JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Emotion Based Music Recommendation System Using Face Detection

Meghna Chhanwal[1],Laksha Wadhwani[2], Dr. Makarand Velankar[3].
[1,2] Students Of MKSSS Cummins College of Engineering for Women Pune
[3] Faculty, MKSSS's Cummins College of Engineering for women, Pune.

Abstract: Understanding human emotions through facial expressions is a significant aspect of human-computer interaction and artificial intelligence. This research delves into the development of a sophisticated emotion detection system employing a tandem approach utilizing Haar Cascade and Convolutional Neural Network (CNN) algorithms. The study initially explores the intricacies of facial emotion recognition, acknowledging the complexity posed by human facial features. Leveraging the Facial Expression Recognition (FER) dataset comprising over 26,000 images, this research trains and tests a model on a diverse range of emotions such as anger, happiness, neutrality, sadness, and surprise. Haar Cascade, renowned for its object detection capabilities, serves as a foundational tool for facial recognition. Its utilization involves meticulous image analysis, harnessing features, and integral image calculations to discern facial attributes. Additionally, the integration of CNN, recognized for its prowess in pattern recognition, enriches the system's capacity to interpret emotions from facial cues. This CNN architecture comprises convolutional, pooling, and fully connected layers, optimizing the extraction and understanding of intricate emotional expressions. Moreover, this paper examines the limitations and computational challenges inherent in both algorithms. Haar Cascade's computational intensity in processing large datasets and CNN's need for substantial computational resources underscore the nuanced trade-offs in algorithm selection. The research findings showcase the system's performance in recognizing emotions from facial expressions. Despite initial successes in training the CNN model with a considerable dataset, observations revealed a stabilization in learning after a specific number of iterations, indicating potential saturation in model improvement. this paper elucidates the efficacy and limitations of integrating Haar Cascade and CNN algorithms for facial emotion recognition. The findings contribute to the ongoing discourse on emotion detection systems, underscoring the need for nuanced approaches to harness the potential of facial expression analysis in human-computer interaction and emotional

Keyword: Convolutional Neural Network, Pattern Recognition, FER Dataset, Haar Cascade, Music Recommendation

1.Introduction

We learned about an interesting system that uses two algorithms called Haar Cascade and CNN. Haar Cascade is like a sharp-eyed detective who's good at spotting things, especially faces. Meanwhile, CNN is like a super brainy expert who recognizes faces accurately. These tools work together to figure out different emotions like happiness, sadness, anger, surprise, and just a neutral expression by looking at how your face changes. Once the system knows what you're feeling, it plays music from a YouTube playlist that matches your mood. I think this is cool because the playlist is saved on a website using Python tools, so the website can play songs that suit your feelings. It's like having a personal DJ for your emotions.

2.LITERATURE SURVRY

Facial emotion recognition challenges machines due to intricate human facial features; diverse systems using CNNs, RNNs, or hybrids attempt live video emotion detection, with varying accuracies were explored [1]**.** Haar feature calculation aims to identify image features by analyzing differences in pixel values within darker and lighter areas. These features scan for edges, intensity changes, and diagonal shifts throughout the image. However, this process involves extensive mathematical computations, with a single rectangle requiring 18-pixel value additions. Scaling this across various feature sizes and the entire image poses significant computational challenges, even for high-powered systems [2]. Music has been with humans since ancient times, helping in both happy and sad moments. It not only affects our feelings but also has good effects on our bodies, which is really important now with many people having mental health problems worldwide. Even though there are lots of music apps, there aren't many that focus on helping people feel better by understanding their emotions. Most apps suggest music based only on what's in the songs or what others like, without thinking about how the person feels right then [3,4,5]. In music prediction, unique features from the music signal guide classification for better recommendations. Unlike text-based suggestions, music relies on audio traits like pitch affected by emotions and surroundings. Understanding these nuances is vital for suggesting songs that suit individual tastes, ensuring more accurate and enjoyable recommendations [6,7]. CNN (Convolutional Neural Network) is a smart tool that's really good at spotting patterns. It has different layers, each doing its own special job, like finding important details in things. Convolution is like mixing two things together to make something new that shows changes in their

shape. It's a bit like blending colors to get a different shade. Once this smart tool figures out how someone is feeling, there's a special part that finds music that matches that feeling. Happy? It might suggest fun songs. Feeling a bit down? It could recommend calming music. Then, the person can pick and listen to their favorite songs from those suggestions [8,14]. Haar feature-based cascade classifiers are tools used to spot objects in images, especially for recognizing faces and facial expressions. They work by learning from pictures, distinguishing what's in them, and then applying that knowledge to find similar things in other images. To train, it uses both positive (showing what it should detect) and negative (showing what it shouldn't detect) images. By looking at these pictures, it learns what features to look for. These features are values calculated by comparing the sum of pixels in a white area to the sum in a black area of the image. This method helps detect faces of different people in various environments [9]

3.METHODOLOGY

3.1 Dataset Description

The Facial Expression Recognition dataset is a collection of images obtained from Kaggle, a popular data science platform. This dataset contains a total of 26087 images. Out of these, 20,819 images are designated for training purposes, which means they are used to teach a machine learning model to recognize different facial expressions. The remaining 5,268 images are utilized as a testing dataset to evaluate how well the model performs on new, unseen data. Each image in this dataset belongs to one of five classes or categories, representing different facial expressions: 'Angry,' 'Happy,' 'Neutral,' 'Sad,' and 'Surprise.'[16].

3.2.Face Detection

Face detection is like a super-smart tool in computers that looks at pictures or videos to find where people's faces are. It uses special tricks to figure out where the eyes, nose, and mouth are, so it can tell if there's a face in the picture and where it is.

3.3 Algorithm

we are using two major algorithms

- Haar Cascade Algorithm for Object Detection
- Convolutional Neural Network Algorithm for Emotion Detection

3.3.1 Face Detection Using Haar Cascade Algorithm

Positive images as shown in Fig.1 contain the object of interest, serving as examples to teach the classifier what the object looks like, while negative images as shown in Fig.2 lack the object and help the classifier learn what the object is not, aiding in differentiation from the background. When using Haar cascades in OpenCV to find faces, sometimes the detections aren't perfect. You might see some wrong detections or miss some faces completely. To fix this, we can adjust two important settings: scale Factor and min Neighbors. The scale Factor decides how much the image size changes during detection, while min Neighbors looks at neighboring areas before deciding if it's a face [10]. Imagine you're a detective searching for a specific suspect in a crowded city. You wouldn't question every person you see, right? Instead, you'd likely start by scanning for broad clues like height, build, and hair color. If someone matches those initial filters, you'd move on to a more detailed investigation, checking for specific features like scars or tattoos [15]. That's essentially what the Viola-Jones classifier does with images. It uses a series of increasingly complex filters, like a detective's checkpoints, to quickly rule out unlikely areas and focus on potential matches, making object detection faster and more accurate [8].

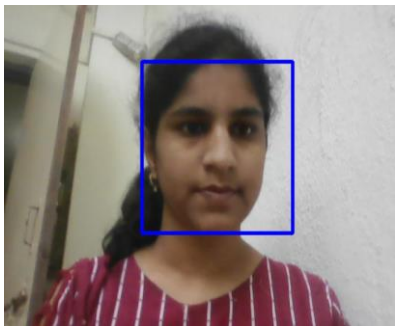

Fig 1: Positive Image

Fig 2: Negative Image

Haar-like features, the building blocks of Haar cascade object detection, are defined as rectangular regions within a detection window, constructed from adjacent rectangles with differing pixel sums. They capture edges and intensity variations by subtracting the pixel sums within white rectangles from those within black rectangles. To accelerate feature calculation, integral images are employed. These precomputed representations store the cumulative sum of pixels above and to the left of each pixel, enabling efficient Haar-like feature calculation in constant time using just four lookups within the integral image.

Integral Image Calculation:

Create a new image with the same dimensions as the original image. For each pixel (x, y), calculate its integral image value using the formula: I(x, y) = I(x-1, y) + I(x, y-1) - I(x-1, y-1) + image(x, y).

Feature value calculation:

Sum of pixels in A = (Top-right of A) + (Bottom-left of A) - (Top-left of A) - (Bottom-right of A)

Sum of pixels in B = (Top-right of B) + (Bottom-left of B) - (Top-left of B) - (Bottom-right of B)..

Feature value = Sum of pixels in A - Sum of pixels in B.

3.3.2 Emotion Detection Using Convolutional Neural Networks

In the second part, the CNN model is trained on a dataset (FER Kaggle dataset) for recognizing five emotions: Angry, Happy, Neutral, Sad and Surprised [12]. The model is then used to real-time predictions in the webcam video stream. Neural Networks are used to handle a variety of input such that it can classify those in an accurate way. Convolutional Neural Network has mainly three layers:

1. Convolutional Layer.
2. Pooling Layer.
3. Fully Connected Layer.

Fig 3: Image Calculation

We initially see a human face detected, but for the model to work, the computer sees the image as a 2-dimensional matrix, one for each pixel in that image as shown in Fig.3 and 4. CNN allows one to directly learn the visual features and detect their presence in the image simultaneously.

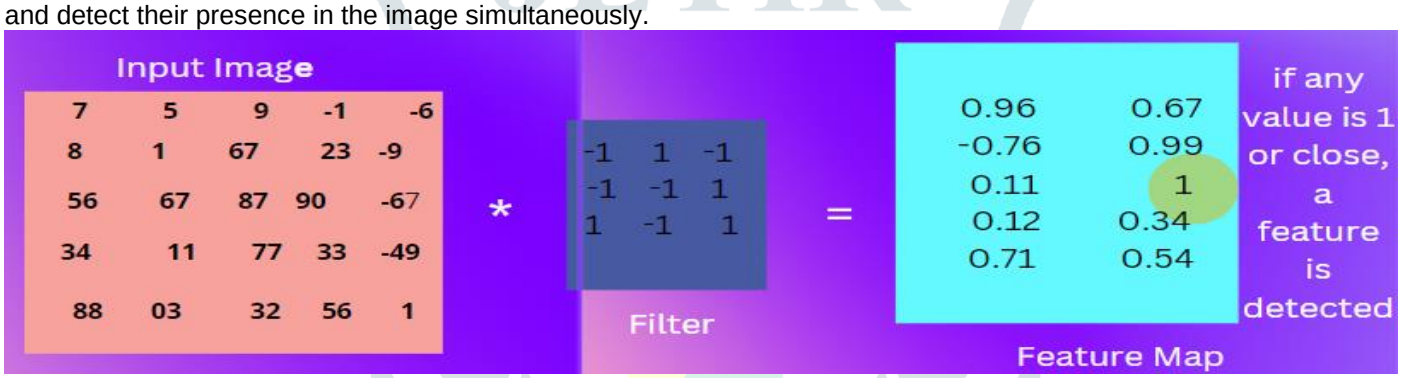

Fig 4: Feature Map

Calculation example: The filter is sliced over the whole 2d array of the in-*put an image, like a patch. The values of the filter are multiplied by the input image, and divided by the total elements in the patch/filter, in our case its 9. The final value is stored in a feature map. The filters can be in 2D or 3D, whenever the value 1 is found in the map, a feature is detected such as, an eye, nose, eyebrows or teeth.

3.3.3 Pooling Layer

Emotion detection using CNNs involves using different layers in a team-like manner to help computers understand emotions in facial images. Picture these layers as specialized groups: the input layer takes in images showing emotions like happiness or sadness, passing them on to convolutional layers that focus on different face parts, detecting patterns linked to emotions. Activation functions add complexity, helping the computer grasp deeper connections in facial details. Pooling layers then simplify things by summarizing crucial parts, making it easier for the computer to manage. Fully connected layers gather this information to better recognize emotions, while the output layer reveals the computer's best guess at the emotions shown in the pictures. Together, these layers guide the computer in learning to spot emotions in faces, which can be valuable in making technology more relatable or in understanding human emotions in various scenarios [13.14].

In the realm of Convolutional Neural Networks (CNNs), the dataset serves as the initial input, comprising pairs of X and Y values, where X represents various data types like images, text, or numbers, while Y denotes corresponding labels or outputs. The CNN's journey as shown in fig. 5, commences with the input layer, which ingests the X values from the dataset. Subsequently, convolutional layers come into play, serving as the backbone of the network by extracting diverse features through the use of multiple filters that scrutinize the data. Following this, pooling layers step in to condense information from previous layers, optimizing efficiency and averting overfitting. The activation function introduces non-linearity, facilitating the network in learning intricate relationships between data and outputs. Culminating the process, the output layer generates final predictions based on extracted features. The critical evaluation comes from the loss function, measuring disparities between network predictions and actual labels, while updates, the adjustment of weights and biases, aim to minimize this disparity over time, enhancing the network's predictive accuracy [11].

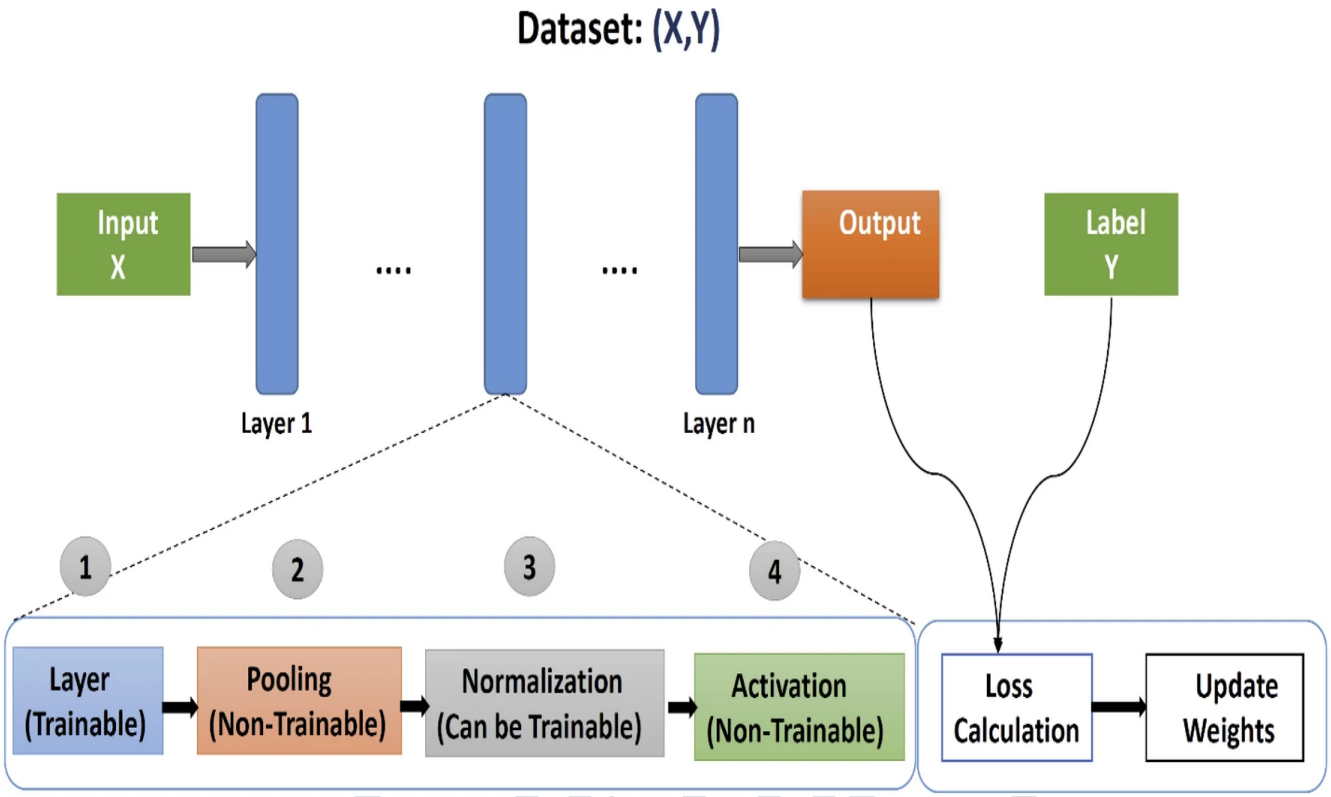

Fig 5: CNN model used

3.4 YouTube Playlist Generation

The process of creating a YouTube playlist and integrating it into a Flask web page involves several crucial steps. Initially, utilizing a specialized YouTube tool enables the creation of a personalized playlist, essentially a curated collection of desired songs. To extract these songs from YouTube, specific programs, such as YouTube-DLC, assist in downloading the content. However, strict adherence to copyright regulations is paramount to avoid using copyrighted material without proper permission, ensuring legal compliance throughout the process.Moving to the development of the web page, Flask serves as the framework to build a dedicated site where users can access and enjoy the playlist while listening to downloaded songs. This involves employing HTML, CSS, and potentially JavaScript to craft an engaging and functional interface. HTML structures the content, defining elements that showcase the playlist and facilitate user interaction. CSS stylizes these elements, enhancing visual aesthetics by adding colors, layouts, and design elements. Additionally, JavaScript may contribute to the webpage's interactivity, allowing for features like play controls, dynamic updates, or user-driven actions.The system architecture, as illustrated in Figure 6, likely outlines the sequential flow of operations, starting from playlist creation, song downloading, Flask-based web development, and finally, user interaction with the webpage to access and listen to the playlist.YouTube-DLC, in particular, stands out as an effective tool for downloading YouTube videos efficiently. Its command-line interface simplifies the downloading process, offering a direct method to acquire videos onto a device without the need for complex graphical interfaces. Despite its simplicity, the tool is highly versatile, accommodating user preferences for video quality and formats. Moreover, its consistent updates from developers ensure alignment with YouTube's changes, reinforcing its reliability and commitment to providing an enhanced user experience over time. Integrating such tools within the Flask-based web application contributes to a seamless user experience in accessing and enjoying the curated playlist content.

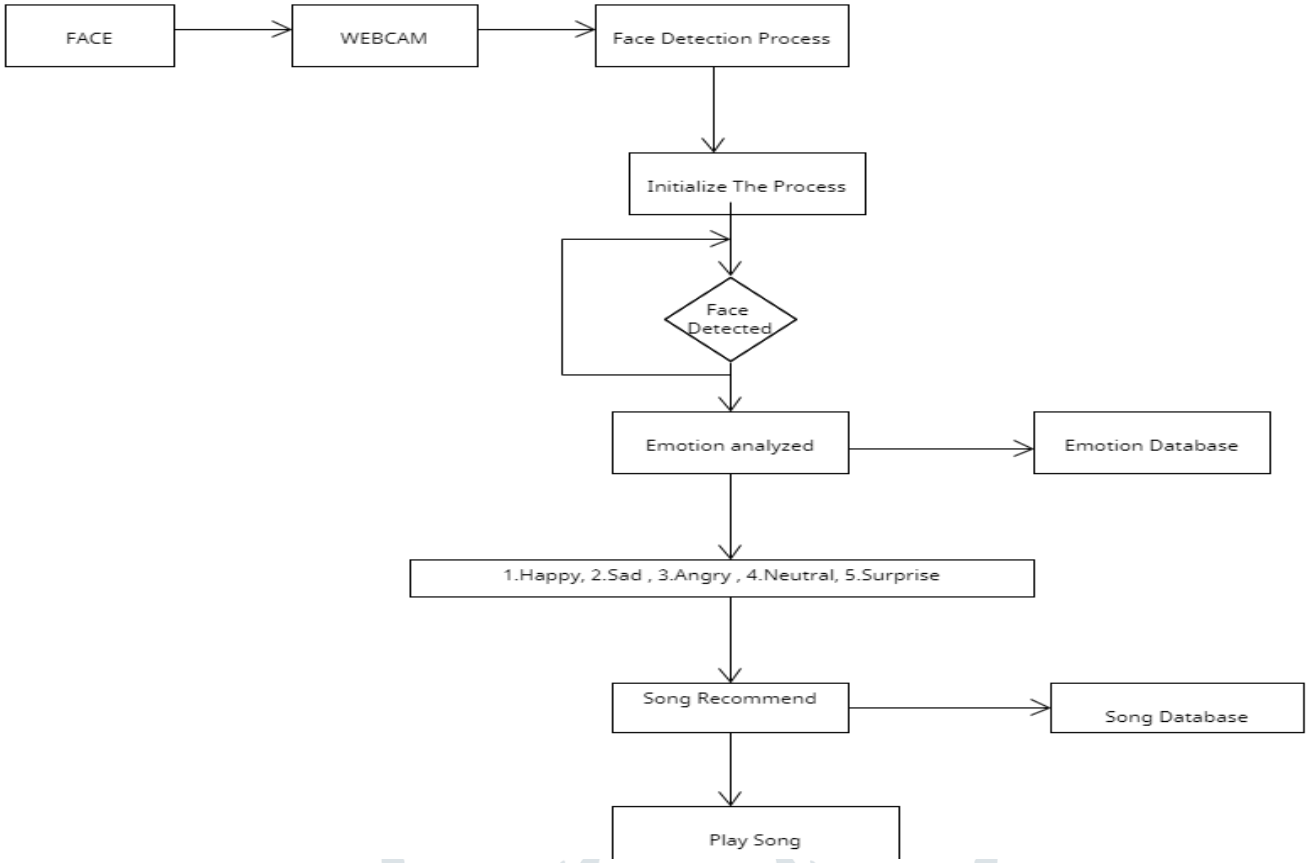

Fig 6: SYSTEM DIAGRAM Of EMOTION BASED MUSIC RECOMMENDATION SYSTEM

1. **Face Webcam**: This is the input source for the system. It captures an image or video frame from the webcam.
2. **Face Detection**: The system searches for a face in the captured image or frame. If a face is found, the process continues. If not, it stops.
3. **Emotion Analyzed**: Once a face is detected, the system analyzes its expression to determine the person's emotion. The diagram shows five possible emotions: Happy, Sad, Angry, Neutral, and Surprise.
4. **Emotion Database**: The system then references an emotion database to associate the detected emotion with a specific feeling or mood.
5. **Song Recommend**: Based on the emotion in the database, the system recommends a song from a song database.
6. **Song Database**: This database stores the songs that the system can recommend.
7. **Play Song**: Finally, the recommended song is played.

4.Result and Accuracy

We used a smart computer system called a Convolutional Neural Network (CNN) to understand emotions from facial expressions. To teach this system, we gave it a bunch of pictures—around 28,821 for learning and 5,268 for testing [12]. Before teaching, we did some things to the pictures to help the computer learn better, like changing their sizes and colors and flipping them around. We made sure all the pictures were the same size and in black and white. Then we labeled each picture with the emotion it showed. The computer was set up in a special way with layers that helped it learn from these pictures. We taught the computer using these pictures for 60 rounds, and it learned a lot in the beginning, but after some time, it didn't improve much more. We stopped because it seemed like the computer had learned as much as it could from these pictures. The graph we made showed that after a certain point, the computer didn't get better at guessing emotions, even though it kept trying a little bit. That's when we knew it had learned as much as it could from those pictures. The color mode is set to “grayscale”, meaning that it contains only one channel, and the presence of brightness is defined by the pixel value. Class mode is set to “categorical” meaning each class out of five, is represented as a binary vector, a way of representing categorical labels. Suitable for multi-class classification tasks. Same specifications are set for testing data. Creating a sequential model, with which we can add layers to the model. (convolutional, pooling and fully connected layers). After all the specifications, the model is for compilation, the parameter “loss” is set to “categorical_crossentropy”, which defines a function for loss used by models while training. Measures the difference between actual class labels and the predicted ones. Optimization: the algorithm called “Adam”, where the learning rate is set to control the step size of the weight updates. Smaller the rate,more stable the model, the metric accuracy specifies the percentage of correctly classified images. This step is necessary before the model fitting

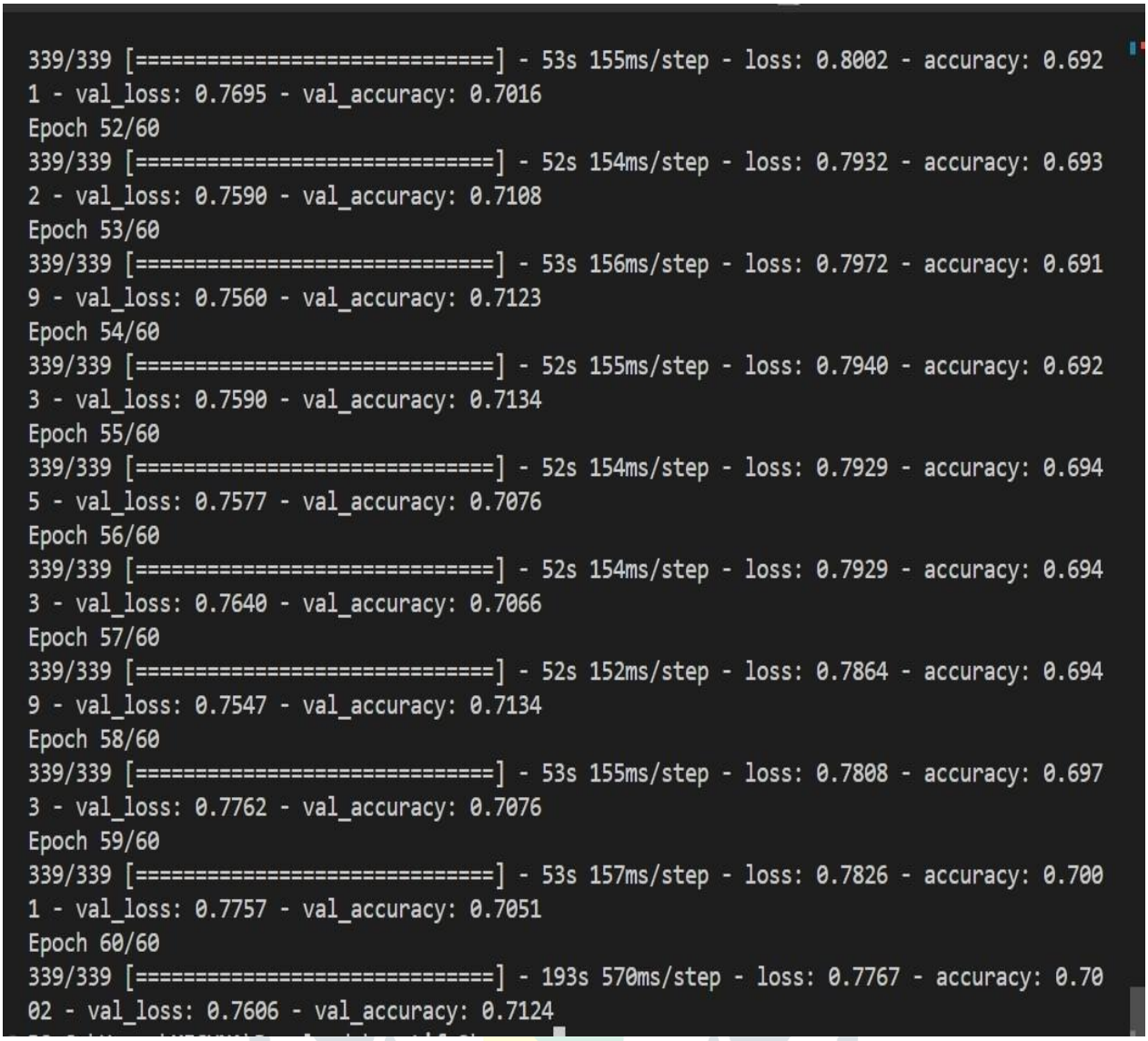

Fig 7: Accuracy and loss

Loss: This measures the difference between the model's predictions and the actual values. Lower loss is better. The training loss is 0.7932 and the validation loss is 0.7560. This suggests that the model is performing slightly better on the validation data than on the training data, which is a good sign. Accuracy measures the percentage of predictions that are correct. Higher accuracy is better. The training accuracy is 0.6 and the validation accuracy is 0.7124 as shown in fig.7. This means that the model is correctly classifying about 60% of the training data and 70% of the validation data.Adjusting batch sizes impacts the training process, influencing both speed and randomness in model training iterations. Set to grayscale, the color mode restricts images to a single channel, simplifying computational complexity. Class mode categorizes emotions into binary vectors, optimizing multi-class classification tasks for the CNN. The model architecture, comprising convolutional, pooling, and fully connected layers, is sequentially structured, providing a framework to add and configure layers for optimal feature extraction and learning.Upon model compilation, categorical cross-entropy is designated as the loss function, measuring discrepancies between predicted and actual labels during training. Employing the Adam optimizer, the learning rate governs weight updates, impacting the stability and convergence speed of the model. The chosen metric, accuracy, evaluates the percentage of correctly classified images, pivotal for assessing the model's performance.Training involves 60 epochs to iteratively update model weights based on computed gradients. However, observations during training reveal a stagnant accuracy and escalating loss after the 57th epoch, indicating model stabilization. Despite an increase in loss and negligible accuracy improvements, continuing training might lead to overfitting, where the model performs well on training data but fails to generalize to unseen data. Consequently, training was halted to prevent unnecessary computational resource consumption, as the model reached a saturation point in learning from the provided dataset.

5.Implementation Result

As every person has unique facial features, it is difficult to detect accurate human emotion or mood. But with proper facial expressions, it can be detected up to a certain extent. The camera of the device should have a

higher resolution. The android application that we have developed runs successfully and following are some of the screenshots captured while using it. Fig.8. displays “sad” mood being detected, Fig.9. displays an “angry” mood being detected, Fig.10. displays “happy” mood being detected, Fig.11. displays “surprise” mood being detected and Fig.12. displays “neutral” mood being detected

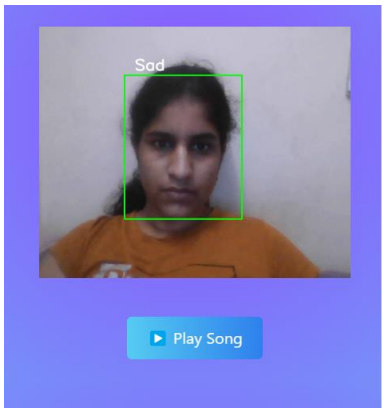

Fig 8: “Sad” mood detected

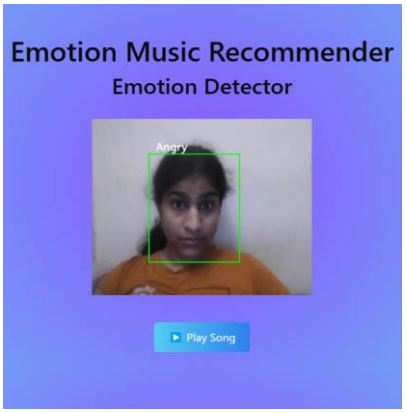

Fig 9:“Angry” mood detected

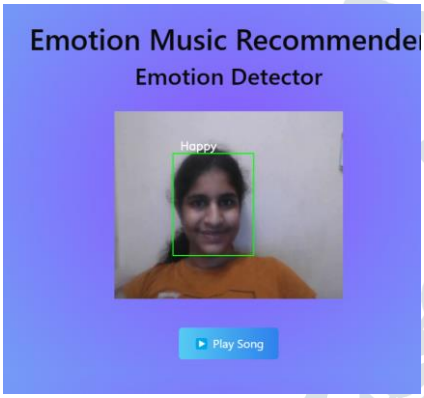

Fig 10: “Happy” mood detected

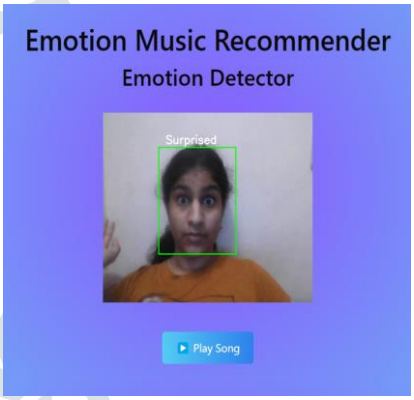

Fig11: “Surprised” mood detected

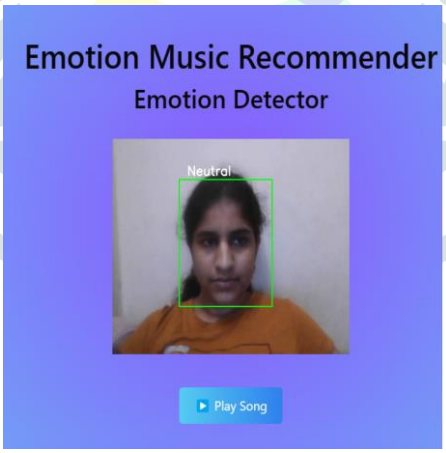

Fig12: “Neutral” mood detected

5.1 Play Song on Flask Page

Creating a system that detects real-time moods and displays accurately curated playlists is a fascinating application. In this instance, let's delve into the "happy" mood playlist displayed in Fig.13. When the system detects a "happy" mood, it retrieves and showcases a playlist tailored to evoke and match that specific emotional state.The accuracy of this playlist relies on the underlying mood detection algorithm's ability to correctly interpret facial expressions or other cues indicating happiness. Once the mood is identified, the system accesses a pre-defined playlist or dynamically generates one, selecting songs known to resonate with the emotion of happiness. These songs might feature upbeat melodies, cheerful lyrics, or compositions that evoke positive emotions.

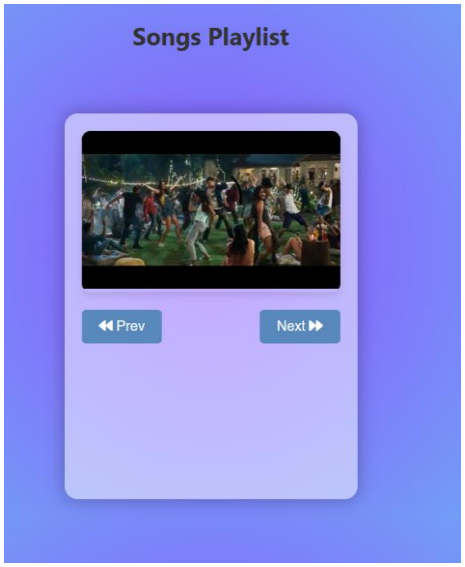

Fig 13: Play Song for "Happy" mood

6.CONCLUSION:

Our study focused on teaching computers to understand emotions by looking at people's faces. We used two special tools: Haar Cascade and Convolutional Neural Network (CNN) algorithms. These tools are like detectives and brainy experts—they work together to recognize emotions like happiness, sadness, anger, surprise, and a neutral expression by examining facial features. We gathered a huge bunch of face pictures—over 26,000 images—to train the computer. At first, our system learned a lot about emotions from these pictures, but it reached a point where it didn't improve much more, even with more training. Haar Cascade is good at spotting emotions in faces, but it needs a lot of computing power, while CNN, which is great at recognizing patterns, requires a big computer to work well. Despite the system's proficiency in understanding emotions from faces, we identified some limitations. Our work marks a step forward in making computers more emotionally aware. By combining Haar Cascade and CNN, we demonstrated how computers can grasp emotions from faces, potentially aiding in technology that better understands and responds to human feelings, like suggesting music based on how you feel.

7.Reference

[1].singh,A.,shoaib,M.,singh,s.,&sharma,S.(2023).Analysis of Real-Time Emotion Detection Techniques.Available at SSRN 4623338

[2].Behera, G. S. (2020, December 25). Face Detection with Haar Cascade — Part II: Detection with the readily available Haar Models.

[3] Velankar, M., Kotian, R., & Kulkarni, P. (2021). Contextual mood analysis with knowledge graph representation for Hindi song lyrics in Devanagari script. arXiv preprint arXiv:2108.06947.

[4].Shivananda, S., Dutt, R., Perumal, B., H, A., & Latha, A. P. (2022, July). Mood-Based Music Recommendation System: VIBY. International Research Journal of Modernization in Engineering Technology and Science, 4(7), 1158.

[5].Shetty, R., Kasbe, S., Jorwekar, K., Kamble, D., & Velankar, M. (2015). Study of Emotion Detection in Tunes Using Machine Learning. International Journal of Scientific and Research Publications, 5(11).

[6].Velankar, M., & Kulkarni, P. (2022). Music recommendation systems: overview and challenges. Advances in Speech and Music Technology: Computational Aspects and Applications, 51-69..

[7].Velankar, M., & Kulkarni, P. (2023). Employing cumulative rewards-based reinforcement machine learning for personalized music recommendation. Multimedia Tools and Applications, 1-14.

[8].Sana, S. K., Sruthi, G., Suresh, D., Rajesh, G., & Subba Reddy, G. V. (2022). Facial emotion recognition based music system using convolutional neural networks

[9].F.M. Javed Mehedi Shamrat, Anup Majumder, Probal Roy Antu, Saykot Kumar Barmon, Itisha Nowrin, Rumesh Ranjan "Human Face Recognition Applying Haar Cascade Classifier" International Conference on Pervasive Computing and Social Networking, Salem, Tamil Nadu, India, 19-20, March 2021.

[10].Rosebrock, A. (2021, April 5). OpenCV Face detection with Haar cascades.

[11].Debnath, T., Reza, M. M., Rahman, A., Beheshti, A., Band, S. S., & Alinejad-Rokny, H. (2022). Four-layer ConvNet for facial emotion recognition with minimal epochs and the significance of data diversity. Scientific Reports, 12(1), 6991.

[12].Dinalankara, L. (2017, August). Face Detection & Face Recognition Using Open Computer Vision Classifiers. University of Plymouth.

[13].Singh, G., & Goel, A. K. (2020). Face Detection and Recognition System using Digital Image Processing. In 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1-6). IEEE.

[14].Mehendale, N. (2020, February 18). Facial emotion recognition using convolutional neural networks (FERC). *SN Applied Sciences, 2*(446)

[15].Kalyankar, A., & Patil, M. (2022). Criminal Identification System Using Haar-Cascade Algorithm. *International Journal of Computer Research and Technology*, 10(6), page range. ISSN: 2320-2882

[16].Manas Sambare, FER2013 Dataset, Kaggle, July 19, 2023. Accessed on: September 9, 2023. [Online], Avalilable at: https://www.kaggle.com/msambare/fer2013