EMOTION BASED MUSIC PLAYBACK SYSTEM

A PROJECT REPORT

Submitted by

SURYA V

(2019202057)

submitted to the Faculty of

INFORMATION AND SCIENCE TECHNOLOGY

in partial fulfillment for the award of the degree
of

MASTER OF COMPUTER APPLICATIONS

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY

CHENNAI 600 025

MAY 2022

ANNA UNIVERSITY

CHENNAI - 600 025

BONAFIDE CERTIFICATE

Certified that this project report titled **"EMOTION BASED MUSIC PLAYBACK SYSTEM"** is the bonafide work of **SURYA V (2019202057)** who carried out project work under my supervision. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on this or any other candidate.

PLACE: CHENNAI
DATE: 31.05.22

MS. T. SINDHU
TEACHING FELLOW
INTERNAL GUIDE
DEPARTMENT OF IST, CEG
ANNA UNIVERSITY
CHENNAI 600025

COUNTERSIGNED

Dr.S. SRIDHAR
HEAD OF THE DEPARTMENT
DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING, GUINDY
ANNA UNIVERSITY
CHENNAI 600025

ABSTRACT

Human expression plays a vital role in determining the current state and mood of an individual, it helps in extracting and understanding the emotion that an individual has based on various features of the face such as eyes, cheeks, forehead or even through the curve of the smile. Music is basically an art form that soothes and calms human brain and body. Taking these two aspects and blending them together our project deals with detecting emotion of an individual through facial expression and playing music according to the mood detected that will alleviate the mood or simply calm the individual and can also get quicker song according to the mood, saving time from looking up different songs and parallel developing a web application that can be used with the help of providing the functionality of playing music according to the emotion detected. The user would not have to waste any time in searching or to look up for songs and the best track matching the user's mood is detected, and songs would be shown to the user according to his/her mood. The image of the user is captured with the help of a webcam and then the image is processed to input into the CNN model that has been trained and exported. The songs are labelled using audio feature and then classified using machine learning algorithm according to moods. The user's picture is taken and then as per the mood/emotion of the user an appropriate list of songs is fetched, displayed and played.

Key Words : Emotion recognition, CNN, Music emotion recognition, Classification, Computer vision.

ACKNOWLEDGEMENT

Its my privilege to express my sincere thanks to my project guide **Ms. T. Sindhu**, Teaching Fellow, Department of Information Science and Technology, College of Engineering, Guindy, Anna University, Chennai for her keen interest, inspiring guidance, constant encouragement and support with my work during all the stages, to bring this thesis into fruition.

I deeply express my sincere thanks to **Dr.S.Sridhar**, Professor and Head of the Department, Department of Information Science and Technology, College of Engineering, Guindy, Anna University, Chennai for extending support.

I would like to express my sincere thanks to the project committee members, **Dr.Saswati Mukherjee**, Professor, **Dr.M.Vijayalakshmi**, Associate Professor, **Dr.E.Uma**, Assistant Professor, Department of Information Science and Technology, Anna University, Chennai for giving their valuable suggestions, encouragement and constant motivation throughout the duration of my project.

SURYA V

TABLE OF CONTENTS

LIST OF FIGURES

LIST SYMBOLS OF ABBREVIATIONS

CNN	Convolutional Neural Network
MER	Music Emotion Recognition
CSV	Comma Separated Value
JSON	Javascript Object Notation
API	Application Programming Interface
HTTP	Hyper Text Transfer Protocol
XML	Extensible Markup Language

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Project Emoplayer [Emotion Based Music Player] is a novel approach which helps user to play the songs according to their emotion / mood. In web application we can apply many features . This application is made for web users only. Firstly user have to click lets start to access the camera application and then our trained cnn model will predict the user emotion and redirect towards the list of songs based on the emotion. user can access this application on desktop too. To access in desktop user need windows7 or further version. The songs are collected from spotify and using audio features it is classified on emotions and saved to json file. The audio is played using in-build web player which fetches the song from spotify.

1.2 MOTIVATION AND PROBLEM STATEMENT

Music listeners have tough time creating and segregating the play - list manually when they have hundreds of songs. It is also difficult to keep track of all the songs sometimes songs that are added and never used , wasting a lot of device memory and forcing the user to find and delete songs manually. User's have to manually select songs every time based on interest and mood. User's also have difficulty to re-organize and playing music when play-style varies. Currently in existing application , music is organized using playlist and different types of moods. When user will click to their current type of mood it might happen that they don't like the play – list created by the developer because here developer is also playing the role of user so it's hard to define that's what kind of play -list user would like .As a music lover I thought to give a effort less music player like emotion recognition to play music

1.3 DEEP LEARNING TECHNIQUES

Deep learning is the subset of machine learning composed of algorithms that permits the machine to train itself and perform a task. These algorithms uses multiple layers to progressively extract the high level feature from the raw input(images, audio, video). In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation. Most modern deep learning models are based on (CNNs). A (CNN) is an algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. It is able to successfully capture the Temporal dependencies in an image through the application of relevant filters. The role of the CNN is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction.

1.4 ORGANIZATION OF THE REPORT

The thesis is organized into 6 chapters, describing each part of the project with detailed illustration and system design diagrams. The chapters are as follows:

Chapter 1: consists of Introduction, Problem statement, Motivation and Objectives etc.

Chapter 2: consists of Literature survey details of the project alongside their detailed methodologies, advantages, disadvantages etc.

Chapter 3: : consist System design of the project with its preliminary design such as overall Architecture diagram and process flow diagram which tells about the modules integration in the project.

Chapter 4: consists of Detailed system design or module description with their input and algorithmic steps involved in each module to derive the output as per the user requirement.

Chapter 5: consists the details about the hardware and software requirement to the project and the experiments that has been performed along with their outcomes. The detailed result of the project is also portrayed in this chapter.

Chapter 6: conclude the project report with all the results and implementation procedure that has been underwent during the project development. The future works and excellence of implemented project is detailed.

The above mentioned six modules are followed up with the references which deliberately explains and list all the reference documents used during the various phases of the project, which includes the journal papers, conference papers, white papers, articles and websites referred for tutorials.

CHAPTER 2

LITERATURE SURVEY

2.1 EMOTION DETECTION

Florence, S. Metilda, and M. Uma. ”Emotional Detection and Music Recommendation System based on User Facial Expression.” IOP Conference Series: Materials Science and Engineering. Vol. 912. No. 6. IOP Publishing, 2020. where the proposed system can detect the facial expressions of the user and based on his/her facial expressions extract the facial landmarks, which would then be classified to get a particular emotion of the user. Once the emotion has been classified the songs matching the user’s emotions would be shown to the user. It could assist a user to make a decision regarding which music one should listen to helping the user to reduce his/her stress levels. The user would not have to waste any time in searching or to look up for songs. The proposed architecture contained three modules, namely, Emotion extraction module, Audio extraction module and Emotion-Audio extraction module. Although it had some limitations like the proposed system was not able to record all the emotions correctly due to the less availability of the images in the image dataset being used. The image that is fed into the classifier should be taken in a well-lit atmosphere for the classifier to give accurate results. The quality of the image should be at least higher than 320p for the classifier to predict the emotion of the user accurately. Handcrafted features often lack enough generalizability in the wild settings.

2.2 ROBERT THAYERS MOOD MODEL

In most existing methods of music mood classification, the moods of songs are divided according to psychologist Robert Thayer's traditional model of mood. The model divides songs along the lines of energy and stress, from happy to sad and calm to energetic, respectively (Bhat et al 359). The eight categories created by Thayer's model include the extremes of the two lines as well as each of the possible intersections of the lines (e.g. happy-energetic or sad-calm).Faster tempos are associated with high-energy songs, and slower tempos with lower energy, sadder songs. Loudness, or intensity of a song can be connected with anger, while softer songs would suggest tenderness, sadness, or fear (Bhat et al 360). Higher overall pitch can be an indicator of happiness, carefree and light moods within a song, while lower pitch implies a darker, sad, and serious tone. Timbre, the tonal component of a piece created by harmonics, is a curious indicator of mood. According to a group of researchers from the BNM Institute of Technology in Bangalore, India, "timbre stimulates human energy levels without regard to rhythmic or harmonic saturation. Sound sources that have simple harmonic profiles have 'darker' timbres and tend to soothe human emotions" (Bhat et al 360). This same group of researchers produced a correlation table of intensity, timbre, pitch, and rhythm in identifying various moods

2.3 HAAR CASCADE ALOGRITHM

M. N. Chaudhari, M. Deshmukh, G. Ramrakhiani and R. Parvatikar, "Face Detection Using Viola Jones Algorithm and Neural Networks,"(2018). It is an machine learning algorithm to categorize objects in an captured image. It is mainly used for object detection.The cascade classifier has different stages of collection which resembles from weak learners. These weak classifiers are the simplest form classifiers that have a name called boosting.If the label ranges in positive state then it goes to the next stage showing the result.These have a positive side and a negative side where they identify the images according to the labels.These have a set of positive images over negative images on various stages. As images with higher resolution has greater quantity are preferred as better quality results.

CHAPTER 3

SYSTEM DESIGN

3.1 SYSTEM ARCHITECTURE

This module consists system design of the project with its preliminary design such as overall architecture diagram and process flow diagram which tells about the modules integration in the project.

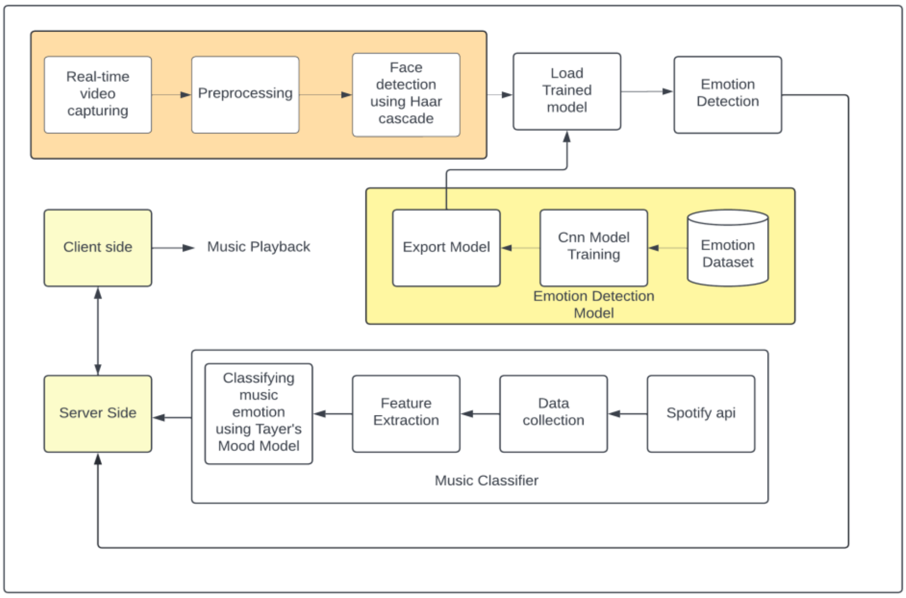


Figure 3.1: Overall Architecture of Proposed System.

Figure 3.1 depicts the overall system architecture. The proposed system works on Emotion based music playback. Emotion dataset are labelled and categorised with 5 labels such as happy, sad, neutral, angry, surprise. Then they are loaded to the CNN model for training the dataset and then export the model. For detecting face we use haar cascade classifier which detect face of a person and then the cropped face image is passed to trined model to predict the emotion of the user.To classify music, data is collected from spotify api and then labelled based on tayer mood model and using classifier like random forest to predict the mood of the song. And then the song is used to map with the appropriate emotions and play music in the website.

3.2 TRAINING DATASET

For Emotion recognition, emotion image datasets are collected from the Kaggle named FER(Facial Emotion Recognition). The data consists of 48x48 pixel gray scale images of faces. The faces have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. The facial expression is categorized into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral).

3.3 DATA PRE-PROCESSING

Dataset preprocessing includes the image augmentation of the dataset. Image augmentation is a process of creating new training examples from the existing ones. To make a new sample, you slightly change the original image. For instance, you could make a new image a little brighter; you could cut a piece from the original image; rotate, scale, shift etc.. you could make a new image by mirroring the original one, etc

3.4 FEATURE EXTRACTION

We build our own CNN model to extract feature from image layer by layer. For this purpose we have used four blocks of convolutional layers with size doubling every layer from 32. Each layer has pooling layer of size of 3*3 with max-pooling. Activation Function used is ‘ELU’. It has two dense or fully-connected layer with 2304 nodes. Finally we have final dense layer or output layer with 5 nodes classifying happy, sad, angry etc..

3.5 WEB DEVELOPMENT

We build a web interface for the user to access via graphical user interface. Initially the user will have to initiate Web cam to capture the video of the user and the video is preprocessed to detect face and predict the emotion using model file stored in the server running on flask. And then the page is redirected to the playlist page based on emotion and it retrieves list of songs from the classified song data and render the content on the html page.

3.6 MUSIC EMOTION RECOGNITION

to classify song based on emotion we to acquire the data from spotify. Firstly retrieve songs from playlist using spotify api and metadata of each song. Pre-process data to remove unwanted and null entries. Label data using Tayer’s mood model and group songs. Split the data to train and test the model using classifier like random forest etc.. and then the model is evaluated and the output is stored in JSON format.

3.7 FLOW-CHART DESIGN

Figure.3.2 shows the prediction workflow and design of the proposed system.

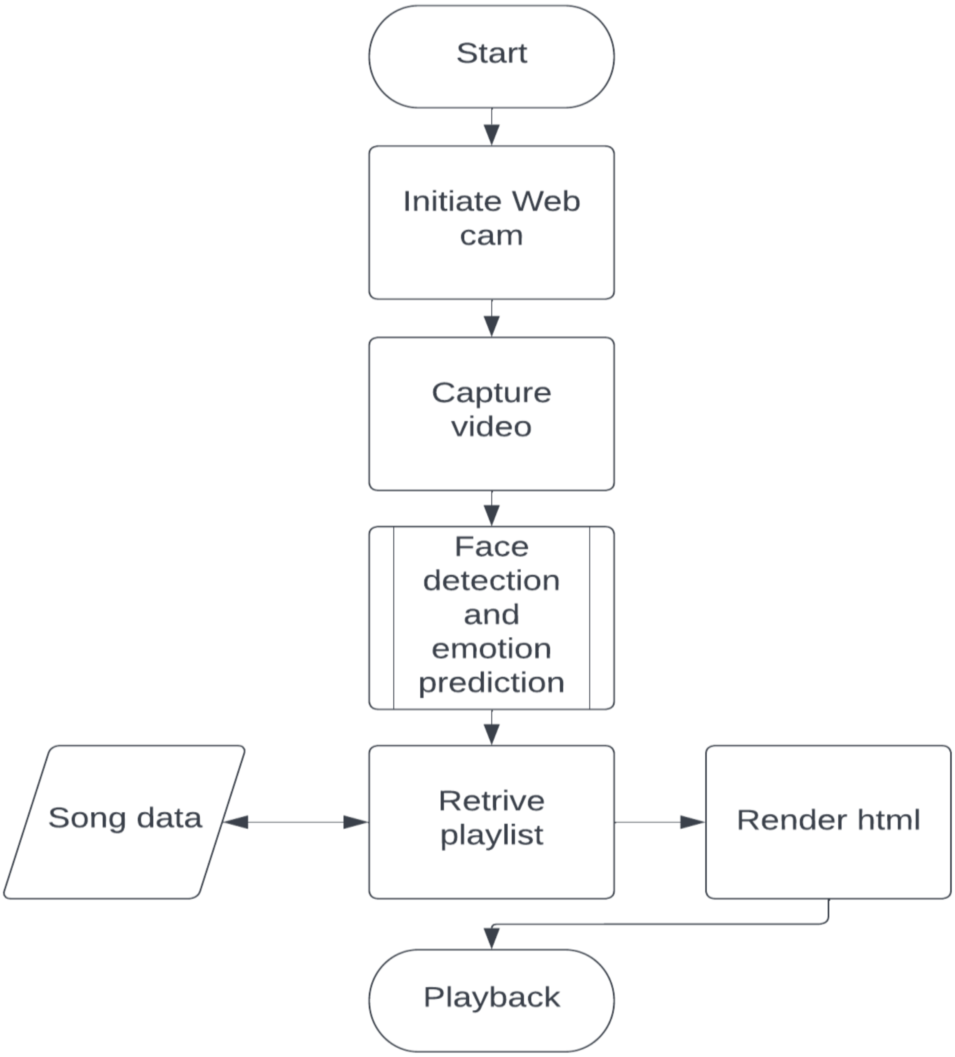


Figure 3.2: Workflow of the proposed system

CHAPTER 4

ALGORITHM IMPLEMENTATION

4.1 FACE CAPTURING

The main objective of this session is to capture images so here we are using the common device i.e, webcam. for that purpose we are using the computer vision library.This makes it easier to integrate it with other libraries which can also use NumPy and it is mainly used as a real time computer vision.In the initial process when execution starts it starts to access the camera stream and captures images for further process and emotion detection.so, in the initial phase of this project in order to capture the images and face detection. We use an haar cascade algorithm which is a pretrained face detection model that could take the images and process and display face detected if it detect face. Then the image is cropped and made input to the CNN model.

4.2 MODEL DETAILS

We build our own CNN model to extract feature from image layer 8 by layer. For this purpose we have used four blocks of convolutional layers with size doubling every layer from 32. Each layer has pooling layer of size of 3*3 with max-pooling. Activation Function used is ‘ELU’. It has two dense or fully-connected layer with 2304 nodes. Finally we have final dense layer or output layer with 5 nodes classifying happy, sad, angry etc

4.2.1 Sequential Layer

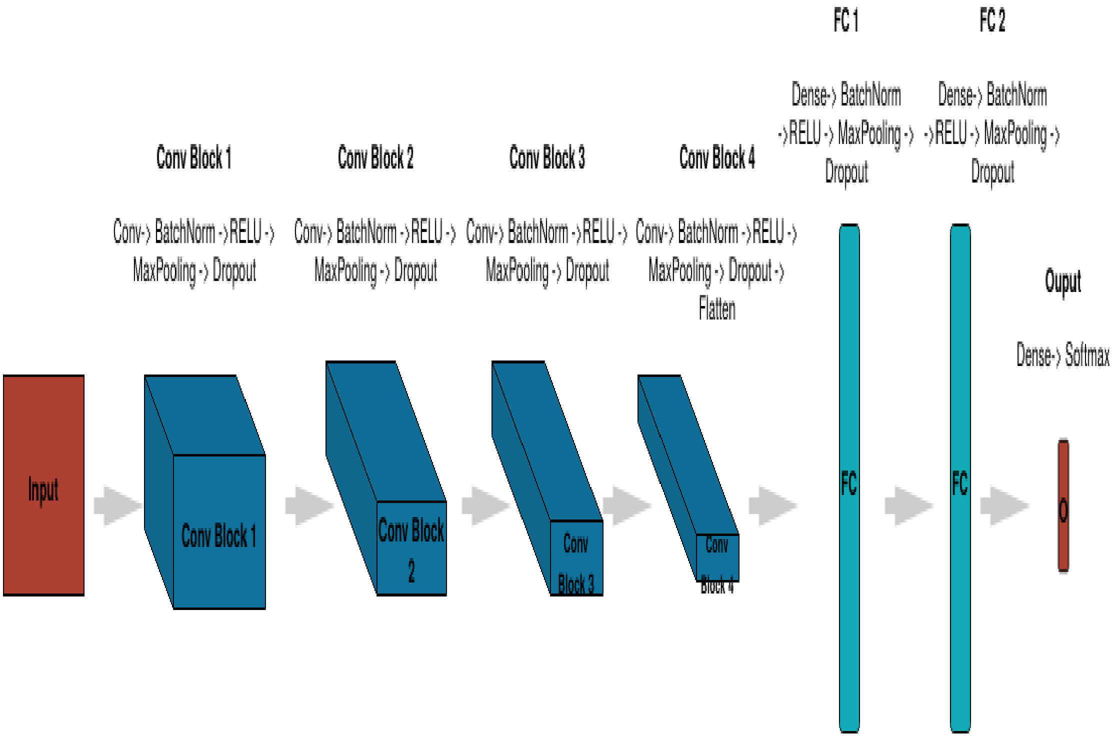


Figure 4.1: CNN Model Architecture

Sequential is a container of Modules that can be stacked together and run at the same time. Sequential layer is used to store feature vector returned by the ResNext model in a ordered way. So that it can be passed to the LSTM sequentially.

4.2.2 Pooling Layer

The pooling operation involves sliding a two-dimensional filter over each channel of feature map and summarising the features lying within the region covered by the filter. For a feature map having dimensions nh x nw x nc, the dimensions of output obtained after a pooling layer is (nh - f + 1) / s x (nw - f + 1)/s x nc

4.2.3 Relu Activation Function

A Rectified Linear Unit is activation function that has output 0 if the input is less than 0, and raw output otherwise. That is, if the input is greater than 0, the output is equal to the input. The operation of ReLU is closer to the way our biological neurons work. ReLU is non-linear and has the advantage of not having any backpropagation errors unlike the sigmoid function, also for larger Neural Networks, the speed of building models based off on ReLU is very fast. Figure 4.2 shows the Relu activation function.

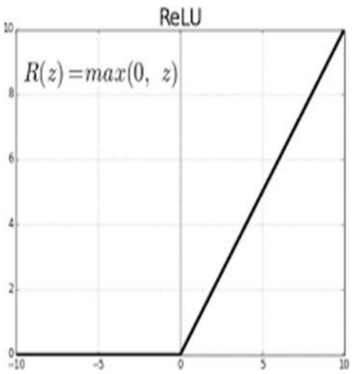


Figure 4.2: Relu Activation Function

4.2.4 Flatten Layer

After we insert convolutional operations and pooling, we would end up with multiple features and dimensions. We have to flatten out the dimensions and feature map into a single column. This single column is a long vector of input data that can be passed onto the ANN (Artificial Neural Network).

4.2.5 Fully Connected Layer

Fully Connected layers in a neural networks are those layers where all the inputs from one layer are connected to every activation unit of the next layer. In most popular machine learning models, the last few layers are full connected layers which compiles the data extracted by previous layers to form the final output. It is the second most time consuming layer second to Convolution Layer

4.2.6 Dropout Layer

Dropout layer with the value of 0.4 is used to avoid overfitting in the model and it can help a model generalize by randomly setting the output for a given neuron to 0. In setting the output to 0, the cost function becomes more sensitive to neighbouring neurons changing the way the weights will be updated during the process of back propagation.

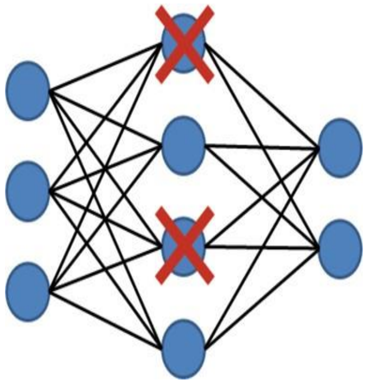


Figure 4.3: Relu Activation Function

4.3 MODEL TRAINING

We build our own CNN model to extract feature from image layer 8 by layer. For this purpose we have used four blocks of convolutional layers with size doubling every layer from 32. Each layer has pooling layer of size of 3*3 with max-pooling. Activation Function used is 'ELU'. It has two dense or fully-connected layer with 2304 nodes. Finally we have final dense layer or output layer with 5 nodes classifying happy, sad, angry etc

4.3.1 Train Test Split

The dataset is split into train and test dataset with a ratio of 70percent 24282 train samples and 30percent of 5937 test samples.

4.3.2 Training

The training is done for 30 epochs with a learning rate of 1e-3 (0.0010),weight decay of 1e-3 (0.0010) using the Adam optimizer.

4.3.3 Adam Optimizer

Adam is an optimization solver for the Neural Network algorithm that is computationally efficient, requires little memory, and is well suited for problems that are large in terms of data or parameters or both. Adam is a popular extension to stochastic gradient descent. To enable the adaptive learning rate Adam optimizer with the model parameters is used.

4.3.4 Cross Entropy

To calculate the loss function Cross Entropy approach is used because we are training a classification problem.

4.3.5 Softmax Layer

A Softmax function is a type of squashing function. Squashing functions limit the output of the function into the range 0 to 1. This allows the output to be interpreted directly as a probability. Similarly, softmax functions are multi-class sigmoids, meaning they are used in determining probability of multiple classes at once. Since the outputs of a softmax function can be interpreted as a probability (i.e.they must sum to 1), a softmax layer is typically the final layer used in neural network functions. It is important to note that a softmax layer must have the same number of nodes as the output later. In our case softmax layer has 5 output nodes i.e Happy, Angry, Sad, Neutral, Surprise, also Softmax layer provide us the confidence(probability) of prediction.

4.3.6 Export Model

After the model is trained, we have exported the model to the flask server running on the local machine. So that it can be used for prediction on real time data.

4.3.7 Model prediction

The model is loaded in the application. The new video for prediction is preprocessed and passed to the loaded model for prediction. The trained model performs the prediction and return if the image is a happy,sad,angry,neutral,surprise along with the confidence of the prediction.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 WEB APPLICATION DASHBOARD MODULE

Figure 5.1 depicts the Dashboard screen provided. It is running on the localhost with flask server.

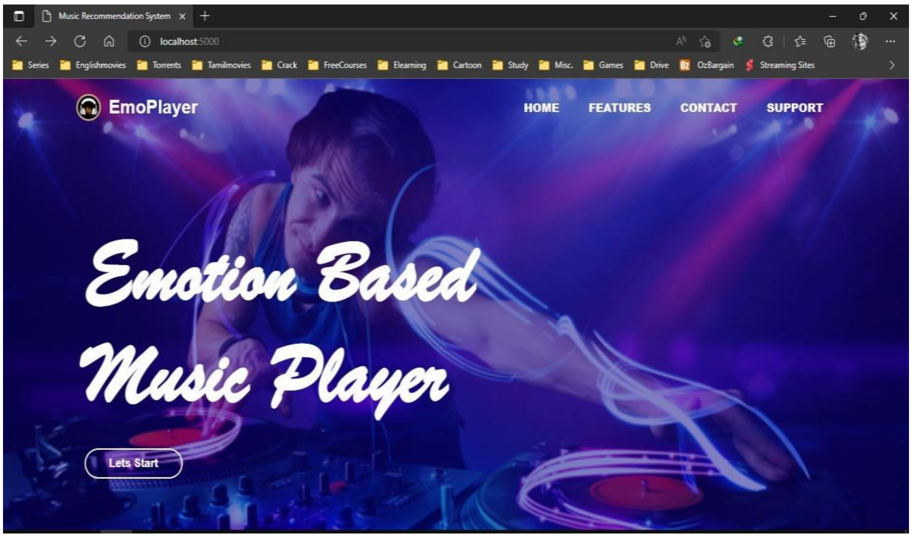
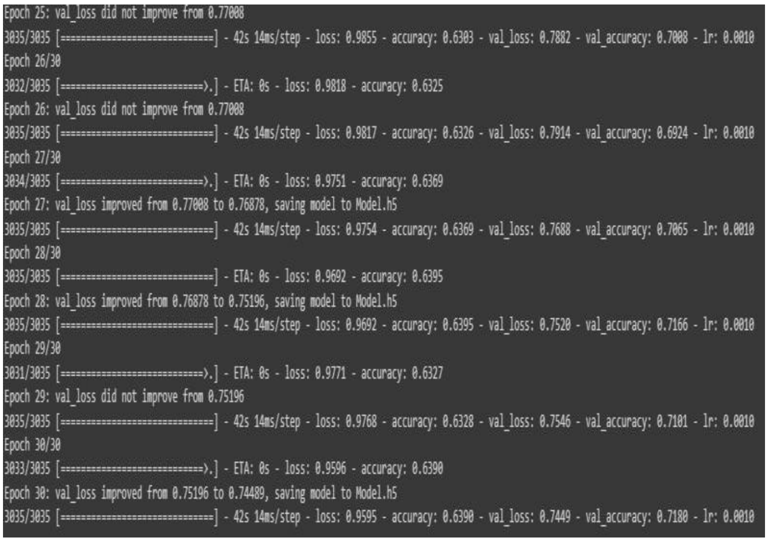


Figure 5.1: Web Application Dashboard Screen

5.2 TRAINING AND TESTING ANALYSIS

Figure 5.2 depicts the model summary and accuracy result. The whole dataset contains 29000 labelled image. The Dataset is divided as 80percent for training and the remaining consider as the testing dataset. Validation accuracy and loss with 30 epoch is 71.8percent and 74percent.



```
Epoch 25: val_loss did not improve from 0.77008
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9855 - accuracy: 0.6303 - val_loss: 0.7882 - val_accuracy: 0.7008 - lr: 0.0010
Epoch 26/30
3032/3035 [============================>.] - ETA: 0s - loss: 0.9818 - accuracy: 0.6325
Epoch 26: val_loss did not improve from 0.77008
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9817 - accuracy: 0.6326 - val_loss: 0.7914 - val_accuracy: 0.6924 - lr: 0.0010
Epoch 27/30
3034/3035 [============================>.] - ETA: 0s - loss: 0.9751 - accuracy: 0.6369
Epoch 27: val_loss improved from 0.77008 to 0.76878, saving model to Model.h5
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9754 - accuracy: 0.6369 - val_loss: 0.7688 - val_accuracy: 0.7065 - lr: 0.0010
Epoch 28/30
3035/3035 [==============================] - ETA: 0s - loss: 0.9692 - accuracy: 0.6395
Epoch 28: val_loss improved from 0.76878 to 0.75196, saving model to Model.h5
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9692 - accuracy: 0.6395 - val_loss: 0.7520 - val_accuracy: 0.7166 - lr: 0.0010
Epoch 29/30
3031/3035 [============================>.] - ETA: 0s - loss: 0.9771 - accuracy: 0.6327
Epoch 29: val_loss did not improve from 0.75196
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9768 - accuracy: 0.6328 - val_loss: 0.7546 - val_accuracy: 0.7101 - lr: 0.0010
Epoch 30/30
3033/3035 [============================>.] - ETA: 0s - loss: 0.9596 - accuracy: 0.6390
Epoch 30: val_loss improved from 0.75196 to 0.74489, saving model to Model.h5
3035/3035 [==============================] - 42s 14ms/step - loss: 0.9595 - accuracy: 0.6390 - val_loss: 0.7449 - val_accuracy: 0.7180 - lr: 0.0010
```

Figure 5.2: Training and Testing

5.3 FACE DETECTION

Face is detected using HAAR cascade classifier and bounding box is draw in the region of interest.

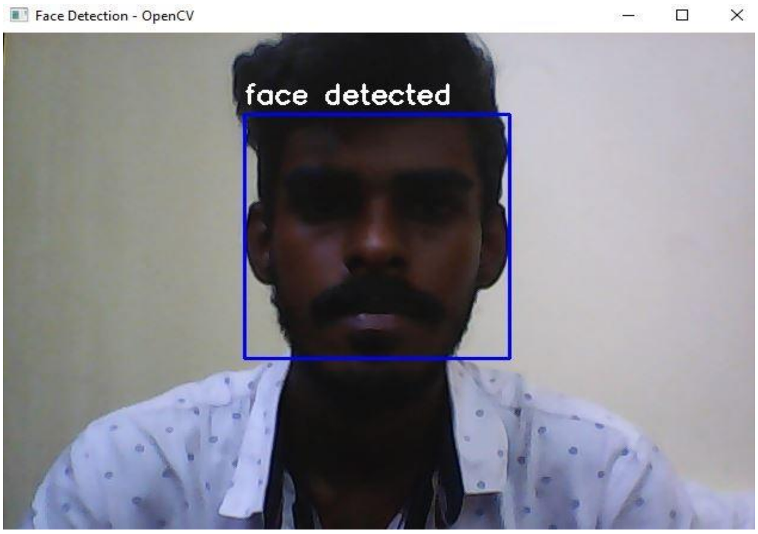


Figure 5.3: Face detection

5.4 EMOTION DETECTION

Emotion of the person is detected and displayed in real time by making bounding box around the face.

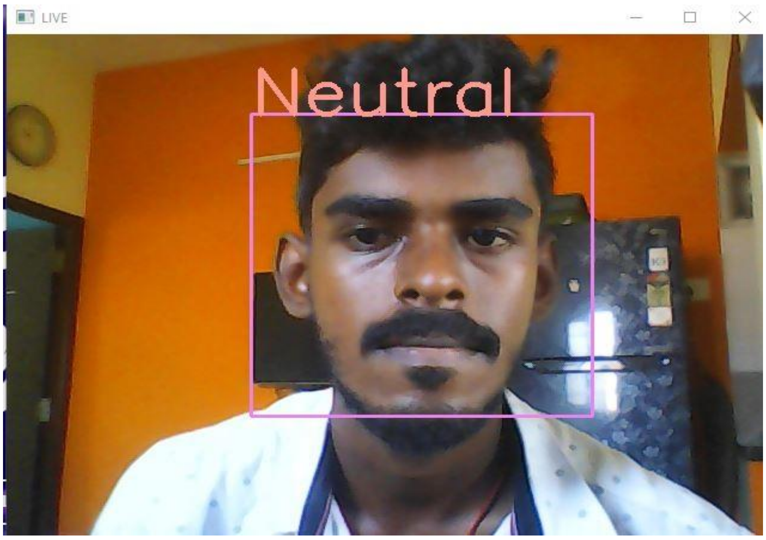


Figure 5.4: Neutral Face

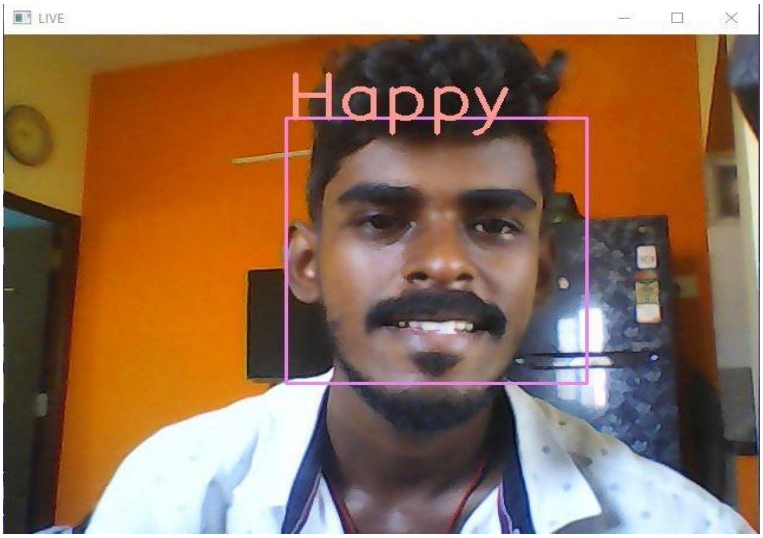


Figure 5.5: Happy Face

Figure 5.6: Angry Face

REFERENCES

10

[1] Rod Johnson, Juergen Hoeller, Keith Donald, Colin Sampaleanu, Rob Harrop, Thomas Risberg, Alef Arendsen, Darren Davison, Dmitriy Kopylenko, and Mark Pollack. ”The spring framework–reference documentation”. *Interface, 21:27*, 2004.

[2] Angular developer guide. magenta`https://angular.io/docs`, 2010-2021.

[3] Brad Green and Shyam Seshadri. *AngularJS*. ” O'Reilly Media, Inc.”, 2013.

[4] Mike Keith and Merrick Schincariol. *Pro EJB 3: Java Persistence API*. Apress, 2006.

[5] N. A. Nik Ahmad and S. A. Syed Zamri. The cross platform application development adapted spring framework to support front-end tendering services. In *2014 International Conference on Computer, Communications, and Control Technology (I4CT)*, pages 58–62, 2014.

[6] D. Ying-kui, W. Yang, G. Ping, P. Yue, Z. LiJuan, and L. Shu. Cloud data monitoring management and visual application system based on spring boot. In *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, volume 1, pages 1143–1146, 2019.

[7] F. Paganelli, S. Turchi, and D. Giuli. A web of things framework for restful applications and its experimentation in a smart city. *IEEE Systems Journal*, 10(4):1412–1423, 2016.

[8] Chunsheng Zhao, Mai Jiang, and Zhiyong He. The design of e-commerce system architecture based on struts2, spring and hibernate. In *The 2nd International Conference on Information Science and Engineering*, pages 3251–3254, 2010.

[9] L. Li and W. Chou. Design and describe rest api without violating rest: A petri net based approach. In *2011 IEEE International Conference on Web Services*, pages 508–515, 2011.

[10] G Ann Campbell and Patroklos P Papapetrou. *SonarQube in action*. Manning Publications Co., 2013.

[11] Andy Hunt and Dave Thomas. *Pragmatic unit testing in Java with JUnit*. The Pragmatic Bookshelf, 2003.

[12] Sujoy Acharya. *Mastering unit testing using Mockito and JUnit*. Packt Publishing Ltd, 2014.