FACIAL EXPRESSION-BASED MUSIC RECOMMENDATION PLATFORM

Mr. Aniruddha Waghchaware[1], Mr. Rohit Rathod[2], Mr. Vinaykumar Injamure[3], Mr. Aniket Lavate[4], Prof. S. S Bhuite[5]

[1,2,3,4]Bachelor of Technology In Computer Science and Engineering Punyashlok Ahilyadevi Holkar University, Solapur, India.

[5]Project Guide Brahmadev dada Mane Institute of Technology, Solapur, India.

DOI: https://www.doi.org/10.58257/IJPREMS34190

ABSTRACT

In our research, we aim to address the challenge many people face when selecting music from a vast collection. We're designing a music recommendation system that tailors suggestions to each user's preferences. One innovative aspect of our approach involves analyzing users' facial expressions to understand their current emotional state. This can provide valuable insights into what type of music would best suit their mood.

We recognize that users often feel overwhelmed by the sheer number of songs in their libraries and struggle to decide what to listen to. Our system aims to alleviate this stress by offering personalized recommendations based on the user's emotional cues. By capturing the user's image via a webcam, we can determine their mood and suggest appropriate songs from their playlist.

By simplifying the decision-making process and offering timely suggestions, our system can help users enjoy their music without the hassle of searching for the perfect track. This not only saves time but also reduces stress levels, contributing to a more enjoyable listening experience.

Keywords: Face Recognition, Feature extraction, Emotion detection, Convolutional Neural Network, Music, Player, camera

1. INTRODUCTION

Human emotions are often expressed through facial expressions. Music has a powerful ability to influence one's mood and emotional state. This project aims to leverage this connection between facial expressions, emotions, and music.

The system uses a webcam to capture the user's facial image. It then employs image processing and computer vision techniques to extract facial features from the captured image. Using these extracted features, the system attempts to detect and recognize the emotion that the user is expressing through their facial expression.

The core idea is to recommend and play music tracks that are well-suited to the user's current emotional state, as detected by their facial expression. By playing music that complements the user's mood, the system aims to provide a calming and pleasant experience, potentially helping to uplift or regulate the user's emotional state.

Facial expression recognition has long been considered one of the most effective ways for humans to interpret and understand the emotions and feelings of others.

In some cases, mood regulation through music can potentially help alleviate conditions like depression or sadness. By recommending music that matches the user's emotional state, the system can serve as a tool for mood enhancement and overall emotional well-being.

2. LITERATURE SURVEY

Londhe et al. [1] proposed a paper focused on studying facial curvature changes and pixel intensity variations. They used Artificial Neural Networks (ANNs) for emotion classification and suggested various approaches for music playlists. Zheng et al. [2] proposed two significant categories for facial feature extraction: appearance-based and geometric-based, including extracting essential facial points like the mouth, eyes, and eyebrows.

Nikhil et al. [3] determined the user's mindset using facial expressions, as humans often express their feelings through expressions, hand gestures, and voice tone, but mostly through facial expressions. An emotion-based music player reduces the user's time complexity. With large song playlists, playing songs randomly may not satisfy the user's mood. This system helps play songs automatically according to the user's mood by capturing their image through a webcam, converting it to binary format (feature-point detection method or Haar Cascade technology), and using a Java program to manage the database and play songs based on the detected mood.

Zeng et al. [5] researched advances in human affect recognition, focusing on approaches for handling audio and visual recordings of affective states. They described affect as a prototype of emotion categories like happiness, sadness, fear, anger, disgust, and surprise. The paper discussed challenges in computing methods for developing automatic, spontaneous affect recognizers and identified problems missed or avoided in uni-modal posed emotion recognition.

Tambe et al. [7] proposed automating interactions between users and music players, learning user preferences, emotions, and activities, and providing song selections accordingly. The device recorded users' facial expressions to determine their emotions and predict their preferred music genre.

Jha et al. [11] proposed an emotion-based music player using image processing, showcasing algorithms and techniques suggested by different authors for connecting music players with human emotions. This approach aimed to reduce user effort in creating and managing playlists while providing a suitable song experience based on the user's current expression.

Anukritine et al. [18] developed an algorithm that provided a list of songs from the user's playlist according to their emotion, focusing on reducing computational time and cost. The algorithm categorized emotions into joy, sadness, anger, surprise, and fear, providing an accurate audio information retrieval approach that extracted relevant information from an audio signal in less time.

Aditya et al. [19] developed an Android application acting as a customized music player for users using image processing to analyze and present songs according to their mood. The application was developed using Eclipse and OpenCV for facial recognition algorithms. Images were captured using the mobile device's front camera, aiming to provide satisfaction to music lovers by extracting their emotions.

Habibzad et al. [20] proposed a new algorithm for recognizing facial emotions, including three stages: pre-processing, feature extraction, and classification. The first part described image processing stages like pre-processing and filtering for extracting facial features. The second part optimized eye and lip ellipse characteristics, and the third part used optimal eye and lip parameters for emotion classification. The results showed faster facial recognition speed compared to other approaches.

Prof. Nutan Deshmukh et al. [21] focused on creating a system that fetches the user's emotion using a camera and automates the result using an emotion detection algorithm. The proposed algorithm's average estimation time of 0.95-1.05 seconds for generating an emotion-based music system was better than existing algorithms and reduced design costs.

3. PROPOSED SYSTEM

The proposed approach combines a traditional computer vision technique and a deep learning model for human facial emotion recognition. Initially, a Haar cascade classifier is employed to localize and extract human faces from the input image. The detected facial regions are then cropped and resized to a standardized dimension of 48x48 pixels through normalization.

The preprocessed facial images serve as inputs to a Convolutional Neural Network (CNN) architecture. CNNs, a variant of feed-forward neural networks, are particularly adept at processing and analyzing visual data by learning hierarchical feature representations. These networks comprise interconnected computational units termed neurons, which operate collectively to identify patterns and classify input images.

The CNN model employed in this system incorporates three fundamental layers: the convolutional layer, the spatial pooling layer, and the fully connected layer. The convolutional layer applies learnable filters to the input image, extracting low-level visual features. The spatial pooling layer performs dimensionality reduction, enhancing the model's robustness to variations in the input. Finally, the fully connected layer integrates the extracted features and maps them to the output classes.

The CNN's output is a single class label ranging from 0 to 6, corresponding to the following facial expressions: 0 - angry, 1 - disgust, 2 - fear, 3 - happy, 4 - sad, 5 - surprise, and 6 - neutral.

4. CONVOLUTIONAL LAYER

The initial stage of the CNN architecture is the convolution layer, responsible for extracting salient features from the input image. Each input image is represented as a matrix of pixel values. The convolution operation involves the application of a filter, which is a smaller matrix of learnable weights. The movement of this filter across the input image is governed by a parameter called stride. A stride of 1 indicates that the filter shifts one pixel at a time, while a stride of 2 means the filter advances two pixels at each step.

The convolution process entails element-wise multiplication of the filter weights with the corresponding pixel values in the input image, followed by summing these products. This operation is performed by sliding the filter across the entire image, generating a feature map as the output. The feature map captures the presence of specific patterns or features in the input image, as detected by the filter.

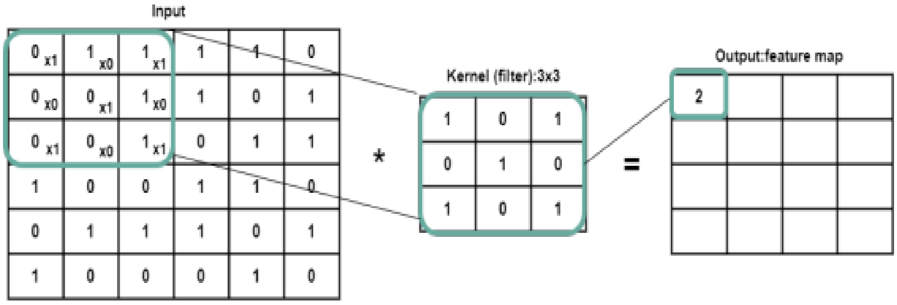


Figure 4.1 : Convolution Operations

4.2. ReLU Layer :

The ReLU (Rectified Linear Unit) layer introduces non-linearity into the CNN model's output. It serves as an activation function, a crucial component that adds complexity and enables the network to learn intricate patterns. The ReLU function operates by applying a simple rule to each element of the input: if the input value is negative, it is replaced with zero; if the input value is positive, it remains unchanged.

Mathematically, the ReLU function can be expressed as

$f(x) = \max(0, x)$,

where x represents the input value. This straightforward computation ensures that negative values are eliminated, while positive values pass through unaltered.

The output of the ReLU layer is a rectified feature map, which preserves the valuable features extracted by the preceding convolution operation while discarding irrelevant or negative information. This rectification process enhances the model's ability to capture and learn meaningful patterns from the input data.

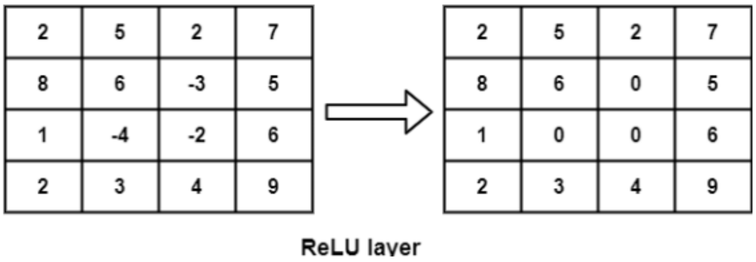


Figure 4.2 : Rectified feature map

4.3. Pooling Layer

The pooling layer in CNNs reduces the size of feature maps, cutting down computational complexity. It retains important info while discarding less relevant details. Three common pooling techniques are:

1) Max pooling: Keeps the highest value in a defined region.
2) Average pooling: Takes the average value in a defined region.
3) Sum pooling: Adds up all values in a defined region.

Pooling is applied to non-overlapping regions, shrinking spatial dimensions. This helps capture essential image features while reducing minor variations' impact. Strategically placed pooling layers enhance the model's ability to recognize patterns and generalize effectively.

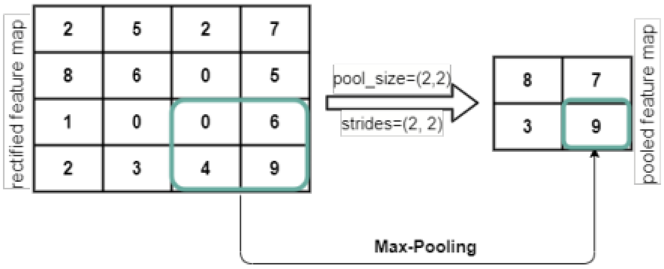


Figure 4.3 : Pooled feature map

After spatial max pooling, the feature map is flattened into a long vector and fed into a fully connected layer for classification.

4.4. Flatten Layer

The flattening layer takes the 2-dimensional feature map and transforms it into a format that can be easily processed by the fully connected layer, which then uses this information to classify the input image into one of the predefined categories.

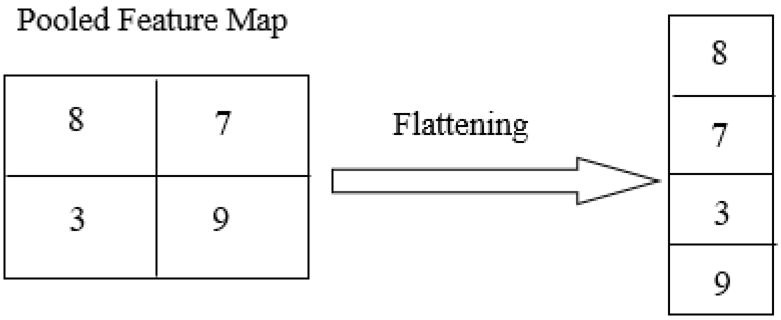


Figure 4.3 : Flattening Layer

4.5. Fully Connected Layer :

The CNN model integrates convolutional layers for extracting features, pooling layers for reducing dimensions, and fully connected layers for classification, with a focus on detecting facial expressions and recognizing emotions such as happiness, sadness, and neutrality.

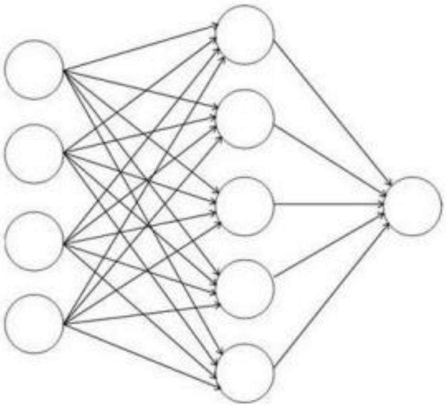


Figure 4.5.1 : Fully Connected Layer

It includes convolution layers to extract features, pooling layers to reduce dimensions, and fully connected layers for classification. We use 3x3 filters in convolution layers with a stride of (2,2) to find important features. Max pooling with a pool size of 2x2 and stride (2,2) is used to decrease feature map dimensions. ReLU activation function adds non-linearity. The output layer uses softmax for result calculation.

This CNN model aims to detect faces and recognize emotions like happy, sad, and neutral. It involves image processing, feature extraction, and classification. Initially, faces are detected, features extracted, and normalized. Then, a classifier categorizes emotions based on a training model. CNNs learn specific properties for each emotion, enhancing emotion detection.

5. METHODOLOGY

5.1. Dataset Description

We built a Convolutional Neural Network (CNN) model using the FER2013 dataset sourced from Kaggle, which consists of 24,176 training images and 6,043 testing images. Each image in the dataset depicts grayscale faces at a resolution of 48x48 pixels and is labeled with one of five emotions: happy, sad, angry, surprise, or neutral. The dataset encompasses both posed and unposed headshots, with faces automatically aligned to be roughly centered and occupy a consistent amount of space. This dataset compilation involved Google image searches for emotions and their synonyms. Given the imbalance in emotion representation within the dataset, we addressed this challenge by implementing a weighted-SoftMax loss approach during training. This technique adjusts the loss function based on the relative frequencies of each emotion class, ensuring a more balanced training process. Additionally, to handle any missing or outlier values within the dataset, we incorporated the categorical crossentropy loss function during training iterations. This helped to effectively manage irregularities in the data and enhance the model's robustness.

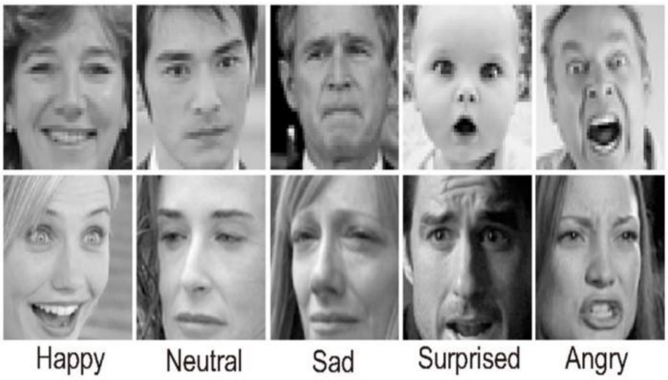


Figure 5.1 : Samples from FER-2013 Dataset

5.2. Emotion Detection Module :

5.2.1. Face Detection

Face detection, a facet of computer vision, involves locating faces or objects within images using specialized algorithms. This process can occur in real-time from video frames or static images. Algorithms, called classifiers, are trained to differentiate between what constitutes a face (1) and what does not (0) in an image. These classifiers, like those in OpenCV, utilize techniques such as Local Binary Pattern (LBP) and Haar Cascades. Specifically, Haar classifiers are adept at face detection, having been trained on diverse face data to ensure accuracy across various face types.

The primary objective of face detection is to precisely identify faces within an image frame while mitigating external factors and noise. It operates on a machine learning paradigm, where a cascade function is trained on a set of input files. This function leverages the Haar Wavelet technique to analyze pixels within the image, breaking them down into squares. This rigorous analysis enables the classifier to accurately discern facial features amidst varying environmental conditions and facial orientations.

5.2.2. Feature Extraction

During feature extraction, we utilize a pre-trained network as a sequential model to extract features from images. We allow the input image to flow through the network until a specific layer is reached, capturing the outputs of that layer as our features. Initially, the early layers of the convolutional network focus on extracting basic features using a small number of filters.

However, as we progress deeper into the network, the number of filters increases, often doubling or tripling in size compared to the previous layer. While these deeper layers extract more complex features, they also require significantly more computational resources.

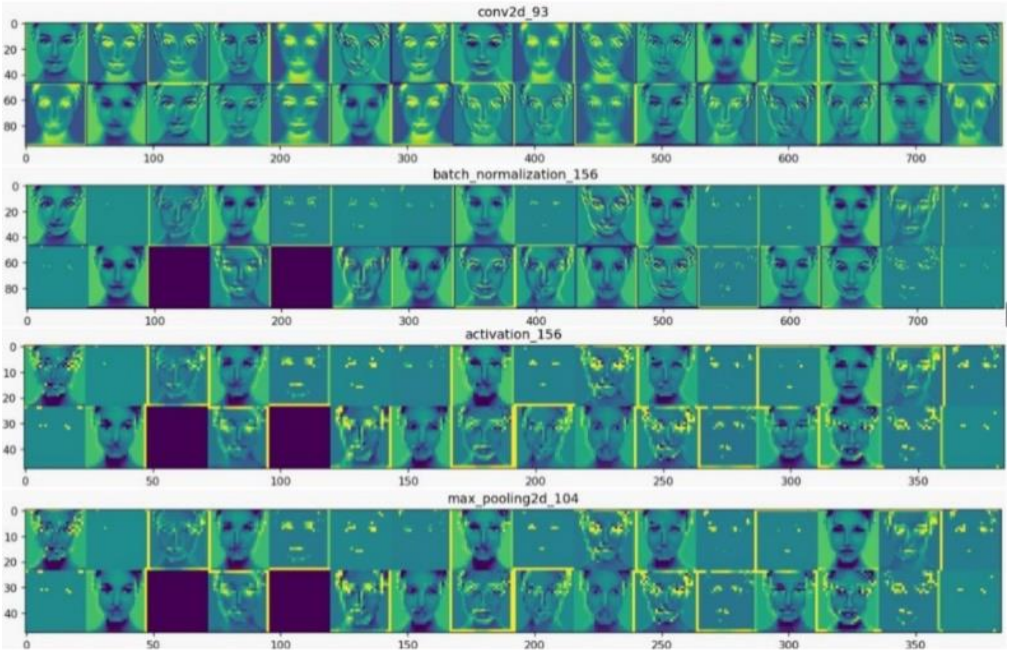


Figure 5.2.2. : Visualization of The Feature Map

5.2.3. Emotion Detection

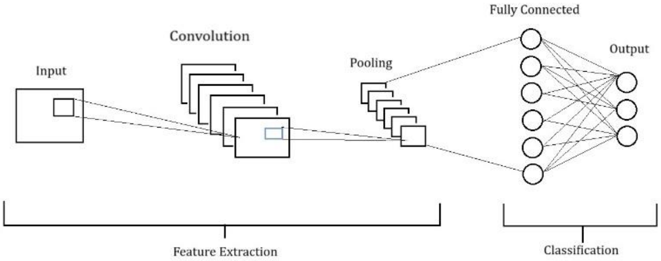


Figure 5.2. 3.1. : CNN Architecture.

In a convolutional neural network (CNNarchitecture, filters or feature detectors are applied to the input image to create feature maps or activation maps using the ReLU activation function. These filters help identify different features in the image, such as edges, lines, and bends. After this, pooling is applied to the feature maps to ensure invariance to translation, meaning that small changes in the input won't affect the pooled outputs.

Pooling methods like min, average, or max can be used, but max-pooling typically gives better results. Finally, the flattened inputs are passed to a deep neural network to determine the class of the object being analyzed.

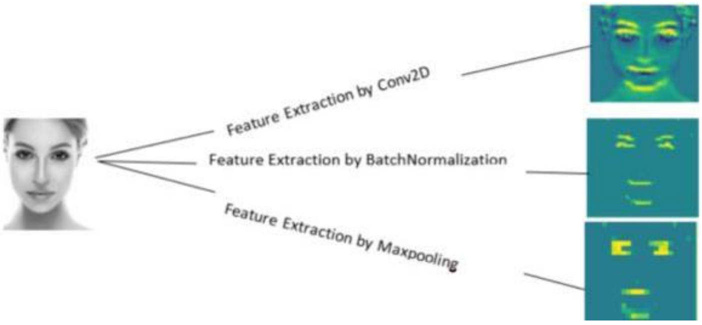

Feature Extraction by Conv2D
Feature Extraction by BatchNormalization
Feature Extraction by Maxpooling

Figure 5.2.3.2. : Feature Extraction by each layer in CNN

In image classification, the image class can be binary or multi-class. Neural networks learn features in a way that's not easily interpretable. We input an image into a CNN model, which returns the classification results. For emotion detection, a pre-trained CNN model predicts emotions from real-time user images, adding a label indicating the predicted emotion.

6. RESULT AND ANALYSIS

The implementation uses OpenCV in Python with a Haar cascade classifier and requires various input images for testing the model. After successful execution, the model can recognize input images with a face detection rate of 95%. Performance analysis in Following figure illustrates the efficient detection of faces in minimal time.

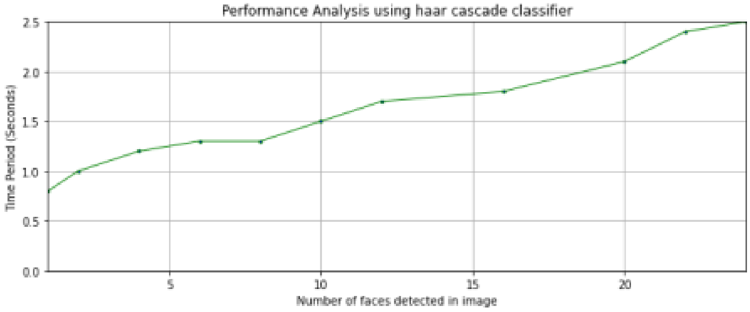


Figure 6 .1: Performance Analysis of Haar cascade classifier.

Training data is processed using the CNN algorithm for feature extraction, leading to the creation of a new trained model capable of predicting facial expressions. The model's accuracy and loss were evaluated across different dataset proportions, revealing overfitting issues. To address this, dropout layers were introduced after each pooling layer and data augmentation was applied to reduce complexity and overfitting. Further experiments confirmed the effectiveness of these adjustments. Training time per epoch with a batch size of 64 was approximately 87 seconds. During training, both training accuracy and validation accuracy increased while losses decreased. By the 106th epoch, validation accuracy reached 0.6559 with a validation loss of 1.04.

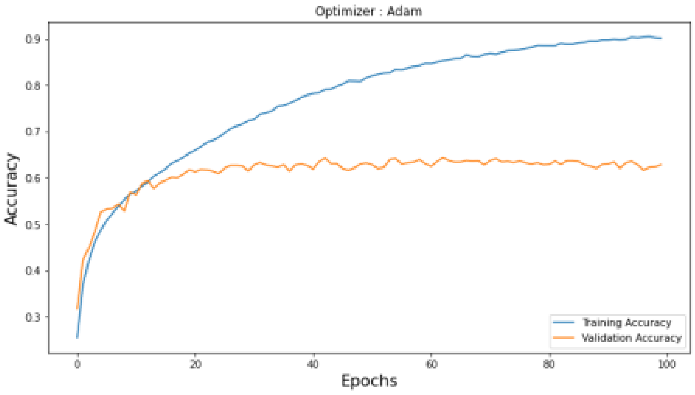


Figure 6.2. : Training and validation accuracy

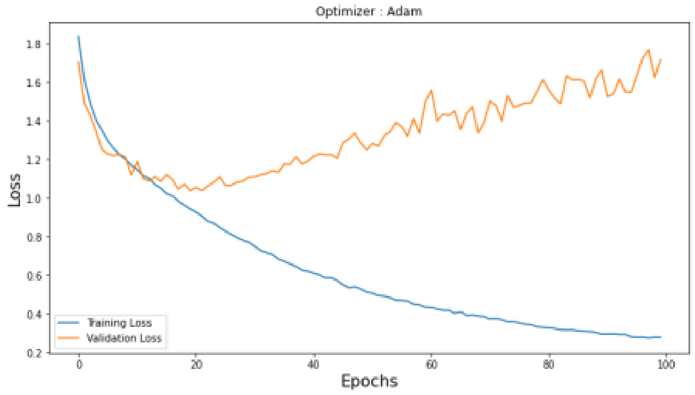


Figure 6.3. : Training and validation loss

The trained model predicts probabilities for each emotion class, as shown in following figure achieving high accuracy for emotions like happy and surprised, moderate accuracy for neutral, and lower accuracy for sadness and fear. Overall, the model effectively predicts emotions, although it may struggle with certain emotions like anger and disgust.

The confusion matrix of the CNN model was generated using the Adam optimizer with a learning rate set to 0.001 and a decay value. This matrix provides quantitative insights into the emotion recognition performance on the FER2013 dataset. The y-labels represent the actual emotions, while the x-labels represent the predicted emotions. Results encompass predictions for seven emotion classes, with the "happy" class achieving the highest prediction accuracy, followed closely by the "surprise" class. However, other emotion classes such as "anger" and "fear" demonstrated lower accuracy compared to "happy" and "surprise" classes.

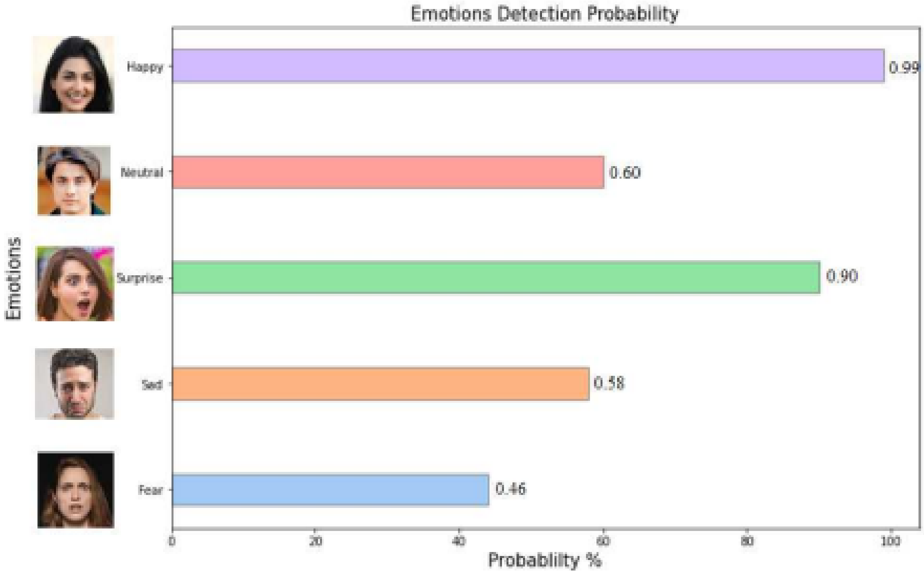


Figure 6.4. : Probability of Emotions Detection

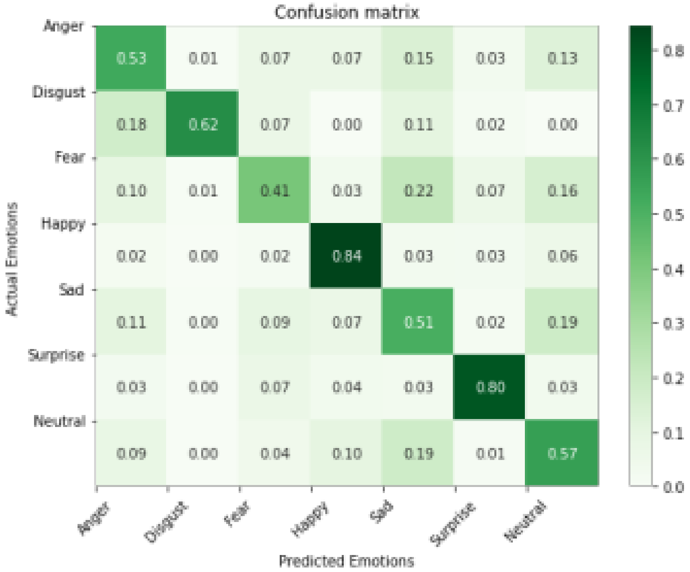
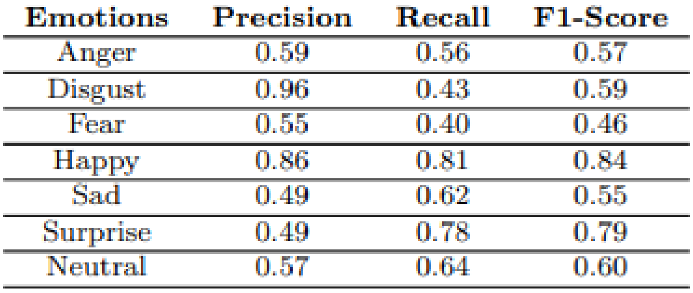


Figure 6.5. : Confusion Matrix of the model

The ROC curve of the model on FER2013 shows strong performance (Fig.19). It visualizes the probability curve, with AUC indicating separability. Different lines represent ROC curves for seven emotion categories. The system detects faces in input images, focusing on the facial region. The softmax output layer computes the final result. The CNN architecture identifies faces, recognizes emotions, and classifies images into happy, sad, neutral, etc., using the dataset



Emotions	Precision	Recall	F1-Score
Anger	0.59	0.56	0.57
Disgust	0.96	0.43	0.59
Fear	0.55	0.40	0.46
Happy	0.86	0.81	0.84
Sad	0.49	0.62	0.55
Surprise	0.49	0.78	0.79
Neutral	0.57	0.64	0.60

Figure 6.5. : Precision, recall, and f1-score of the model

The ROC curve on FER2013 indicates strong model performance, visualizing probability with AUC indicating separability. The system detects faces in input images, focusing on facial regions. The CNN architecture classifies images into emotions like happy, sad, etc. The model successfully detects single-face emotions in Fig.20 and Fig.21, and multiple-face emotions across various expressions. These experiments confirm accurate recognition of different facial expressions.

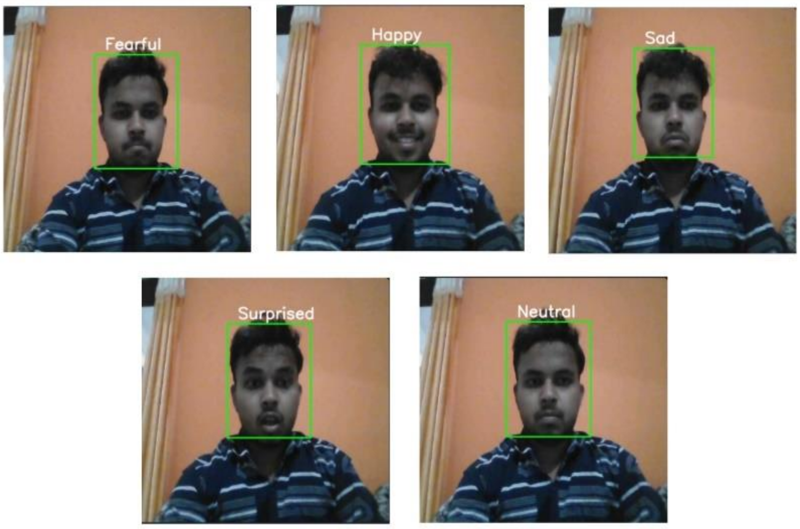


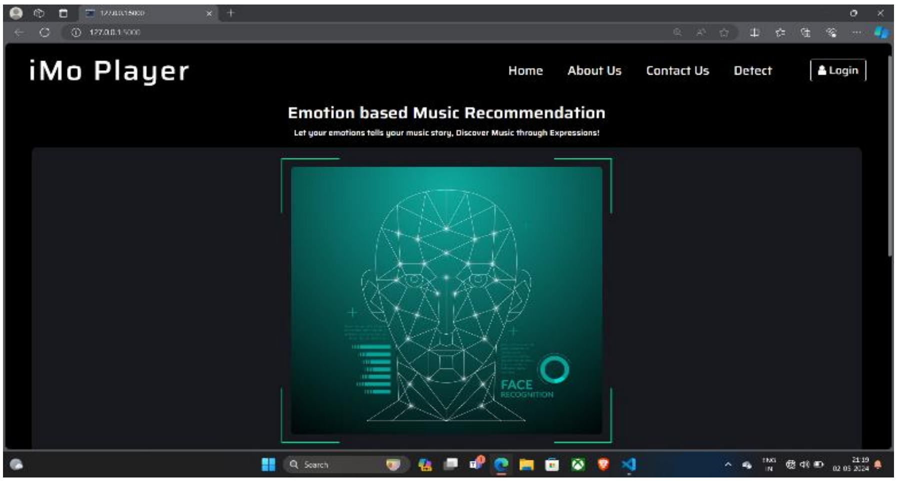
Figure 6.5. : Multi Emotion Detection of a Face

Following table shows the performance comparison of validation accuracy of the previous researcher's methods and the proposed method.

Method	Dataset	# of Facial Expressions	Validation Accuracy
Tümen et al [30]	FER2013	7	57.1%
Wang et al [8]	FER2013	7	58.8%
Knyazev et al [31]	FER2013	7	60.0%
Liu et al [32]	FER2013	7	62.44%
Giannopoulos et al [33]	FER2013	7	64.20%
Haque et al [34]	FER2013	7	63.11%
Our proposed Method	FER2013	7	65.59%

Figure 6.5. : Performance comparison between previous methods and the proposed method

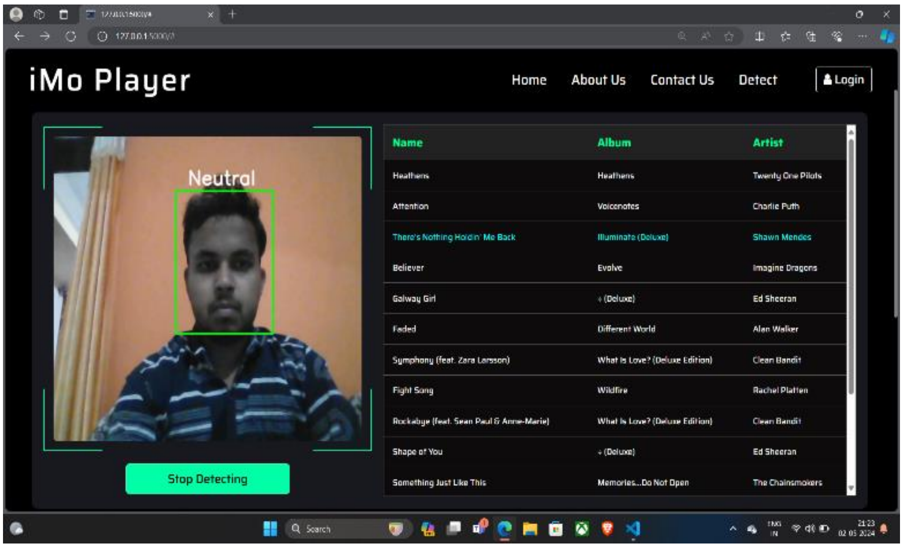
Result & Output:



Emotion based Music Recommendation

Let your emotions tells your music story, Discover Music through Expressions!

Figure 6.6. : Landing Page of iMO Player



Name	Album	Artist
Heathens	Heathens	Twenty One Pilots
Attention	Voicenotes	Charlie Puth
There's Nothing Holdin' Me Back	Illuminate (Deluxe)	Shawn Mendes
Believer	Evolve	Imagine Dragons
Galway Girl	÷ (Deluxe)	Ed Sheeran
Faded	Different World	Alan Walker
Symphony (feat. Zara Larsson)	What Is Love? (Deluxe Edition)	Clean Bandit
Fight Song	Wildfire	Rachel Platten
Rockabye (feat. Sean Paul & Anne-Marie)	What Is Love? (Deluxe Edition)	Clean Bandit
Shape of You	÷ (Deluxe)	Ed Sheeran
Something Just Like This	Memories...Do Not Open	The Chainsmokers

Figure 6.7. : Recognization of Emotion from a face

7. CONCLUSION

The CNN algorithm is highly effective in image processing for precise results, particularly in recognizing facial emotions. Our paper introduces a CNN model designed to detect human facial emotions across seven classes. Face detection is achieved using the Haar-Cascade Classifier. We trained the CNN model using the FER-2013 dataset, splitting the data into training, testing, and validation sets. Through various experiments, our system achieved a validation accuracy of 65.59% at epoch 105, with training and validation losses of 0.6 and 1.0 respectively. In the future, we plan to enhance our approach by incorporating different patterns using deep learning algorithms.

By continuously refining and expanding our approach, we strive to contribute to the advancement of emotion detection technology in diverse applications ranging from healthcare to human-computer interaction.

8. REFERENCES

[1] "A Music Recommendation System Based on Facial Expression Recognition and Emotion Classification" by Junjie Gu, Xiong Guan, and Chengjie Tu. (2021)

[2] "Affective Computing and Sentiment Analysis: Emotion, Metaphor and Terminology" by Khurshid Ahmad, Published by Springer, 2011.

[3] "Music Recommendation System Based on Facial Expression Analysis" by Muhammad Usman, Yasir Saleem, Fahad Ahmad, and Munam Ali Shah. (2019)

[4] "Emotion-Based Music Recommendation Using Facial Expression Recognition" by Prachi Jain, Abhilash Sanap, Pratham Arora, and Mansi Agrawal. (2018)

[5] "Facial Expression Recognition: Emotion Mapping" by Dr. Renu Nagpal, Published by Springer, 2018.

[6] "Music Recommendation System Based on Facial Expression" by Toshiki Hayashi, Hirokazu Kimura, and Takehiro Ohya. (2017)

[7] "Emotion Recognition: A Pattern Analysis Approach" by Amit Konar and Aruna Chakraborty, Published by John Wiley & Sons, 2015.

[8] "Music Recommendation System Based on Facial Emotion Recognition" by Sujata Sinha, Shaili Majumder, and Anuja Mishra. (2021)

[9] "Affect and Emotion in Human-Computer Interaction: From Theory to Applications" by Christian Peter and Russell Beale, Published by Springer, 2008.

[10] "Facial Analysis: Concepts and Methods" by Harry Wechsler, Published by Springer, 2018.

[11] "Affective Computing" by Rosalind W. Picard, Published by MIT Press, 1997.

[12] "Multimodal Emotion Recognition: A Deep Learning Approach" by Stefanos Eleftheriadis, Orestis Zachariadis, and Vassilis Kyritsis, Published by Springer, 2020.

[13] "Emotion Recognition: A Computational Perspective" by Mehdi Ghayoumi, Published by Springer, 2022.

[14] "Emotional Intelligence in Artificial Intelligence: The Next Step in Human-Machine Interaction" by Emad Bataineh and Dr. Sultan Naji, Published by CRC Press, 2022.

[15] "Affective Computing and Intelligent Interaction" by Ana Paiva, Rui Prada, and Rosalind W. Picard, Published by Springer, 2007.