Music Therapy Treatment based on Emotion Detection

Elisha Maria K Orlina
A20520119
eorlina@hawk.iit.edu

Raj Shah
A20524266
rshah114@hawk.iit.edu

Abhishek Jaiswal
A20380004
ajaiswal1@hawk.iit.edu

ABSTRACT

Music is used in one's everyday life to modulate, enhance, and plummet undesirable emotional states like stress, fatigue, or anxiety. Even though in today's modern world, with the ever-increasing advancement in the area of multimedia content and usage, various intuitive music players with various options have been developed, the user still has to manually browse through the list of songs and select songs that complement and suit his or her current mood. This report aims to solve the problem of mental health issues and help people to overcome mental stress and anxiety. Furthermore, this report also aims to unravel the issue of the manual finding of songs to suit one's mood along with creating a high-accuracy CNN model for Facial emotion recognition. Through the webcam, the emotional state can be deduced from facial expressions. To create a neural network model, the CNN classifier was used. This model is trained and tested using OpenCV to detect mood from facial expressions. A playlist will be recommended through the system according to the mood predicted by the model. The model used for predicting the mood obtained an accuracy of 97.42 % with 0.09 loss on the training data from the FER2013 Dataset.

Keywords— Convolutional neural networks (CNNs), Facial emotion recognition, FER2013, Music Therapy, OpenCV.

I. INTRODUCTION

A. *Problem Summary*

Facial expressions play a critical role in revealing a person's emotions [2]. The future of computer vision systems may rely heavily on dynamic user interactions, such as those found in present-day computer systems. Facial emotions have a significant impact on security, entertainment, and human-machine interaction (HMI), and they can be conveyed through a person's lips and eyes.

In today's world, having a vast music playlist is commonplace, as music is a crucial source of entertainment and can even serve as a therapeutic tool in some cases. Music can uplift one's mood, boost happiness, and help reduce anxiety, and it has been an integral part of the human experience for ages [4]. The correlation between music and mood is so apparent that its impact is often known to improve one's emotional and psychological well-being.

This project proposes a system that detects a person's mood through facial emotion recognition and generates a music playlist based on the detected mood. Unlike humans, machines and detecting facial expressions are challenging tasks. However, with advancements in technology, Convolutional Neural Networks (CNNs) have proven to be an efficient solution for facial emotion recognition.

Therefore, this report and project try to solve problems that are untouched by other available systems.

B. *Previous Works*

Various methods exist for emotional analysis, such as facial expressions, body movements, gestures, and speech. Research has explored different approaches for detecting and classifying emotional and physiological states expressed through facial features [6]. Typically, facial digital images are pre-processed, and various feature extraction and classification algorithms are used. Convolutional neural networks (CNNs) are a promising option for facial emotion recognition, and researchers have worked to improve their performance by adjusting their hyperparameters and architecture.

One study used the Random Search algorithm to create and train models with multiple hyperparameter combinations and architectures. The resulting CNN-based method achieved an accuracy of 72.16 percent when applied to the FER2013 dataset. Facial emotional processing plays a vital role in many fields, including engagement with humans and computers, system vision, online game testing, and customer studies. Facial expressions serve as a form of nonverbal communication, providing insight into a person's inner emotions and feelings. Researchers have developed and tested various deep-learning methods to recognize facial expressions and sentiments accurately.

Some studies propose using facial and semantic analysis to recommend songs to users based on their moods. Other studies have explored the use of deep learning architectures for facial expression identification and have examined various datasets and models' accuracies.

This study [5] explores the various datasets commonly used for identifying facial emotions, as well as the different deep-learning modules used for detecting emotions from facial expressions. The study highlights several CNN models and

their respective accuracies, along with the datasets they were applied to. Additionally, the research provides an overview of the latest advancements in facial expression recognition for detecting emotions using various deep-learning architectures.

We also analyzed one particular model based on deep learning and board learning with an accuracy of approx 93%. The below workflow gives us a quick overview of how this model works in the real-world [13].

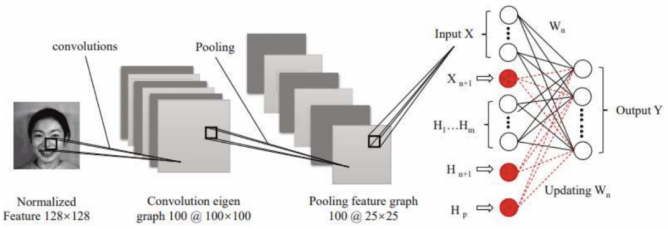

Fig. 1. Structure of CNNBL with Efficient Incremental Reconstruction

II. Methodology

A. Data Processing

To predict the mood or emotion of the user, Convolutional Neural Networks (CNNs) are employed to develop the model. The FER 2013 Dataset is utilized to train the model, which consists of grayscale facial images of faces with a dimension of 48 pixels in height and width.

Fig. 2. A sample of FER 2013 Dataset

The dataset is composed of a training set containing 28,709 examples and a public test set comprising 3,589 examples. Additionally, the faces in the dataset are automatically registered, ensuring that they are centered in each image and occupy roughly the same amount of space.

B. Model Training & Hyperparameter Tuning

After getting the data and performing resizing and rescaling we went ahead and separated the data into training, testing, and validation sets. The dataset consisted of around 35 thousand images split into 7 different moods, Happy, Disgust, Sad, Fear, Anger, Neutral, and Surprise. 80% of the data was set aside for training, 10% for validation, and 10% for testing.

The first CNN architecture was with 3 Convolutional layers along with a Maxpooling layer and finally the flatten and dense layers. We used 'relu' as the activation function and for the dense layer, 'softmax' was used. The model was intially trained for 150 epochs and achieved an accuracy of 44.83%. While upon using the Keras tuner we trained the model with the best model as given by the Keras tuner. An accuracy of 55.19 % was achieved, around 10% more than the first model.

C. Batch Normalization

The process of improving the speed and stability of a deep neural network involves adding additional layers, a technique known as "batch normalization" [12]. This method involves standardizing and normalizing the input of a previous layer through a new layer.

Batch normalization is a two-step process, whereby the input is initially normalized, followed by re-scaling and off-setting. This approach enhances the performance of the neural network, making it more efficient and reliable.

Batch Normalization is a two-step process.

- Input normalization
- Re-scaling and offsetting

Normalization of the input:

Normalization refers to the process of transforming data to have a mean of zero and a standard deviation of one.

- **Step: 1**
  In this phase, we have a batch input from layer h, and the first step is to calculate the mean of this hidden activation.

$$\mu = \frac{1}{m}\sum_{i=1}^{m} h_i \tag{1}$$

  Here, m is the number of neurons at layer h.

- **Step: 2**
  Now, the next step is to calculate the standard deviation of the hidden activations, using-

$$\mu = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(h_i - \mu)^2} \tag{2}$$

- **Step: 3**
  Moreover, the mean and standard deviation have already been calculated and will be utilized for normalizing the hidden activations. This will be accomplished by subtracting the mean from each input and then dividing the result by the sum of the standard deviation and the smoothing term (ϵ).

$$h_i(norm) = \frac{h_i - \mu}{\sigma + \epsilon} \tag{3}$$

- **Step: 4**
  In the final step of the process, the input undergoes rescaling and offsetting. This is achieved through the use of two BN algorithm components - γ (gamma) and β (beta). The purpose of these parameters is to rescale (γ) and shift (β) the vector that contains the results of the previous operations.

$$h_i = \gamma h_i(norm) + \beta \tag{4}$$

  The neural network ensures that the two parameters, γ and β, are learn-able and optimized throughout the training process, allowing for accurate normalization of each batch.

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_12 (Conv2D)          (None, 46, 46, 64)        640

 max_pooling2d_12 (MaxPoolin  (None, 23, 23, 64)       0
 g2D)

 batch_normalization_12 (Bat  (None, 23, 23, 64)       256
 chNormalization)

 conv2d_13 (Conv2D)          (None, 21, 21, 128)       73856

 max_pooling2d_13 (MaxPoolin  (None, 10, 10, 128)      0
 g2D)

 batch_normalization_13 (Bat  (None, 10, 10, 128)      512
 chNormalization)

 conv2d_14 (Conv2D)          (None, 8, 8, 256)         295168

 max_pooling2d_14 (MaxPoolin  (None, 4, 4, 256)        0
 g2D)

 batch_normalization_14 (Bat  (None, 4, 4, 256)        1024
 chNormalization)

 flatten_4 (Flatten)         (None, 4096)              0

 dense_8 (Dense)             (None, 256)               1048832

 dense_9 (Dense)             (None, 7)                 1799

=================================================================
Total params: 1,422,087
Trainable params: 1,421,191
Non-trainable params: 896
_________________________________________________________________
```

Fig. 3. The Summary of CNN Model Architecture

D. Best Model Parameters

The below-represented model showcased the best accuracy.

- Part 1 - The use of Conv2D 64 Size, 128 Size, and 256 Size along with MaxPooling Size (3, 3) means that we are applying convolutional layers with 64, 128, and 256 filters respectively, each with a 3x3 kernel size.
  After each convolutional layer, we are applying a Max-Pooling layer with a 3x3 pooling window to downsample the feature maps.

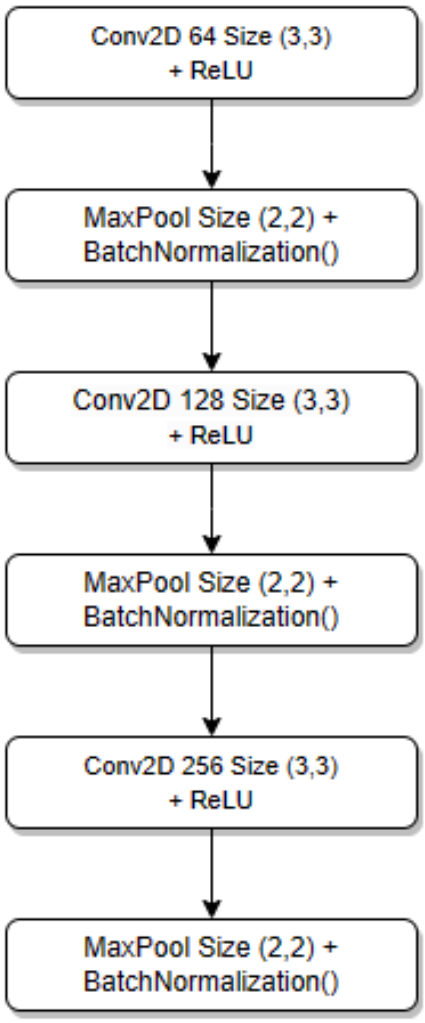

Fig. 4. CNN Architecture of Proposed Model

- Part 2 - Dense 256 and ReLu

  After applying CNNs and Maxpooling (Using Conv2D 64 Size, 128 Size, and 256 Size along with MaxPooling Size (3, 3) we moved on to this part wherein we tried to flatten followed by Dense 256 and ReLU activation function.
  In addition to the convolutional and pooling layers, CNNs typically include one or more dense (or fully connected) layers that are responsible for the final classification of the input image. Dense layers take the flattened feature maps produced by the convolutional and pooling layers and transform them into a vector of class probabilities.
  In this case, we have not specified the exact number or size of the dense layers used in the CNN, but it is common to use several dense layers with decreasing

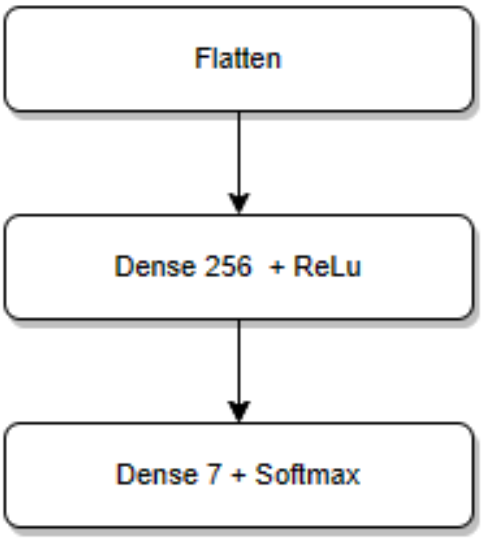

Fig. 5. CNN Architecture of Proposed Model

size, followed by a final output layer with the Softmax activation function. The Softmax function normalizes the output of the previous layer into a probability distribution over the possible classes, making it suitable for multi-class classification tasks.

The exact architecture of the CNN and the number and size of the dense layers used can vary depending on the specific task and the complexity of the input data. In general, larger and more complex datasets may require deeper and more complex CNN architectures with more filters, pooling layers, and dense layers to achieve high accuracy. However, deeper and more complex architectures also require more computational resources and may be more prone to overfitting if the training dataset is not large enough.

Overall, the combination of convolutional and pooling layers with dense layers and the Softmax activation function is a powerful and widely used approach for image classification tasks, and has achieved state-of-the-art results on many benchmark datasets.

E. System Flow

To use the proposed system, the user first captures their image using their webcam. With Web RTC, the user's webcam is accessed to show their face, and the user can choose when to capture the image by clicking the capture button. The captured image is then displayed on a canvas on the website, and if the user is satisfied with it, they can click the upload button to transmit the image to the server via AJAX. The image is delivered as a base 64 string and later converted back to an image.

Next, the image is analyzed to detect if there is a face, and if found, the face is cropped from the original image using Haar cascade frontal face detection. Image resizing is then applied to change the size of the input image to 48 x 48 for the CNN classifier. The Trained CNN model then predicts the emotion through the image and we move on to the next step.

After the emotion detection, the proposed system will generate a personalized playlist for the user based on their emotional state. The playlist can contain songs that are known to positively influence the user's mood, which can enhance the effectiveness of the therapy process.

A complete process is represented in the system flow-

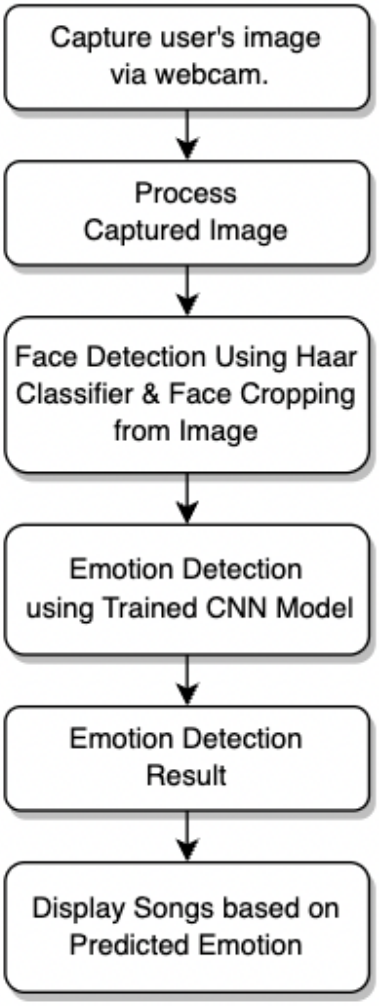

Fig. 6. Flow of the Proposed System

III. RESULT

To evaluate the proposed model's overall performance, initially it underwent 150 epochs of training on the FER

2013 dataset. After EDA and visualization, we noticed the data for one of the feature was lacking in comparison to the other features. Therefore, we augmented through using various augmentation techniques like - horizontal mirroring, rotations of ±10°, image zooms of ±10%, and horizontal and vertical shifting of ±10%.

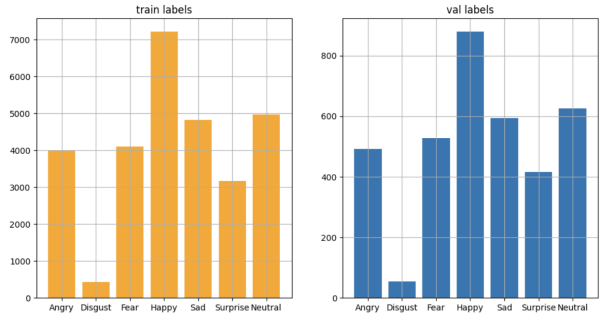

Fig. 7. Before Data Augmentation

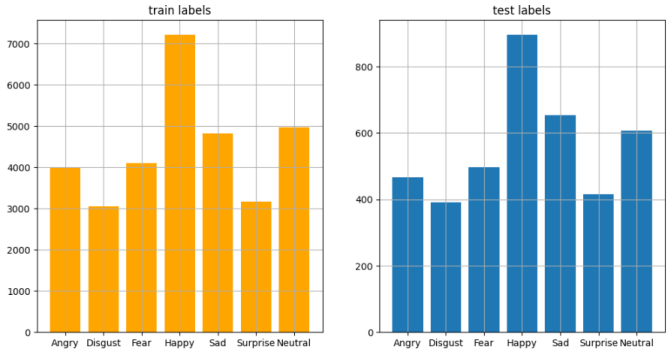

Fig. 8. After Data Augmentation

During this training stage, the Adam optimizer was chosen with a learning rate of 0.0001 and a batch size of 25. After completion of the training, the model achieved an accuracy of 55% with a loss of 1.4 on the validation data while an ideal 97% accuracy with a loss of 0.09 on the training data from FER2013.

A. *Confusion Matrix*

In this case, the confusion matrix suggests that the model was able to correctly detect 6 out of 10 moods in real life, which means that the model's accuracy is 60%. This is a moderate level of accuracy, but there is still room for improvement in future work.

To improve the accuracy of the model, several approaches can be considered. One approach is to collect more data for training the model, as a larger and more diverse dataset can help the model learn more robust features and generalize better to new data. Another approach is to try different model architectures, hyperparameters, or optimization techniques to improve the model's performance.

Overall, the goal of future work is to accurately predict all moods and make the model a reliable tool for music therapy-based systems. This can have significant implications for mental health and well-being, as music therapy has been shown to be an effective intervention for a range of emotional and psychological conditions.

```
array([[210,   3,  62,  31,  89,  19,  53],
       [105, 182,  35,  14,  42,   0,  14],
       [ 55,   2, 191,  28, 107,  54,  59],
       [ 43,   2,  46, 610,  75,  37,  82],
       [ 69,   4,  88,  39, 315,  31, 107],
       [ 17,   0,  47,  20,  11, 308,  12],
       [ 57,   1,  66,  54, 118,  25, 286]])
```

Fig. 9. Confusion Matrix

IV. Conclusion

The proposed system for music therapy treatment is a game-changer in the field of emotional wellness. By utilizing facial expression detection and personalized playlist generation, the system eliminates the burdensome task of manually searching for music playlists based on mood. This innovative approach has the potential to revolutionize therapy sessions, making them more efficient, personalized, and impactful.

With the incorporation of convolutional neural network (CNNs) models and batch normalization techniques, the system achieved an impressive accuracy rate of 97.42% on the training data. This outstanding performance demonstrates the potential of this system to accurately detect the user's emotions and tailor the playlist accordingly.

The potential for this system is vast, as it could be further customized to cater to different languages and cultures, creating a more inclusive therapy experience. The automation of mapping songs to moods through audio emotion recognition further streamlines the process, making it more efficient and effective.

In conclusion, this system has the potential to change the way we approach emotional wellness by creating a more personalized and effective music therapy experience.

V. Future Aspirations

To further enhance the model's performance, audio emotion recognition could be integrated with facial emotion recognition. This would result in a more comprehensive and accurate assessment of the user's emotional state, leading to an even more effective therapy experience.

Moreover, the system could be customized to include music in different languages, based on the user's preference and spoken language. By detecting the language spoken by the

user, the system could generate songs in that language, making the therapy process more culturally relevant and personalized.

Automating the mapping of songs to moods could also be achieved using audio emotion recognition, streamlining the therapy process and making it more efficient. Additionally, the creation of a user account system would allow users to save recommended songs to their accounts for future use, further enhancing the user experience.

Overall, this system for music therapy treatment based on emotion detection has great potential to revolutionize the way we approach therapy, by making it more personalized, efficient, and effective.

VI. Source Code & Data

- Code Base: Github

VII. Acknowledgment

We would like to express our sincere gratitude to *Prof. Yan Yan* for providing us with guidance and support throughout this project. His teachings, insights, and feedback have been invaluable in shaping the final project and report.

References

[1] Sharmeen M. Saleem Abdullah, Adnan Mohsin Abdulazeez "Facial Expression Recognition Based on Deep Learning Convolution Neural Network: A Review", Journal of Soft Computing and Data Mining, 2021.

[2] Swapnil V Dhatrak, Ashwini G Bhange, Abhishek K Sourabh, Prof. Yogesh A Handge "A Survey Emotion Based Music Player through Face Recognition System" International Journal of Innovative Research in Science, Engineering and Technology, 2019.

[3] Wafa Mellouk, Wahida Handouzi "Facial emotion recognition using deep learning: review and insights", The 2nd International Workshop on the Future of Internet of Everything (FIoE) 2020.

[4] MinSeop Lee, Yun Kyu Lee, Myo-Taeg Lim and Tae-Koo Kang "Emotion Recognition Using Convolutional Neural Network with Selected Statistical Photoplethysmogram Features" Applied Sciences MDPI, 2020.

[5] Thompson, David. 2022. "Mental Health - an Important Conversation in the Tech Industry." Tech Times. February 4, 2022.

[6] Ahlam Alrihaili, Alaa Alsaedi, Kholood Albalawi, Liyakathunisa Syed "Music Recommender System for users based on Emotion Detection through Facial Features", Developments in eSystems Engineering (DeSE), 2019.

[7] A. T. Lopes, E. de Aguiar, A. F. De Souza, and T. OliveiraSantos, "Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order," Pattern Recognit., vol. 61, pp. 610–628, 2017

[8] G. Zeng, J. Zhou, X. Jia, W. Xie, and L. Shen, "Hand-crafted feature guided deep learning for facial expression recognition," in Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), 2018

[9] M. Koujan, L. Alharbawee, G. Giannakakis, N. Pugeault, and A. Roussos, "Real-time facial expression recognition 'in the wild' by disentangling 3D expression from identity," in Proc. 15th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), 2020

[10] Z. Du, S. Wu, D. Huang, W. Li, and Y. Wang, "Spatio-temporal encoderdecoder fully convolutional network for video-based dimensional emotion recognition," IEEE Trans. Affect. Comput., early access, 2019

[11] https://www.analyticsvidhya.com/blog/2021/03/introduction-to-batch-normalization

[12] https://machinelearningmastery.com/batch-normalization-for-training-of-deep-neural-networks/

[13] Luefeng Chen, Min Li, Xuzhi Lai, Kaoru Hirota, Witold Pedrycz "CNN-based Broad Learning with Efficient Incremental Reconstruction Model for Facial Emotion Recognition", IFAC PapersOnLine 53-2, 2020.