ABSTRACT

Every individual human has completely different faces, however their expression tells us the same story and it notably plays a significant role in extraction of an individual's emotion and behaviour. music is the purest form of art and medium of expression, which is known to have a greater connection with a person's emotion. Music plays an important role in our life we often listen to music based on the mood and situation and we create lot of playlists of it. Emotion based music system is a modern approach that provides the user a automatically generated playlist based on users mood and behaviour. This system itself selects songs according to the current mood of the user. This system has noble ability to lift one's mood. The proposed model will extract facial expressions of user to determine the current emotion by web cam Once emotion is detected, playlist of songs suitable to mood of the user will be presented to him. The system involves the image processing, facial detection processing, mood classification and data base collection.

KEYWORDS: Face recognition, Emotion and mood detection, Mood extraction, Computer vision, Deep learning techniques.

TABLE OF CONTENTS

CONTENTS	PAGE NO
Certificate	ii
Declaration	iii
Acknowledgement	iv
Vision & Mission, PEOs	v
POs and PSOs	vi
Course Objectives & Outcomes	vii
Abstract	viii
Table of Contents	ix
List of Figures	x
List of Tables	xi
List of Abbreviations	xi
Chapters	
1 Introduction	1 - 4
1.1 Problem statement	
1.2 Existing system	
1.3 Proposed system	
1.4 Scope of	
2 System Requirement Specifications	5 - 6
2.1 Software Requirements	
2.2 Hardware Requirements	
2.3 System Architecture / Block Diagram	
3 Design & Implementation	7 - 12
3.1 Design	
3.2 Environmental Setup	
4 Test & Results	13 - 17
1.1 Test cases	
1.2 Results	
5 Conclusion & Future Enhancements	18
References	19
6 Appendix A: source/pseudo code	20-21

List of Figures

Figure No.	Figure Name	Page No
1.1	Existing system	3
2.1	Flow chart of proposed system	6
3.2.1	Python installation	9
3.2.2	PyCharm download	10
3.2.3	PyCharm overview	11
3.2.4	Installation of modules	11
3.2.5	Streamlit application setup	12

List of Tables

Table No.	Table Name	Page No.
4.1	System functional testing results	14
4.2	Emotions saved in proposed system	15
4.3	Testing results for neutral expression	16
4.4	Testing results for sad expression	16
4.5	Testing results for surprise expression	17
4.6	Testing results for happy expression	17
4.7	Summary of results tested	18

CHAPTER 1

INTRODUCTION

Emotions are the bodily feelings associated with mood, temperament, personality, or character. Paul Ekman had developed the classifications of basic emotions which are anger, disgust, fear, happiness, sadness, and surprise in 1972. A facial expression can be expressed through the motions or from one or more motions, movements or even positions of the muscles of the face. These movements transmit of the emotional status of an individual. Facial expression can be adopted as voluntary action as individual can control his facial expression and to show the facial expression according to his will.

In today ‘s world, with the increasing advancements in the field of multimedia and technology, various music players have been developed with features like fast forward, reverse, variable playback speed, genre classification, streaming playback with multicast streams and including volume modulation, etc. These features might satisfy the user ‘s basic requirements, but the user has got to face the task of manually browsing the playlist of songs and choose songs supported their current mood and behaviour. Emotion based music player is approach that helps the user to automatically play songs according to the emotions of the user. It recognizes the facial emotions of the user and plays the songs according to their emotion. The webcam captures the image of the user. It then extracts the facial features of the user from the captured image. Facial expression categorized into happy, sad, neutral, surprised etc... The foremost concept of this project is to automatically play songs based on the emotions of the user. It aims to provide user-preferred music with respect to the emotions detected. Facial expression analysis includes both detection and interpretation of facial motion and recognition of expression. The three approaches which enabled the automatic facial expression analysis (AFEA) include:

i) face acquisition.

ii) facial data extraction and representation.

iii) facial expression recognition.

1.1 PROBLEM STATEMENT

The significance of music on an individual's emotions has been generally acknowledged. After the day’s toils and hard works, both the primitive and modern man able to relax and ease him in the melody of the music. Studies had proof that the rhythm itself is a great tranquilizer.

However, most people facing the difficulty of songs selection, especially songs that match individuals’ current emotions. Looking at the long lists of unsorted music, individuals will feel more demotivated to look for the songs they want to listen to. Most user will just randomly pick the songs available in the song folder and play it with music player. Most of the time, the songs played does not match the user’s current emotion.

To develop an application to recognize user emotion through facial recognition and provide songs on user based emotion.

1.2 EXISTING SYSTEM

Currently, there are many existing music player applications. Some of the interesting applications among them are:

Savaan and Spotify - These application gives good user accessibility features to songs and recommends user with other songs of similar genre.

Moodfuse - In this application user should manually enter mood and genre that wants to be heard and moodfuse recommends the songs-list.

Fig – 1.1 EXISTING SYSTEMS

1.3 PROPOSED SYSTEM

This proposed system majorly focuses on introducing a system that is resolves most of the drawbacks of the existing system and generates playlists on the basis of the current emotional state of the individual using the facial expression of a human face therefore reducing the time and labour of the human as otherwise it needs to be done manually and thus avoiding the employment of additional hardware which in turn results in reduction in design cost. The aim is to build a system or player that has the capacity to segregate the list of songs and generate playlist according to the emotional state of the individual automatically with the need to be monitored by the humans. And which can process the query fast enough that is it is computationally fast and returns very accurate and precise results proving much superior efficient than the existing system.

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.

1.4 SCOPE OF STUDY

Currently there is no commonly used application or system which able to detect the emotion of individual and play music according to the emotion detected. This system will propose a new lifestyle to all music lovers which will ease them when searching for playlists. The target users will be the music lovers. English will be the main medium of language used in the proposed model and specifically aimed to detect some basic emotion such as normal, happy, sad or surprise. The evaluation of this system will base on the accuracy in detecting the correct facial expression as well as playing the right category of songs.

The scope of study will be as follow:

Study on the different method in expression detection. With the improvement of technology in image processing, more and more experts did researches or introduced different technique in processing a specific area or small area on an image. All these techniques can be applied to the facial expression processing. Research had to be done in order to understand each technique which will then be useful in the project development.

Get information on the tools appropriate for the facial expression detection to build the proposed model for this project. Different tools (software and hardware) are studied on their feasibility and functionality as well as user friendliness in order to figure out the most suitable and applicable tools to develop it.

CHAPTER 2

SYSTEM REQUIREMENT SPECIFICATIONS

2.1 SOFTWARE REQUIREMENTS

i. Programming language

Python idle 3.10.7 2022

- Main development software which includes interface design and coding.

PyCharm community edition (from jet brains)

- To development of project and easy access to build a python project.

ii. streamlit web application:

- Streamlit is an open-source app framework in Python language.
- It helps us create web apps for data science Streamlit is an open-source app framework in Python language.
- It helps us create web apps for data science and machine learning in a short time.
- It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, NumPy, pandas, Matplotlib etc.

2.2 Hardware requirements

i. Personal Laptop (Lenovo ThinkPad)

Intel® Core (TM) i5-7300U CPU @ 2.60GHz 2.71

4 Gb RAM

ii. Build in webcam/External Webcam

To capture the user's facial expression.

iii. Operating System

Windows 10 pro

2.3 System architecture

Flow chart

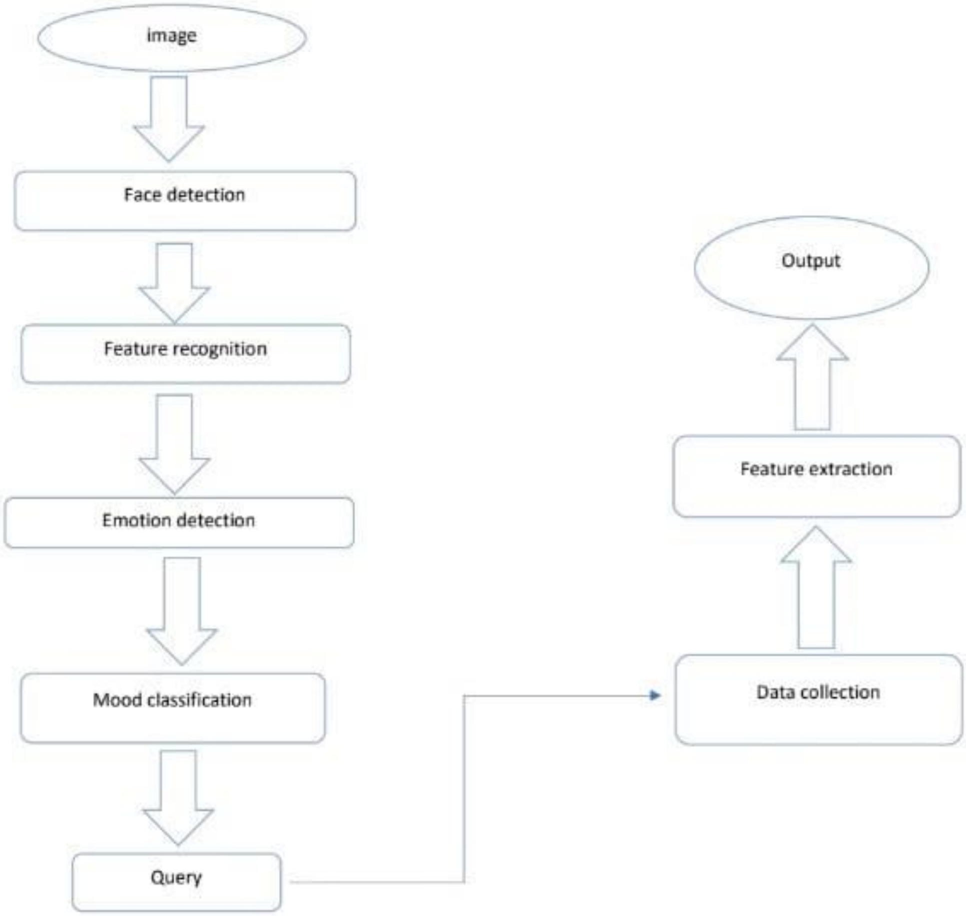


Fig -2.1 - FLOW CHART OF PROPOSED SYSTEM

CHAPTER 3

Design & Implementation

3.1 Features

Face recognition module

Facial recognition is a way of identifying or confirming an individual's identity using their face. Facial recognition systems can be used to identify people in photos, videos, or in real-time. Facial recognition is a category of biometric security. Other of biometric software include voice recognition, fingerprint recognition, and eye retina or iris recognition. The technology is mostly used for security and law enforcement, though there is increasing interest in other areas of use. Many people are familiar with face recognition technology through the FaceID used to unlock iPhones (however, this is only one application of face recognition). Typically, facial recognition does not rely on a massive database of photos to determine an individual's identity — it simply identifies and recognizes one person as the sole owner of the device, while limiting access to others. Facial technology systems can vary, but in general, they tend to operate as follows: Face detection, Face analysis, Converting the image to data, Finding a match.

Emotion classification module

Emotion classification the means by which one may distinguish or contrast one emotion from another, is a contested issue in emotion research and in affective study . Researchers have approached the classification of emotions from one of two fundamental viewpoints:

Sadness: The eyelids droop while the inner corners of the brows rise.

Surprise: Both the upper eyelids and brows rise, and the jaw drops open.

Anger: Both the lower and upper eyelids squeeze in draw together. The jaw pushes forward while lip pressed on each other.

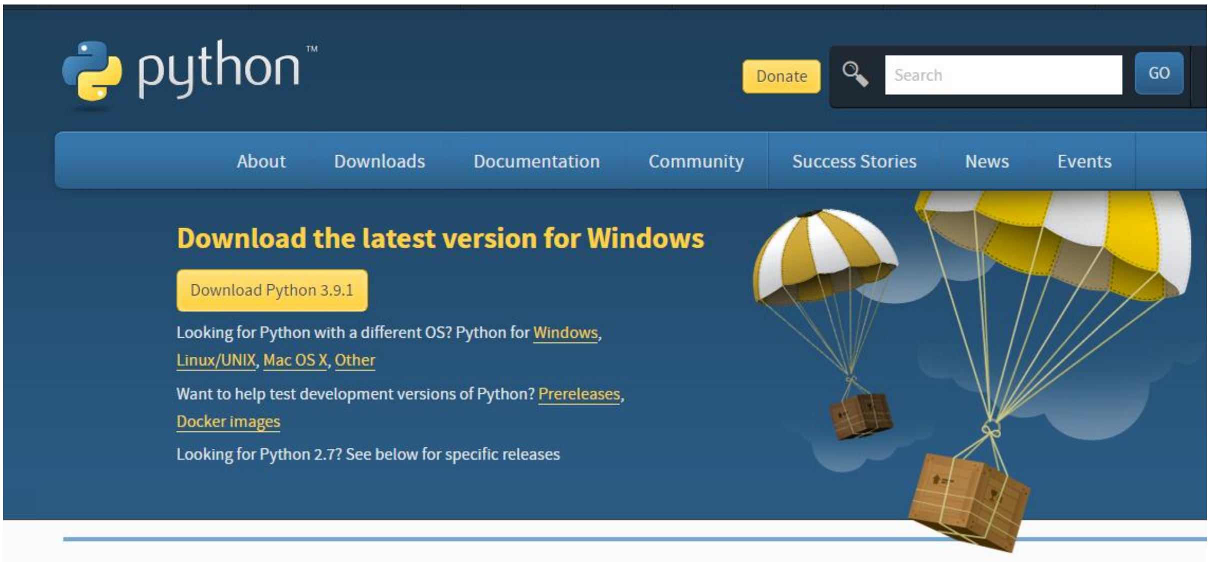
Fear: The eyes widen and the upper lids rise. The brows draw together while the lips extend horizontally.

Happiness: The corners of the lips lifted and shaped a smile, the cheeks rise up and the outside corners of the brows pull down.

3.2 Environmental setup

Installing Python:

1. To download and install Python visit the official website of Python https://www.python.org/downloads/ and choose your version.



Download the latest version for Windows

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Prereleases, Docker images

Looking for Python 2.7? See below for specific releases

Fig-3.2.1 PYTHON INSTALLATION

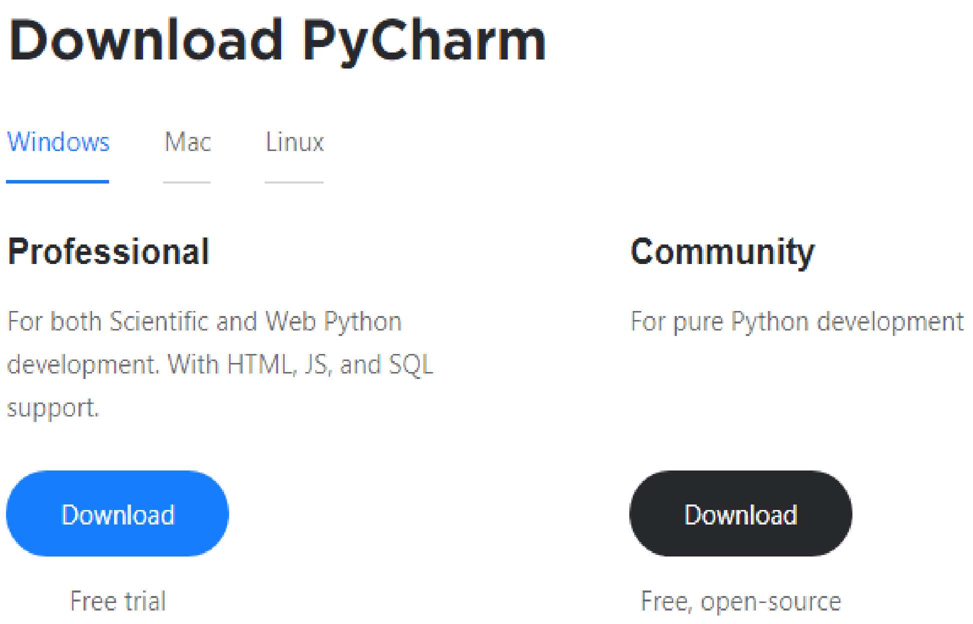
2. Once the download is complete, run the exe for install Python. Now click on Install Now.

3. You can see Python installing at this point.

4. When it finishes, you can see a screen that says the Setup was successful. Now click on "Close".

Installing PyCharm:

1. To download PyCharm visit the website https://www.jetbrains.com/pycharm/download/ and Click the "DOWNLOAD" link under the Community Section.



Download PyCharm

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

Free trial

Community

For pure Python development

Free, open-source

Fig-3.2.2 PYCHARM DOWNLOAD

2. Once the download is complete, run the exe for install PyCharm. The setup wizard should have started. Click “Next”.
3. On the next screen, Change the installation path if required. Click “Next”.
4. On the next screen, you can create a desktop shortcut if you want and click on “Next”.
5. Choose the start menu folder. Keep selected JetBrains and click on “Install”.
6. Wait for the installation to finish.
7. Once installation finished, you should receive a message screen that PyCharm is installed. If you want to go ahead and run it, click the “Run PyCharm Community Edition” box first and click “Finish”.
8. After you click on "Finish," the Following screen will appear.

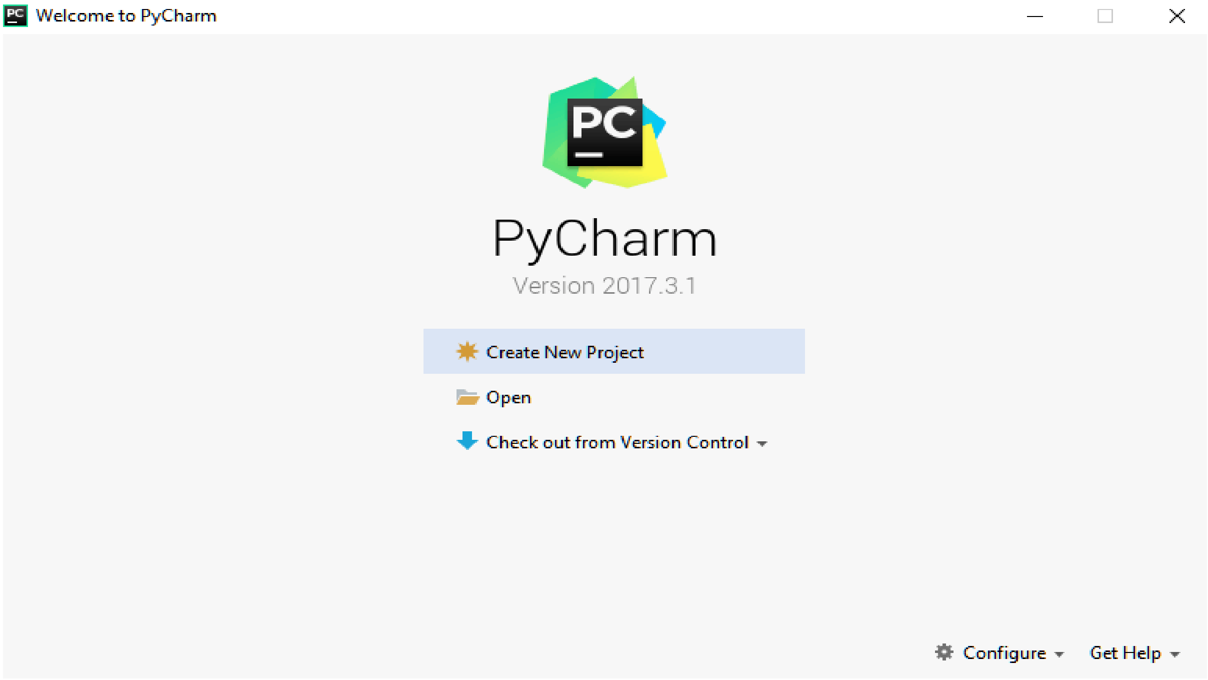
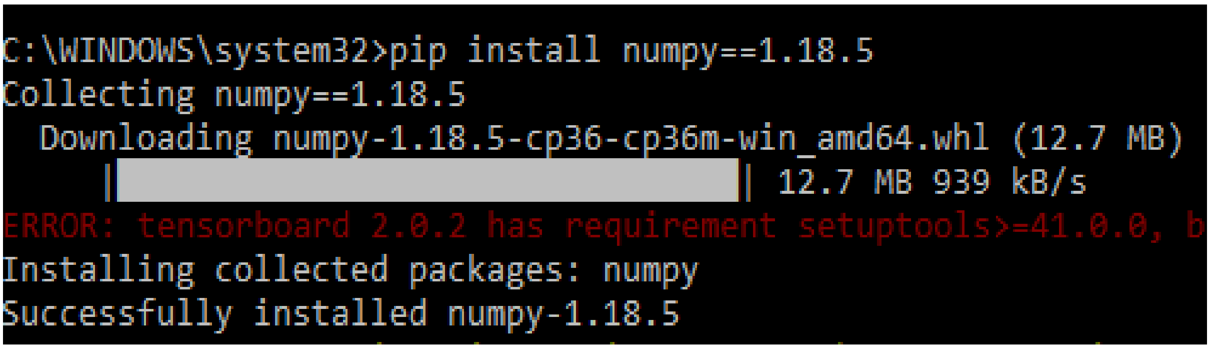


Fig-3.2.3 PYCHARM OVERVEIW

9. You need to install some packages to execute your project in a proper way.

10. Open the command prompt/ anaconda prompt or terminal as administrator.

11. The prompt will get open, with specified path, type “pip install package name” which you want to install (like NumPy, pandas, seaborn, scikit-learn, matplotlib. pyplot).

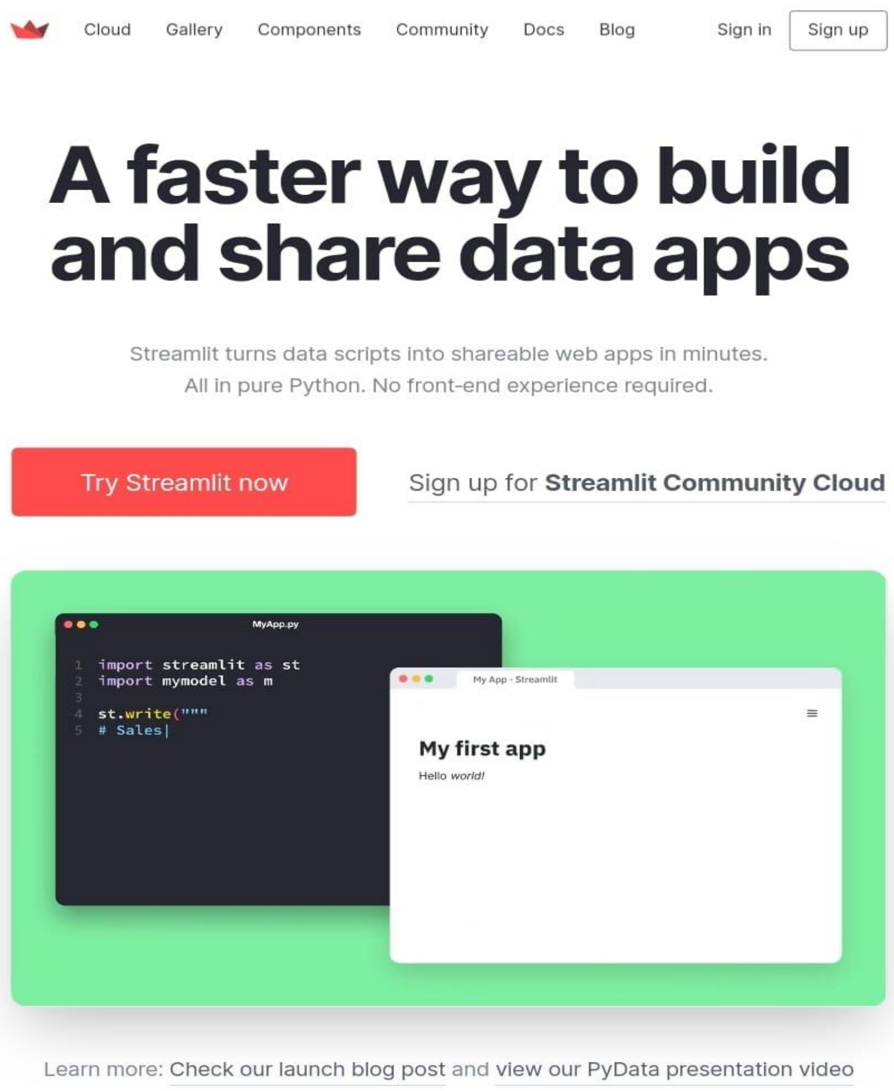


```
C:\WINDOWS\system32>pip install numpy==1.18.5
Collecting numpy==1.18.5
  Downloading numpy-1.18.5-cp36-cp36m-win_amd64.whl (12.7 MB)
     |████████████████████████████████| 12.7 MB 939 kB/s
ERROR: tensorboard 2.0.2 has requirement setuptools>=41.0.0, b
Installing collected packages: numpy
Successfully installed numpy-1.18.5
```

Fig-3.2.4 INSTALLATION OF MODULES

Setup of streamlit

Streamlit is an open-source app framework in Python language. It helps us create web apps for data science Streamlit is an open-source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, NumPy, pandas, Matplotlib etc.



A faster way to build and share data apps

Streamlit turns data scripts into shareable web apps in minutes.
All in pure Python. No front-end experience required.

```
import streamlit as st
import mymodel as m

st.write("""
# Sales|
```

My first app

Hello *world!*

Learn more: Check our launch blog post and view our PyData presentation video

Fig-3.2.5 STREAMLIT APPLICATION SETUP

The main thing to choose the code language as python is all features and functions are inbuilt and python is user understandable and user friendly and we have used PyCharm community edition to build this project as it is an efficient edition which works in python idle interpreter and for front end development for our project we have used the streamlit web application were this application also trains us to build any software related projects most the inbuilt function code is collected from the streamlit application itself and most importantly for front end application in streamlit there is no use of javascript,css,html all the code and functions are defined in python itself the streamlit turn the data scripts into sharable webpages.

Implementation

1. System

- **1.1 Capture Images:** Capture the images using OpenCV, webrtc platform
- **1.2 Train Images:**
    - The data can't be trained by the unknown users. The authentication process is implemented here where the training accessibility will be held with only the owner of the application.

2.User:

- **2.1 Enter language:** The user needs to enter his/her preferred language.
- **2.2 Enter singer name:** The user needs to enter his/her preferred singer name.
- **2.3 Capture Images:** The user requests the application to capture images of his/her face without disguise.
- **2.4 View Results:**

  The captured image detects the mood and apply all the inputs and send an query to web browser then the info gets forwarded to YouTube environment and display the songs.

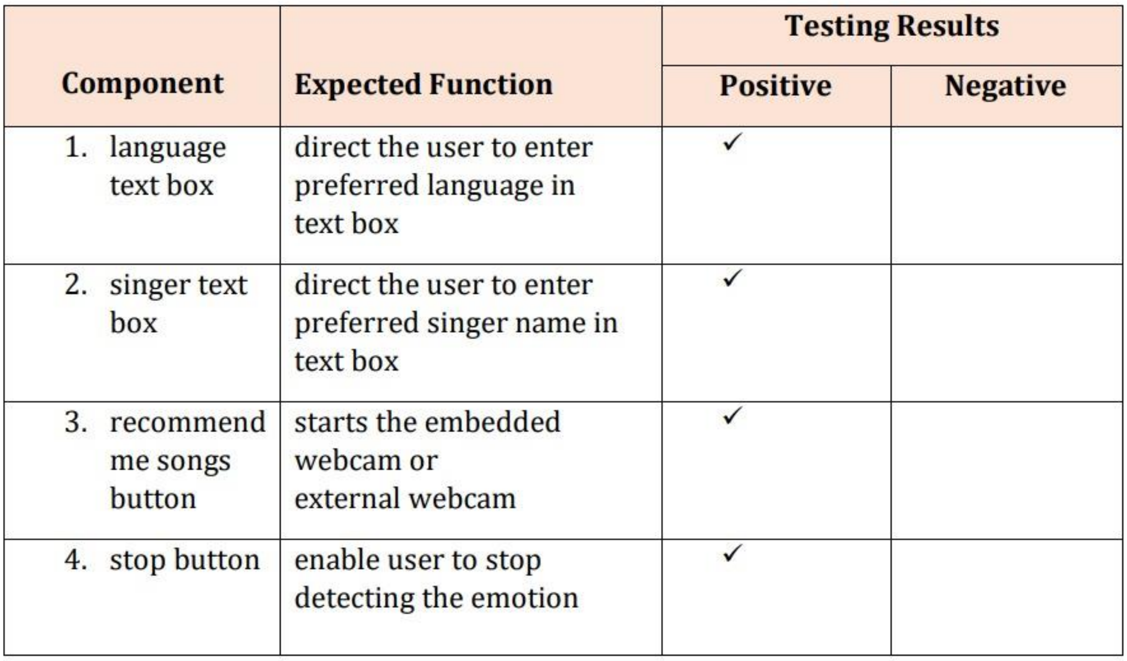
CHAPTER 4

TESTS AND RESULTS

4.1 Test cases

The user carried out system testing once the completion of the system development. The purpose of this testing is to check the functionalities system, whether if it is usable and well-functioned. The results from the functional testing can be seen in the table below.

Table 4.1: System functional testing results



Component	Expected Function	Testing Results	
		Positive	Negative
1. language text box	direct the user to enter preferred language in text box	✓	
2. singer text box	direct the user to enter preferred singer name in text box	✓	
3. recommend me songs button	starts the embedded webcam or external webcam	✓	
4. stop button	enable user to stop detecting the emotion	✓	

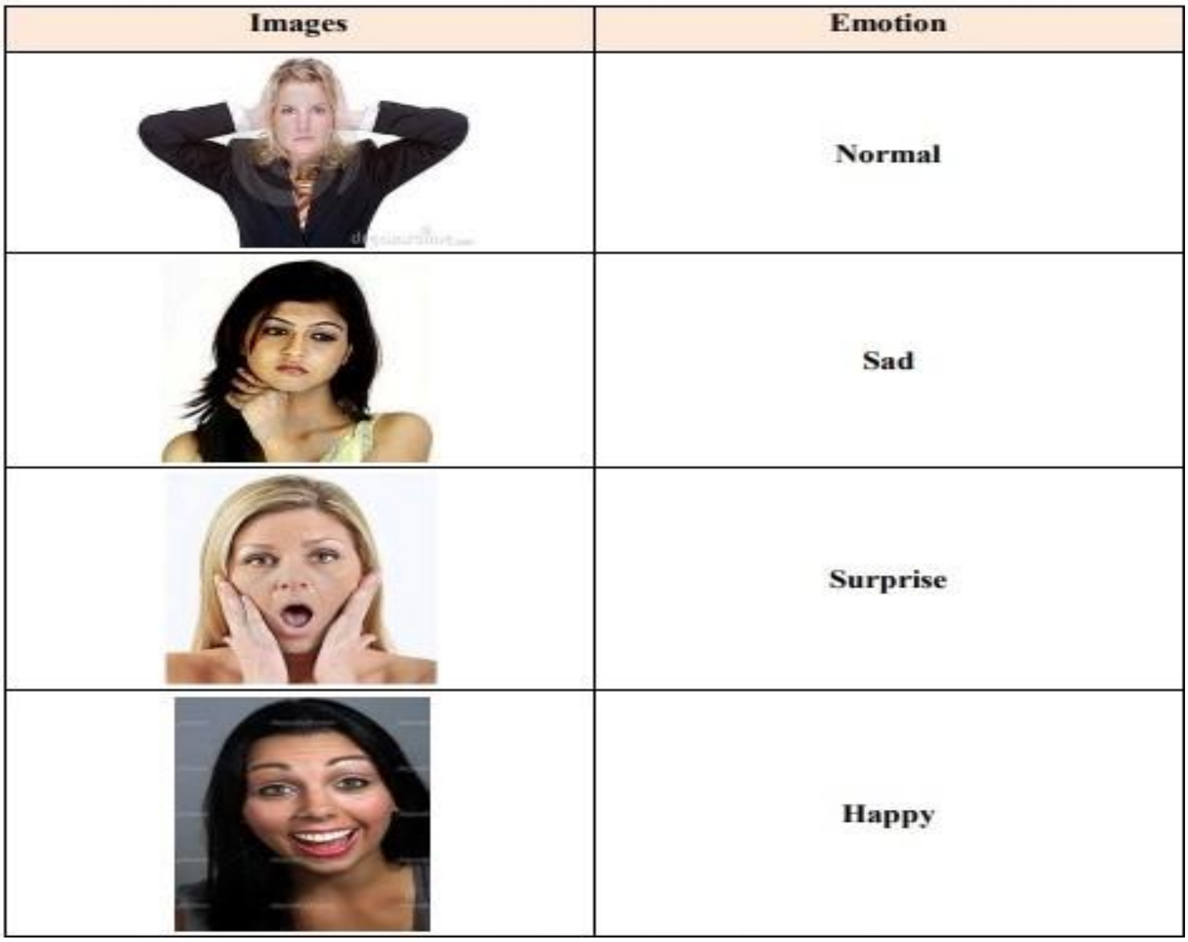
All the website front end test cases got positive for every expected function of components given below these components are generated by GUI programming.

The term GUI refers to graphic user interface A graphics-based operating system interface that uses icons, menus and a mouse (to click on the icon or pull down the menus) to manage interaction with the system. Developed by Xerox, the GUI was popularized by the Apple Macintosh in the 1980s.

EMOTION ACCURACY TESTING RESULTS

Set of images for the each emotions (normal, sad, surprise and happy) are saved in the proposed model for the comparison purposes. The newly load images will be compared with the saved dataset in order to detect the emotion of the users. Table below showed the set of images that saved in the proposed model.

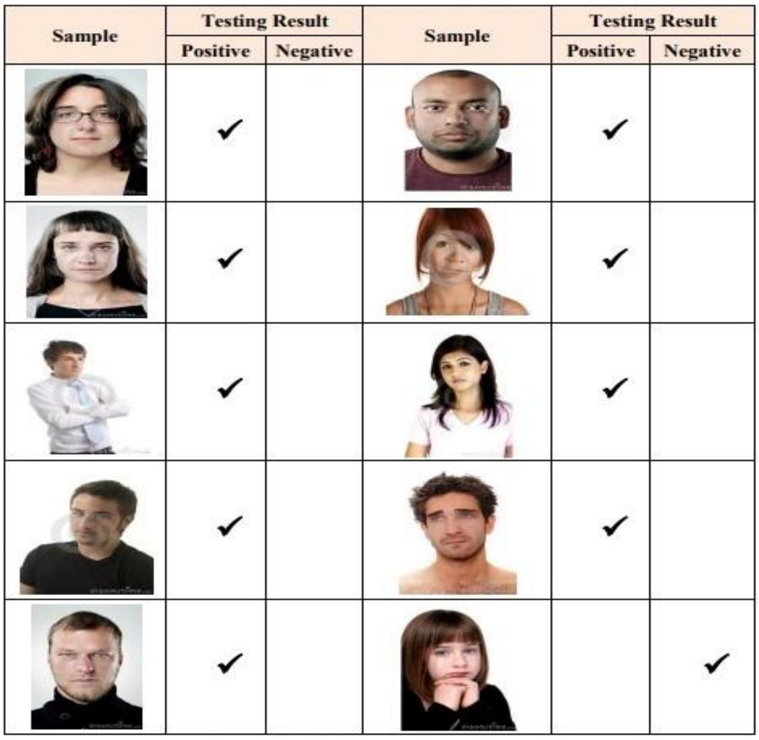
Table 4.2: Emotions saved in the proposed model



Images	Emotion
	Normal
	Sad
	Surprise
	Happy

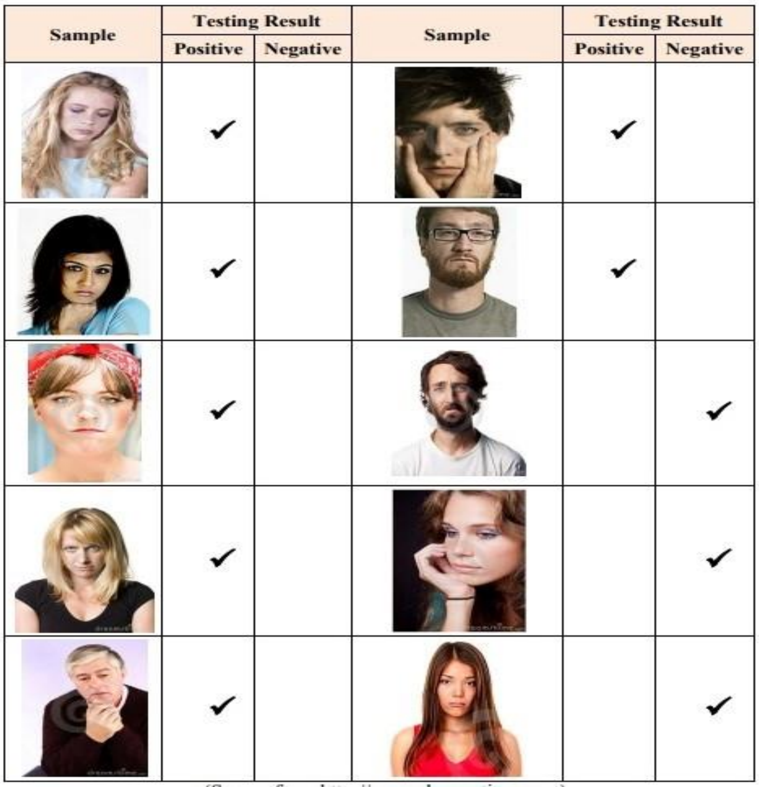
The proposed model is tested with set of persons of similar emotion to test on it accuracy in detecting the emotion. Ten persons are tested for each category of emotions and the results are shown as tables below.

Table 4.3: The testing results for "Neutral" Expression



Sample	Testing Result		Sample	Testing Result	
	Positive	Negative		Positive	Negative
	✓			✓	
	✓			✓	
	✓			✓	
	✓			✓	
	✓				✓

Table 4.4: The testing results for "Sad" Expression



Sample	Testing Result		Sample	Testing Result	
	Positive	Negative		Positive	Negative
	✓			✓	
	✓			✓	
	✓				✓
	✓				✓
	✓				✓

Table 4.5: The testing results for “Surprise” Expression

Sample	Testing Result		Sample	Testing Result	
	Positive	Negative		Positive	Negative
	✓			✓	
	✓			✓	
	✓			✓	
	✓				✓
	✓				✓

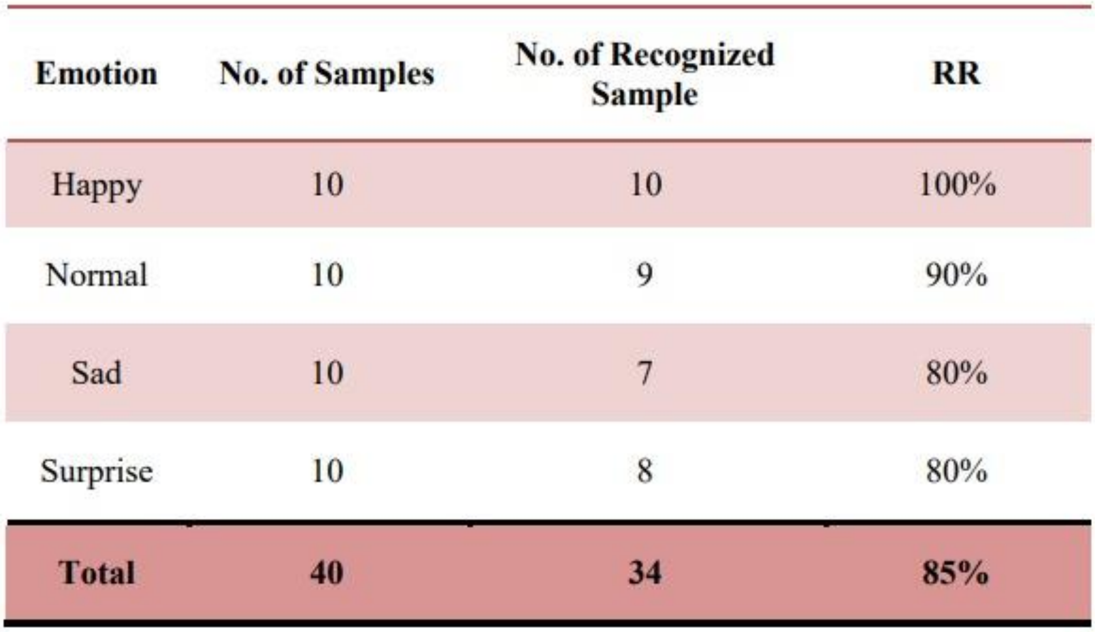
Table 4.6: The testing results for “Happy” Expression

Sample	Testing Result		Sample	Testing Result	
	Positive	Negative		Positive	Negative
	✓			✓	
	✓			✓	
	✓			✓	
	✓			✓	
	✓			✓	

4.2 Result

Summary of Result are shown in the table below:

Table 4.7: The summary of the results tested



Emotion	No. of Samples	No. of Recognized Sample	RR
Happy	10	10	100%
Normal	10	9	90%
Sad	10	7	80%
Surprise	10	8	80%
Total	**40**	**34**	**85%**

In order to find the RR (Recognition Rate), the following formula is applied to the results collected as below:

$$RR = \frac{Classified\ Character}{Total\ Number\ of\ Character} \text{ x } 100\%$$

RR=34/40*100

= 85%

Based on the result above, it shows that the proposed model has the recognition rate (RR) of 85%.

CHAPTER 5

Conclusion & Future Enhancements

5.1 CONCLUSION

The Emotion-Based Music Player is used to automate and give a better music player experience for the end user. The application solves the basic needs of music listeners without troubling them as existing applications do it uses technology to increase the interaction of the system with the user in many ways. It eases the work of the end-user by capturing the image using a camera, determining their emotion, and suggesting a customized play-list through a more advanced and interactive system. Facial expressions are captured using an inbuilt camera. The accuracy of the emotion detection algorithm used in the system for real time images is around 85-90%, while for static images it is around 98-100%. The proposed algorithm on an average calculated estimation takes around 0.95-1.05 sec to generate an emotion-based music playlist. Thus, it yields better accuracy in terms of performance and computational time and reduces the designing cost, compared to the algorithms used in the literature survey. The application can be improved by modifying and adding few functionalities.

5.2 Future Enhancements

To increase the accuracy in the emotion detection. It can be done by increasing the numbers of facial features used in emotion detection.

Face recognition, Image capturing can be made more efficient in low light environment.

To build a mobile application like Spotify where we can store songs for appropriate emotion and provide the user a great using experience.

We would like to develop this application for recommendation of movies and web series.

REFERENCES

[1] Ekman, P. & Friesen, W. V. (1969): The repertoire of nonverbal behavior: Categories, origins, usage, and coding: Semiotica, Vol. 1: 49–98.

[2] Barbara Raskauskas (2009). Why Do People Listen to Music? Retrieved on October 9 2012 from http://voices.yahoo.com/whypeople-listen-music-2608185.html.

[3] Emily Sohn (2011) Why Music Makes You Happy? Retrieved on October 9 2012 from http://news.discovery.com/human/musicdopamine-happiness-brain-110110.html.

[4] Mary Duenwald (2005). The Physiology of Facial Expressions. Retrieved on October 9, 2012, from http://discovermagazine.com/2005/jan/physio logy-of-facial-expressions

[5] Joseph C. Hager (2003). Introduction To the DataFace Site: Facial Expressions, Emotion Expressions, Nonverbal Communication, Physiognomy. Retrieved 10 October 2012, from http://face-andemotion.com/dataface/general/homepage.jsp

[6] W.K. Teo, Liyanage C De Silva and Prahlad Vadakkepat. (2004): Facial Expression Detection and Recognition System: Journal of The Institution of Engineers, Singapore. Vol 44.

[7] Jagdish Lal Raheja, Umesh Kumar. (2010): Human Facial Expression Detection From Detected In Captured Image Using Back Propagation Neural Network: International Journal of Computer Science & Information Technology (IJCSIT). Vol.2(1).

[8] Zhengyi Zhang. (1998): Feature-Based Facial Expression Recognition: Sensitivity Analysis and Experiments with a Multi-Layer Perceptron: Journal of Pattern Recognition and Artificial Intelligence. Vol. 13(6): 893- 911.

[9] Eva Cerezo1, Isabelle Hupont, Critina Manresa, Javier Varona, Sandra Baldassarri, Francisco J. Perales, and Francisco J. Seron. (2007): Real-Time Facial Expression Recognition for Natural Interaction: In J. Martí et al. (Eds.): IbPRIA 2007, Part II, LNCS 4478, (pp. 40–47). Springer-Verlag Berlin Heidelberg.

[10] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," Image Vis. Comput., vol. 27, no. 6, pp. 803–816, May 2009.

s[11] Antoinette L. Bouhuys, Gerda M. Bloem, Ton G.G.Groothuis. (1994): Induction Of Depressed And Elated Mood By Music Influences The Perception Of Facial Emotional Expressions In Healthy Subjects: Journal of Affective Disorders. Vol. 33: 215- 226.

[12] Daniel T. Bishop, Costas I. Karageorghis, and Georgios Loizou. (2007): A Grounded Theory of Young Tennis Players' Use of Music to Manipulate Emotional State: Journal of Sport & Exercise Psychology: 584-607.

APPENDIX

```
import streamlit as st
from streamlit_webrtc import webrtc_streamer
import av
import cv2
import numpy as np
import mediapipe as mp
from keras.models import load_model
import webbrowser
model  = load_model("demo.h5")
label = np.load("labels.npy")
holistic = mp.solutions.holistic
hands = mp.solutions.hands
holis = holistic.Holistic()
drawing = mp.solutions.drawing_utils
st.header("Emotion Based Music system")
if "run" not in st.session_state:
  st.session_state["run"] = "true"
try:
  emotion = np.load("emotion.npy")[0]
except:
  emotion=""
if not(emotion):
  st.session_state["run"] = "true"
else:
  st.session_state["run"] = "false"
class EmotionProcessor:
  def recv(self, frame):
    frm = frame.to_ndarray(format="bgr24")
    ##############################
    frm = cv2.flip(frm, 1)
    res = holis.process(cv2.cvtColor(frm, cv2.COLOR_BGR2RGB))
    lst = []
    if res.face_landmarks:
      for i in res.face_landmarks.landmark:
        lst.append(i.x - res.face_landmarks.landmark[1].x)
        lst.append(i.y - res.face_landmarks.landmark[1].y)
      if res.left_hand_landmarks:
        for i in res.left_hand_landmarks.landmark:
          lst.append(i.x - res.left_hand_landmarks.landmark[8].x)
          lst.append(i.y - res.left_hand_landmarks.landmark[8].y)
      else:
        for i in range(42):
          lst.append(0.0)
      if res.right_hand_landmarks:
        for i in res.right_hand_landmarks.landmark:
          lst.append(i.x - res.right_hand_landmarks.landmark[8].x)
          lst.append(i.y - res.right_hand_landmarks.landmark[8].y)
      else:
        for i in range(42):
          lst.append(0.0)
      lst = np.array(lst).reshape(1,-1)
      pred = label[np.argmax(model.predict(lst))]
      print(pred)
      cv2.putText(frm, pred, (50,50),cv2.FONT_ITALIC, 1, (255,0,0),2)
      np.save("emotion.npy", np.array([pred]))
    drawing.draw_landmarks(frm, res.face_landmarks,
holistic.FACEMESH_TESSELATION,landmark_drawing_spec=drawing.DrawingSpec
(color=(0,0,255),thickness=1,circle_radius=1),connection_drawing_spec=drawing.Draw
ingSpec(thickness=1))
    drawing.draw_landmarks(frm, res.left_hand_landmarks,
hands.HAND_CONNECTIONS)
    drawing.draw_landmarks(frm, res.right_hand_landmarks,
hands.HAND_CONNECTIONS)
  ##############################
    return av.VideoFrame.from_ndarray(frm, format="bgr24")
lang = st.text_input("Language")
singer = st.text_input("singer")
if lang and singer and st.session_state["run"] != "false":
  webrtc_streamer(key="key",desired_playing_state=
True,video_processor_factory=EmotionProcessor)
btn = st.button("Recommend me songs")
if btn:
  if not(emotion):
    st.warning("Please let me capture your emotion first")
    st.session_state["run"] = "true"
  else:

webbrowser.open(f"https://www.youtube.com/results?search_query={lang}+{emotion
}+songs+{singer}")
    np.save("emotion.npy", np.array([""]))
st.session_state["run"] = "false"
```