A

PROJECT BASED LEARNING REPORT

ON

"MUSIC4MOOD: ML DRIVEN SONG RECOMMENDER"

FOR PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
PROJECT BASED LEARNING SUBJECT
OF S.E. E&TC – 2019 COURSE, SPPU, PUNE

By

Harsh Umesh Bhat	**S190053027**
Shreyas Shreepad Chandolkar	**S190053038**
Atharva Jitendra Chirputkar	**S190053046**
Kaustubh Chintamani Damle	**S190053050**

GUIDE
Prof. Lalit P. Patil

DEPARTMENT OF
ELECTRONICS AND TELECOMMUNICATION ENGINEERING
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
PUNE – 43

ACADEMIC YEAR: 2021 – 2022

Department of Electronics and Telecommunication Engineering

Pune Institute of Computer Technology, Pune – 43

CERTIFICATE

This is to certify that the Project Based Learning Project Report entitled

MUSIC4MOOD: ML DRIVEN SONG RECOMMENDER

has been successfully completed by

Harsh Umesh Bhat	S190053027
Shreyas Shreepad Chandolkar	S190053038
Atharva Jitendra Chirputkar	S190053046
Kaustubh Chintamani Damle	S190053050

Is a bona fide work carried out by them under the supervision of Prof. Lalit P. Patil and it is approved for the partial fulfillment of the requirements for Project Based Learning Subject of S.E. E&TC – 2019 Course of the Savitribai Phule Pune University, Pune.

Prof. Lalit P. Patil
Project Guide
Dept. of E&TC

Prof. Dr. Mousami V. Munot
HOD, E&TC

Place: Pune
Date: 12/05/2022

ACKNOWLEDGEMENT

This has been a tough yet fruitful venture, all of us strived hard to reach the end and the payoff was definitely worth the effort as not only did we complete the project but also learned teamwork and life skills along the way.

So, henceforth, we would like to express our gratitude here. The first people that come to mind in guiding us through this journey are the magnanimous personas, respected E&TC HoD Prof. Dr. Mousami V. Munot, and our principal, Dr. R. Sreemathy Ma'am for treating us with this opportunity. I, personally am indebted to my courteous teammates who aided me till the end of this venture and shared the burden of the project equally throughout the project

We would like to thank our doting mentor, Prof. Lalit P. Patil, who had always been there for us, reviewing our project every week, pointing out our flaws, and criticizing us while helping us grow at the same time.

We would also like to specially mention Mr. Akshit Madan for his worthy guidance and immense knowledge.

Harsh Umesh Bhat

Shreyas Shreepad Chandolkar

Atharva Jitendra Chirputkar

Kaustubh Chintamani Damle

ABSTRACT

World renowned applications such as Netflix and Spotify are well-known for their robust recommendation software. Such recommendation software work on the basis of identifying what the user wants based on the user's history. Our goal in this project is to build a recommendation system that would provide personalized content in the form of music by correctly identifying the mood of the user. This system will take input in the form images captured from the user's device and will give the output in the form of a song after analyzing and determining the user's mood. Concepts used are Basics of Data Analysis, Machine Learning Fundamentals, and Neural Networking

.

Harsh Umesh Bhat

Shreyas Shreepad Chandolkar

Atharva Jitendra Chirputkar

Kaustubh Chintamani Damle

List of figures:

List of tables:

Abbreviations

Symbol/word	meaning
ML	Machine Learning
CNN	Convolutional Neural Network: A deep learning algorithm that specializes in image classification.
TFLite	TensorFlow Lite: TensorFlow Lite is a mobile library for deploying models on mobile, microcontrollers and other edge devices.
SDK	Software Development Kit

CONTENTS

CHAPTER 1

Introduction

1.1 Background

In today's world, filled with competition and stress, music has been proved to be an excellent medicine. People from all backgrounds and age limits have been listening to music to get relaxation and peace of mind. Music is directly related to our emotions and many times the music we listen to depends on our present emotional state. The 7 major emotions that human showcases are: Happy, sad, angry, surprise, disgust, neutral, and fear. If we are able to track the emotions and then recommend specific music to relate to our emotions then that would certainly make the music listening experience wonderful.

1.2 Relevance

The adoption of machine learning and deep learning in electronics and telecommunication field has been revolutionary. The functions work efficiently to increase the accuracy and quality when images are transferred. Data analysis and prediction being a part of our curriculum, the application we present provides a great platform to learn and get hands-on experience.

1.3 Literature Survey

The given research paper tries to explain the importance of facial emotion recognition and gives detailed comparison between different ML models used for this purpose. The Convoluted Neural Networks (CNN) algorithm turns out to have the best accuracy on the given dataset. [1]

The research paper provides in-depth comparison between different android app development technologies such as Flutter, Java etc. Considering all the pros that Flutter offers over others, we decided to choose Flutter as our tech-stack. [2]

1.4 Motivation

Music is integral part of life of many people and hence, recommending apt music with respect to a person's mood, age and time plays a major role in uplifting someone's mood. It not only increases the satisfaction of listening to music but

also enhances user experience and that is the goal as this has widespread use-cases which make it favourable for business.

1.5 Aim of the Project

ML is a powerful tool to solve many real-life applications that involve image classification. Our aim is to build an Android application that recognizes facial emotion of person and recommends playlists according to his/her mood. We aim to create a solution that is:
1) Easier to use
2) Intuitive and attractive
3) The results obtained should be reliable.

1.6 Scope and Objectives

Since music is a part of our life and we have every second person listening to music every day, a presented solution will have tremendous usage. The recommendation system proves to be very beneficial as the task of selecting certain music is automated. This application is being developed in flutter and can be deployed for both Android as well as iOS users.

CHAPTER 2

Description of Project

2.1 Technical Approach

To establish a precise and modular workflow, our project is divided into 2 broad parts. Part 1 focuses on Machine Learning Model development and deployment, and Part 2 focuses on the front-end, routing, and UI development.

Part 1 – Machine Learning

As stated in [1] after rigorous research and analysis, it was concluded that deep learning models - particularly the Convolutional Neural Network (CNN) algorithm works best in the case of the Facial Emotion Recognition problem.

Convolutional Neural Networks:

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

- Convolutional layer
- Pooling layer
- Fully-connected (FC) layer

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colours and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object.

Convolutional Layer
The convolutional layer is the core building block of a CNN, and it is where the majority of computation occurs. It requires a few components, which are input data, a filter, and a feature map. Let's assume that the input will be a colour image, which is made up of a matrix of pixels in 3D. This means that the input will have three dimensions—a height, width, and depth—which correspond to RGB in an image. We also have a feature detector, also known as a kernel or a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.
The feature detector is a two-dimensional (2-D) array of weights, which represents part of the image. While they can vary in size, the filter size is typically a 3x3 matrix; this also determines the size of the receptive field. The filter is then applied to an area of the image, and a dot product is calculated between the input pixels and the filter.

This dot product is then fed into an output array. Afterward, the filter shifts by a stride, repeating the process until the kernel has swept across the entire image. The final output from the series of dot products from the input and the filter is known as a feature map, activation map, or a convolved feature.

Pooling Layer

Pooling layers, also known as down sampling, conduct dimensionality reduction, reducing the number of parameters in the input. Similar to the convolutional layer, the pooling operation sweeps a filter across the entire input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array. There are two main types of pooling:

- Max pooling: As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.
- Average pooling: As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.

While a lot of information is lost in the pooling layer, it also has a number of benefits for CNN. They help to reduce complexity, improve efficiency, and limit the risk of overfitting.

Since the algorithm is meant to be working on a mobile application, we went with the MobileNetV2 algorithm which is specially curated for such machine learning problems.

MobileNetV2:

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depth wise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers.

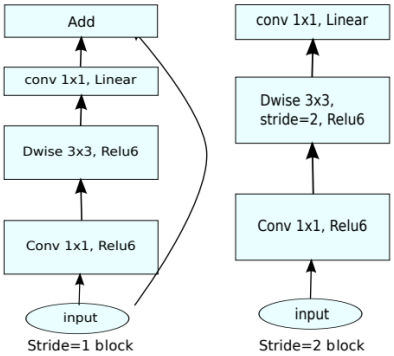


Fig. No. 2.1 MobilenetV2

Source: Towards data science

To mark out the facial area for better results of the model, we use the haar cascades facial recognition algorithm to serve this purpose.

Haar Cascades Classifiers:
Haar Cascade classifiers are an effective way for object detection. This method was proposed by Paul Viola and Michael Jones in their paper Rapid Object Detection using a Boosted Cascade of Simple Features. Haar Cascade is a machine learning-based approach where a lot of positive and negative images are used to train the classifier.

Positive images – These images contain the images which we want our classifier to identify.
Negative Images – Images of everything else, which do not contain the object we want to detect.

After the development of the model, deployment is the next important stage of the machine learning pipeline. We did it using FastApi and Tflite.

FastApi:

- FastAPI is way faster than Flask, not just that it's also one of the fastest python modules out there.
- Unlike Flask, FastAPI provides an easier implementation for Data Validation to define the specific data type of the data you send.
- Automatic Docs to call and test your API (Swagger UI and Redoc).
- FastAPI comes with built-in support for Asyncio, GraphQL, and Websockets.

Part B – Frontend part

Flutter:

- Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web from a single codebase.
- Main features of Flutter
  Simplified and agile development
  Single language for development
  Google support
- We have developed our application using Flutter stack and its packages.
- Libraries used with flutter:
  Image Picker, Path-provider, tflite
- Navigation is created by using the package – path-provider
- UI design is made user-friendly with minimal options and less complexity.

2.2 Block diagram

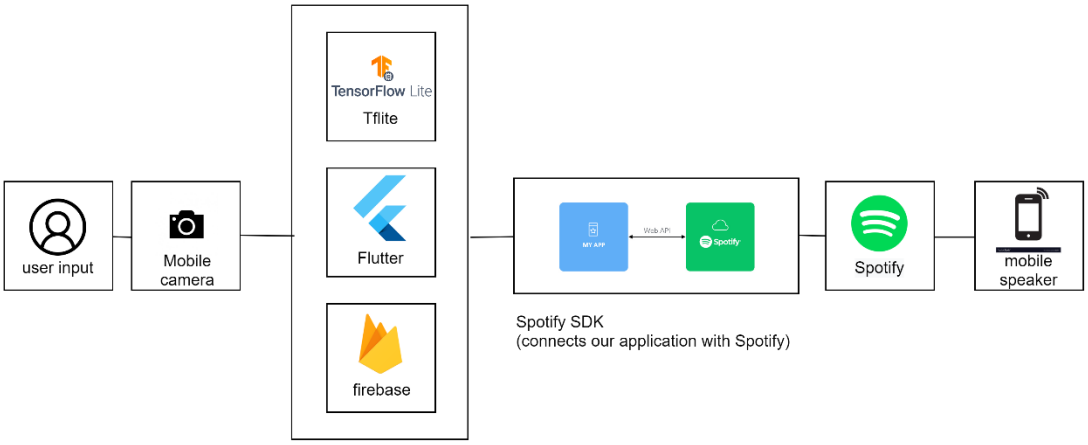


Fig. No. 2.2 Block diagram

The block diagram of the application is as follows:

In the first stage, a user would take a photograph of his/her face by means of the flutter app as is shown in the block diagram 2.2.

The camera plugin then interacts with the flutter code that in-turn, makes calls to the backend Api That was prepared with TFLite and FastApi.

The application then sends the data to the Firebase backend handler that keeps the track of a user.

The application then sends a request to the Spotify SDK after the model is processed upon the camera input provided by the user. The Spotify SDK then recommends the playlist and lets the user enjoy songs by means of the play button.

2.3 Flow chart

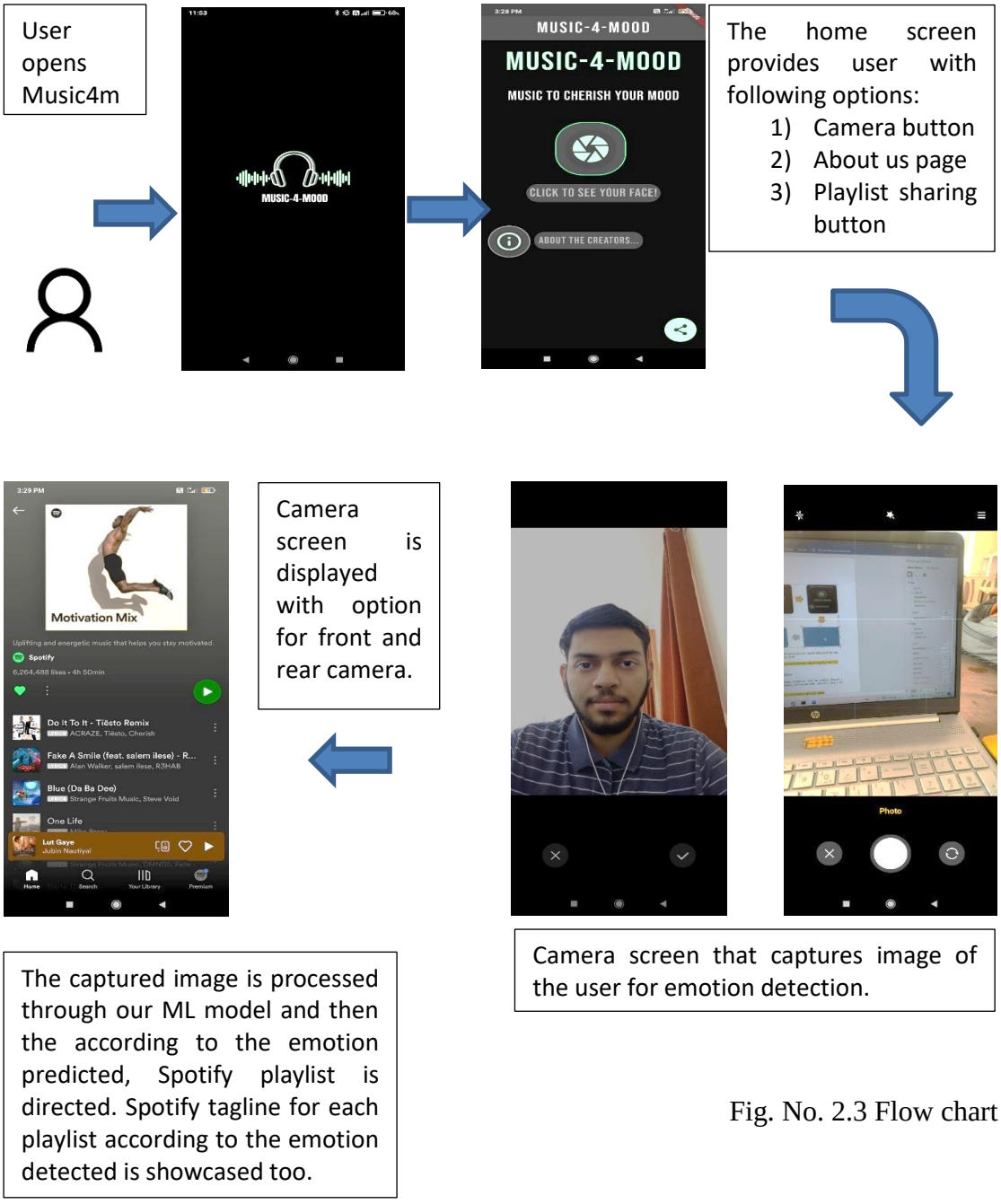


Fig. No. 2.3 Flow chart

Fig. No. 2.3 Flow chart depicts the overall flowchart of out Music4mood: ML driven song recommender application. The first step starts with capturing face of the user using camera. Then the image is processed through our ML model to predict the facial emotion. The emotion predicted is mapped with the Spotify playlist based on the emotion.

2.4 Software resources

Software resources with libraries:

Flutter: version - 2.10.5

Tflite package: version - 1.1.2

Spotify: version - 8.6.98.900

CHAPTER 3

System Design

3.1 Circuit diagram

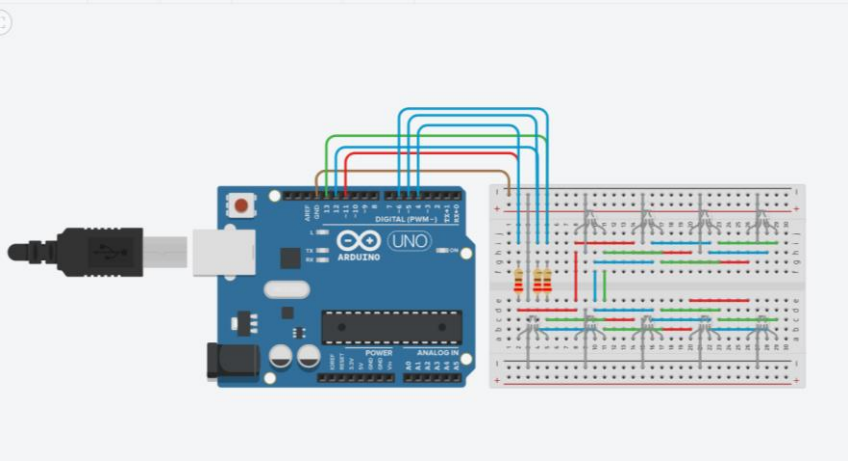


Fig. No. 3.1 Arduino powered RGB LED circuit in Tinkercad.

[Tinkercad is used to simulate the RGB LED circuit powered by Arduino microcontroller development board.]

Fig. No. 3.1 consists of the following electronic components:

Arduino UNO microcontroller development board, Jumper wires, Resistors (3, 100 Ohms), RGB LED (8). The Arduino provides an operating voltage of 5V. The input voltage recommended is from 7 to 12 V. The input voltage that can be provided is 6 to 20 V. The RGB LEDS have varying voltage supply such as for red LED its 2 V, green 3.2 V, blue 3.2 V. The combined recommended voltage supply for RGB LED is from 3 to 5 V. The maximum forward current is 25mA and backward current is 100mA. The resistor value is calculated accordingly. Using V = IR the values for resistors lie in the range 100 Ohms to 220 Ohms.

3.2 Design calculation

5V / 220 = 22 mA, that 40-mA max is to be avoided. Most LEDs are plenty bright with 10 mA and aren't that much brighter at 20 mA.

CHAPTER 4

Implementation and Testing

4.1 Implementation

Machine Learning Algorithm and deployment –

The Machine Learning Model is the integral part of our project. We used following libraries for training the Machine Learning model –

Tensorflow, Numpy, Pandas, OpenCV, and Keras.

NumPy: NumPy arrays facilitate advanced mathematical and other types of operations on large numbers of data. Typically, such operations are executed more efficiently and with less code than is possible using Python's built-in sequences.

Keras: Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research.

OpenCV: OpenCV provides a real-time optimized Computer Vision library, tools, and hardware. It also supports model execution for Machine Learning (ML)

Pandas: Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

The Camera Plug-in:
The camera plug-in is another important package that helped us use live cameras on the Flutter application. It can be called by writing a code as follows:

We have used the above-mentioned tech stack keeping mind the advantages that the stack offers. The Tensorflow library offers vast variety of algorithms to train our model upon which forms our peak usage.

The image captured is pass on to the flutter image picker library then the image is stored in firebase and then passed on to the ML model for predicting the emotion. The emotion is tracked using CNN MobilenetV2 algorithm. Here Tensorflow is used exclusively to train the model and also use the model to predict the emotion. Firebase forms the background storage for storing the image. Spotify SDK helps to connect the playlist recommended for a particular emotion with our application. Here routing process takes place and finally the user is passed on to the Spotify app for listening the song.

4.2 Testing and debugging

Issue: Camera plug in (video feed) throws 'unimplemented error' in newer releases of flutter.

The latest updated version of flutter proved to be hurdle for camera plugin package. The camera was not detected and a very uncertain error ‘unimplemented error’ was thrown. The problem persisted for long time as many users and developers from all over the world, brought this to attention through platforms such as GitHub, Stackoverflow, etc. Initially we were trying to capture the live video feed and then predict the emotion for the video feed.

Solution: Opted to access camera through 'image picker' another package of Flutter which allows you to take a picture through the camera. We solved by capturing just a single frame, a picture and then transferred the picture to our ML model to predict the emotion.

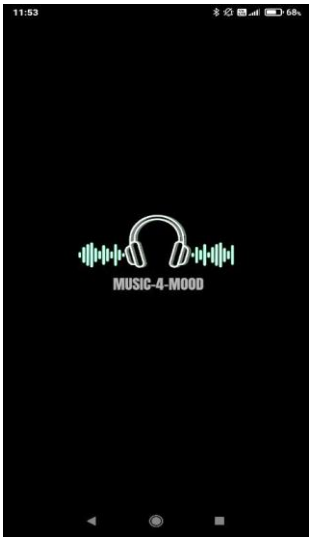
CHAPTER 5

Results and conclusion

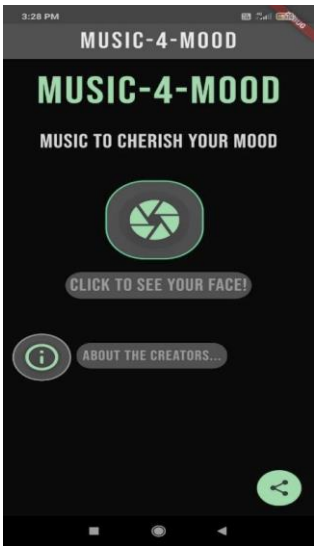
5.1 Results

The following are the result pages of our application (Application page).

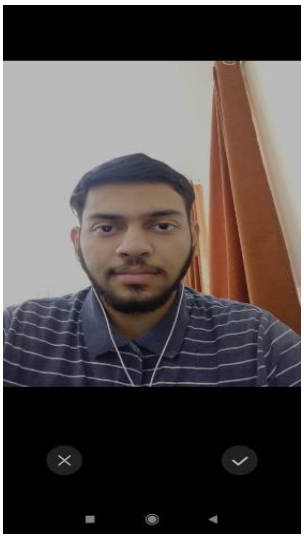
Opening screen (5.1.a)



Home screen (5.1.b)



Capturing user face (5.1.c)



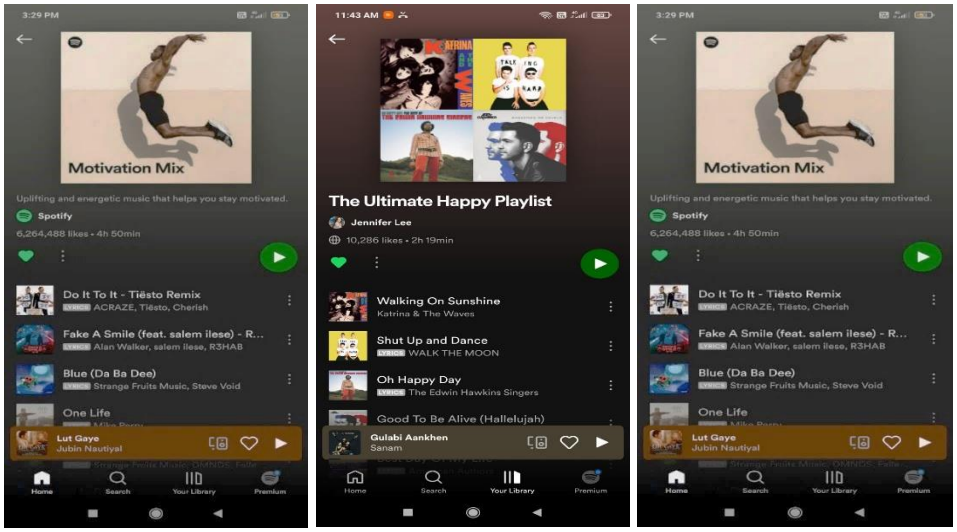
Neutral

Sad

Happy

Playlist recommended on Spotify (5.1.d)

Neutral Happy Sad



Here is the outcome of our application. The process runs smoothly that starts with 5.1.a where the application starts. 5.1.b is the home screen for our application. 5.1.c captures images through camera. We have taken 3 sample cases to predict the facial emotion for recommending song playlist. 5.1.d depicts the playlist recommended by the ML model based on the emotion predicted by our ML model. There are 3 playlists Motivational mix, The ultimate Happy playlist, Motivational mix 2.

[1]

Algorithm and Features	Number of Emotions	Accuracy
CNN Network: FENet/ FeNet with BNLayer No. of Iterations: 5000->35000	7	87.35

Table. No. 5.1.1 Accuracy table of research paper

Fig. No. 5.1.1 showcases the accuracy chart of ML model used in the research paper. The number of emotions were 7 and accuracy achieved was 87.35 %. They had used Convolutional Neural Network with FENet algorithm with BNLayer.

The CNN algorithm used to predict the emotion showed the following results:

Algorithm and Features	Number of Emotions	Accuracy
CNN Network: MobileNet	3	90.50

Table. No. 5.1.2 Accuracy table of our ML model

Table. No. 5.1.2 depicts the accuracy we have achieved in our ML model for Facial emotion recognition. We have used CNN and MobileNet algorithms to train the model. The number of emotions are 3 mainly happy, sad, neutral.

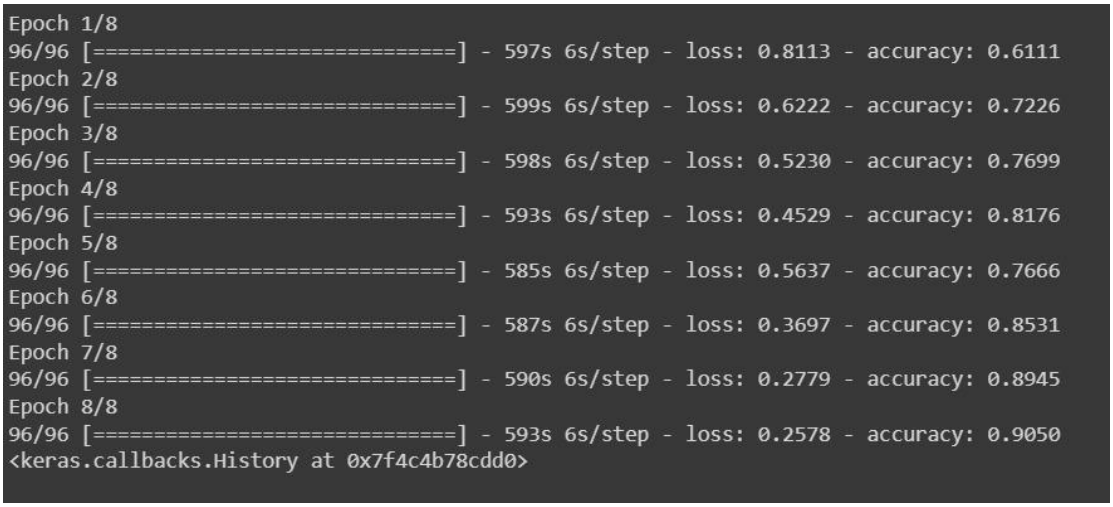


Fig. No. 5.1.2 Accuracy of ML model

Fig. No. 5.1 Depicts the accuracy of our ML MobileNetV2 model. The accuracy showed is 90.50%. The 10% error is due to lack of dataset to train the model. The data set used for training our model consisted of 3000 images.

5.2 Conclusion

We successfully implemented and developed an android and cross platform mobile application for recommending music playlist on Spotify by detecting user face and predicting the present emotion. A customized and condition-based music was the recommended that enriched the user experience. A simple user-friendly UI helped user navigate through the app with utmost ease. Results predicted by ML proved to be reliable for recommending accurate song playlist.

5.2 Future Scope

Currently, the ML model predicts three different emotions mainly happy, sad, and neutral. 7 emotions are showcased by humans, we intend to develop an ML model that predicts all 7 emotions with perfect accuracy for each and every emotion. We also intend to prepare an API to set up the backend with different technologies to make our application robust. The main userbase for our application are common people, as this app finds use on daily basis, so publishing the application in the local market would certainly add greater value.

References

[1] *Raut, N., 2018. Facial Emotion Recognition Using Machine Learning*.

[2] *Wasilewski, K. and Zabierowski, W., 2021. A Comparison of Java, Flutter and Kotlin/Native Technologies for Sensor Data-Driven Applications. Sensors, 21(10), p.3324.*