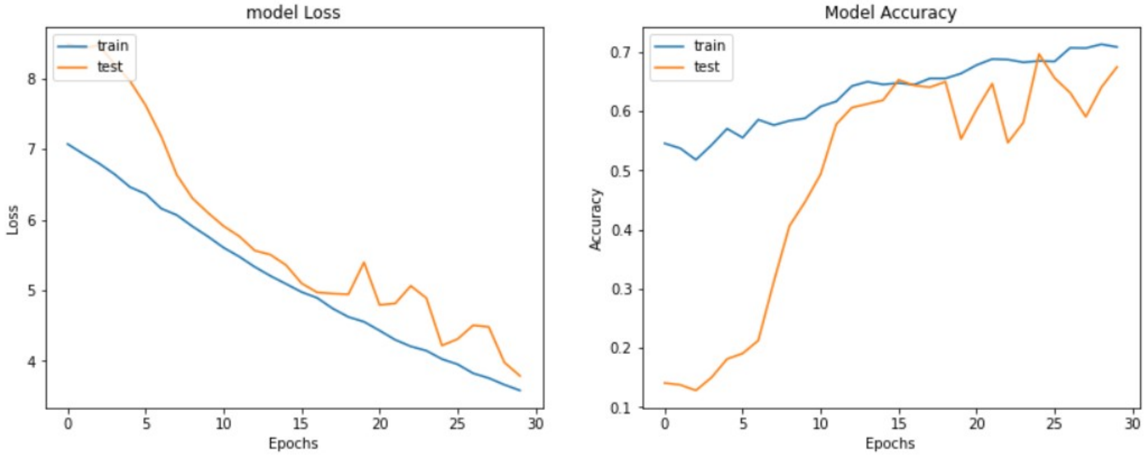


Figure 4.4: 30 Epochs with Model 2

Then finally we increased the number of epochs to 60 to match the number of epochs we used to train our models with FER-2013 dataset which yielded the best result after training both our models with our dataset. For model 1, the final accuracy score was 69.19% with a validation accuracy of 72.08%. In the Loss vs epochs graph, the training loss curve dropped sharply during first 10 epochs and maintained a lower loss level in the remaining 50 epochs. Even though the testing loss curve began with a lower level, it showed a rise in loss level during the first 10 epochs and then dropped to the lowest level in the next 10 epochs. The testing loss curve then maintained the lowest level for the remaining 40 epochs. In the Accuracy vs epochs graph, the training accuracy curve has shown a significant rise during the first 10 epochs and then a slow rise for the next 50 epochs. The testing curve rose sharply during the first 20 epochs, surpassing the accuracy level of the training curve. The testing curve maintained its highest accuracy throughout the remaining 40 epochs, staying above the testing curve.

For model 2, the final accuracy was 66.55% with a validation accuracy of 72.36% as shown in Figure. In the Loss vs epochs graph, the training curved showed a steady drop throughout the 60 epochs. The testing curve, however, did not display a steady drop but reached the same loss level as training curve did during the 60 epochs. In the Accuracy vs epochs graph, the training curve showed a sharp rise for the first 10 epochs and then a steady rise for the next 50 epochs. The testing curve started with a poor accuracy for the first 15 epochs and then rose sharply in the next 5 epochs. The curve then showed an unsteady rise for the remaining 40 epochs, crossing the training curve during 57th epoch.

The validation accuracy surpassed the training accuracy during 60 epochs for both the models.

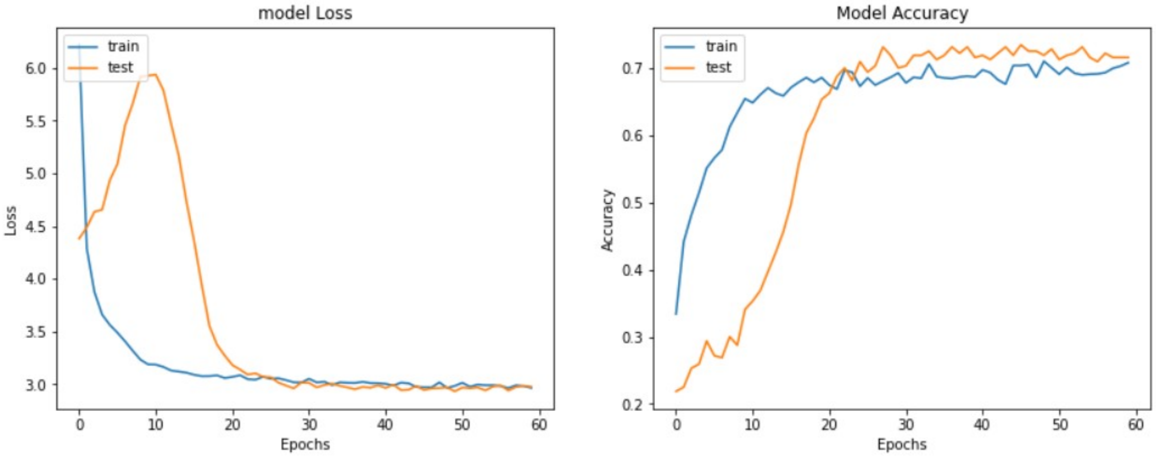


Figure 4.5: 60 Epochs with Model 1

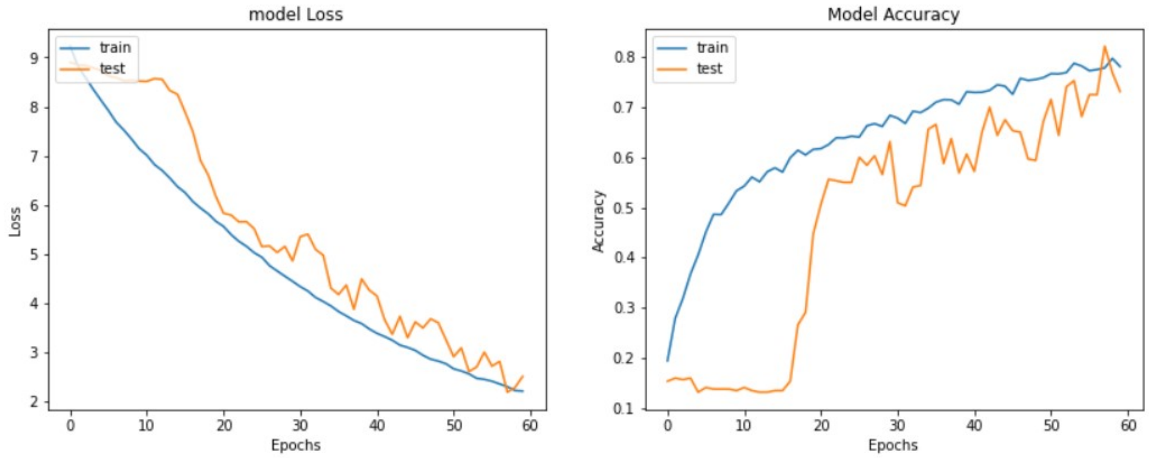


Figure 4.6: 60 Epochs with Model 2

From the data, we see that:

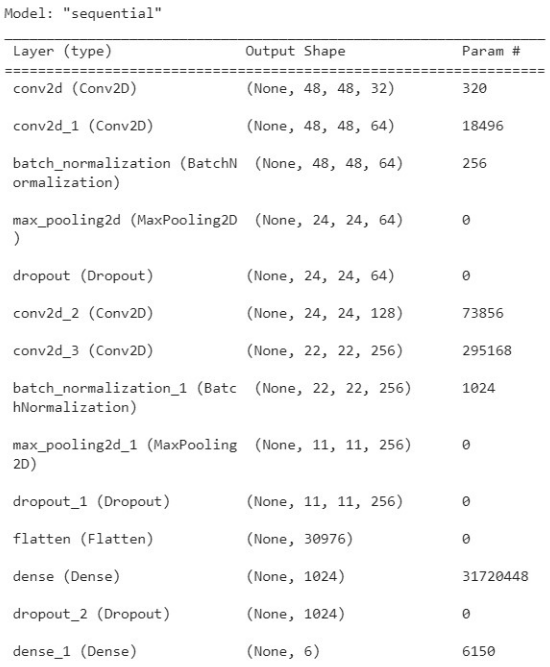
Model 1	FER-2013 (60 epochs)	Eye images (10 epochs)	Eye images (30 epochs)	Eye images (60 epochs)
Training Accuracy	83.78%	36.59%	77.09%	69.19%
Validation Accuracy	65.35%	30.20%	75.50%	72.08%

Model 2	FER-2013 (60 epochs)	Eye images (10 epochs)	Eye images (30 epochs)	Eye images (60 epochs)
Training Accuracy	76.56%	22.74%	64.04%	66.55%
Validation Accuracy	67.21%	13.68%	68.38%	72.36%

The low accuracy of the model for our dataset may be because of our shortcomings when it came to the amount of training data, we could provide for the model which can be overcome by simply using a superior dataset. The margin of error for our dataset can also be because of the quality of images we provided which may have lacked certain features thus resulting in a low accuracy score. However, the models showed a considerable result, given the concise dataset the we provided to train and test, when compared to the vast dataset of FER-2013.

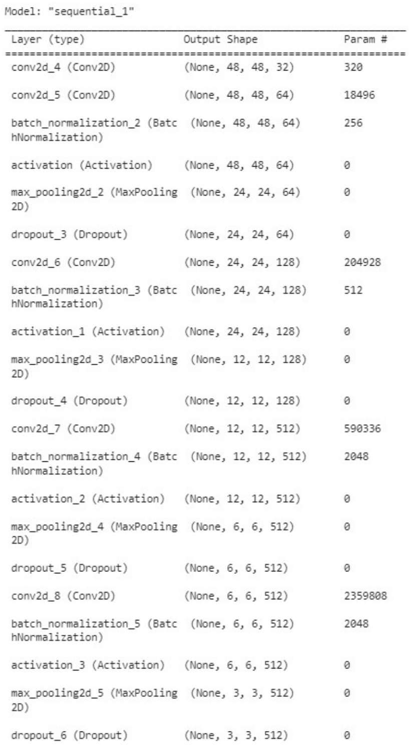
4.5 Model Summary

We have used two CNN models for detecting moods using eye images. The summary of these models are given below through figure x and is also represented through flowcharts.



```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 48, 48, 32)        320

 conv2d_1 (Conv2D)           (None, 48, 48, 64)        18496

 batch_normalization (BatchN  (None, 48, 48, 64)       256
 ormalization)

 max_pooling2d (MaxPooling2D  (None, 24, 24, 64)       0
 )

 dropout (Dropout)           (None, 24, 24, 64)        0

 conv2d_2 (Conv2D)           (None, 24, 24, 128)       73856

 conv2d_3 (Conv2D)           (None, 22, 22, 256)       295168

 batch_normalization_1 (Batc  (None, 22, 22, 256)      1024
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 11, 11, 256)      0
 2D)

 dropout_1 (Dropout)         (None, 11, 11, 256)       0

 flatten (Flatten)           (None, 30976)             0

 dense (Dense)               (None, 1024)              31720448

 dropout_2 (Dropout)         (None, 1024)              0

 dense_1 (Dense)             (None, 6)                 6150
```

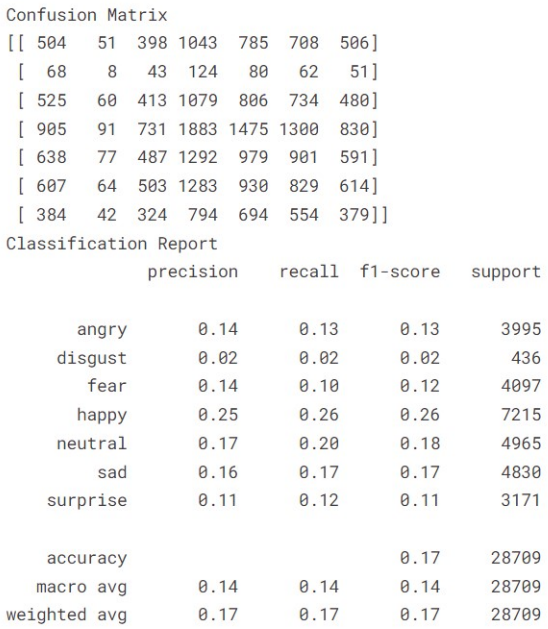
Figure 4.7: Model 1 Summary



```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 48, 48, 32)        320

 conv2d_5 (Conv2D)           (None, 48, 48, 64)        18496

 batch_normalization_2 (Batc  (None, 48, 48, 64)       256
 hNormalization)

 activation (Activation)     (None, 48, 48, 64)        0

 max_pooling2d_2 (MaxPooling  (None, 24, 24, 64)       0
 2D)

 dropout_3 (Dropout)         (None, 24, 24, 64)        0

 conv2d_6 (Conv2D)           (None, 24, 24, 128)       204928

 batch_normalization_3 (Batc  (None, 24, 24, 128)      512
 hNormalization)

 activation_1 (Activation)   (None, 24, 24, 128)       0

 max_pooling2d_3 (MaxPooling  (None, 12, 12, 128)      0
 2D)

 dropout_4 (Dropout)         (None, 12, 12, 128)       0

 conv2d_7 (Conv2D)           (None, 12, 12, 512)       590336

 batch_normalization_4 (Batc  (None, 12, 12, 512)      2048
 hNormalization)

 activation_2 (Activation)   (None, 12, 12, 512)       0

 max_pooling2d_4 (MaxPooling  (None, 6, 6, 512)        0
 2D)

 dropout_5 (Dropout)         (None, 6, 6, 512)         0

 conv2d_8 (Conv2D)           (None, 6, 6, 512)         2359808

 batch_normalization_5 (Batc  (None, 6, 6, 512)        2048
 hNormalization)

 activation_3 (Activation)   (None, 6, 6, 512)         0

 max_pooling2d_5 (MaxPooling  (None, 3, 3, 512)        0
 2D)

 dropout_6 (Dropout)         (None, 3, 3, 512)         0
```

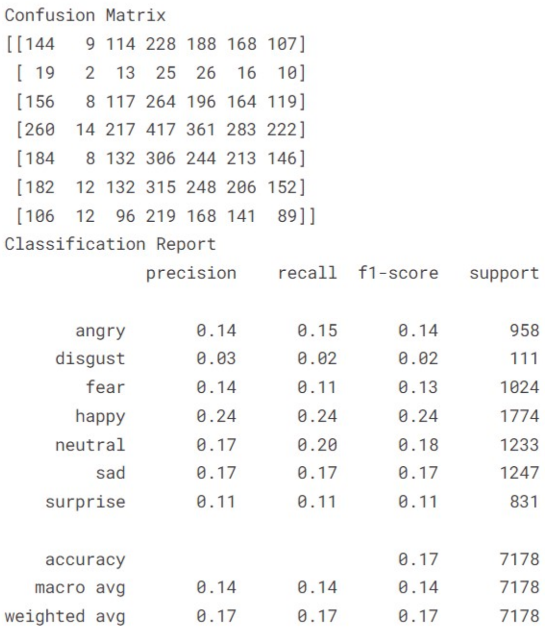
Figure 4.8: Model 2 Summary

The following figures show the confusion matrix and classification report of FER-2013 dataset and our own dataset on model 2.



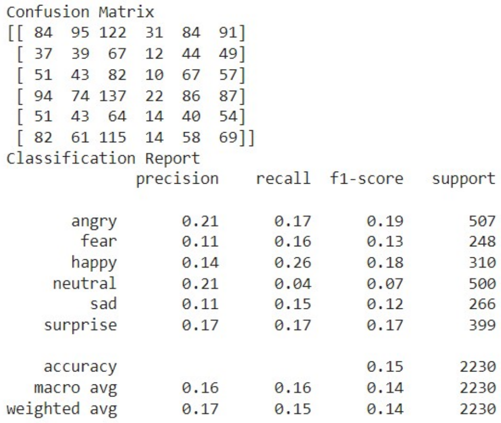
```
Confusion Matrix
[[ 504   51  398 1043  785  708  506]
 [  68    8   43  124   80   62   51]
 [ 525   60  413 1079  806  734  480]
 [ 905   91  731 1883 1475 1300  830]
 [ 638   77  487 1292  979  901  591]
 [ 607   64  503 1283  930  829  614]
 [ 384   42  324  794  694  554  379]]
Classification Report
              precision    recall  f1-score   support

       angry       0.14      0.13      0.13      3995
     disgust       0.02      0.02      0.02       436
        fear       0.14      0.10      0.12      4097
       happy       0.25      0.26      0.26      7215
     neutral       0.17      0.20      0.18      4965
         sad       0.16      0.17      0.17      4830
    surprise       0.11      0.12      0.11      3171

    accuracy                           0.17     28709
   macro avg       0.14      0.14      0.14     28709
weighted avg       0.17      0.17      0.17     28709
```

Figure 4.9: Confusion Matrix and Classification Report of FER-2013 training dataset



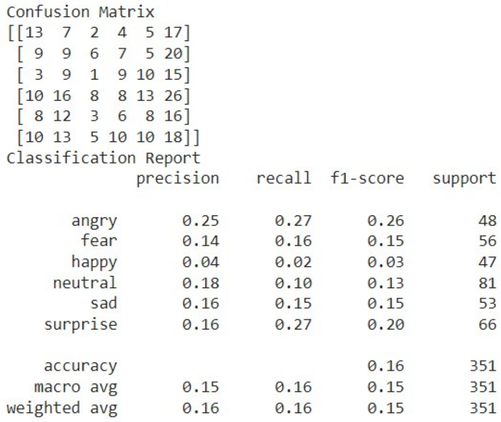
```
Confusion Matrix
[[144   9 114 228 188 168 107]
 [ 19   2  13  25  26  16  10]
 [156   8 117 264 196 164 119]
 [260  14 217 417 361 283 222]
 [184   8 132 306 244 213 146]
 [182  12 132 315 248 206 152]
 [106  12  96 219 168 141  89]]
Classification Report
              precision    recall  f1-score   support

       angry       0.14      0.15      0.14       958
     disgust       0.03      0.02      0.02       111
        fear       0.14      0.11      0.13      1024
       happy       0.24      0.24      0.24      1774
     neutral       0.17      0.20      0.18      1233
         sad       0.17      0.17      0.17      1247
    surprise       0.11      0.11      0.11       831

    accuracy                           0.17      7178
   macro avg       0.14      0.14      0.14      7178
weighted avg       0.17      0.17      0.17      7178
```

Figure 4.10: Confusion Matrix and Classification Report of FER-2013 testing dataset



```
Confusion Matrix
[[ 84  95 122  31  84  91]
 [ 37  39  67  12  44  49]
 [ 51  43  82  10  67  57]
 [ 94  74 137  22  86  87]
 [ 51  43  64  14  40  54]
 [ 82  61 115  14  58  69]]
Classification Report
              precision    recall  f1-score   support

       angry       0.21      0.17      0.19       507
        fear       0.11      0.16      0.13       248
       happy       0.14      0.26      0.18       310
     neutral       0.21      0.04      0.07       500
         sad       0.11      0.15      0.12       266
    surprise       0.17      0.17      0.17       399

    accuracy                           0.15      2230
   macro avg       0.16      0.16      0.14      2230
weighted avg       0.17      0.15      0.14      2230
```

Figure 4.11: Confusion Matrix and Classification Report of our training dataset



```
Confusion Matrix
[[13  7  2  4  5 17]
 [ 9  9  6  7  5 20]
 [ 3  9  1  9 10 15]
 [10 16  8  8 13 26]
 [ 8 12  3  6  8 16]
 [10 13  5 10 10 18]]
Classification Report
              precision    recall  f1-score   support

       angry       0.25      0.27      0.26        48
        fear       0.14      0.16      0.15        56
       happy       0.04      0.02      0.03        47
     neutral       0.18      0.10      0.13        81
         sad       0.16      0.15      0.15        53
    surprise       0.16      0.27      0.20        66

    accuracy                           0.16       351
   macro avg       0.15      0.16      0.15       351
weighted avg       0.16      0.16      0.15       351
```

Figure 4.12: Confusion Matrix and Classification Report of our testing dataset

Chapter 5

Conclusion

We attempted to reliably determine a person's mood with eye-tracking using CNN. We aimed to see if there is a link between mood detection using only eye-images. We hoped to train our model to utilize eye photos and follow their pattern to assess a person's mood using our dataset of around 3000 images. The goal was to make mood identification easier in these modern times by laying emphasis just on the eyes. We chose CNN for image processing since it produces the most accurate results, and does not require any external feature extraction. We initially tested our model with the FER-2013 dataset, which contains around 30,000 images, then compared the results with our own dataset. To see the rising accuracy rate of our model, we ran it with a different number of epochs. Finally, we raised the number of epochs to 60, which corresponded to the number of epochs we used to train our model using FER-2013, which produced the best results after training with our dataset with a training accuracy of 66.55% and validation accuracy of 72.36% in model 2. However, model 1 showed better results during 30 epochs with a training accuracy of 77.09% and a validation accuracy of 75.50%. The training accuracy for model 1 dropped to 69.19% and validation accuracy dropped to 72.08% when it was run for 60 epochs. The model's poor accuracy for our dataset might be due to our limitations in terms of the quantity of training data we could supply, which could be solved by simply choosing a better dataset. After extracting the features from the fed images, the model categorizes the photos into its respective moods.

Bibliography

[1] B. Tursky, D. Shapiro, A. Crider, and D. Kahneman, "Pupillary heart rate and skin resistance changes during a mental task," *Journal of experimental psychology*, vol. 79, pp. 164–167, Feb. 1969. DOI: 10.1037/h0026952.

[2] R. P. Hobson, J. Ouston, and A. Lee, "Emotion recognition in autism: Coordinating faces and voices," *Psychological Medicine*, vol. 18, no. 4, pp. 911–923, 1988. DOI: 10.1017/S0033291700009843.

[3] T. Partala, M. Jokiniemi, and V. Surakka, "Pupillary responses to emotionally provocative stimuli," in *Proceedings of the 2000 Symposium on Eye Tracking Research amp; Applications*, ser. ETRA '00, Palm Beach Gardens, Florida, USA: Association for Computing Machinery, 2000, pp. 123–129, ISBN: 1581132808. DOI: 10.1145/355017.355042. [Online]. Available: https://doi.org/10.1145/355017.355042.

[4] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behavior Research Methods, Instruments, & Computers*, vol. 34, pp. 455–470, 2002.

[5] S. Craig, A. Graesser, J. Sullins, and B. Gholson, "Affect and learning: An exploratory look into the role of affect in learning with autotutor," *Journal of Educational Media*, vol. 29, no. 3, pp. 241–250, 2004. DOI: 10.1080/1358165042000283101. eprint: https://doi.org/10.1080/1358165042000283101. [Online]. Available: https://doi.org/10.1080/1358165042000283101.

[6] W.-B. Horng, C.-Y. Chen, Y. Chang, and C.-H. Fan, "Driver fatigue detection based on eye tracking and dynamk, template matching," *IEEE International Conference on Networking, Sensing and Control, 2004*, vol. 1, pp. 7–12, 2004.

[7] A. Mehrabian, *Communication without words*, 2nd ed. Routledge, 2008, pp. 193–200.

[8] S. Alghowinem, R. Goecke, M. Wagner, J. Epps, M. Breakspear, and G. Parker, "From joyous to clinically depressed: Mood detection using spontaneous speech.," Jan. 2012.

[9] S. Alghowinem, M. AlShehri, R. Goecke, and M. Wagner, "Exploring eye activity as an indication of emotional states using an eye-tracking sensor," in *Intelligent Systems for Science and Information: Extended and Selected Results from the Science and Information Conference 2013*, L. Chen, S. Kapoor, and R. Bhatia, Eds. Cham: Springer International Publishing, 2014, pp. 261–276, ISBN: 978-3-319-04702-7. DOI: 10.1007/978-3-319-04702-7_15.

[10] C. Aracena, S. Basterrech, V. Snáel, and J. Velásquez, “Neural networks for emotion recognition based on eye tracking data,” in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, 2015, pp. 2632–2637. DOI: 10.1109/SMC.2015.460.

[11] Rajakumari and S. T. Selvi, “Hci and eye tracking : Emotion recognition using hidden markov model,” 2015.

[12] M. Gulraj and N. Ahmad, “Mood detection of psychological and mentally disturbed patients using machine learning techniques,” *IJCSNS International Journal of Computer Science and Network Security*, vol. 16, pp. 63–67, Aug. 2016.

[13] S. Bazrafkan, T. Nedelcu, P. Filipczuk, and P. Corcoran, “Deep learning for facial expression recognition: A step closer to a smartphone that knows your moods,” Jan. 2017, pp. 217–220. DOI: 10.1109/ICCE.2017.7889290.

[14] H. Tang, W. Liu, W.-L. Zheng, and B.-L. Lu, “Multimodal emotion recognition using deep neural networks,” Oct. 2017, pp. 811–819, ISBN: 978-3-319-70092-2. DOI: 10.1007/978-3-319-70093-9_86.

[15] R. Bhatia. “Why convolutional neural networks are the go-to models in deep learning.” (2018), [Online]. Available: https://analyticsindiamag.com/why-convolutional-neural-networks-are-the-go-to-models-in-deep-learning/.

[16] K. Maladkar. “Overview of convolutional neural network in image classification.” (2018), [Online]. Available: https://analyticsindiamag.com/why-convolutional-neural-networks-are-the-go-to-models-in-deep-learning/.

[17] Z. Kilimci, S. Akyokus, M. Uysal, and A. Güven, “Mood detection from physical and neuro-physical data using deep learning models,” *Complexity*, vol. 2019, Dec. 2019. DOI: 10.1155/2019/6434578.

[18] J. Z. Lim, J. Mountstephens, and J. Teo, “Emotion recognition using eye-tracking: Taxonomy, review and current challenges,” *Sensors*, vol. 20, no. 8, 2020. DOI: 10.3390/s20082384.

[19] M. Mishra. “Convolutional neural networks, explained.” (2020), [Online]. Available: https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939.

[20] P. Tarnowski, M. Kołodziej, A. Majkowski, and R. Rak, “Eye-tracking analysis for emotion recognition,” *Computational Intelligence and Neuroscience*, vol. 2020, pp. 1–13, Sep. 2020. DOI: 10.1155/2020/2909267.

[21] M. Mandal. “Introduction to convolutional neural networks (cnn).” (2021), [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/?fbclid=IwAR3hIlaYxYjbRX-rhZPyecgtGyy0751c6u1qoR
mb7a8VKhS9hTRHR0_Q.