International Journal of Cyber Behavior, Psychology and Learning
Volume 11 • Issue 3 • July-September 2021

Statistical Analysis of Human Emotions to Suggest Suitable Music as per Individual's Mood:

An Application of AI and ML for NextGen Smart Cities

Rohit Rastogi, Dayalbagh Educational Institute, India & ABES Engineering College, India
https://orcid.org/0000-0002-6402-7638

Prabhat Yadav, ABES Engineering College, India

Jayash Raj Singh Yadav, ABES Engineering College, India

ABSTRACT

There is music recommendation software and music providers that are well explored and commonly used, but those are generally based on simple similarity calculations and manually tagged parameters. This project proposes a music recommendation system based on emotion detection of users, automatic computing, and classification. Music is recommended based on the emotion expressed and temper of the user. Like artists and genre, emotion of the user can also be a crucial recommendation point for music listeners. The different moods in which the system will classify the images are happy, neutral, and sad. The system will pre-sort the songs according to their genre in the above-mentioned categories. This research project gives us advancement in the music industry with the help of machine learning and artificial intelligence and will reduce the hassle of selecting songs in our leisure time and will automatically play songs by detecting the emotion of the user. This data can be used to play the songs that match the current mood detected from the provided input by the user.

KEYWORDS

Artificial Intelligence (AI), Emotion Recognition, Emotion-Based Music Player (EMP), Face, Face Recognition, Facial Expression, Feature Extraction, Machine Learning (ML), Music Suggestion, Python

1. INTRODUCTION

1.1 Motivation

The Research team have always loved to use smart devices, be it google assistance, Alexa, Cortana, or Siri. Authors were fascinated by the way they reacted when asked something, using voice and how beautifully they responded to that, how a machine can identify different voices and faces among billions of voices and faces and it was always a desire to create something similar.

In today's era of IoT and AI, one thing which authors thought was lacking is that they are unable to judge one's mood or emotion and respond accordingly. It is always lovable to listen to music but the traditional ways of choosing and making playlist are outdated, in today's era when most of the

DOI: 10.4018/IJCBPL.2021070104

things can be connected to an AI why can't the music apps be like that, which can play according to one's emotion and mood, therefore, this research work and project was thought a long time ago but due to our own limitation of not knowing how to use ML or Python we could not complete the project then, but when this opportunity came up, the team were excited to make that wish of communicating with the devices using self and family faces a reality.

As music lover's, old ways of playing and selecting songs is time-consuming, therefore this project could solve these issues and therefore authors decided to work on this project.

1.2 Project Objectives

Music Listeners must waste a lot of time and effort in segregating and creating a playlist manually for different moods, they must waste a lot of time deciding what to play and what not to play according to their mood at that time. In the Present existing application's, processes of music organization occur using a playlist, and playlist's songs cannot be altered or modified in single click; you must manually go through each song then have to delete or add. In present times, no applications can be found that allows us music listeners to play songs without having to select the songs manually. This research project objective is to overcome all the above hardships faced by music listeners and to save their time which goes wasted while selecting songs.

This music player based on emotion makes it mandatory for the user to have an account to be able to use the application. The user of the application needs to give permission to the application to be able to get direct access to the camera of the device on which the application is running currently, to capture their live photo and then accordingly the song which is fed in the player is played without the user intervention according to his/her mood.

This project makes the life of all music listeners easier and aims to automate the process of music playing and makes the process a lot of fun. It plays the songs based on the current emotional state of the person who is using the application and in turn automates the process of playing music.

1.3 Scope of Project

Music player based on emotion is a useful application for all music lovers and listeners and can be accessed by any pc or mac device for now. The application can be accessed by anyone who has created a profile by signup in the application. The application can do the following functions, for now, these include:

1.Create a user account (sign in or sign up)
2.Playing songs by following ways:
   a. Randomly
   b. By creating a Queue.
   c. By Identifying emotion
      3. Capturing emotion using the camera

1.4 Scope of Research work

1. The project is still in the early stages and therefore adding and removing songs is still not recommended for most users as someone needs a little knowledge of JS and HTML to do so. In the future, this process of adding and removing songs can be a lot easier.
2. The application cannot be run on android or iOS, it can only be run on Windows, Linux, and macOS. In the future, this application may be available for android and iOS.
3. This application cannot be integrated with other AI as of now but in the future, this feature may be available.

4. The application only identifies correct emotions 66% of the time due to its small dataset of 414 images used to train the algorithm but the efficiency can be increased by using a larger dataset and training under different lighting conditions and different circumstances.
5. This can be used to deal with a depressed person and provide him/her assistance by integrating with AI.
6. There are a lot more things that can be achieved by this project but are beyond the scope of this report.

1.5 Need of Emotion Detection

Emotion plays an important role in our day to day communication and most of our non verbal communication comprises our emotions so understanding of emotion is vital for our interpersonal relationships (Varghese, A., et al., 2015).The recognition of emotion has been in discussion for very long various research have been conducted as to how we can interact with computers using our emotion. Understanding emotions could greatly increase our ability to deal and react with people expressions and it could help in treating various mental illness like depression etc. (Litman, D., et al., 2004). Emotion Detection can help in creating friendly robots, AI machines and apps which could help ease people's lives by providing emotional support and these could help reduce cases of depressions and suicides (A. Yashwanth, et al., 2020).

1.6 Global Market of Emotion Detection

The market of Emotion detection is huge and is growing at a rapid pace it is a billion dollar industry, as of 2020 the global market of emotion detection is nearly about USD 19.5 billion which is expected to grow to a size of USD 37.1 billion in a span of 6 years at a CAGR (Compound Annual Growth Rate) of 11.3% (Markets and Markets Survey Report, 2020), several other research conducted even predict a very high growth of about USD 65 billion till 2023 (Sightcorp report on emotion detection, 2019), the major factors contributing to growth of emotion detection is need of automated emotion detection AI to analyze emotional states, rapidly rising market of IOT(Internet of things),demand in Automotive AI industry, growing need of socially smart and intelligent artificial agents and machines (Markets and Markets Survey Report, 2020).

1.7 Software and Companies and Progress of Emotion Recognition

The companies use emotion recognition in various ways and the progress made in this field is appreciable, adopting emotion recognition into business could help the companies understand the perspective of the customers and their users which could help them give better services and build better products or improvise based on the emotion of the customer/user (Sightcorp report on emotion detection, 2019).

The various ways in which the companies used emotion recognition are:

1. *Market Research:* The companies do research on mood of the people and based on the data collected they make changes in their product or services. Like film industry takes in data of emotions of people watching movies in real time and then based on those data they modify the movie or select theme for their movies which help in making the movie a success.
2. *Digital Advertising:* In the modern-day advertising industry has gained a lot of success and this has contributed to the advertisement based on preferences of people rather than a large advertisement which is not focused. The emotion recognition helps companies to identify which product to be recommended to which person or which person is likely to buy which product based on his/her emotion. This itself created a billion dollar industry.
3. *One-to -one interviews:* Not all companies employ the emotion recognition to sell products, but some apply it to choose the best candidate suitable for a job by looking at his/her personality or

state of emotional intelligence and this has helped companies to get better people for different jobs(Sightcorp report on emotion detection, 2019).

4. *Health Care industry:* Emotion recognition helps in treating various problems and diseases and the health industry on a large scale to make different health products (Sightcorp report on emotion detection, 2019).

The uses of emotions recognition in industry and business are far beyond what could be discussed here.

1.8 Applications of Emotion Recognition Based Software

There is a vast field of applications of emotion recognition and its related software, as we humans are influenced by emotions and our emotions decide our actions and how we react to them. These emotions can be used to do more productive work by use of various of software that are designed to make use of our emotional state (Kołakowska, A., et al., 2013). Few applications of emotion recognition software are playing music based on the emotional state of a person through a music app integrated with AI, providing emotional support to a person by use of an emotional intelligent artificial agent or chatbot, dealing with people with mental illness using emotional recognition software and machines, advertising industry uses software to promote products and services based on emotion of a person (E., Kodhai et al., 2020).

2 RELATED PREVIOUS WORKS (LITERATURE REVIEW)

There is ample research and work carried out in the field of emotion detection from faces. Some relevant work that was surveyed was mainly industry standard and leading research in the field. Jones and the viola proposed a robust way to detect human faces in the image They introduced a new way for representation of images known as "Integral Images". They implemented an efficient and simple classifier by using the algorithm Adaboost for selecting from a large set of features, a smaller number of critical features. Lastly, they introduced the combining classifiers method in a cascade-like structure which quickly neglects and discards background areas of the image (Abat, F. et al., 2011; Davis, E, 2009).

After searching a lot, the research team found no application in the market which plays the songs just by looking at the mood or facial expression of person but there are various good music applications which give the users various features, some of them are:

1. *Spotify*- This application gives the user accessibility features like playing songs from the top song list or already built playlist and recommending users based on their previous songs, which they have listened to, they also provide a feature like playing the songs in a loop or playing songs in queue.
2. *Saavan*- This application is also like Spotify and gives the user accessibility features like playing songs from the top song list or already built playlist and recommending users based on their previous songs, which they have listened to, they also provide a feature like playing the songs in a loop or playing songs in queue (Z. eduard, et al., 2020).
3. *YouTube Music* - All the above features which are provided by Saavan or Spotify are also provided by YouTube music apart from these features YouTube music comes with inbuilt support of google home (Which is google assistance) and can be controlled by using voice and it also makes playlist automatically for the user and recommends songs or music based on previous choice of the listener.

4. *Amazon Music*- Apart from all features mentioned above in various music applications. It comes with inbuilt support of Alexa which is amazon AI assistance and can be controlled by using voice and can relate to Echo Dot which is an AI assistance device of Amazon.

Special mentions are: moodfuse, stereomood, musicovery.

These all provide various features like creating a playlist, playing songs from the playlist in the loop, and a lot of other features but cannot play by just looking at facial features.

2.1 Emotion Based Music Player

As a means of communication songs, have always been a favored way for expressing and understanding people's feelings. Emotional planning programs can do much to help us understand their intention and meaning research, however into the field of related to emotion recognition and emotion recognition-based songs did not produce the best results. Through this paper, the team introduces a country music or song player, EMP, which recommends songs based on emotion and previous and current behavior of the user.

EMP provides high quality music recommendations by incorporating emotional content within our music recommendations program. This music player has three main modules: Music Split Module, Recommendation's and Emotion's Module. The module of emotion captures and takes the face of the user as input and then uses the provided algorithm to identify the emotion of the user based on the image taken with closer to 90.2% accuracy. The SplitMusic Module uses audio features to get amazing 97.6% results while splitting songs into four different sensory subjects. The Module for recommendations suggests users song by inserting a map of their feelings into the emotional song type, taking into account users' preferences (Gilda, S., et al., 2017; B. Balasubramaniam et al., 2019).

2.1.1 Research Gap

In the above musicplayer, the users must manually categories the song in different folders of emotions and this task is time consuming, further features like playing songs in queue is not available.

2.2 Automated Agent for Music Selection

This paper aims to suggest a clever agent who filters the collection of music and songs based on the emotional feature presented by each song and gives suggestion of a playlists that is right for the music listener based on his or her current emotion/mood. The user's local album can be assembled in different ways to represent different emotions conveyed by the songs. These counts are based on song lyrics, as well as an music. Whenever a user wishes to produce an emotional playlist, the user automatically downloads them. This photo is aimed at facial recognition and emotional awareness, to see the feelings of the user. Music like these emotions is recommended to the user as a playlist (Ramanathan, R. et al., 2017; (Z. Rzayeva, et al., 2019).

2.2.1 Research Gap

The songs are not played automatically they are just categories in different playlist based on different emotions captured while user was listening any song, but it is the user who must select the song and play it manually.

2.3 Labelling and Classification of Music by ML Algorithm's

As a genre of music, moodoremotion is a useful attribute in the music industry and would enable listeners to be able to automatically organize their music according to emotions and produce labels for song. Describing effort in organizing song categories emotionally with the help of various matador typing machines, namely Fisherface algorithm, and SVM (Support Vector Machine) classifiers, Random Forest. Inorder to be able to automatically assign several emotions built for the removal of

whether a song faces a challenge in this emotion. Duplicate confirmation of repetition in each phase of each emotion and from the tyres of classifier: Fisherface, SVM (Support Vector Machine) and Random Forest. For this study, it is possible that emotions recognition can work well, although the procedure certainly can to refine further (Doris, B., 2006; S. Sharma, et al., 2020).

2.3.1 Research Gap

The above research gives an idea of how to label different songs based on different mood of user while user was listening to them, it does not use the face captured to play songs or create playlist.

2.4 AI Based Application to Detect the Music as per Face and Mood

The research proposed on the project Emo Music player is a novel approach towards a great step of using machine learning and artificial intelligence in providing a different experience to music listeners based on their emotions. It detects the expression of the human face and plays music accordingly. Emotions are scanned through webcam and processed using Support Vector Machine (SVM) algorithm. The SVM algorithm matches the expression of the listener with the pre-stored dataset. The dataset used in this research is trained over 212 images of models with different face expressions. Webcam captures the user's image then extracts the attributes which are necessary for expression comparison. The extracted attributes are matched with the trained model with a dataset to provide values say 1 for happy 0 for not happy and then the song is played according to the user's expression (Hemanth, P. et al., 2018; Y. Gupta, et al., 2021).

2.4.1 Research Gap

The above research aims to suggest music to user based on his/her current emotional state, it does nothing more than suggesting songs. Furthermore, it uses SVM which gives lower results as compared to Fisher face Algorithm under low light conditions and SVM algorithm is not suitable for large data sets.

3. EXPERIMENT SETUP AND METHODOLOGY

The various features which are available in today's music players are:

1. Manually selecting a song
2. Shuffling the playlist
3. Creating playlists and adding songs
4. Music queues

But in all these features user have to physically select a song based on his/her mood. Users listen music based on their mood and temper for that they have to manually select the songs. Using the traditional music players would be a hectic work for some users as they have to select and play each and every song based on their mood. In today's world of artificial intelligence and machine learning music players have been evolving day by day to give ease to the listener with better music quality, rewind and suggestions of music based on recently listened, location etc.

Although these features are more than enough for music listeners, still the user has to pick each song and play or add to his/her created playlist manually based on their mood and the type of music they want to listen. This research proposes a new advanced era of music suggestion for music lovers that will completely change the music listening experience of the user. By this an image is captured of the user face which then is processed under fisherface algorithm to detect the mood of the listener and then the music playlist is suggested.

3.1 Steps Followed

1. Gathering data for dataset.
2. Preparing that data for analysis.
3. Choosing a model for face recognition.
4. Training the model for emotion detection.
5. Evaluation of emotion.
6. Hyper parameter tuning of dataset.
7. Prediction of music based on emotion.

3.2 Requirements

3.2.1 Hardware requirements

- RAM: 500 MB (1 GB recommended)
- Processor: 1.2 GHz(minimum)
  ●Storage:200MB (500MB recommended)

3.2.2. Browser

- Chrome 51 or higher (latest recommended)

3.2.3. Database

- A .csv extension file which is auto generated by the python code is required to store username and password.

3.2.4. Software requirements

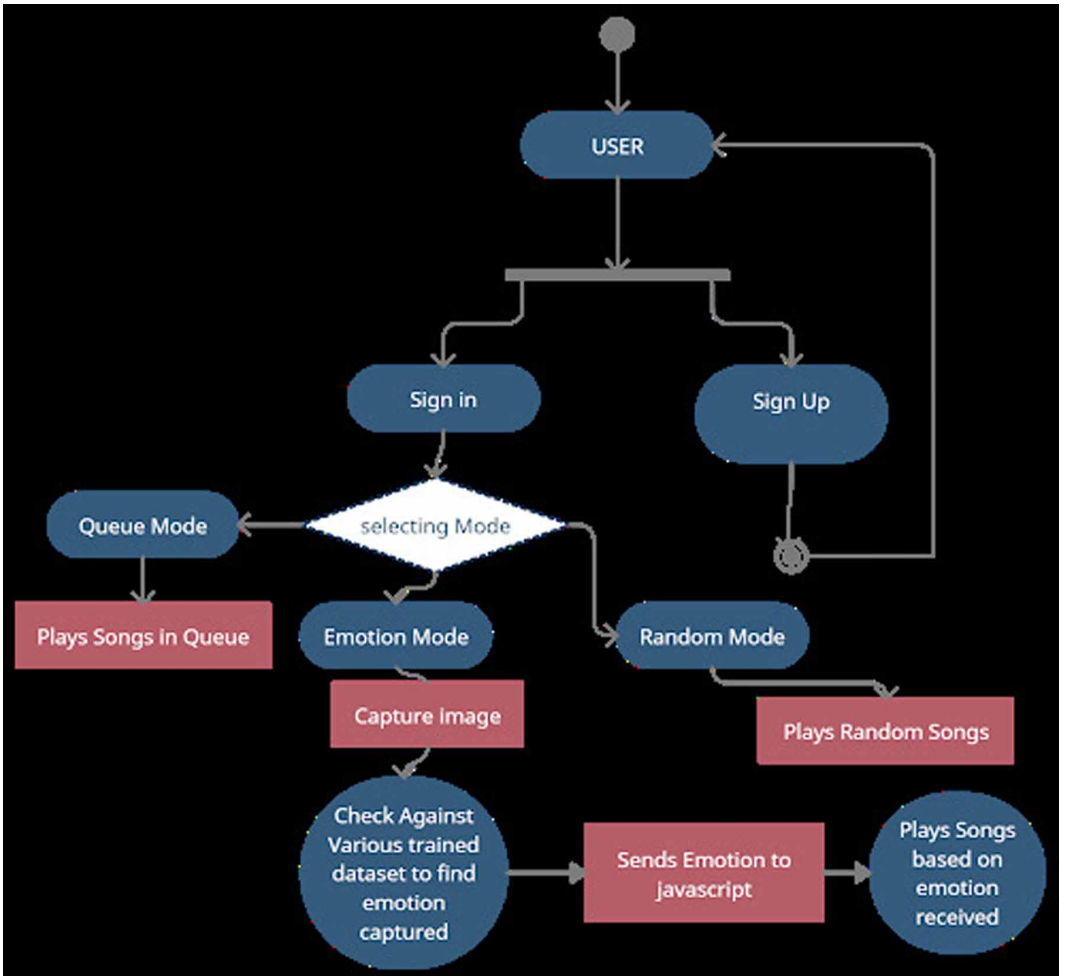
●OS requirement:
●Windows 7 or higher
●Ubuntu 14 or above
●macOS 10.12 or above recommended
●Python2.8 or above (python 3 or above recommended)

3.3 Libraries Used

1. *OpenCV*- This Python library can be used to create real-time computer vision applications. It mainly focuses on video processing, image processing, and identifying face and objects in the image and in this project, we have used this to deal with expressions of the face.
2. *Argparse*- This python library is used to create CLI (Command Line Interface) so that the user can give instruction to the python code from the command line itself.
3. *Eel*- Eel library makes it easier to work with HTML and python helps to interact with python using chrome, helps to create a beautiful user interface. More like using HTML, CSS, JS as frontend, and python as backend.
4. *Tkinter*- This python library is used to make a graphical user interface (GUI) in python.

Other python libraries used are - time, os, glob, random.

Figure 1. Activity Diagram of the emotion based software



3.4 Diagrams

3.4.1 Activity Diagram

(As per Figure 1) This is the activity diagram of how the user can use or access the emotion-based music players and its various features that are provided in the software. The user must firstly create and it then the user can access the application and play songs using various modes provided.

3.4.2 Flow Chart

(As per Figure 2) This Flowchart shows how the control moves in the software from user to music player and how the music is played based on the emotion of the user and how the user can select various modes available in the player.

3.4.3 Gantt Chart

(As per Figure 3) This Gantt chart shows our approach of how we were successfully able to complete the project and its research in time and how much time we took to complete each task in the project.

Figure 2. Flow Diagram for the Emotion based Software

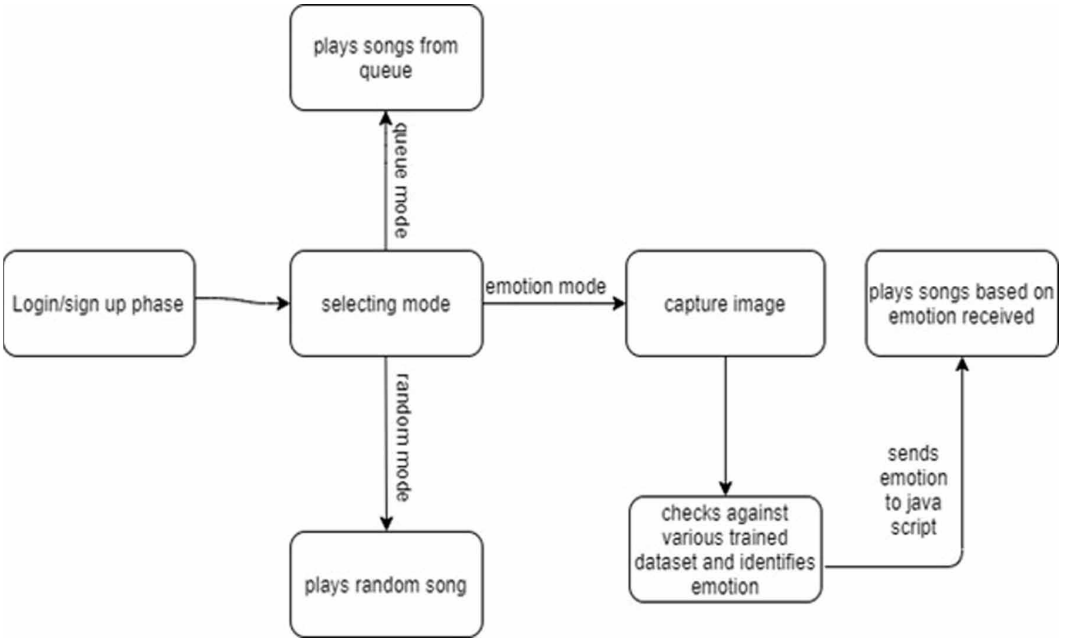
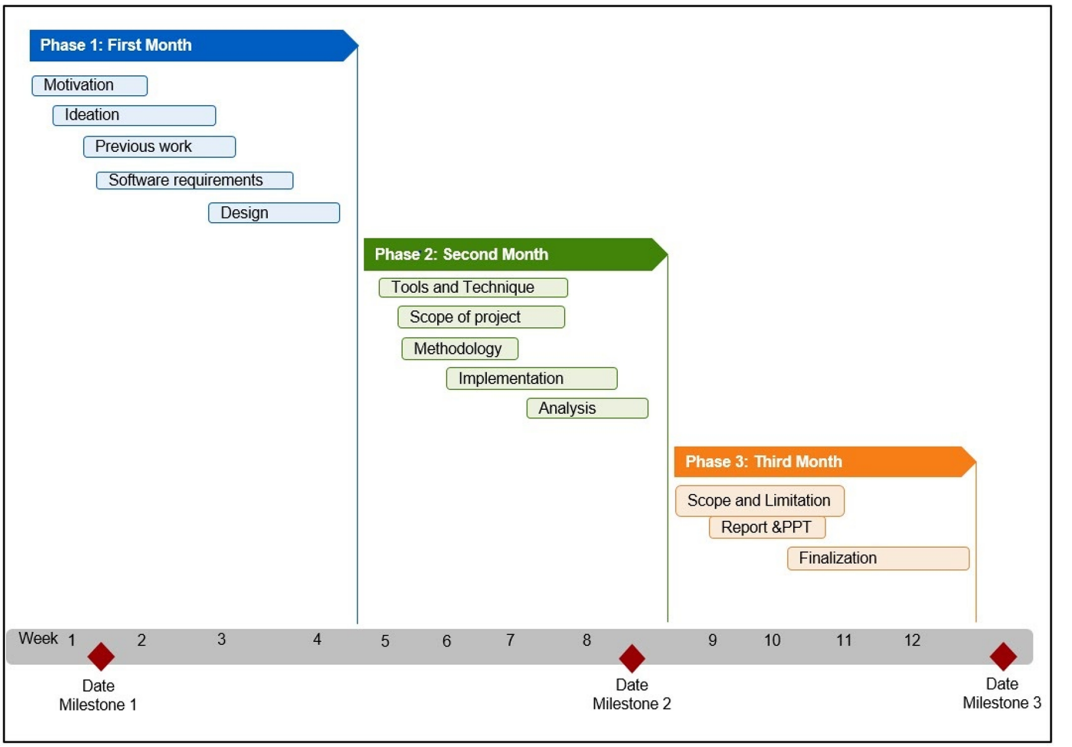


Figure 3. Gantt Chart for the Research and Software Development



4. RESULTS AND DISCUSSIONS

The unequal number of images for various emotions causes some emotions to become dominant and others remain suppressed due to some emotion's dataset being large and some smaller. This error could be solved by taking an equal number of the dataset for all emotions and then training the model.

The overall system can be divided into various logical stages like capturing an image, detection of a face, detection of landmark points on the detected face, classifying those feature points with the help of the OpenCV haarcascade_frontalface.xml, and then generating the playlist according to that recognized mood. During the training phase, a dataset of images will be created to train the model. xml while after implementation of the system a single captured image can be given to a trained model file to predict the mood. The different moods in which the system will classify the images are happy, neutral, and sad. The system will pre-sort the songs according to their genre in the above-mentioned categories.

Model Training for different Emotions

1. Different images are collected for different emotions- The team collected a total of 414 images for anger, sadness, happy and neutral emotion from various sources google images, stock photos, and other various sites than trained the data set all those images sources are as follows.

www.pexels.com, www.shutterstock.com, www.pixabay.com.

2. Cropping the images to face only images and excluding the background and other objects in the image using the crop function in python made by using OpenCV library.
3. Turning the image into a grayscale image to make the processing easier and faster using OpenCV grayscale function.
4. After that extracting the faces grayscale now extract the facial expression using haarcascade_ frontalface.xml from OpenCV library and save them in a model.xml file.

After having trained the model for all faces, capture an image of the user, and check the image features against our trained model if it matches against an emotion then return that emotion and the songs are played accordingly.

This image will be captured and processed by the python code and converted into a 350x350 grayscale image for checking its emotion against the trained model.

As per Figure 4, this is the output image that is received and is checked against the trained model and emotion is returned for the same image.

For the current image anger is returned.

As per Figure 5, the input image is cropped to face and converted into a grayscale image for easy processing as less resources will be used to process a grayscale image than a colour image.

Figure 6, shows how the algorithm is trained and songs are played in the music app for different emotion (instead of SVM classifier we have used fisherface classifier from the opencv library available in python) Few examples of trained faces from the dataset are:

All images sizes in the dataset are 350x350 pixels in grayscale

4.1 Angry Gesture

As per Figure 7, this figure shows few of the images used in the dataset of angry images. They have been cropped and grayscale for easy processing by the CPU. A total of 164 images have been trained for angry emotion.

Figure 4. Input Image Captured by the Software

Figure 5. Output of the Image, processed for Emotion

4.2 Happy Gesture

As per Figure 8, this figure shows few of the images used in the dataset of Happy images. They have been cropped and grayscale for easy processing by the CPU. A total of 100 images have been trained for happy emotion.

4.3 Sad Gestures

As per Figure 9, this figure shows few of the images used in the dataset of sad images. They have been cropped and grayscale for easy processing by the CPU. A total of 33 images have been trained for sad emotion.

Figure 6. Mentioned steps for Face recognition and Mood Detection

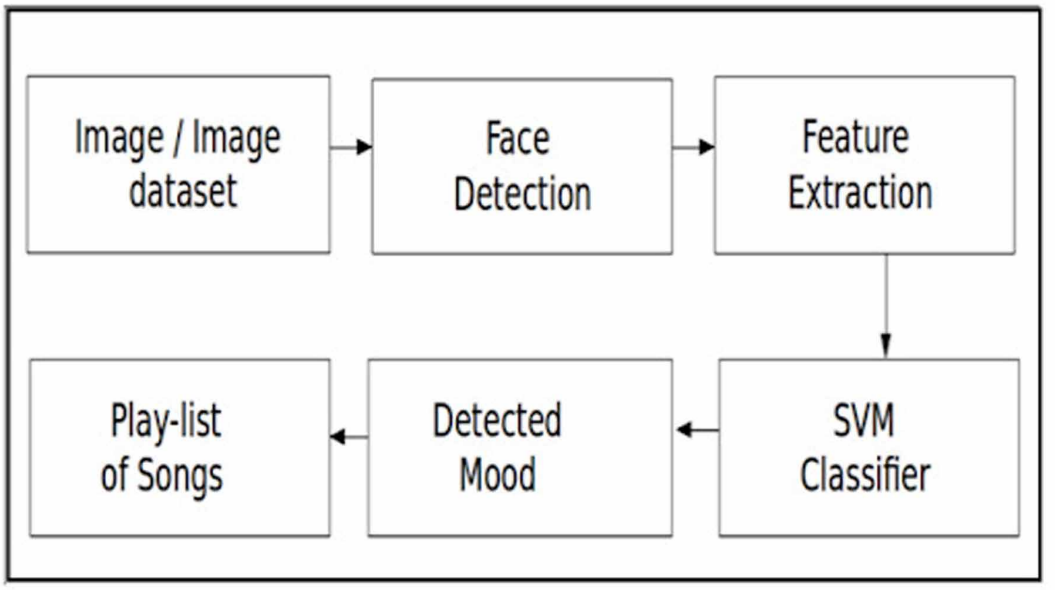


Figure 7. Angry Emotion Trained Faces

4.4 Neutral Gestures

As per Figure 10, a total of 121 images have been trained for neutral emotion.

Combining all these images we form a complete dataset of images against which we can train our model and increase its efficiency by increasing the number of images in the dataset.

Figure 8. Happy emotion trained faces

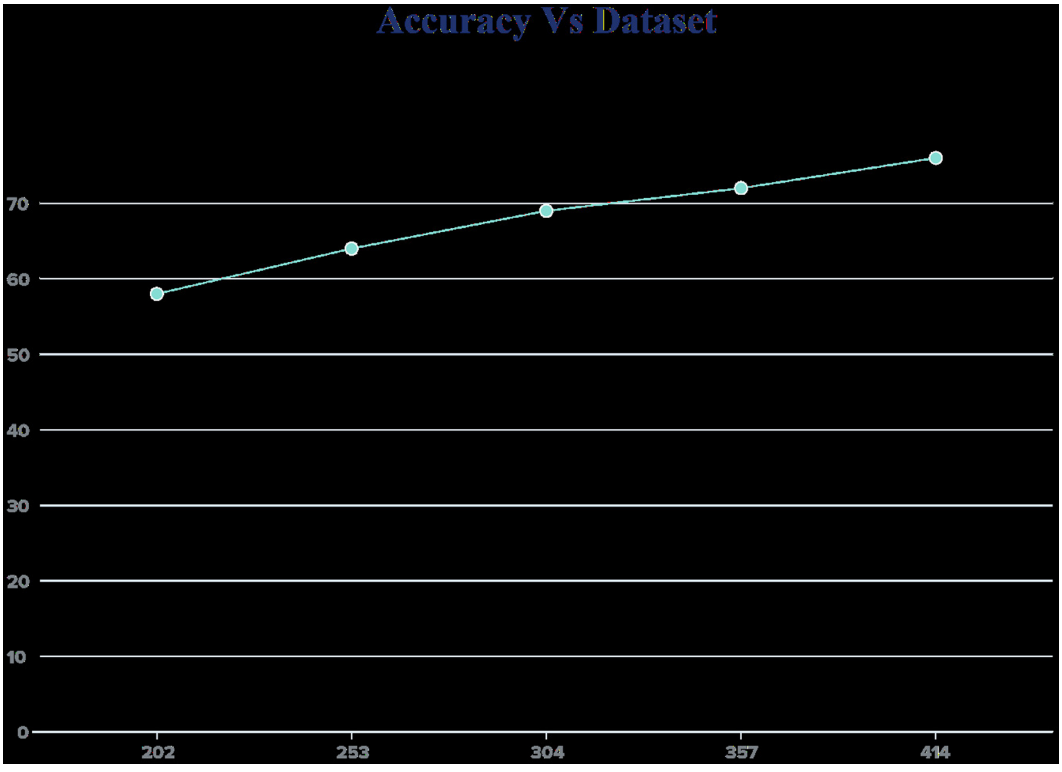
Figure 9. Sad Emotion Trained Faces

4.5 Accuracy Discussions

As per Figures 11, 12, and 13, as we can see from the graph obtained under different numbers of images in the dataset the accuracy changes, if the number of images in the dataset are less the accuracy is low as the images number increases so does the accuracy. When images in dataset were 202 the accuracy was closer to 58%, as the images were increased to 253 the accuracy was increased to 64%, when 304 accuracy was closer to 69%, when images number was 357 accuracy was 72%, when images number 414 accuracy was closer 76%. From the results it's easy to conclude that as we go on

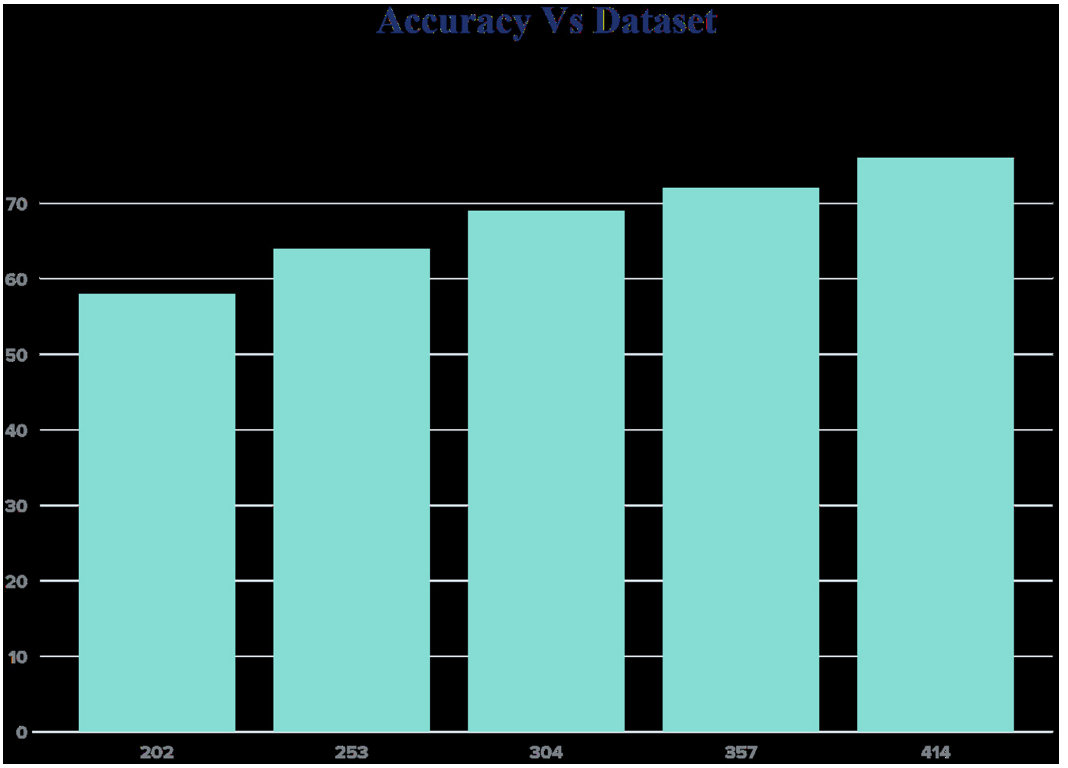
Figure 10. Neutral emotion trained faces

Figure 11. Graph to display the accuracy vs. dataset on images line Chart



increasing the images number we can increase the accuracy. Which would help in better recognition of emotion and suggesting and playing songs accordingly.

Figure 12. Graph to display the accuracy vs. dataset on images bar Chart



Accuracy also depends on light conditions if the luminous intensity increases the accuracy for proper emotion detection increases and as light intensity decreases so does the accuracy.

5. RECOMMENDATIONS

- Music listeners do not have to select the songs; it will be an automated process.
- Easy to Use.
- We don't have to add music in a playlist; it will be done by software itself.
- Free For All.
- Music listeners do not have to select and play each and every song themselves.
- Account Creation to keep user data safe.

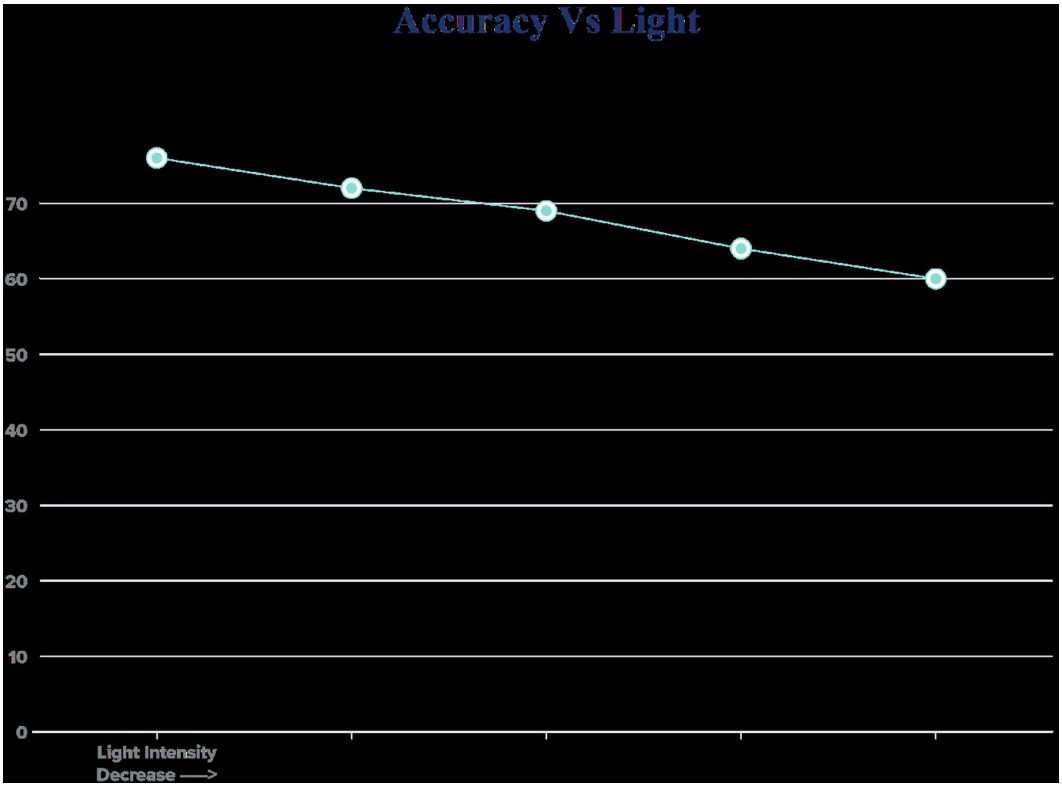
Some generic questions which can run in the mind of audiences and readers of this manuscript has been answered in advance in below tabular structure.

Some technical and legal queries have been addressed (Pl. refer Table 1).

6. NOVELTIES

As per Table 2, from the research done in current music player industries it is clearly evident that none of the music giants like Spotify, Savan, amazon uses emotion detection-based music player but this research has potential to revolutionize the music industry to a whole new level of artificial intelligence. Even Though the research is in a budding state and may have less availability of music

Figure 13. Graph to display the accuracy vs. Lightning conditions of dataset images



collection and stability but in future this research has potential to completely change the music listening experience of the user.

7. FUTURE RESEARCH DIRECTIONS AND LIMITATIONS

The efficiency of the project code is closer to 76%, the face emotion captured correctly out of 105 trials was 89, and 16 times the captured emotion was false due to various reasons. The prominent reason being the smaller dataset which is of size only 414 images. The lower dataset results in poor identification of the user emotion. This error could be solved by using a larger dataset of images for model training and could increase efficiency tremendously (Y. Cheng, et al., 2021).

The Quality of the webcam affects the output of the code if the webcam is of a low quality than the captured emotion may not be true due to poor expression capture by the webcam. This error could be solved by using a good quality webcam for capturing the user's image.

The lightning condition affects whether the captured image will be truly identified for emotion or not if the captured image is too bright or I too dim light the emotion may not be identified correctly. Having a moderate lightning condition could easily solve this issue. The lighting conditions should not be too dark or too bright (R. Rahul, et al., 2021).

7.1 Future Scope

- Developing Android application of the project
- Developing Desktop application of the project

Table 1. Explanation of user related specific features and comments

Specific Features and Comments	Performance and Evaluation by Proposed software	Remarks
How is proposed work different from those apps..	1. User do not have to select the playlist manually based on the mood it will be selected automatically, and a song which is most preferred by the user will be played from the genre, over time song choices will improve. 2. Users do not have to add music in a playlist, it will be done by software itself from vast numbers of songs available online and from users' behavior of songs selection.	Applications like Wynk Music, YouTube, etc., come with a playlist based on genre, artist, mood, etc., like the romantic playlist, rock songs playlist, etc. Users do not want to select songs manually.
How efficient is proposed system in capturing emotions.	On an average the accuracy of the current version of the proposed system is closer to 84-87% under different condition of lighting and datasets size, it can be increased by employing good quality devices such as better camera for capturing face, increasing the dataset size, and training the model and doing so requires better hardware resources which we could not employ due to our own limitations of less funds.	
People can even fake mood. Is the proposed project handling such false positive as well….	Using complex methods such as monitoring users voice pattern, using radio waves to monitor heartbeat, and breathing pattern, false positive can be handled to some extent, but employing it in our software is not feasible. we are just using a camera to capture face, using other devices mentioned above could increase our cost to many folds, even as humans, sometimes we ourselves cannot differentiate between fake and real emotions.	
Possible ethical issues for the proposed application. how can the users face data be protected…	The user data in current version of proposed application is not using any server resources to store or process any kind of data, it is achieved locally on the users device .However in future we will be using servers but we will instantly destroy any image that might be processed over our servers and if the user doesn't want to give permission to use server to process the image, we will do so locally on user device but it will use lots of users resources and it will be user's decision.	It has been well taken care off and executed.

Table 2. Comparative Analysis of Emotion Detection Software with Existing Products

Software	Playlist	Library	Quality	Queue	Emotion/Mood
Proposed and Presented Research	Y	Y	Y	Y	Y
Spotify	Y	Y	Y	Y	N
Hungama music	Y	Y	Y	N	N
Gaana	Y	Y	Y	Y	N
Jio savan	Y	Y	Y	Y	N

- Developing Web Application of the project
- Reducing Execution time the program
- Developing cloud storage for data set to reduce burden of user device

7.2 Limitations

1. Adding and removing songs is a tedious job as someone with knowledge of JS, Html can only achieve it.
2. This expression detection is not accurate because of the small set of data used to train the model.
3. This can't be run on android or IOS as for now.

8. CONCLUSION

Emotion based music player software could bring about a change in the way we look at music players and software as a whole, implementing it in a perfect way could be a long way to go. The research could be beneficial for people who are interested in emotion recognition to suggest music. There are obviously limitations to what could be done and what could not be, as these types of software and research progress these limitations can be overcome.

From the graph in Figures 11,12 and 13, it is evident that the accuracy depends on datasets as well as lightning condition when using the software as we increase the size of dataset the accuracy increases and vice versa. In case when luminous intensity while capturing the image of user is low than the accuracy is low and when luminous intensity is high while capturing the image of subject the accuracy is higher. The results of the software are not perfect but it could be improved over time, the accuracy we could achieve was closer to 76% but using good quality webcam's and large dataset and good lighting conditions we could drastically increase the accuracy of the software .

ACKNOWLEDGMENT

Completion of this work would not have been possible without the help of so many people whose names could not possibly have been recorded. Their contributions are sincerely appreciated. However, the party would like to express its deepest gratitude and indebtedness especially to Prof. (Dr.) Pradeep Kr. Singh, HoD-CSE and Prof. (Dr.) Shailesh Tiwari, Director-ABESEC for giving us this opportunity to work on this project and guide us throughout. Owed to Mr. Rupesh Yadav from DPSG for giving us guidance on working on Tkinter. Thanks to Harish Kumar from ABESIT for providing his guidance and valuable support. To all the relatives and friends who somehow shared their valuable support, thank you. Above all, the Almighty, the author of all knowledge and wisdom, in His infinite love.

REFERENCES

Abat, F., & Maaoui, C. (2011). Human-computer interaction using emotion recognition from facial expression. *UKSim 5th European Symposium on Computer Modeling and Simulation,* 196-201. doi:10.1109/EMS.2011.20

Balasubramanian, B., Diwan, P., Nadar, R., & Bhatia, A. (2019). Analysis of Facial Emotion Recognition. *Proceedings of 3rd International Conference on Trends in Electronics and Informatics (ICOEI),* 945-949, doi:10.1109/ICOEI.2019.8862731

Cheng, Y., & Meng, H. (2021). Research and implementation of network information security management system based on face recognition. *Proceedings of IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE),* 294-302. doi:10.1109/ICBAIE52039.2021.9389893

Davis, E. (2009). A Machine Learning Toolkit. *Journal of Machine Learning Research*, *10*(1), 1755–1758.

Doris, B. (2006). EmoMusic. *Classifying Music According to Emotion.*, *1*, 215–120.

Eduard, Z., Lucian-Mihai, C., Mihai, N., & Mihai, D. (2020). Detection of Emotional States Through the Facial Expressions of Drivers Embedded in a Portable System Dedicated to Vehicles. *Proceedings of 28th Telecommunications Forum (TELFOR)*, 1-4.

Gilda, S., Zafa, H., Soni, C., & Waghurdekar, K. (2017). Smart music player integrating facial emotion recognition and music mood recommendation. *Proceedings of International Conference on Wireless Communications*, 154-158. doi:10.1109/WiSPNET.2017.8299738

Gupta, Y., Prasad, A., Touti, S., Sachdev, K., Jaiswal, V., & Naranje, V. (2021). Real-time face recognition: A survey. *Proceedings of International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 430-434. doi:10.1109/ICCIKE51210.2021.9410792

Hemanth, P., Adarsh, Aswani, C.B., Ajith, P., & Veena, A. K. (2018). Emotion Based Music Player. *International Research Journal of Engineering and Technology, 5*(4).

Jones. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference*, 1, 511-518. doi:10.1109/CVPR.2001.990517

Kodhai, E., Pooveswari, A., Sharmila, P., & Ramiya, N. (2020). Literature Review on Emotion Recognition System. *Proceedings of System Computation Automation and Networking (ICSCAN) 2020 International Conference*, 1-4.

Kołakowska, A., Landowska, A., Szwoch, M., Szwoch, W., & Wróbel, R. M. (2013). Emotion recognition and its application in software engineering. *6th International Conference on Human System Interactions (HSI)*, 532-539. doi:10.1109/HSI.2013.6577877

Litman, D., & Riley, K. (2004). Predicting student emotions in computer-human tutoring dialogues. *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 351-357. doi:10.3115/1218955.1219000

Markets and Markets. (2020). Emotion Detection and Recognition Market by Component (Solutions [Facial Expression Recognition, Speech & Voice Recognition] Services), Technology, Application Area, End User. *Vertical, Region - Global Forecast to 2026.* https://www.marketsandmarkets.com/Market-Reports/emotion-detection-recognition-market-23376176.html

Rahul, R., Ranga, P. V., Shravan, S., & Ajjaiah, H. B. M. (2021). Instant Facial Recognition of Criminals Using a Smart Wearable Tech with Gesture Control from a IoT (Internet of Things) Cloud Database. *Proceedings of 6th International Conference on Inventive Computation Technologies (ICICT),* 190-194. doi:10.1109/ICICT50816.2021.9358637

Ramanathan, R., Kumaran, R., Rohan, R., Gupta, R., & Prabhu, V. (2017). An Intelligent Music Player Based on Emotion Recognition. *Proceedings of 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, 1-5. doi:10.1109/CSITSS.2017.8447743

Rzayeva, Z., & Alasgarov, E. (2019). Facial Emotion Recognition using Convolutional Neural Networks. *Proceedings of IEEE 13th International Conference on Application of Information and Communication Technologies (AICT),* 1-5. doi:10.1109/AICT47866.2019.8981757

Sharma, S., Bhatt, M., & Sharma, P. (2020). Face Recognition System Using Machine Learning Algorithm. *Proceedings of 5th International Conference on Communication and Electronics Systems (ICCES),* 1162-1168, doi:10.1109/ICCES48766.2020.9137850

Sightcorp. (2019). *Emotion recognition, face analysis, mood estimation*. https://sightcorp.com/blog/3-ways-companies-are-using-emotion-detection-technology/

Varghese, A., Cherian, J., & Kizhakkethottam, J. (2015). Overview on emotion recognition system. *International Conference on Soft-Computing and Networks Security (ICSNS)*, 1-5. doi:10.1109/ICSNS.2015.7292443

Yashwanth, A., Vishwanath, B., & Sairam, R. U. (2020). Interviewee Performance Analyzer Using Facial Emotion Recognition and Speech Fluency Recognition. *Proceedings of Innovation in Technology (INOCON) 2020 IEEE International Conference*, 1-5.

APPENDIX

Data Set Description and Meta Data

This Research Project (Emotional Music Player) is a novel way that helps the user to play songs automatically according to the user's feelings. Identifies the user's facial expression and plays songs according to their mood. Feelings are detected using the Fisherface algorithm. The human face is an important part of the human body and in particular plays a key role in shaping one's personality and emotional state. The webcam captures the user's image, and then displays the user's face features in the captured image. The facial expression is sadly divided, happy, neutral, angry. Depending on the mood, the music will be played from pre-defined tracks.

A total of 121 images have been trained for neutral emotion. Combining all these images we form a complete dataset of images against which we can train our model and increase its efficiency by increasing more number of images in the dataset. All images sizes in the dataset are 350x350 pixels in grayscale. Instead of SVM classifier we have used fisherface classifier from the opencv library available in python, A total of 121 images have been trained for neutral emotion. The images have been cropped and grayscale for easy processing by the CPU. A total of 33 images have been trained for sad emotion. The unequal number of images for various emotions causes some emotions to become dominant and others remain suppressed due to some emotions dataset being large and some smaller. This error could be solved by taking an equal number of the dataset for all emotions and then training the model. A total of 100 images have been trained for happy emotion. A total of 164 images have been trained for angry emotion. These all images have been cropped to the face of the user and grayscale for easy processing to use less resources. Combining all these images we form a complete dataset of images against which we can train our model and increase its efficiency by increasing the number of images in the dataset.

Snapshots

1. Code for Login Window:

This code is used to generate the given windows

1. Login window so that the user can enter the credentials and start the app.

2. Signup window for users who want to create their ID.

This code also handles the wrong username and password is entered by the user by displaying the following windows:

1.If the user enters a wrong password then this window will appear

2.If the user enters a wrong username then this window will appear

2. Code for Music App Interface

this code creates the main app window which the user can use the select different modes for the player.

Figure 14. code for login window Part-1

```
from tkinter import messagebox
import csv
import tkinter as tk
def first():
    global wn
    wn = tk.Tk()
    wn.geometry("800x800")
    wn.title("Music APP")
    background_image = tk.PhotoImage(file="background1.png")
    background_label = tk.Label(wn, image=background_image)
    background_label.place(relx=0,rely=0,relwidth=1,relheight=1)
    background_image1 = tk.PhotoImage(file="background.png")
    background_label1 = tk.Label(wn, image=background_image1)
    background_label1.place(relx=0.3,rely=0.071)
    wn.resizable(0,0)
    label = tk.Label(wn, text="USER LOGIN", font=("Courier", 30, "normal"), borderwidth=0)
    label.place(relx=0.5, rely=0.02, anchor="center")

    label1 = tk.Label(wn, text="USER ID", font=("Courier", 20, "normal"))
    label1.place(relx=0.48, rely=0.3, anchor="center")
    first.sender = tk.Entry(wn, width=30, borderwidth=8, font=("Arial", 15, "normal"))
    first.sender.place(relx=0.5, rely=0.4, anchor="center")

    label1 = tk.Label(wn, text="PASSWORD", font=("Courier", 20, "normal"))
    label1.place(relx=0.48, rely=0.55, anchor="center")
    first.password = tk.Entry(wn, width=30, borderwidth=8, font=("Arial", 15, "normal"),show="*")
    first.password.place(relx=0.5, rely=0.65, anchor="center")

    label2 = tk.Button(wn, text="CREATE ID", font=("Arial", 12, "normal"), command=create_Id)
    label2.place(relx=0.48, rely=0.9, anchor="center")

    label3 = tk.Button(wn, text="LOGIN", font=("Arial", 12, "normal"), command=login)
    label3.place(relx=0.48, rely=0.75, anchor="center")

    wn.mainloop()
def create_Id():
    global wn
    wn.destroy()
    wn = tk.Tk()
    wn.geometry("800x800")
    background_image = tk.PhotoImage(file="background1.png")
    background_label = tk.Label(wn, image=background_image)
    background_label.place(relx=0,rely=0,relwidth=1,relheight=1)
    background_image1 = tk.PhotoImage(file="background.png")
    background_label1 = tk.Label(wn, image=background_image1)
    background_label1.place(relx=0.3,rely=0.071)
    label = tk.Label(wn, text="CREATING USER ID", font=("Courier", 30, "normal"), borderwidth=0)

    wn.resizable(0,0)
    label.pack()
```

This code generates the following windows for the user

If the users select emotion mode than the webcam starts and captures the image of the user and identifies emotion from the image and plays songs based on emotion mainly three windows are generated:

For Angry face,
output window
For Neutral Face,
Output window
For Happy Face,
output window
For Sad Face,
output window

Figure 15. code for login window part-2

```
    label1.place(relx=0.48, rely=0.3, anchor="center")
    create_Id.Id = tk.Entry(wn, width=30, borderwidth=8, font=("Arial", 15, "normal"))
    create_Id.Id.place(relx=0.5, rely=0.4, anchor="center")

    label = tk.Label(wn, text="PASSWORD", font=("Courier", 20, "normal"))
    label.place(relx=0.48, rely=0.6, anchor="center")
    create_Id.password = tk.Entry(wn, width=30, borderwidth=8, font=("Arial", 15, "normal"),show="*")
    create_Id.password.place(relx=0.5, rely=0.7, anchor="center")

    label3 = tk.Button(wn, text="SIGNUP", font=("Arial", 12, "normal"), command=signup)
    label3.place(relx=0.48, rely=0.85, anchor="center")
    label3 = tk.Button(wn, text="BACK TO LOGIN PAGE", font=("Arial", 10, "normal"), command=first1)
    label3.place(relx=0.48, rely=0.95, anchor="center")
    wn.mainloop()
def login():
    global user
    ml = [first.sender.get(), first.password.get()]
    c=0
    f=open("userdata.csv","r")
    r=csv.reader(f)
    for i in r:
        if(len(i)>0):
            if(i[0]==ml[0]):
                c=1
                if(i[1]==ml[1]):
                    wn.destroy()
                    import capture
                    capture.main()
                else:
                    messagebox.showinfo("message", "LOGIN FAILED: Try again")
                    wn.destroy()
                    first()
    if(c==0):
        messagebox.showinfo("message", "ID doesn't exists : Create One")
        wn.destroy()
        first()
        f.close()
def first1():
    wn.destroy()
    first()


def signup():
    f=open("userdata.csv","a+")
    w=csv.writer(f)
    ml = [create_Id.Id.get(), create_Id.password.get()]
    w.writerow(ml)
    f.close()
    wn.destroy()
    first()
```

3.Code for updating the trained model or Training the model

This code is used to train the model which identifies emotion. It generates the following windows

It first asks the user to maintain an angry face for 5 sec then a happy face for 5 sec then a sad face and then a neutral face in the same way than it trains the model based on the images received.

Figure 16. Login window so that the user can enter the credentials and start the app

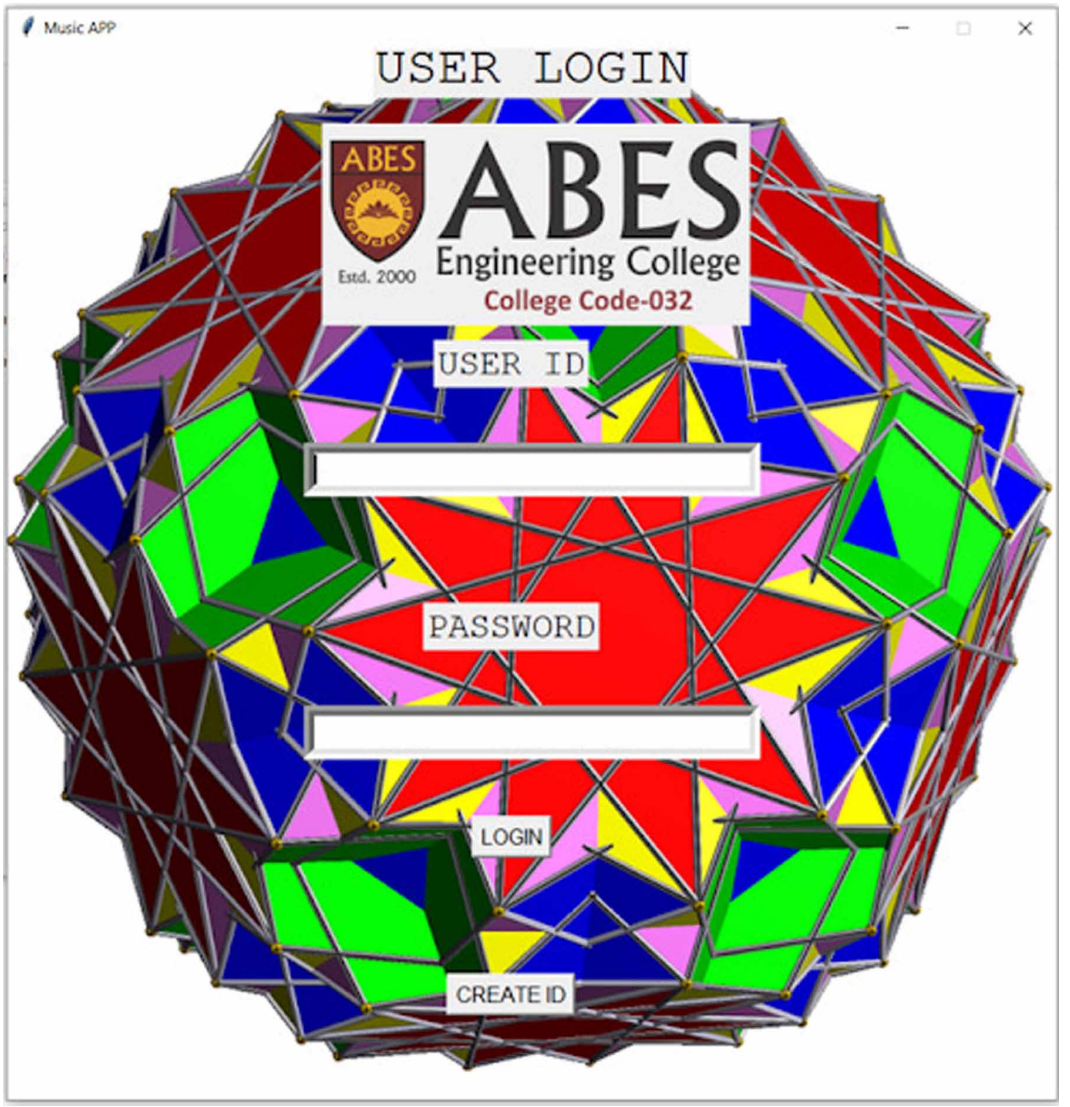


Figure 17. Signup window for users who want to create their ID

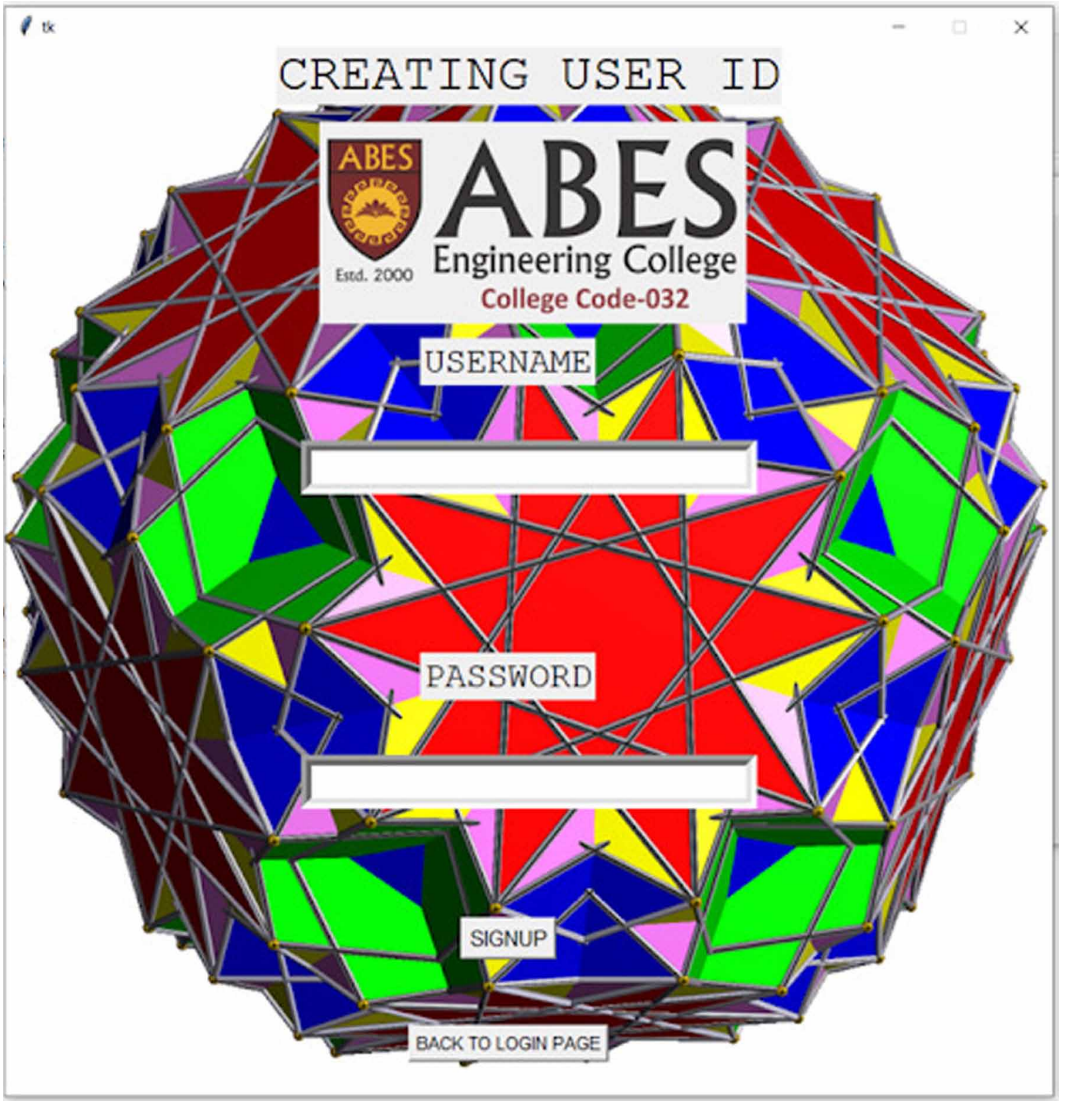


Figure 18. If the user enters a wrong password then this window will appear

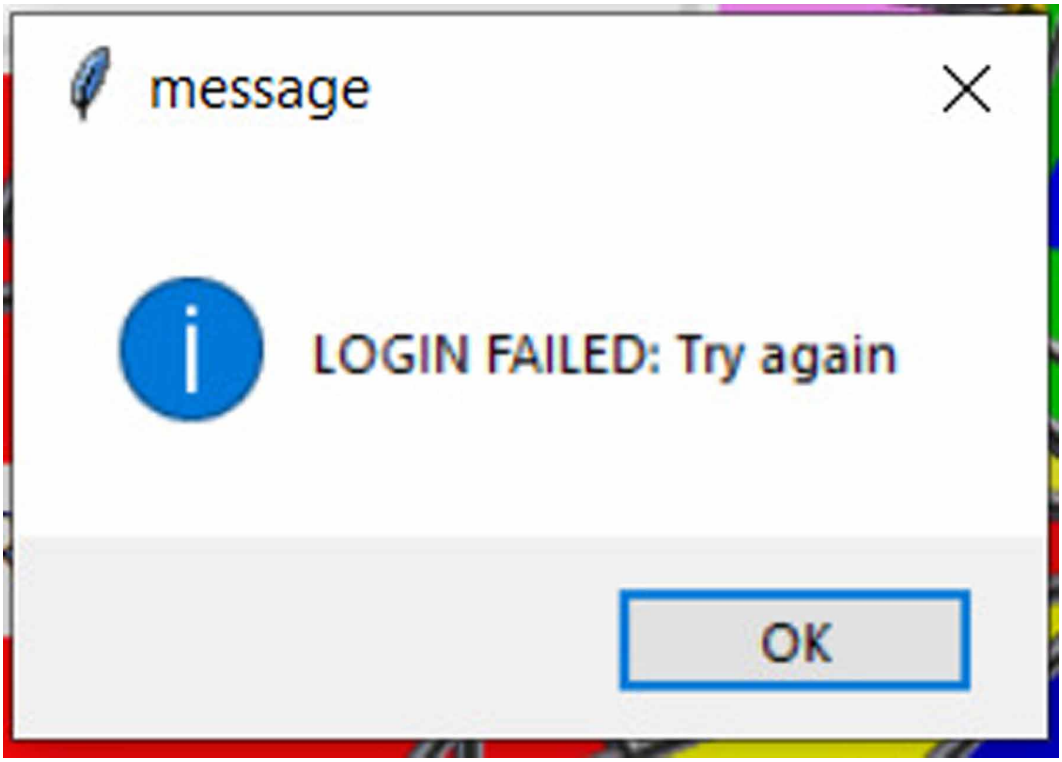


Figure 19. If the user enters a wrong username then this window will appear

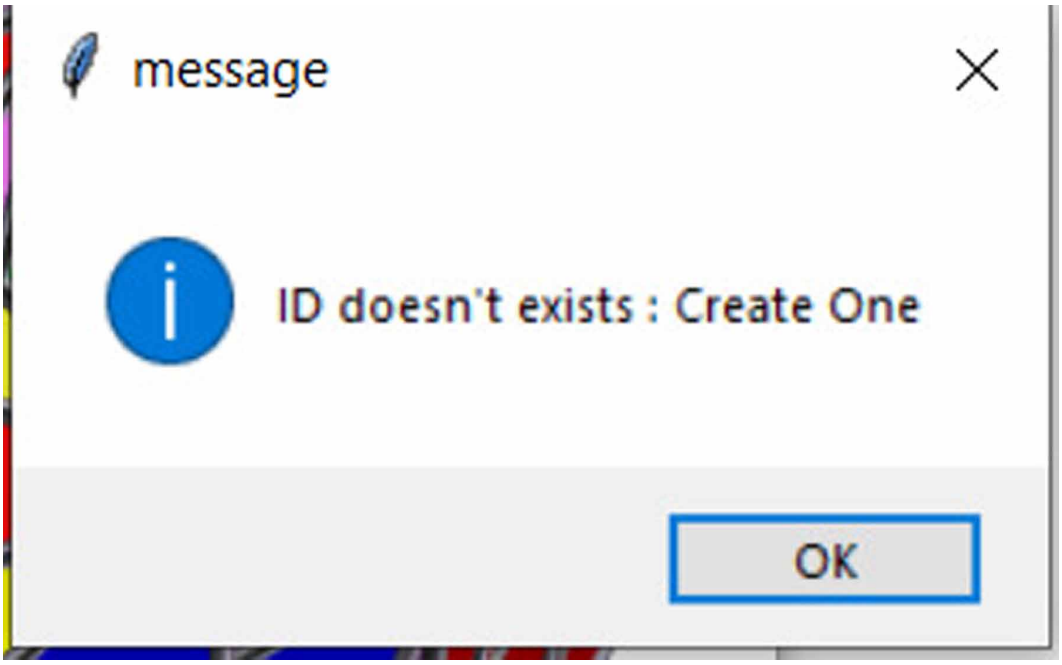


Figure 20. Code for music app interface-Part1

```
'''code Authors:
1.Prabhat Yadav(ABESEC) &
2.Jayash Raj Singh Yadav(ABESEC)
project:Music Player Based on Emotions
Created For partial fulfillment of Mini Project
Version:Beta'''
'''Start Of Code'''
def main():
        import cv2 #(computer vision 2 library for image processing reading and writing the image and a lot more image processing)
        import argparse #( For command line interface in python recommended commandline parsing modu in the Python standard librar It is what you
        import time #(Thetime module provides many ways to represent time time in code, such as objects, numbers, and strings. its other functiona
        import os #(It helps to interact with the operating system we are using)
        import Update_Model #(contains different user build functions definations for working with images data set )
        import glob #(use to retrieve file from pathnames matching a certain pattern.)
        import random #(Random library from python having various functions most of which are dependent on random() function of the same library.)
        import eel #(eel library  makes it easier to work with HTML and python helps to interact with python using chrome,helps to create beautifu
        import light #(user defined library used for capturing webcam image)
        eel.init('WEB')#(This will start a webserver on the default settings (http://localhost:8000).)
        emotions=["angry", "happy", "sad", "neutral"]
        fishface = cv2.face.FisherFaceRecognizer_create()
        font = cv2.FONT_HERSHEY_SIMPLEX #normal size sans-serif font
        parser1=argparse.ArgumentParser(description="Options for  music player based on emotions(Updating model)")
        parser1.add_argument("--update", help="Call for taking new images, retraining the model.", action="store_true") #store_true option automat
        args=parser1.parse_args()
        facedict={}
        video_capture=cv2.VideoCapture(0)
        facecascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")#it is a function of opencv which is pointing to the location whe
        def crop(clahe_image, face):
                for (x, y, w, h) in face:
                        faceslice=clahe_image[y:y+h, x:x+w]
                        faceslice=cv2.resize(faceslice, (350, 350))
                        facedict["face%s" %(len(facedict)+1)]=faceslice
                return faceslice
        def grab_face():
                ret, frame=light.nolight()
                cv2.imwrite('test.jpg', frame)
                cv2.imwrite("images/main%s.jpg" %count, frame)
                gray=cv2.imread('test.jpg',0)
                clahe=cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8,8))
                clahe_image=clahe.apply(gray)
                return clahe_image

        def detect_face():
                clahe_image=grab_face()
                face=facecascade.detectMultiScale(clahe_image, scaleFactor=1.1, minNeighbors=15, minSize=(10, 10), flags=cv2.CASCADE_SCALE_IMAGE)
                if len(face)>=1:
                        faceslice=crop(clahe_image, face)
                else:
                        print("No/Multiple faces detected!!, passing over the frame")
```

Figure 21. Code for music app interface-Part2

```
def save_face(emotion):
        print("\n\nLook "+emotion+" until the timer expires same emotion for some time.")
        print('\a')


        for i in range(0, 5):
                print(5-i)
                time.sleep(1)

        while len(facedict.keys())<16:
                detect_face()

        for i in facedict.keys():
                path, dirs, files = next(os.walk("dataset/%s" %emotion))
                file_count = len(files)+1
                cv2.imwrite("dataset/%s/%s.jpg" %(emotion, (file_count)), facedict[i])
        facedict.clear()

def update_model(emotions):
        print("Update mode for model is ready")
        checkForFolders(emotions)

        for i in range(0, len(emotions)):
                save_face(emotions[i])
        print("Collected the images, looking nice! Now updating the model...")
        Update_Model.update(emotions)
        print("Model train successful!!")

def checkForFolders(emotions):
        for emotion in emotions:
                if os.path.exists("dataset/%s" %emotion):
                        pass
                else:
                        os.makedirs("dataset/%s" %emotion)

def identify_emotions():
        prediction=[]
        confidence=[]

        for i in facedict.keys():
                pred, conf=fishface.predict(facedict[i])
                cv2.imwrite("images/%s.jpg" %i, facedict[i])
                prediction.append(pred)
                confidence.append(conf)
        output=emotions[max(set(prediction), key=prediction.count)]
        print("You seem to be %s" %output)
        facedict.clear()
        return output;
count=0
@eel.expose
```

Figure 22. Code for music app interface-Part3

```
@eel.expose
def getEmotion():

        count=0
        while True:
                count=count+1
                detect_face()
                if args.update:
                        update_model(emotions)
                        break
                elif count==10:
                        fishface.read("model2.xml")
                        return identify_emotions()
                        break

eel.start('main.html')#open a browser to http://localhost:8000/main.html.
'''End Of Code'''
```

Figure 23. the main app window which the user can use the select different modes for the player

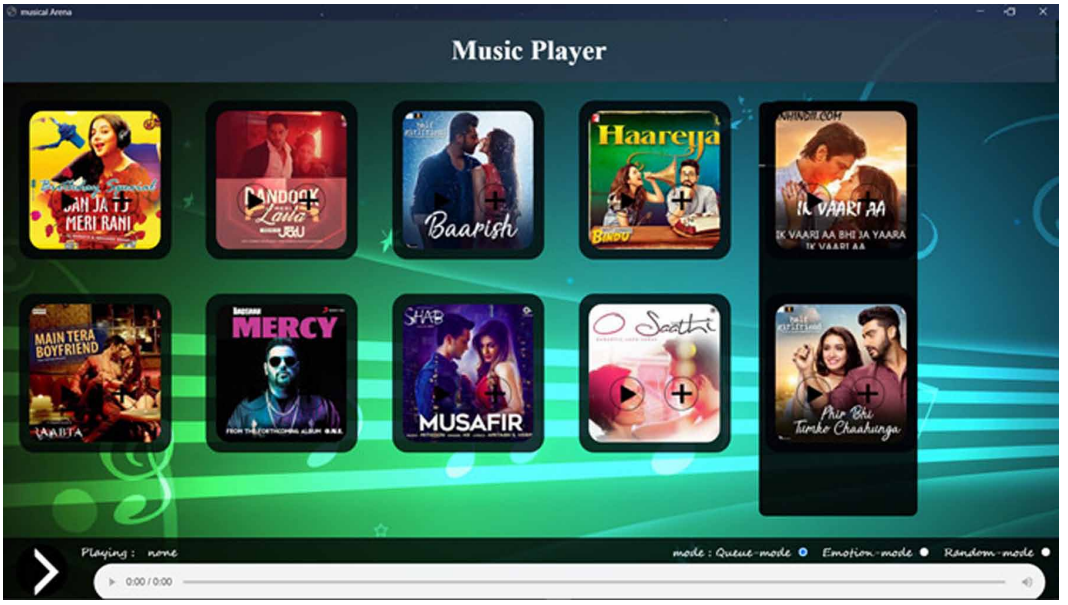


Figure 24. Output window for angry face

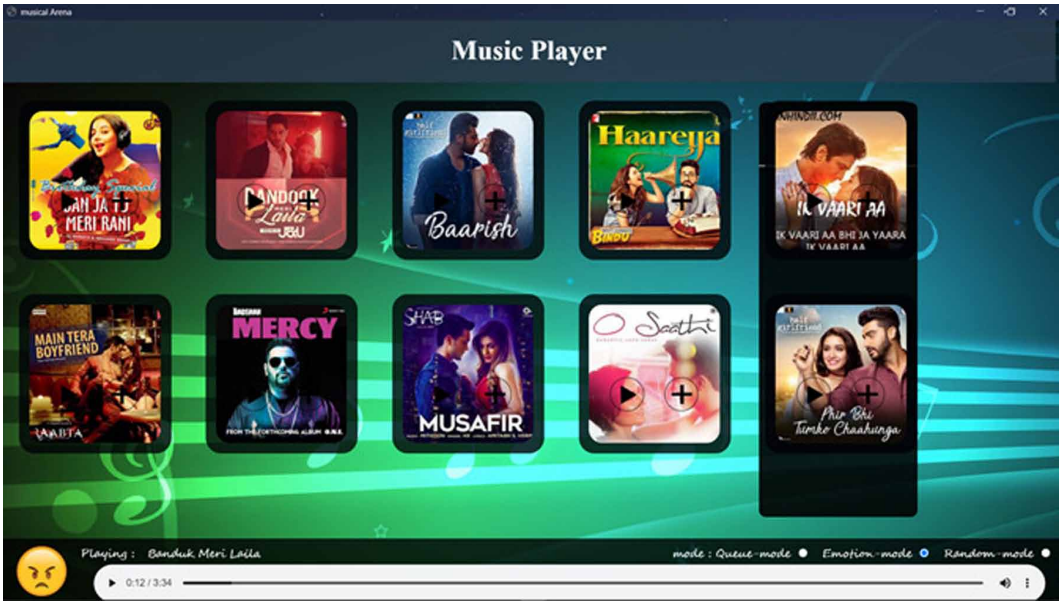


Figure 25. Output window for neutral face

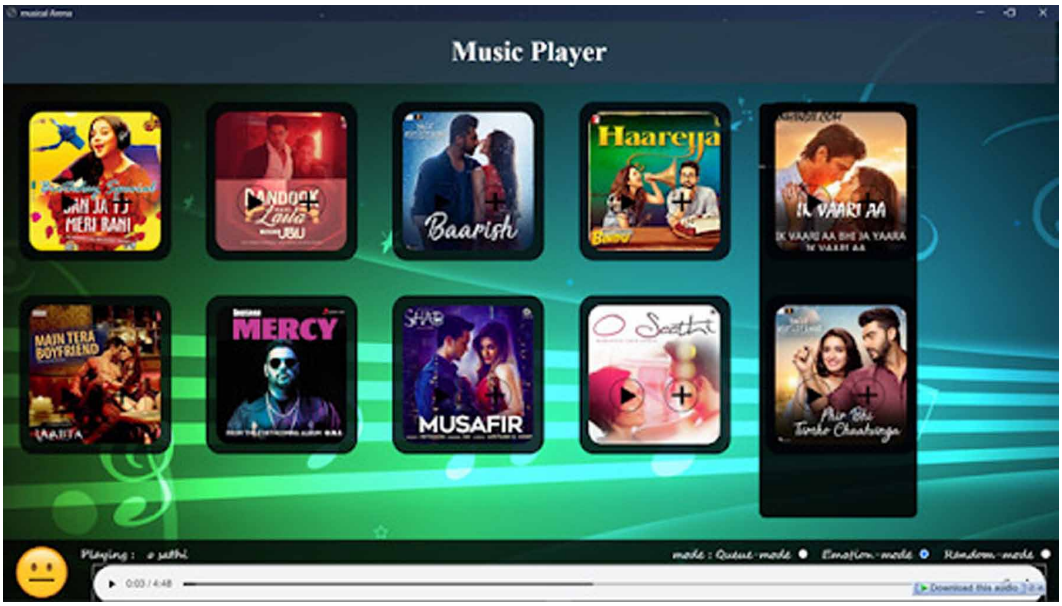


Figure 26. Output window for happy face

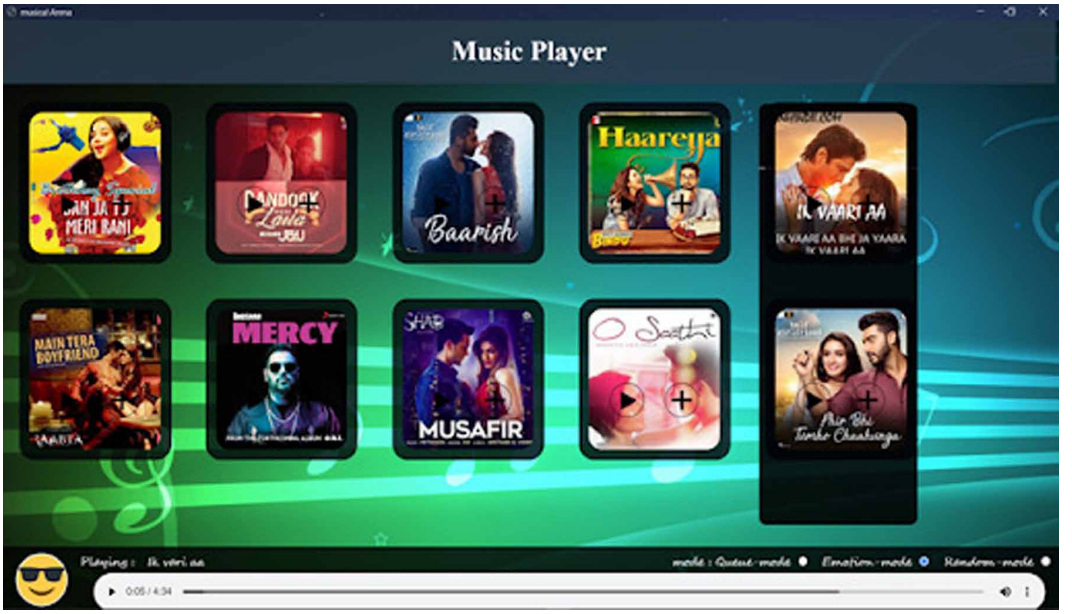


Figure 27. Output window for sad face

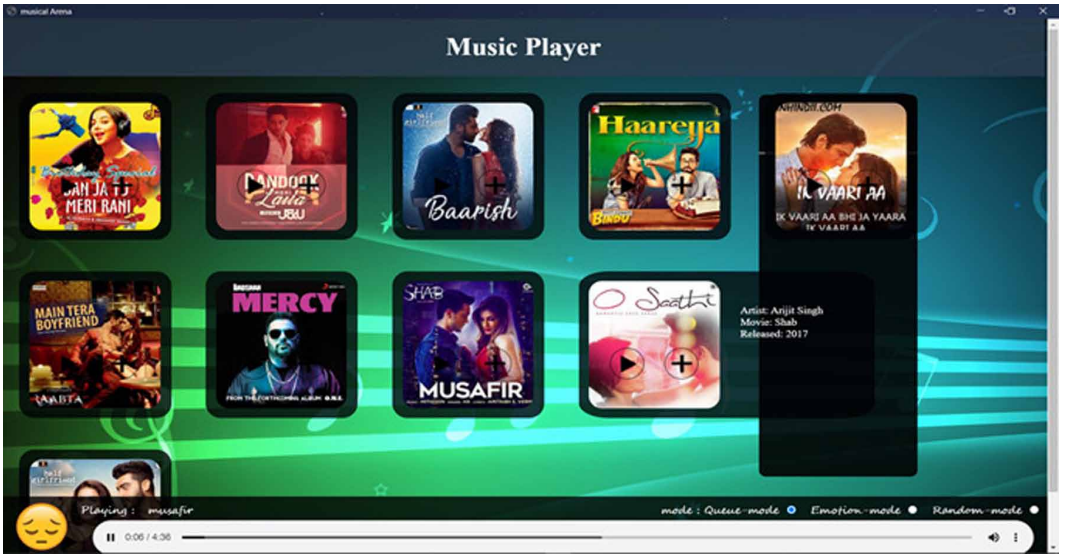


Figure 28. Code for updating the trained model or Training the model

```
import numpy as np#NumPy is a general-purpose array-processing package. It provides a hi
import glob#the glob module is used to retrieve files/pathnames matching a specified pat
import random#Almost all functions of the random module depend on the basic function ran
import cv2 # computer vision 2, processing and working with images

fishface=cv2.face.FisherFaceRecognizer_create()# function of open cv library for recogni
data={}

def update(emotions):
    run_recognizer(emotions)
    print("Saving model...")
    fishface.save("model2.xml")
    print("Model saved!!")

def make_sets(emotions):
    training_data=[]
    training_label=[]

    for emotion in emotions:
        training=sorted(glob.glob("dataset/%s/*" %emotion))
        for item in training:
            gray=cv2.imread(item,0)
            #gray=cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
            training_data.append(gray)
            training_label.append(emotions.index(emotion))
    return training_data, training_label

def run_recognizer(emotions):
    training_data, training_label=make_sets(emotions)
    print("Training model...")
    print("The size of the dataset is "+str(len(training_data))+" images")
    fishface.train(training_data, np.asarray(training_label))
```

Figure 29. Code captures the image of a user in a different mood and trains that image to increase the efficiency of identifying the emotion

```
Look angry until the timer expires same emotion for some time.

5
4
3
2
1


Look happy until the timer expires same emotion for some time.

5
4
3
2
1


Look sad until the timer expires same emotion for some time.

5
4
3
2
1


Look neutral until the timer expires same emotion for some time.

5
4
3
2
1
Collected the images, looking nice! Now updating the model...
Training model...
The size of the dataset is 444 images
Saving model...
Model saved!!
Model train successful!!
```

Rohit Rastogi received his B.E. degree in Computer Science and Engineering from C.C.S. Univ. Meerut in 2003, the M.E. degree in Computer Science from NITTTR-Chandigarh (National Institute of Technical Teachers Training and Research-affiliated to MHRD, Govt. of India), Punjab Univ. Chandigarh in 2010. Currently he is pursuing his Ph.D. In computer science from Dayalbagh Educational Institute, Agra under renowned professor of Electrical Engineering Dr. D.K. Chaturvedi in area of spiritual consciousness. Dr. Santosh Satya of IIT-Delhi and dr. Navneet Arora of IIT-Roorkee have happily consented him to co supervise. He is also working presently with Dr. Piyush Trivedi of DSVV Hardwar, India in center of Scientific spirituality. He is an Associate Professor of CSE Dept. in ABES Engineering. College, Ghaziabad (U.P.-India), affiliated to Dr. A.P. J. Abdul Kalam Technical Univ. Lucknow (earlier Uttar Pradesh Tech. University). Also, He is preparing some interesting algorithms on Swarm Intelligence approaches like PSO, ACO, and BCO. Rohit Rastogi is involved actively with Vichaar Krnati Abhiyaan and strongly believe that transformation starts within self.

Prabhat Yadav is BTech Second Year Student at ABESEC Ghaziabad.

Jayash Raj Singh Yadav is BTech Second Year Student at ABESEC Ghaziabad.