Emotion Recognition using Novel CNN Architecture

M.Anil kumar, K.Varun Krishna, D.Dedeepya, K.Sri Teja

K.L University Hyderabad, R.V.S Nagar, Moinabad-Chilkur Rd, near AP Police Academy, Aziznagar, Telangana 500075

Abstract-----This paper presents a novel approach to facial emotion recognition, a challenging task in computer vision. The study aims to enhance both accuracy and processing efficiency by employing a standalone neural network for the accurate classification of human emotions from facial expressions. The research utilizes transfer learning with EfficientNetB3 as the base model and the FER2013 dataset, comprising 28,709 training and 7,178 testing images. Due to significant class imbalances in the training data, which adversely affect model performance, oversampling techniques are employed. The proposed model demonstrates promising results, achieving an accuracy of 0.9330 and a loss of 0.3046. The study contributes to the ongoing efforts to improve emotion classification techniques in the field of computer vision.

Keywords*: Facial emotion recognition, FER2013, EfficientNetB3, Oversampling.*

1. Introduction

This paper introduces an Emotion-Based Music Recommender System, an innovative application of artificial intelligence (AI) that leverages machine learning algorithms to analyze user data and emotions, and recommends music accordingly. The system identifies the user's current emotional state by recognizing patterns in their listening history and social media activity, and subsequently suggests songs or playlists that align with the identified mood.

The system's key components include:

Data Collection: Gathering data from diverse sources such as the user's listening history, social media activity, and sensor data (e.g., heart rate, facial expressions). **Emotion Detection**: Utilizing machine learning algorithms to analyze the collected data and detect the user's emotional state. This can involve facial recognition technology or natural language processing (NLP) to analyze facial expressions and social media activity, respectively. **Music Recommendation**: Recommending music that matches the identified emotional state. This could involve analyzing the user's listening history to suggest similar songs or artists, or recommending songs or playlists known to be associated with the identified emotion. **Feedback Loop**: Refining recommendations over time by learning from the user's interactions, preferences, and emotional states.The proposed system offers several benefits, including helping users discover new music, enhancing the listening experience by matching music to the user's current emotional state, and improving user engagement and satisfaction with music streaming platforms by providing a personalized and intuitive experience. However, the system also faces challenges, primarily in accurately detecting the user's emotional state due to the complexity and subjectivity of human emotions. There is also a risk of perpetuating stereotypes or limiting the user's music choices by associating certain emotions with specific genres or artists. This paper aims to address these challenges and explore potential solutions.

In addition to the aforementioned components and benefits, the Emotion-Based Music Recommender System also has potential applications in various fields. For instance, it could be used in mental health therapy, where music is often used as a form of treatment. By accurately identifying a patient's

emotional state, the system could recommend music that could help in managing stress, anxiety, and other emotional issues.Moreover, the system could also be integrated into smart home systems to create an ambient environment that matches the user's mood, enhancing their overall living experience. For example, if the system detects that the user is feeling stressed, it could recommend calming music to help the user relax. However, the development of such a system also raises ethical and privacy concerns. The system requires access to personal data, such as social media activity and sensor data, to function effectively. Therefore, it is crucial to ensure that the system complies with data privacy laws and regulations, and that users are informed about the data the system collects and how it is used. Future research could focus on addressing these challenges and exploring ways to improve the accuracy of emotion detection, such as incorporating more sophisticated machine learning algorithms or using additional data sources for emotion detection. Furthermore, studies could also investigate the impact of such systems on user behavior and music consumption patterns. This could provide valuable insights into the effectiveness of emotion-based music recommendation systems and their potential influence on the music industry.

2. Related work

Emotion recognition using Convolutional Neural Networks (CNNs) has been a topic of interest in many previous studies. For instance, a study proposed a model that combines the bidirectional long short-term memory (BiLSTM)–Transformer and a 2D convolutional neural network (CNN) for emotion recognition from speech[1]. This model processes audio features to capture the sequence of speech patterns, while the 2D CNN handles Mel-Spectrograms to capture the spatial details of audio[1].

Another study focused on emotion detection from facial expressions using a CNN architecture[3]. Similarly, a real-time emotion recognition model based on CNN was proposed, which detects changes in people's emotions[4]. This model was developed using popular libraries such as TensorFlow and Keras[4].In another research, a novel CNN architecture was proposed to classify the driver's emotional state into seven fundamental emotions, including happiness, fear, surprise, anger, disgust, sadness, and neutrality[5]. This study highlights the potential of CNNs in emotion recognition, particularly in real-world applications like driving[5].These previous works demonstrate the versatility and effectiveness of CNNs in emotion recognition, providing a solid foundation for the development of your proposed novel CNN architecture for emotion recognition.

Another research introduced a novel technique called facial emotion recognition using convolutional neural networks (FERC). The FERC is based on a two-part convolutional neural network (CNN): The first part removes the background from the picture, and the second part concentrates on the facial feature vector extraction.

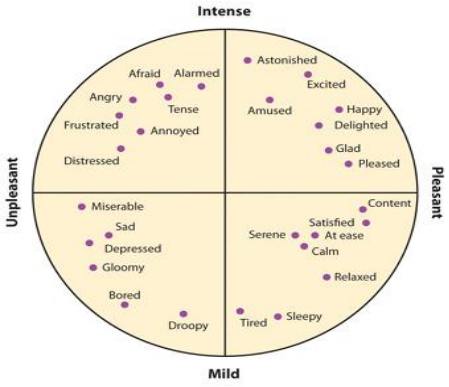

Fig 1: 2-D Valence-Arousal Space

Moreover, a paper described how to utilize a deep learning algorithm, the Convolutional Neural Networks (CNN) with InceptionV3 architecture, commonly used for Image Recognition, to classify seven different emotion classes (angry, afraid, disgusted, happy, neutral, sad, and surprised) with high accuracy4.

These studies highlight the versatility of CNNs in emotion recognition and provide a solid

foundation for the development of your proposed novel CNN architecture for emotion recognition. They also underscore the potential of such systems in various real-world applications, from enhancing user experiences in digital platforms to supporting mental health interventions. The potential of Convolutional Neural Networks (CNNs) in emotion recognition is vast, and their application extends beyond the realms of entertainment and mental health. For instance, in the field of education, emotion recognition systems could be used to detect student engagement and adapt learning materials accordingly. In the automotive industry, such systems could enhance safety by monitoring driver emotions and alerting them when they exhibit signs of stress or fatigue.

However, they also point to the challenges in this field, particularly in accurately detecting and interpreting complex and nuanced emotional states. Future research could focus on addressing these challenges and exploring ways to further improve the accuracy and efficiency of emotion recognition systems.

The paper by Isha Talegaonkar et al. presents the importance of Human-Computer Interaction (HCI) in a digital world and highlights the role of facial expressions in non-verbal communication. The proposed system aims to develop a Facial Expression Recognition (FER) system using Convolutional Neural Networks (CNN) to classify facial expressions in real-time. The system can be utilized for emotion analysis during activities like watching movie trailers or video lectures. The conclusion states that a CNN model was created and experimented with, achieving satisfactory train and test accuracies, which are, 0.7989 and 0.6012 respectively. The model is capable of real-time emotion classification using a webcam. Overall, the paper introduces a relevant and promising approach for FER using CNN, highlighting its potential applications and presenting positive results.[1] Facial expressions are a fundamental means of human communication, and deep learning methods in artificial intelligence have been employed to enhance human-computer interactions.

However, accurately recognizing and understanding facial expressions can be challenging.

This research by Lutfiah Zahara et al. proposes a real-time facial emotion prediction and classification system using the Convolutional Neural Network (CNN) algorithm and the OpenCV library, specifically TensorFlow and Keras, implemented on a Raspberry Pi. The system involves three main processes: face detection, facial feature extraction, and facial emotion classification. The experimental results using the FER-2013 dataset and CNN method achieved a facial expression prediction accuracy of 65.97%.[2] While many researchers focus on improving accuracy, the study by Gede Putra Kusuma, Jonathan, and

Andreas Pangestu Lim aims to enhance the processing efficiency of emotion classification by utilizing a single standalone neural network. The proposed approach involves a modified Convolutional Neural Network (CNN) based on the VGG-16 classification model, which was pre-trained on the ImageNet dataset and fine-tuned for emotion classification. The classification process is carried out on the FER2013 dataset, consisting of over 35,000 face images with in-the-wild settings, representing 7 distinct emotions, and divided into 80% training, 10% validation, and 10% testing data. The proposed approach achieves an accuracy of 69.40%, outperforming many standalone-based models.[3]

In this study, Yousif Khaireddin and Zhuofa Chen try to enhance the accuracy of FER2013 using CNN. They have achieved the highest classification accuracy using a single network on the FER2013 dataset. They have employed the VGGNet architecture, carefully fine-tuned its hyperparameters, and experimented with various optimization methods. Notably, their model achieves a state-of-the-art single-network accuracy of 73.28% on the FER2013 dataset without the need for additional training data.[4]

Benyoussef Abdellaoui et al. proposed a custom CNN model. They applied the Keras Tuner model optimizer for the FER2013 dataset. Keras Tuner is an open-source library within the TensorFlow ecosystem that provides an easy-to-use API for automating the process of hyperparameter tuning in Keras deep learning models, enabling users to find the optimal configuration for their models. They got a training accuracy of 0.8313 and a validation accuracy of 0.53.[5]

Despite the increasing interest in real-time facial emotion recognition for human-computer interaction, existing datasets in this field suffer from various issues such as unrelated photos, imbalanced class distributions, and misleading images that can adversely impact accurate classification. To address these problems, Abou Zafra et al. make use of new dataset called 3RL in this project, consisting of approximately 24,000 labeled images representing five basic emotions: happiness, fear, sadness, disgust, and anger. In comparison to other well-known datasets like FER and CK+, experiments conducted using commonly used algorithms like SVM and CNN demonstrate significant improvements in generalization on the 3RL dataset, achieving an accuracy of up to 91.4%, while results on FER2013 and CK+ datasets range from approximately 60% to 85%.[6] 8

The project by Mengyu Rao, Ruyi Bao, and Liangshun Dong is a comparative study between CK+ and FER2013 datasets. They used four different models for the comparison. These models are VGG19, ResNet18, ResNet50, and Xception. They also apply Hybrid Data Augmentation by applying horizontal flips and adding Gaussian noise. The final results show that the CK+ dataset gives better accuracies in all four models than FER2013. [7] The research by Ozioma Collins Oguine et al. proposes a Hybrid Architecture that combines the Haar Cascade Face Detection algorithm with a CNN Model. The CNN architecture processes input images of size 48x48x1 from the FER 2013 dataset. With the suggested modifications outlined in the paper, the proposed model achieved an average predictive accuracy of 70%. The weighted average accuracy on the test dataset was also 70%.[8]

3. Proposed System

3.1 Data Analysis and Visualization

Data analysis involves exploring and examining datasets to understand their structure, patterns, and relationships. It often includes tasks such as data cleaning, data manipulation, and statistical analysis. Python provides powerful libraries such as Pandas, NumPy, and SciPy that facilitate these tasks. Pandas, in particular, are widely used for data manipulation, transformation, and analysis, offering flexible data structures and data analysis tools. Visualization, on the other hand, involves representing data visually using graphs, charts, and plots. Python provides several libraries for data visualization, the most popular being Matplotlib and Seaborn. Matplotlib offers a wide range of customizable plots, while Seaborn provides higher-level functions for creating aesthetically pleasing statistical visualizations. Additionally, libraries like Plotly and Bokeh enable interactive and web-based visualizations. We have plotted Bar charts and Pie charts for visualizing and better understanding of the dataset we are going to work on.

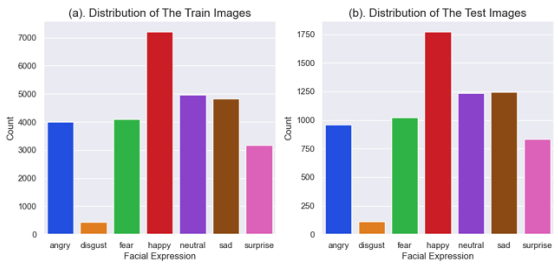

Fig2: Bar charts for Data Visualization

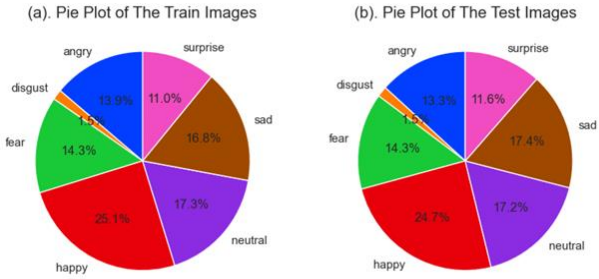

Fig3: Pie charts for Data Visualization

3.2 Data Pre-processing:

Data pre-processing is a crucial step in the data analysis pipeline. It involves preparing and transforming raw data into a format that is suitable for analysis and modeling. Data pre-processing helps in improving the quality and reliability of the data, removing inconsistencies, handling missing values, and ensuring that the data is in a consistent and usable form.

Oversampling the minority class helps to provide more training examples for the model to learn from and can improve the model's ability to accurately classify the minority class.

However, it is essential to be cautious with oversampling, as blindly applying it can lead to overfitting or artificially inflating the importance of the minority class. Careful evaluation and consideration of different oversampling techniques are necessary to ensure that the model benefits from the increased representation of the minority class without introducing biases or degrading the performance of the majority class.

This project oversamples the minority classes. The goal is to increase the representation of the minority class in the training dataset by creating additional synthetic samples. Overall, our code performs oversampling by duplicating or creating synthetic samples to increase the representation of the minority class in the training dataset. The data after oversampling:

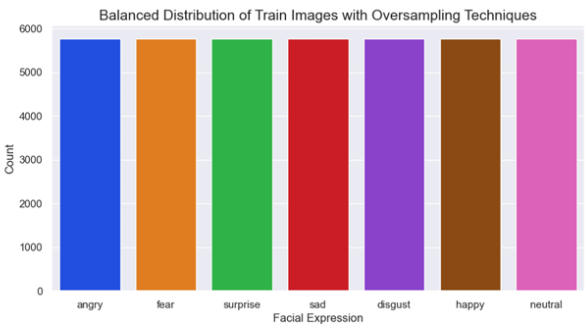

Fig4: Bar chart of Train Data after Oversampling

3.3 Model Development

We have applied transfer learning in this project, that is, we have taken a pre-trained model as the base model and upgraded the same. This project makes use of EfficientNetB3. EfficientNetB3 is a specific variant of the EfficientNet architecture. It refers to the third model in the EfficientNet series, which is known for its balance between model size and performance.

EfficientNetB3 is designed by scaling the base EfficientNet architecture using the compound scaling method. It has more parameters and is deeper compared to EfficientNetB0 and EfficientNetB1, but it is still more efficient than larger models like EfficientNetB4 or EfficientNetB7.

The scaling of EfficientNetB3 involves increasing the depth, width, and resolution of the network. The depth is increased by adding more layers, while the width is increased by widening the channels in each layer. The resolution is increased by using larger input image sizes. EfficientNetB3 has been pre-trained on a large-scale dataset such as ImageNet, allowing it to learn general features from a diverse range of images. It can be fine-tuned or used as a feature extractor for various computer vision tasks, including image classification, object detection, and image segmentation.

Compared to earlier versions of EfficientNet, EfficientNetB3 typically offers improved performance on image classification benchmarks while maintaining a relatively efficient model size and computational cost.
The model description is as below:

- We provide EfficientNetB3 as our base model.
- After the base model, we have added the BatchNormalizatoin layer which normalizes the activations of the previous layer. It helps in stabilizing and accelerating the training process.
- After this, we add a Dense layer.
- Now we apply Dropout. This layer randomly sets a fraction of the input units to 0 at each update during training to prevent overfitting.
- Finally, we add one more Dense layer which will be the model's Output layer.

3.4 Model Evaluation

We measure the Accuracy and the Loss of our model. Accuracy is a metric that measures the proportion of correctly classified samples out of the total number of samples. It provides an overall assessment of how well the model is performing in terms of correct predictions. Accuracy is often used in classification tasks where the goal is to assign the correct label or class to each input. Loss, also referred to as the loss function or objective function, quantifies the difference between the predicted outputs of the model and the actual ground truth labels. It represents the error or discrepancy between the predicted values and the true values.

3.5 Saving the model

We save this model as an HDF5 (.h5) file. These files are often used to save and load trained models. When training a deep learning model, the weights, architecture, and other necessary parameters of the model can be saved in a .h5 file format.

This allows you to save the model's state and use it later for prediction, fine-tuning, or sharing with others.

3.6 WebApp

The application we described consists of multiple modules that work together to create a web-based system for real-time face detection, track recommendation, and streaming video. Here's a combined explanation of each module's purpose:

i. Spotipy:

• Spotipy is a module used to establish a connection to Spotify and retrieve tracks using the Spotipy wrapper.

• It provides functionality to authenticate with Spotify, access user playlists, search for tracks, and retrieve track information.

ii. haarcascade:

• The haarcascade module is used for face detection.

• It includes pre-trained classifiers (haarcascade XML files) that can detect faces in images or video frames.

• These classifiers utilize Haar-like features and machine learning techniques to identify facial features.

iii. camera.py:

• The camera.py module is responsible for video streaming, frame capturing, prediction, and track recommendation.

• It utilizes the webcam or camera input to capture video frames.

• It performs real-time face detection using the haarcascade module.

• For each detected face, it makes predictions and recommends tracks based on the detected emotion or facial expression.

iv. main.py:

• main.py is the main Flask application file.

• It defines routes and handles HTTP requests from the web page.

• It interacts with the camera.py module to initiate video streaming, capture frames,

and receive predictions and track recommendations.

• It sends the processed data to the web page for display and interaction.

v. index.html:

• index.html is an HTML file located in the 'templates' directory.

• It serves as the web page for the application.

• It provides the user interface and displays the video stream, detected faces, predicted emotions, and recommended tracks.

• It includes basic HTML and CSS code for structuring and styling the web page.

vi. utils.py:

• utils.py is a utility module used for video streaming from the web camera.

• It employs threads to enable real-time video capture and processing.

• It assists in achieving smooth and responsive streaming by utilizing concurrent

execution.

Together, these modules work in tandem to create a web-based application that streams video from a webcam, detects faces in real-time, predicts emotions or facial expressions, and recommends tracks based on the detected emotion. Users can interact with the application through the web page and view the video stream along with the corresponding predictions and recommendations.

4. Flowchart:

The proposed methodology involves data analysis and visualization using Python. The data analysis part includes exploring and examining datasets to understand their patterns and relationships, facilitated by Python libraries like Pandas, NumPy, and SciPy. Visualization involves creating graphical representations of the data using libraries like Matplotlib and Seaborn for static plots, and Plotly and Bokeh for interactive visualizations. The dataset is further understood by plotting Bar and Pie charts.

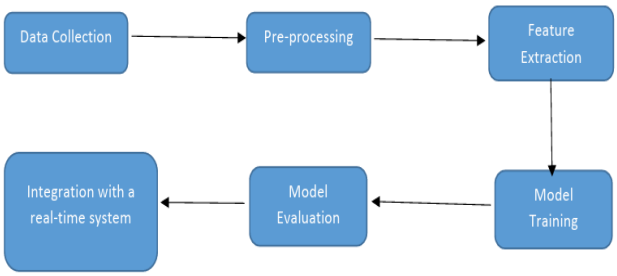

Fig5: architecture of proposed model

5. Datasets

The dataset comprises a collection of images depicting faces, with each image consisting of grayscale pixels arranged in a 48x48 grid. These images have undergone an automated registration process, ensuring that the face within each image is approximately centered and occupies a consistent amount of space. The main objective of this task is to assign an emotion category to each face based on the displayed facial expression.

There are seven distinct emotion categories used for classification: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. The purpose is to develop a model that can accurately predict the emotion category for any given face image.

Fig 6: Collection of images

To facilitate model development and evaluation, the dataset has been divided into two sets: a training set and a public test set. The training set comprises 28,709 examples, which will be used to train and fine-tune the emotion classification model. The public test set consists of 3,589 examples, which will be used to assess the performance of the trained model and determine its ability to generalize to unseen data.

By analyzing and learning from this dataset, the goal is to create a model capable of accurately categorizing facial expressions into one of the seven specified emotions, thereby enabling the recognition and understanding of emotions from facial images.

6. Results

In this project, we used transfer learning by taking EfficientNetB3 as the base model. After training the model we calculate the Accuracy and Loss of our model.

Accuracy	0.9330
Loss	0.3046

Table 1: Accuracy and Loss values proposed model

We have plotted the Accuracy and Loss graphs for our model:

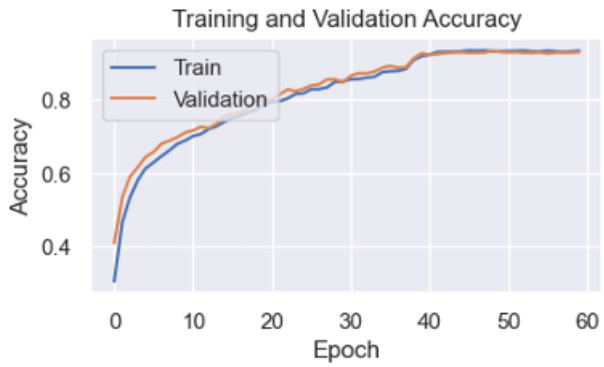

Fig 7: Accuracy

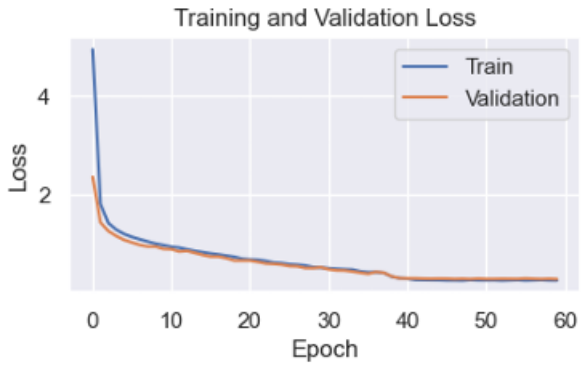

Fig 8: Loss

We have also plotted Confusion matrix as shown in the below validation table .

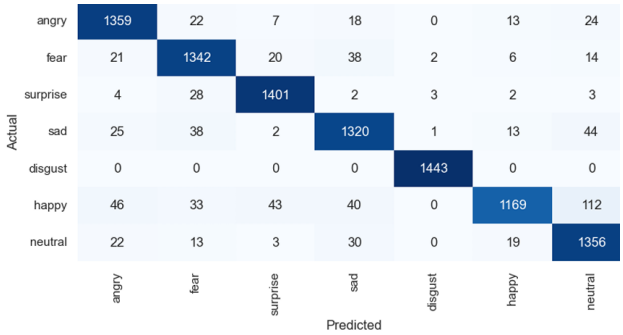

Fig 9: Confusion Matrix o Validation Data

7. Conclusion

This paper presented The Emotion-Based Music Recommender uses real-time face recognition and Spotipy integration to offer personalized music recommendations. By analyzing users' facial expressions, it detects their emotions and suggests music that aligns with their mood. The web app interface provides easy access to this system. This technology revolutionizes music engagement by adding emotional intelligence to recommendations, offering a unique music discovery experience tailored to each individual's emotions. In addition to personalization, this system introduces a new dimension of emotional intelligence to music discovery. It allows users to explore music in a way that is deeply connected to their emotional state. Furthermore, it opens up new avenues for users to discover and connect with a variety of genres, artists, and tracks they might not have found otherwise. This technology truly redefines the music listening experience.

REFERENCES

[1] Talegaonkar, Isha and Joshi, Kalyani and Valunj, Shreya and Kohok, Rucha and Kulkarni, Anagha, Real Time Facial Expression Recognition using Deep Learning (May 18, 2019). Proceedings of International Conference on Communication and Information Processing (ICCIP) 2019,

[2] L. Zahara, P. Musa, E. Prasetyo Wibowo, I. Karim and S. Bahri Musa, "The Facial Emotion Recognition (FER-2013) Dataset for Prediction System of Micro-Expressions Face Using the Convolutional Neural Network (CNN) Algorithm based Raspberry Pi," 2020 Fifth International Conference on Informatics and Computing (ICIC), Gorontalo, Indonesia, 2020, pp. 1-9

[3] Kusuma Negara, I Gede Putra & Jonathan, Jonathan & Lim, Andreas. (2020). Emotion Recognition

on FER-2013 Face Images Using Fine-Tuned VGG-16. Advances in Science, Technology and

Engineering Systems Journal. 5. 315-322. 10.25046/aj050638.

[4] Yousif Khaireddin and Zhoufa Chen. (2021). Facial Emotion Recognition: State of the Art Performance on FER2013.

[5] Abdellaoui, Benyoussef & Moumen, Aniss & Idrissi, Younes & Remaida, Ahmed. (2021). Training the Fer2013 Dataset with Keras Tuner. 409-412.

[6] Rahmeh Abou Zafra, Lana Ahmad Abdullah, Rouaa Alaraj, Rasha Albezreh, Tarek Barhoum, Khloud Al Jallad. (2022). An experimental study in Real-time Facial Emotion Recognition on new 3RL dataset.

[7] by Mengyu Rao, Ruyi Bao, and Liangshun Dong. (2022). Face Emotion Recognization Using Dataset Augmentation Based on Neural Network.

[8] Ozioma Collins Oguine, Kanyifeechukwu Jane Oguine, Hashim Ibrahim Bisallah, Daniel Ofuani.

(2022). Hybrid Facial Expression Recognition (FER2013) Model for Real-Time Emotion Classification and Prediction.

[9] Sera Kim 1 andSeok-Pil Lee 2,*:A BiLSTM–Transformer and 2D CNN Architecture for Emotion Recognition from Speech 25 September 2023

[10] Mehendale, N. Facial emotion recognition using convolutional neural networks (FERC). SN Appl. Sci. 2, 446 (2020).

[11]J. Timothy, R. Prasetio, D. Tandi, H. Lucky and I. A. Iswanto, "Facial Emotion Recognition with InceptionV3 CNN Architecture," 2023 International Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS), Jakarta Selatan, Indonesia, 2023

[12]Bautista, John Lorenzo, Yun Kyung Lee, and Hyun Soon Shin. 2022. "Speech Emotion Recognition Based on Parallel CNN-Attention Networks with Multi-Fold Data Augmentation" Electronics 11, no. 23: 3935.

[13]Mehmet Akif ÖzdemirMehmet Akif Özdemir :Real Time Emotion Recognition from Facial Expressions Using CNN Architecture Profile image of Reza SadighzadehReza SadighzadehProfile Mehmet Akif ÖzdemirMehmet Akif Özdemir

[14] Jaiswal, S., Nandi, G.C. Robust real-time emotion detection system using CNN architecture. Neural Comput & Applic 32, 11253–11262 (2020).

[15]D. Sahana, K. S. Varsha, Snigdha Sen, R. Priyanka Soft Computing: Theories and Applications, 2023, Volume 627

[16]Kishan K. Nayak; Prabhudev; Rohan; Venkatesh; Raghavendra Nayak Continuous emotion recognition from facial expressions using CNN architecture Volume 2742, Issue 1 13 February 2024

[17]Iyer, A., Das, S.S., Teotia, R. et al. CNN and LSTM based ensemble learning for human emotion recognition using EEG recordings. Multimed Tools Appl 82, 4883–4896 (2023).

[18]Nazmin Begum*1, Dr. Md Shoaibuddin Madni2 and Dr. Ismath Unnisa3 REAL TIME FACE AND EMOTION RECOGNITION USING CNN wjert, 2023, Vol. 9, Issue 3, 60-75.