**FaceMelody - Mood-Aware Music Recommendation System With Facial Recognition**

#### A Project Report Submitted in partial fulfillment of the requirements for the award of the degree of

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

#### By

**PRAVALLIKA (2010030046)**

##### TAHSEEN BEGUM (2010030168)

##### M. RUSHITHA (2010030362)

**K. KAVYA (2010030550)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**K L DEEMED TO BE UNIVERSITY AZIZNAGAR, MOINABAD, HYDERABAD-500075**

**OCTOBER 2023**

##### BONAFIDE CERTIFICATE

This is to certify that the project titled **FaceMelody – Mood Aware Music Recommendation System with Facial Recognition** is a bonafide record of the work done by

##### PRAVALLIKA (2010030046) TAHSEEN BEGUM (2010030168)

##### M. RUSHITHA (2010030362)

##### K.KAVYA (2010030550)

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **COMPUTER SCIENCE AND ENGINEERING** of the **K L DEEMED TO BE UNIVERSITY, AZIZNAGAR, MOINABAD, HYDERABAD-500075**, dur-

ing the year 2023-2024.

##### Mr. Chiranjeevi Nuthalapati Dr. Arpita Gupta

Project Guide Head of the Department

Project Viva-voce held on

##### Internal Examiner External Examiner

**ABSTRACT**

The proposed FaceMelody analyzes the user’s facial expressions in real time. Through the user’s webcam, the system captures emotional cues, such as joy, sadness, or excitement, and processes this data to understand the user’s current mood.

FaceMelody leverages a sophisticated recommendation engine that correlates the detected mood with a vast music database, ensuring that the music selection aligns with the user’s emotional context.

The recommendation engine considers a variety of factors, including tempo, genre, lyrics, and historical preferences to provide a personalized and emotionally resonant playlist. The system also adapts in real-time, allowing users to change their music selection as their mood evolves. Whether a user is looking to lift their spirits, relax, or reflect, FaceMelody is designed to cater to these emotional shifts.

##### ACKNOWLEDGEMENT

We would like to thank the following people for their support and guidance without whom the completion of this project in fruition would not be possible.

**Mr. Chiranjeevi Nuthalapati**, our project guide, for helping us and guiding us in the course of this project.

**Dr. Arpita Gupta**, the Head of the Department, Department of Computer Science And Engineering.

Our internal reviewers, **Ms. N Anuradha**, **Mr. Faizan Ahmed**, **Dr. Rajib Debnath** for their insight and advice provided during the review sessions.

We would also like to thank our individual parents and friends for their constant support.

**TABLE OF CONTENTS**

##### Title Page No.

##### ABSTRACT ii

##### ACKNOWLEDGEMENT iii

##### TABLE OF CONTENTS iv

##### LIST OF TABLES vii

##### LIST OF FIGURES viii

##### Introduction 1

* 1. Background of the Project 1
     1. Subsection Title 1
  2. Problem Statement 1
  3. Objectives 2
  4. Scope of the Project 2

##### Literature Review 3

* 1. An Example On How To Add Pictures 3
  2. Fixed Image 4
  3. Example on Table Usage 5
  4. Overview of related works 5
  5. Advantages and Limitations of existing systems 5

##### Proposed System 6

|  |  |  |
| --- | --- | --- |
| 3.1 | System Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
|  | 3.1.1 Subsection Title . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
| 3.2 | Design of the System . . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
|  | 3.2.1 Subsection Title . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
| 3.3 | Algorithms and Techniques used . . . . . . . . . . . . . . . . . . . . . | 7 |
|  | 3.3.1 Subsection Title . . . . . . . . . . . . . . . . . . . . . . . . . . | 7 |

##### Implementation 8

* 1. [Tools and Technologies used 8](#_TOC_250015)
     1. [Subsection Title 8](#_TOC_250014)
  2. [Modules and their descriptions 8](#_TOC_250013)
     1. [Subsection Title 8](#_TOC_250012)
  3. [Flow of the System 9](#_TOC_250011)
     1. [Subsection Title 9](#_TOC_250010)

1. Results and Analysis 10
   1. [Performance Evaluation 10](#_TOC_250009)
      1. [Subsection Title 10](#_TOC_250008)
   2. [Comparison with existing systems 10](#_TOC_250007)
      1. [Subsection Title 10](#_TOC_250006)
   3. [Limitations and future scope 11](#_TOC_250005)
      1. [Subsection Title 11](#_TOC_250004)
2. Conclusion and Recommendations 12
   1. [Summary of the Project 12](#_TOC_250003)
   2. [Contributions and achievements 12](#_TOC_250002)
   3. [Recommendations for future work 12](#_TOC_250001)

References 13

[Appendices 14](#_TOC_250000)

A Source code 15

1. Screen shots 17
2. Data sets used in the project 18

**List of Tables**

* 1. Sensor Specifications 5
  2. Modeled Hosts Configuration 5

**List of Figures**

* 1. Figure Title 3
  2. Figure Title 4
  3. Inertial Measurement Unit 4

B.1 Figure Title 17

**Chapter 1 Introduction**

* 1. **Background of the Project**

Music has been a significant part of human culture and expression for centuries. It holds the power to evoke emotions, create memories, and provide solace. With the digital revolution, music consumption has become more personalized than ever before, thanks to the advent of music streaming platforms and recommendation systems. However, these systems are often limited to algorithms that analyze user history, preferences, and other contextual data, neglecting a crucial element of human experience: emotions.

* + 1. **Subsection Title**

Emotions play a pivotal role in how music affects us. A song that resonates with one person during a moment of happiness may not have the same impact during a moment of sadness. This interplay between music and emotions has been the subject of numerous studies and has given rise to the field of music psychology. It has long been recognized that music can either enhance or alter one's emotional state.

##### Sub Sub Section Title

The FaceMelody transforms the music listening experience, enhancing user engagement and emotional well-being. This project represents a harmonious blend of technology and human emotion to create a more personalized and emotionally resonant music experience.

* 1. **Problem Statement**

This project is aimed at developing a music recommendation system that is capable of recognizing the user's mood through facial recognition technology. This system will analyze user's facial expressions in real-time to determine their emotional states and preferences. By combining facial data with mood-specific music databases, this system will deliver personalized music recommendations that align with user's emotions, enhancing their listening experience and fostering emotional well-being.

* 1. **Objectives**
* **Facial Emotion Recognition**: The system uses a pre-trained convolutional neural network (CNN) model to recognize facial emotions. The model takes an input image of a face and predicts the emotion associated with it. The emotions recognized include anger, disgust, fear, happiness, neutrality, sadness, and surprise.
* **Music Recommendation**: The system recommends music based on the user's detected emotion. Each emotion is associated with a specific music genre or playlist. The system retrieves the corresponding music playlist from a CSV file and presents it to the user.
* **Video Streaming**: The system captures video frames from the user's webcam in real-time. It processes each frame to detect facial emotions and recommends music accordingly. The video streaming is performed using OpenCV, a popular computer vision library.
  1. **Scope of the Project**
* **Facial Emotion Recognition**: Implementing facial recognition technology to detect and analyze facial expressions in real time. Emotions such as happiness, sadness, anger, and more will be recognized.
* **Emotion - to - Music Mapping**: Creating an emotion-to-music database that associates specific emotional states with music genres, moods, or specific songs.
* **User Interface**: Developing a user-friendly interface, such as a mobile app or web platform, to access and interact with the FaceMelody.
* **Real - time Emotion Analysis**: Ensuring the system can process and interpret facial expressions continuously in real-time to adapt music recommendations on the fly.
* **Music Database Integration**: Connecting FaceMelody with music streaming services to access a vast library of songs and genres.

**Chapter 2 Literature Review**

* 1. **An Example On How To Add Pictures**

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

Example of using a label to refer to a Figure 2.1

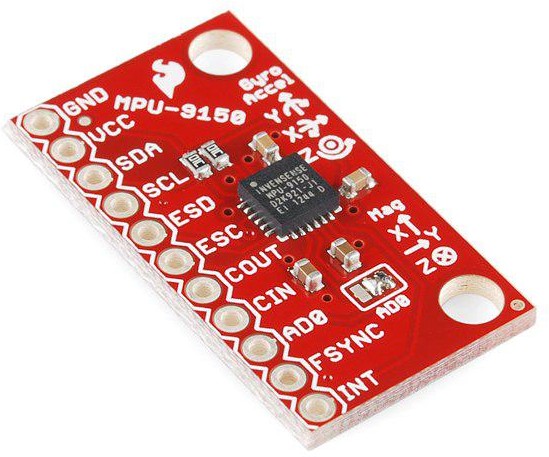


Figure 2.1: Figure Title

* 1. **Fixed Image**

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

I want my image fixed here



Figure 2.2: Figure Title

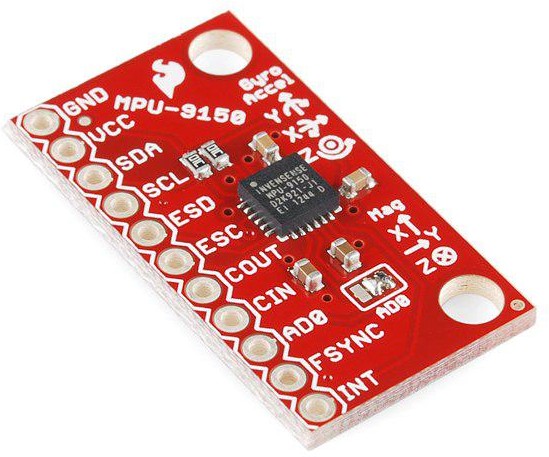


Figure 2.3: Inertial Measurement Unit

* 1. **Example on Table Usage**

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

To refer to the table use the same 2.1

Acceleration Range: 2G Dual-Core 80 Mhz Processor Gyroscope Range: 250 degrees/second 4MB Flash Memory Magnetometer Range: 1200 micro Teslas 802.11 b/g/n connectivity

Communication Protocol: I2C

NodeMCU Processor

MPU 9150 Inertial Measurement Unit

Table 2.1: Sensor Specifications

* 1. **Overview of related works**

We will analyze a Flask application that integrates a camera. The application allows users to view a live video feed and interact with a table of music data. We will explore an emotion-based music recommendation system using Python. The system uses facial emotion recognition to detect the user's emotion and recommends music based on their emotional state.

* 1. **Advantages and Limitations of existing systems**

**Advantages**

**Personalization**: Facial expression-based systems can provide highly personalized music recommendations by taking into account the user's current emotional state. This can lead to a more engaging and enjoyable music listening experience.

**Emotion Detection**: They can detect and respond to a wide range of emotions, including happiness, sadness, excitement, and relaxation, enabling a more comprehensive emotional connection with the user.

**Mood Regulation**: Such systems can help users regulate their emotions by recommending music that matches their desired mood, which can be beneficial for mental health and well-being.

**Limitations**

**Limited Accuracy**: Facial expression analysis is not always accurate in determining a user's emotional state. Users may not always express their emotions through their facial expressions, or there may be misinterpretations.

**Privacy Concerns**: Analyzing facial expressions for music recommendations can raise privacy concerns. Users may be uncomfortable with their facial data being used for such purposes.

**Data Quality**: The effectiveness of these systems depends on the quality of the facial expression data and the accuracy of emotion detection algorithms. Inaccurate data can lead to poor recommendations.

**Chapter 3 Proposed System**

* 1. **System Requirements**

**Software Requirements**

The major software requirements of the project are as follows:

**Language:** Python

**Operating system:** Windows XP

**Tools:**

* + 1. **Hardware Requirements**

The hardware requirements that map to the software are as follows:

**RAM:** 8.00 GB (7.78 GB usable)

**Processor:** Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.10 GHz

**System-type:** 64-bit operating system, x64-based processor

**Version:** 22H2

**Edition:** Windows 10 Pro

* 1. **Design of the System**

The emotion recognition model is trained on the FER 2013 dataset. It can detect 7 emotions. The project works by getting a live video feed from a webcam, and passing it through the model to get a prediction of emotion. Then according to the emotion predicted, the app will fetch a playlist of songs from Spotify through spotipy wrapper and recommend the songs by displaying them on the screen.

* + 1. **Code Structure**

The provided code consists of several import statements, class definitions, and function definitions. Let's discuss the main components of the code:

**Import Statements:** The code imports various libraries and modules required for the emotion recognition system. These include numpy, cv2 (OpenCV), PIL (Python Imaging Library), tensorflow.keras, pandastable, and datetime.

**Music and Emotion Dictionaries:** The code defines two dictionaries: "emotion\_dict" and "music\_dist". The "emotion\_dict" maps emotion labels to their corresponding names, while the "music\_dist" maps emotion labels to the paths of CSV files containing music recommendations for each emotion.

**FPS and WebcamVideoStream Classes:** The code defines two classes: "FPS" and "WebcamVideoStream". The "FPS" class calculates the frames per second (FPS) of the video stream, while the "WebcamVideoStream" class reads frames from the video stream in a separate thread to boost performance.

**VideoCamera Class:** The code defines the "VideoCamera" class, which is responsible for capturing video frames, performing emotion recognition, and recommending music. The class uses the face cascade classifier to detect faces, extracts the facial region of interest (ROI), and feeds it to the emotion model for prediction. It also retrieves the corresponding music recommendations from the CSV files based on the detected emotion.

**Spotipy:** Spotipy is a Python library that allows us to interact with the Spotify Web API. It provides methods to authenticate with Spotify, fetch playlist data, and retrieve information about tracks, albums, and artists.

**Spotify Web API:** The Spotify Web API is a RESTful API that provides access to various Spotify resources, such as playlists, tracks, albums, and artists. It allows developers to retrieve and manipulate data from the Spotify music catalog.

**Emotion Analysis:** Emotion analysis, also known as sentiment analysis, is the process of determining the emotional tone of a piece of text or audio. In our case, we will analyze the emotions associated with the tracks in a Spotify playlist.

**Emotion Recognition:** Emotion recognition involves training a model to recognize and classify human emotions based on facial expressions. The model is trained on a dataset of labeled facial images, where each image is associated with a specific emotion label (e.g., happy, sad, angry, etc.).

**WebcamVideoStream:** This class represents a separate thread for video streaming from a web camera. It utilizes the OpenCV library (cv2) to capture frames from the camera.

**Thread:** The Thread class from the threading module is used to create a separate thread for reading frames from the video stream. This allows the video streaming to run concurrently with other tasks.

**VideoCapture:** The VideoCapture class from the OpenCV library is used to capture video frames from the web camera. It takes the camera source as an argument (default is 0, which represents the default camera).

The WebcamVideoStream class provides a convenient way to stream video from a web camera using a separate thread in Python. By utilizing a separate thread, we can perform other tasks concurrently while continuously reading frames from the video stream.

**Facial Emotion Detection:** FaceMelody uses computer vision techniques to analyze facial expressions and detect emotions such as happiness, sadness, anger, and surprise. This is done by capturing frames from the webcam and applying machine learning algorithms to classify the emotions based on facial features.

**Song Recommendations:** Once the user's emotions are detected, FaceMelody recommends songs that align with their emotional state. These recommendations are based on a pre-defined database of songs and their associated emotional characteristics. The application selects songs that are known to evoke similar emotions in listeners.

**Real-time Updates:** FaceMelody continuously updates the song recommendations based on the user's changing emotions. It constantly analyzes the facial expressions captured by the webcam and adjusts the recommendations accordingly. This ensures that the suggested songs are always relevant to the user's current emotional state.

* 1. **Algorithms and Techniques used**

**Convolutional Neural Networks (CNNs):** CNNs are a type of deep learning model that is particularly effective for image recognition tasks. They consist of multiple layers of convolutional and pooling operations, followed by fully connected layers for classification.

CNN model is used for facial emotion recognition. The model architecture consists of several convolutional and pooling layers, followed by fully connected layers. The model is loaded with pre-trained weights from a file.

A pre-trained convolutional neural network (CNN) model is used to recognize facial emotions. The model takes an input image of a face and predicts the emotion associated with it.

By training a CNN model on a dataset of labeled facial images, we can develop a powerful tool for emotion recognition.

* + 1. **Techniques**

**ImageDataGenerator:** The ImageDataGenerator class in Keras is a powerful tool for data augmentation and preprocessing. It allows us to generate batches of augmented image data on the fly, which helps improve the performance and generalization of our model.

**Face Cascade Classifier:** It's a technique used for face detection in images and videos. The code initializes a face cascade classifier using the "haarcascade\_frontalface\_default.xml" file. This classifier is used to detect faces in the video frames. The "VideoCamera" class, is responsible for capturing video frames, performing emotion recognition, and recommending music. The class uses the face cascade classifier to detect faces, extracts the facial region of interest (ROI), and feeds it to the emotion model for prediction. It also retrieves the corresponding music recommendations from the CSV files based on the detected emotion.

**Chapter 4 Implementation**

## Tools and Technologies used

**OpenCV (cv2):** An open-source computer vision library that provides various tools for image and video analysis. The video streaming is performed using OpenCV(a popular computer vision library). The WebcamVideoStream class utilizes the OpenCV library (cv2) to capture frames from the camera. The VideoCapture class from the OpenCV library is used to capture video frames from the web camera. It takes the camera source as an argument (default is 0, which represents the default camera).

**PIL (Python Imaging Library):** A library for opening, manipulating, and saving many different image file formats in Python.

**Pandastable:** A tool for creating graphical data tables for pandas data frames.

### Technologies

**Keras:** Convolution Neural Network along with keras library is used for emotion recognition.

**TensorFlow.keras:** Part of the TensorFlow framework, it's a high-level neural networks API that is based on the Keras library.

**Spotipy:** Spotipy is a Python library that allows us to interact with the Spotify Web API. It provides methods to authenticate with Spotify, fetch playlist data, and retrieve information about tracks, albums, and artists. The Spotify Web API is a RESTful API that provides access to various Spotify resources, such as playlists, tracks, albums, and artists. It allows developers to retrieve and manipulate data from the Spotify music catalog. Emotion analysis, also known as sentiment analysis, is the process of determining the emotional tone of a piece of text or audio. In our case, we will analyze the emotions associated with the tracks in a Spotify playlist.

**Flask:** The application is built using Flask which integrates a camera. It allows users to view a live video feed and interact with a table of music data.

## Modules and their Descriptions

**Modules(Python Libraries)**

Numpy

DateTime

Spotipy

### Description of Modules

**NumPy:** A fundamental package for scientific computing with Python, used for numerical operations and data manipulation.

**DateTime:** A module used to work with dates and times in Python.

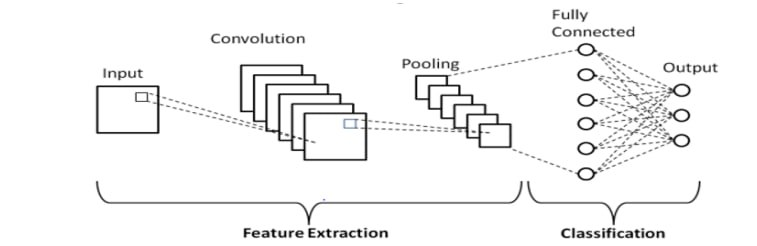
**Spotipy:** Spotipy is a Python library that allows us to interact with the Spotify Web API. It provides methods to authenticate with Spotify, fetch playlist data, and retrieve information about tracks, albums, and artists. The Spotify Web API is a RESTful API that provides access to various Spotify resources, such as playlists, tracks, albums, and artists. It allows developers to retrieve and manipulate data from the Spotify music catalog. Emotion analysis, also known as sentiment analysis, is the process of determining the emotional tone of a piece of text or audio. In our case, we will analyze the emotions associated with the tracks in a Spotify playlist.

## Flow of the System

FaceMelody is an innovative web application that combines facial emotion detection with song recommendations. By analyzing the user's facial expressions in real time, it provides personalized song suggestions that match their emotional state. The provided code defines the structure and styling of the web application, as well as the JavaScript code for updating the song recommendations dynamically. With FaceMelody, users can discover new music that resonates with their emotions and enhances their listening experience.

### Subsection Title

A project We will input an user image how.



**Chapter 5 Results and Analysis**

## Performance Evaluation

Performance evaluation of a music recommendation system based on face emotion recognition involves assessing the system's effectiveness in recommending music that matches the emotional state of the user.

### Subsection Title

**Model Architecture:** The model architecture is a sequential model consisting of Conv2d, Maxpool2d, Dropout and Dense layers:

Conv2D layers throughout the model have different filter size from 32 to 128, all with activation 'relu'

Pooling layers have pool size (2,2)

Dropout is set to 0.25 as anything above results in poor performance

Final Dense layer has 'softmax' activation for classifying 7 emotions

Used 'categorical\_crossentropy' for loss with 'Adam' optimizer with 'accuracy' metric

**Note:-** This model architecture gives good enough accuracy. A bit more tinkering with hyperparameters might lead to better accuracy.

**Image Processing and Training:** The images were normalised, resized to (48,48) and converted to grayscale in batches of 64 with the help of 'ImageDataGenerator' in Keras API.

Training took around 13 hours locally for 75 epochs with an accuracy of ~88 %

## Comparison with existing systems

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

### Subsection Title

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

## Limitations and future scope

**Privacy Concerns:** Face emotion recognition raises privacy concerns, as it involves processing and storing facial images. Users might be uncomfortable with their facial data being used for recommendations.

**Real-Time Processing:** The real-time processing requirement for live recommendations can be resource-intensive. This could limit the system's performance on devices with limited processing power.

**Cultural Differences:** Emotion expressions can vary across cultures. A system trained on one cultural group may not work as effectively for others.

### Future Scope

**Multimodal Emotion Recognition:** Combine facial emotion recognition with other modalities such as voice or text sentiment analysis to improve accuracy.

**Context Awareness:** Incorporate contextual information, such as location, time of day, or recent activities, to enhance recommendations based on emotional states.

**Personalization:** Implement user profiles and historical data to provide more personalized recommendations tailored to an individual's preferences.

**Chapter 6**

**Conclusion and Recommendations**

## Summary of the Project

In conclusion, the development of music recommendation systems based on face emotion recognition, leveraging Convolutional Neural Networks (CNN) and OpenCV, represents a significant stride in the field of personalized and emotionally intelligent music streaming. This innovative system harnesses the power of computer vision to enhance user experiences by tailoring music recommendations to individual emotional states. Throughout this journey, we've explored the fundamental components and techniques involved in this system. It combines the capabilities of CNNs to accurately recognize facial emotions and OpenCV for real-time image processing. By categorizing and analyzing the emotions detected, it provides users with music selections that align with their current emotional context.

## Contributions and achievements

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

## Recommendations for future work

* Instead of CSVs, create a database and connect it to the application. The DB will fetch songs for recommendations and new songs can be updated directly onto the database.
* Add a feature that will update specified playlists for better and more recent recommendations, a specific day over a fixed duration say every sunday, and append it to the database.
* Directly play the song or redirect to the song on Spotify when the user clicks on it.
* Rewrite code such that Video Streaming is done on the client side instead of server side so as it make the app deployable.

**Bibliography**

* + 1. Roy, Dharmendra, Anjali CH, G. Kavya Sri, B. Tharun, and K. Venu Gopal. "Music Recommendation Based on Current Mood Using Ai & Ml." (2023).
    2. Kambare, Mrs PP, and S. T. Patil. "FACIAL IMAGE BASED EMOTION DETECTION AND MUSIC RECOMMENDATION SYSTEM.".
    3. Mahadik, Ankita, Shambhavi Milgir, Janvi Patel, Vijaya Bharathi Jagan, and Vaishali Kavathekar. "Mood based music recommendation system." INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 10 (2021).
    4. Mortensen, Magnus, Cathal Gurrin, and Dag Johansen. "Real-world mood-based music recommendation." In Information Retrieval Technology: 4th Asia Infomation Retrieval Symposium, AIRS 2008, Harbin, China, January 15-18, 2008 Revised Selected Papers 4, pp. 514-519. Springer Berlin Heidelberg, 2008.
    5. Adjei, J.K., Blackman, C., Blackman, C., 2015. Explaining the role of trust in cloud computing services. Info 17.
    6. Afroz, S., Navimipour, N.J., 2017. Memory designing using quantum dot cellular automata: systematic literature review, classification, and currenttrends. J. Circuits Syst. Comput. 26 (12), 1730004 (2017) [34 pages].
    7. Alhanahnah, M., Bertok, P., Tari, Z., 2017. Trusting cloud service providers: trust phases and a taxonomy of trust factors. IEEE Cloud Comput. 4,44–54.

# Appendices

# Appendix A Source code

A project description is a high-level overview of why you’re doing a project. The docu- ment explains a project’s objectives and its essential qualities. Think of it as the elevator pitch that focuses on what and why without delving into how.

1

2

3

4

5

6

7

8

9

10

d e f g e t p a r a m e t e r s ( data , c h u n k s i z e = 410 ) : # S t o r e t h e a c t i v i t y l a b e l t o add l a t e r a c t i v i t y = d a t a [ ’ A c t i v i t y ’ ]

’ ’ ’

D e f i n e a d i c t i o n a r y of f u n c t i o n

’ ’ ’

f u n c d i c t = *{*

’ min ’ : np . min ,

’ max ’ : np . max ,

’ d i f f ’ : lambda x : np . max ( x ) − np . min ( x ) , ’ s t d ’ : np . s t d ,

’ i q r ’ : s t a t s . i q r ,

’ rms ’ : lambda x : np . s q r t ( np . mean ( np . s q u a r e ( x ) ) ) , ’ mad ’ : lambda x : x . mad ( ) ,

’ mediad ’ : mediad

*}*a g g r e g a t i o n s = *{*

’X’ : f u n c d i c t ,

’Y’ : f u n c d i c t , ’Z ’ : f u n c d i c t

*}*d a t a g r o u p s = [ ]

’ ’ ’

Transform t h e d a t a s e t i n t o r o l l i n g windows of 410 r e a d i n g s each and s t o r e them i n a Pandas d a t a group .

’ ’ ’

f o r i i n r a n g e ( i n t ( d a t a . shape [ 0 ] / ( c h u n k s i z e / 2 ) ) − 1 ) :

temp = d a t a . i l o c [ i n t ( i \*( c h u n k s i z e / 2 ) ) : i n t ( ( i + 2 ) \*( c h u n k s i z e / 2 ) ) ] temp [ ’ k ’ ] = i

d a t a g r o u p s . append ( temp )

d a t a g r o u p s = pd . c o n c a t ( d a t a g r o u p s ) . groupby ( ’ k ’ , a s i n d e x = F a l s e ) #Run t h e a g g r e g a t i o n s on a l l d a t a g r o u p s

s t a t s d a t a = d a t a g r o u p s . agg ( a g g r e g a t i o n s )

s t a t s d a t a . columns = [ ’ ’ . j o i n ( c o l ) . s t r i p ( ) f o r c o l i n s t a t s d a t a . columns . v a l u e s ]

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

a c t i v i t y = a c t i v i t y . r e s e t i n d e x ( drop =True )

#Add a c t i v i t y l a b e l

s t a t s d a t a = pd . c o n c a t ( [ s t a t s d a t a , a c t i v i t y [ : l e n ( s t a t s d a t a ) ] ] , a x i s = 1 )

d e l s t a t s d a t a [ ’ k ’ ] r e t u r n s t a t s d a t a

34

35

36

37

38

**Appendix B Screen shots**

**B.1 Fixed Image**



Figure B.1:Dashboard

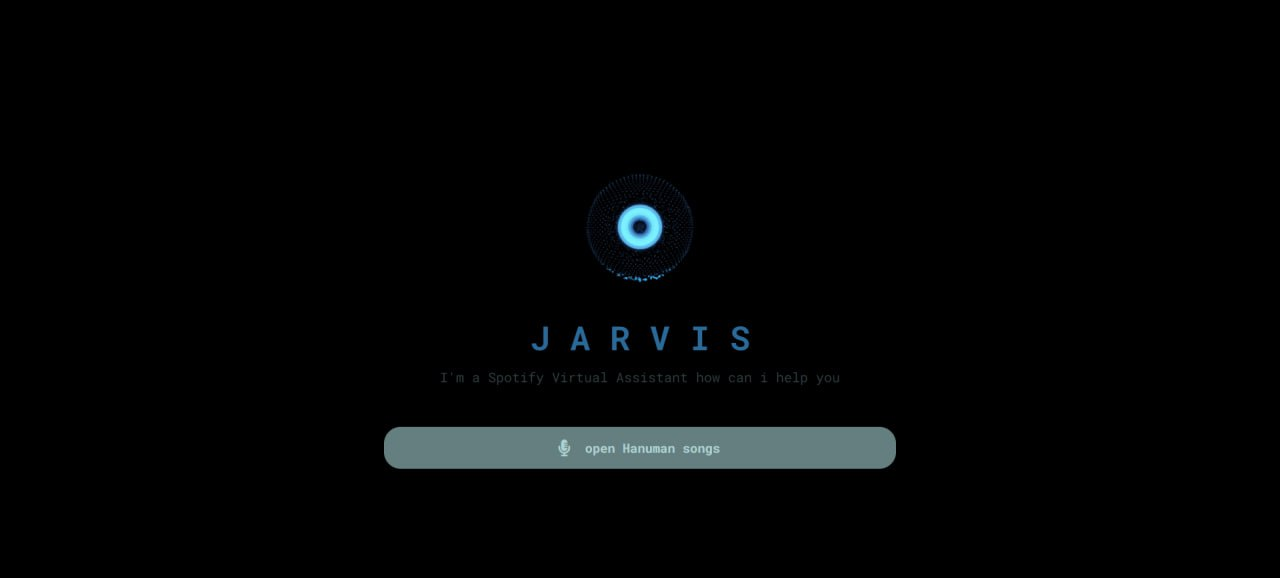


Figure B.2: Voice Ass

Voice assistant: The Spotify Web API is a RESTful API that provides access to various Spotify resources, such as playlists, tracks, albums, and artists. It allows developers to retrieve and manipulate data from the Spotify music catalog.

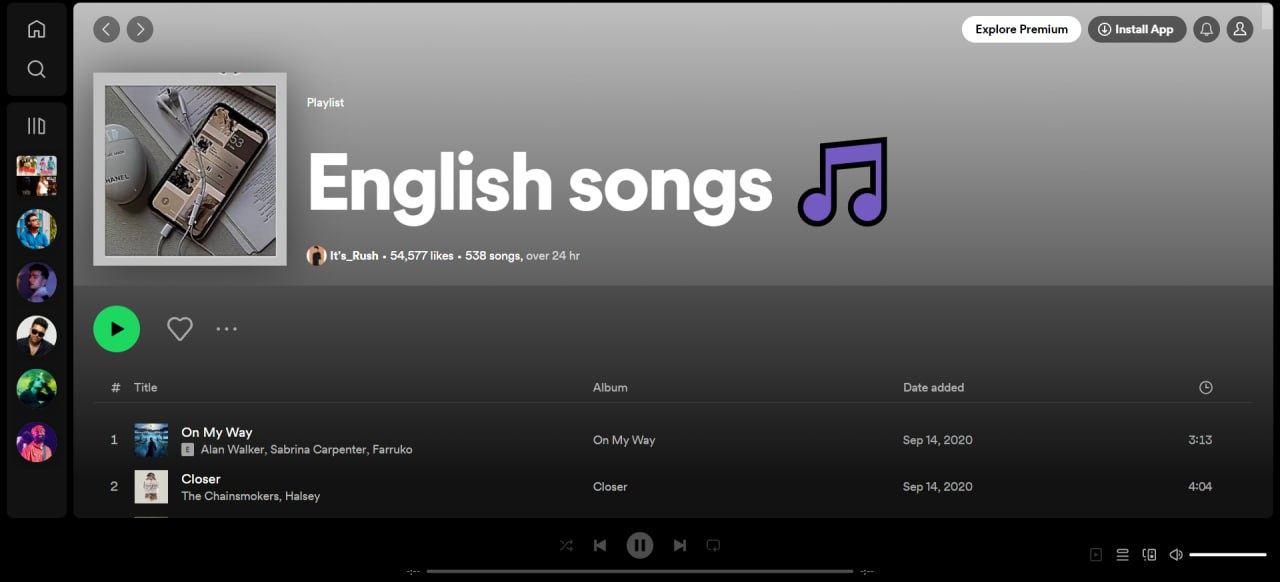
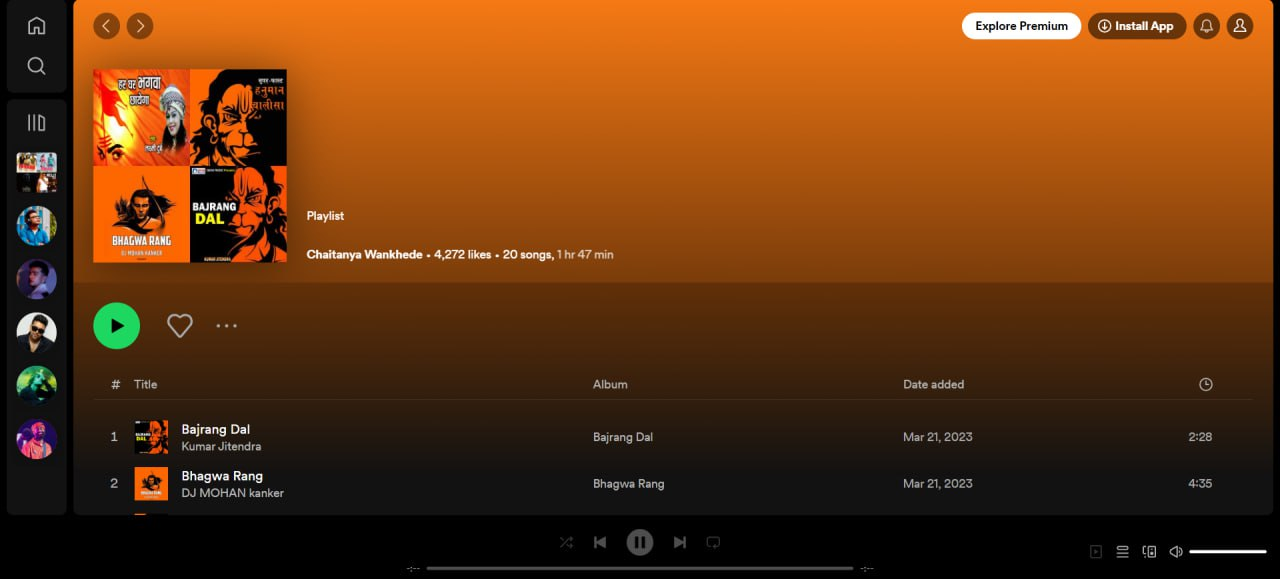


Figure B.3: Songs

We will explore how to analyze emotions in Spotify playlists using Python. We will use the Spotipy library, which is a Python wrapper for the Spotify Web API, to fetch playlist data and extract information about the tracks in the playlist.

Spotipy: Spotipy is a Python library that allows us to interact with the Spotify Web API. It provides methods to authenticate with Spotify, fetch playlist data and extract information about the tracks in the playlist.

Figure B.4: Songs

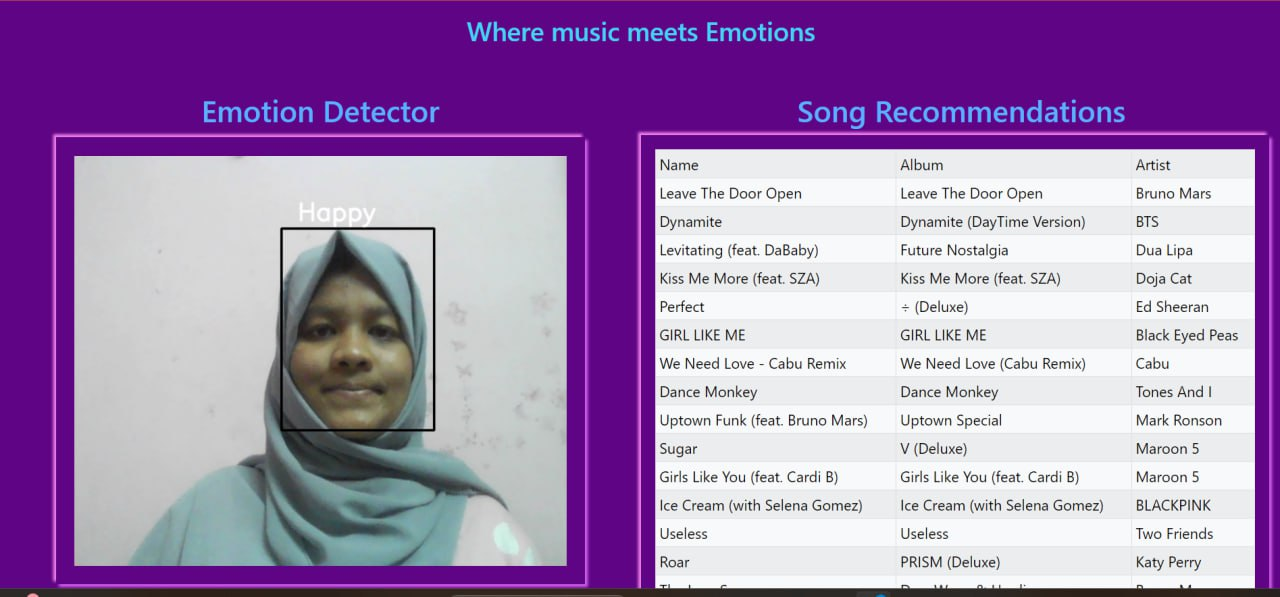


Figure B.5: Happy Face

The WebcamVideoStream class provides a convenient way to stream video from a web camera using a separate thread in Python. By utilizing a separate thread, we can perform other tasks concurrently while continuously reading frames from the video stream. This class can be useful in various applications that require real-time video processing or analysis.

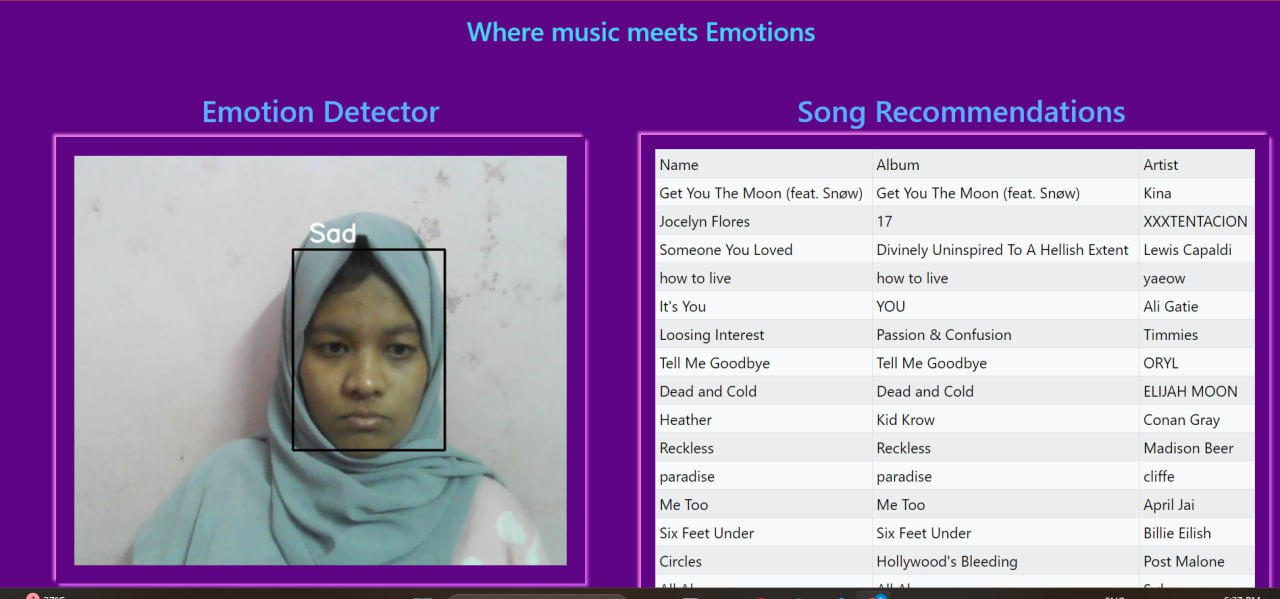


Figure B.6: Sad face

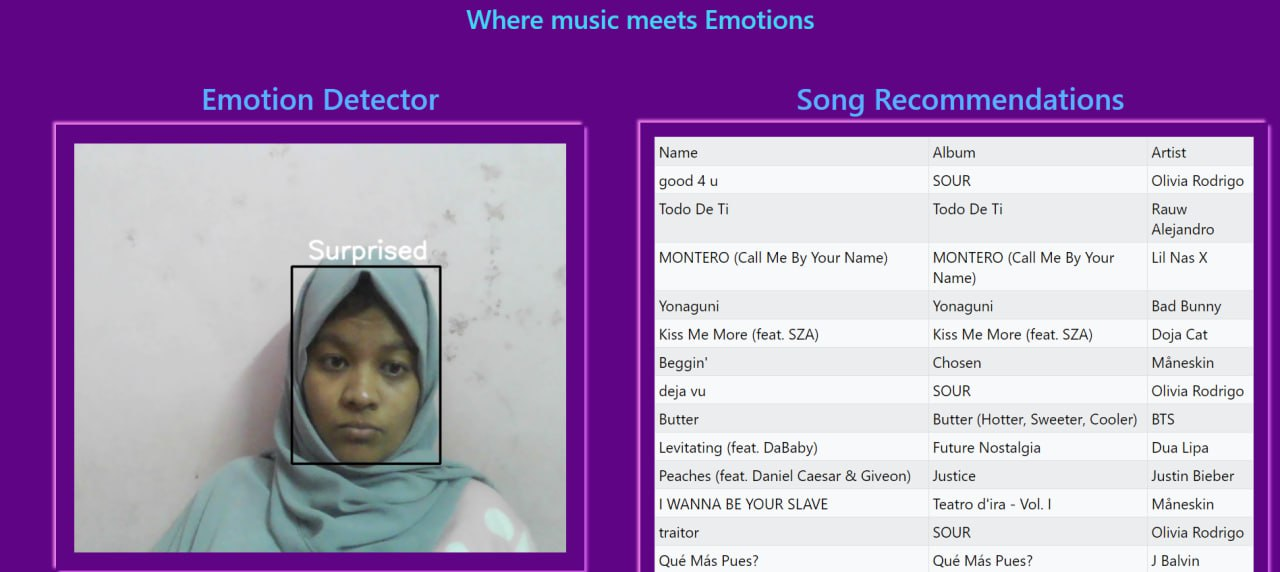


Figure B.7: Surprised Face

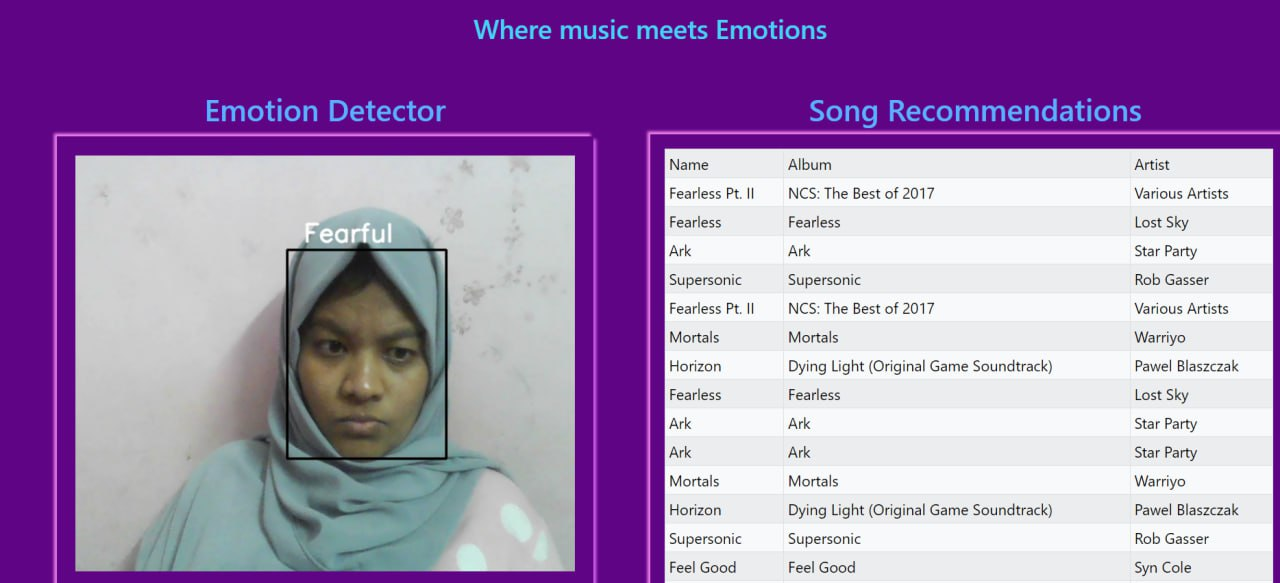


Figure B.8: Fearful Face

The code defines the "VideoCamera" class, which is responsible for capturing video frames, performing emotion recognition, and recommending music. The class uses the face cascade classifier to detect faces, extracts the facial region of interest (ROI), and feeds it to the emotion model for prediction. It also retrieves the corresponding music recommendations from the CSV files based on the detected emotion.

**Appendix C**

**Data sets used in the project**

We have used Facial Emotion Recognition (FER-2013) [Dataset](https://www.kaggle.com/datasets/msambare/fer2013/data). The dataset is split into testing and training sets. 80% of the data is used in the training set to train the model. 20% of the data is available in the testing set. Both training and testing sets are again classified into 7 different folders based on emotions.



Figure C.1: Dataset

The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centred and occupies about the same amount of space in each image.

The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples and the public test set consists of 3,589 examples.