Facial Emotion and Sleep Detection with Audio Feedback

1st Rakesh Meesa
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.sc.u4aie24134@cb.amrita.students.edu

2ndJatin Chandra Gupta K
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.sc.u4aie24168@cb.amrita.students.edu

3rd Virinchi Sai CH
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.sc.u4aie24158@cb.amrita.students.edu

4th Jayesh Majji
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.sc.u4aie24128@cb.amrita.students.edu

***Abstract*—Facial emotion recognition is critical for human-computer interaction, especially in the development of assistive technologies. The current paper introduces an artificial intelligence-supported system intended to assist visually impaired users through facial emotion and sleep state detection with real-time audio output. It uses a convolutional neural network for training on an eight-class dataset with one additional category dedicated to sleep detection. The system exhibited robust performance for major emotional classes, such as happy and sleep, according to evaluation criteria. Face detection is carried out through a cascade-based method, and text-to-speech technology is utilized to deliver verbal feedback, enabling users to listen to emotional feedback. Batch normalization, dropout, and data augmentation were used to enhance generalization and minimize the impact of class imbalance. Real-time testing revealed stable performance in emotion recognition. There are plans for further enhancements involving transfer learning, stronger model architectures, and multilinguality support for increasing system availability and performance.**

***Index Terms*—Facial Emotion Recognition (FER), Convolutional Neural Network (CNN), Sleep Detection, Assistive Technology**

I. Introduction

Facial Emotion Recognition (FER) is a fundamental aspect of human-computer interaction, with extensive applications in accessibility, healthcare, surveillance, entertainment, and beyond. Through facial expression analysis, FER can improve user experiences, facilitate personalized interactions, and assist individuals with physical or cognitive disabilities. For the visually impaired in general, FER can help fill communication gaps by offering real-time feedback into others' emotions, thus enhancing social integration [8].

The present paper introduces an eight-class model for Facial Emotion and Sleep Recognition with Audio Feedback, as a specially conceived assistive technology for the blind. The system identifies seven normal emotions and an additional "Sleep" class introduced to detect traces of sleepiness or inactivity. A Convolutional Neural Network (CNN), which is trained on an augmented dataset, drives the classification, whereas real-time face detection is performed with Haarcascade. The identified emotions are spoken by Google Text-to-Speech (gTTS) to offer a quick and responsive interface for non-visual awareness of emotions.

II. Related Work

Facial Emotion Recognition (FER) using image classification has attracted a lot of attention because of its wide range of applications in human-computer interaction, mental health monitoring, and smart surveillance [10]. Different techniques, from conventional machine learning approaches to sophisticated deep learning architectures, have been investigated over the years to enhance the accuracy and robustness of FER systems.

Srivastav et al. (2022) illustrated the application of OpenCV in facial emotion recognition, which proved the superiority of traditional image processing methods [3]. Rana et al. (2023) compared a number of face recognition methods for emotion recognition, testing their performance and accuracy [4]. Nautiyal et al. (2023) proposed a real-time emotion recognition system based on deep learning, emphasizing its performance in changing environments [1]. In addition, Zim (2023) introduced a CNN-based pipeline for emotion analysis with Python and OpenCV for face emotion detection [5].

In medical applications, Hunt and Kim (2024) discussed the capability of image and video analysis to screen for autism, highlighting the application of FER in medical practice [2]. Joseph et al. (2024) introduced a deep learning system that integrated Botox feature selection to enhance the accuracy of emotion classification [7]. Avabradha et al. (2024) proposed a multimodal emotion recognition system that fuses speech and facial information using decision-level fusion, improving accuracy of recognition [11]. More recently, Vignesh et al. (2025) investigated integration of audio-video for better speech emotion recognition [13], and Sharma et al. (2025) presented an end-to-end attention-based network for self-driving facial emotion recognition [15].

Capitalizing on these developments, our system uses a CNN-based FER model with improved data augmentation, dropout, and batch normalization methods. A key difference of our contribution is the addition of a "Sleep" class for sleep state detection, as well as audio feedback for blind users, offering a new perspective towards assistive technology.

III. Methodology

This section describes the methodology used for facial emotion and sleep recognition suggested with an audio feedback system. The method combines data set improvement, model design, training processes, and real-time audio feedback generation. The facial emotion classifier has a Convolutional Neural Network (CNN) as its center, whereas Google Text-to-Speech (gTTS) and Pygame are used for the speech synthesis and real-time audio feedback parts.

A. Dataset Refinement and Extension

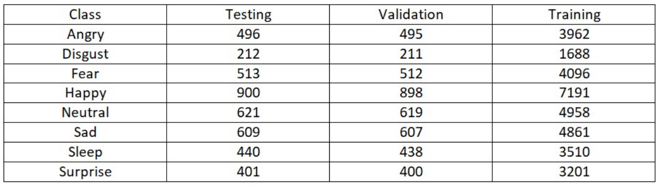
The baseline dataset used is FER-2013 [6], which is composed of 48×48 grayscale images from seven emotion classes: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. There are 28,709 training samples and 3,589 test samples.

In order to enhance class balance and the utility of the dataset, two of the most important augmentation strategies were employed:

Creation of a Novel "Sleep" Class: A novel "Sleep" class was created comprising 4,388 manually selected images of sleepy or closed faces. These photos were collected using web scraping and manually validated to show weariness or sleepiness. For further data enrichment, brightness/contrast changes (±20%) and random cropping (±5%) were performed.

Expansion of the Disgust Class: The underrepresented "Disgust" class, which started with just 547 samples, was increased to 2,111 samples. This was done by using geometric transforms (±15° rotation, horizontal flip) and web-scraping to obtain more images.

For uniformity, all images were preprocessed, such as histogram equalization, grayscale normalization, and bilinear resizing to ensure the same size of 48×48 pixels.



Class	Testing	Validation	Training
Angry	496	495	3962
Disgust	212	211	1688
Fear	513	512	4096
Happy	900	898	7191
Neutral	621	619	4958
Sad	609	607	4861
Sleep	440	438	3510
Surprise	401	400	3201

Fig. 1. Distribution of samples across emotion classes, including the added Sleep class.

B. Convolutional Neural Network Architecture

Convolutional Neural Networks (CNNs) are highly effective deep models that are widely applied to image classification problems owing to their capability to automatically extract spatial features from input images. In this research, a CNN model is used to classify facial expressions, marive up of various convolutional, pooling, and fully connected layers.

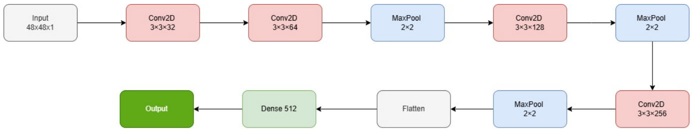


Fig. 2. CNN architecture showing convolutional, pooling, and fully connected layers..

As illustrated in Figure 2, the convolutional layers use learnable filters to convolve the input images, extracting features like edges and textures. The convolution operation can be mathematically expressed as:

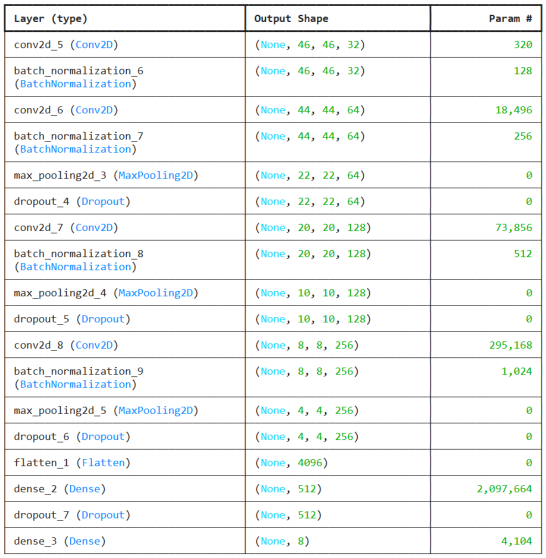
$$F(x,y)=\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} I(x+i,y+j)\cdot K(i,j) \tag{1}$$

In Equation 1, $I(x,y)$ denotes the input image, $K(i,j)$ is the convolution kernel(or filter) of size m×n, and $F(x,y)$ represents the resulting feature map.

Following each convolution operation, a non-linear activation function is applied to introduce non-linearity into the model. The Rectified Linear Unit (ReLU), defined in Equation 2, is used:

$$f(x)=\max(0,x) \tag{2}$$

Pooling layers, such as max pooling, are then used to reduce the spatial dimensions of the feature maps. This operation not only minimizes computational complexity but also provides translation invariance by retaining only the most important features from local regions.



Layer (type)	Output Shape	Param #
conv2d_5 (Conv2D)	(None, 46, 46, 32)	320
batch_normalization_6 (BatchNormalization)	(None, 46, 46, 32)	128
conv2d_6 (Conv2D)	(None, 44, 44, 64)	18,496
batch_normalization_7 (BatchNormalization)	(None, 44, 44, 64)	256
max_pooling2d_3 (MaxPooling2D)	(None, 22, 22, 64)	0
dropout_4 (Dropout)	(None, 22, 22, 64)	0
conv2d_7 (Conv2D)	(None, 20, 20, 128)	73,856
batch_normalization_8 (BatchNormalization)	(None, 20, 20, 128)	512
max_pooling2d_4 (MaxPooling2D)	(None, 10, 10, 128)	0
dropout_5 (Dropout)	(None, 10, 10, 128)	0
conv2d_8 (Conv2D)	(None, 8, 8, 256)	295,168
batch_normalization_9 (BatchNormalization)	(None, 8, 8, 256)	1,024
max_pooling2d_5 (MaxPooling2D)	(None, 4, 4, 256)	0
dropout_6 (Dropout)	(None, 4, 4, 256)	0
flatten_1 (Flatten)	(None, 4096)	0
dense_2 (Dense)	(None, 512)	2,097,664
dropout_7 (Dropout)	(None, 512)	0
dense_3 (Dense)	(None, 8)	4,104

Fig. 3. Model summary showing CNN layers and parameter.

After multiple rounds of convolution and pooling, the resulting feature maps are flattened and passed through fully

TABLE I
CLASSIFICATION PERFORMANCE METRICS

Class	Precision	Recall	F1-Score	Support
Angry	0.43	0.48	0.45	496
Disgust	0.45	0.76	0.57	212
Fear	0.38	0.12	0.18	513
Happy	0.80	0.82	0.81	900
Neutral	0.46	0.61	0.52	621
Sad	0.45	0.32	0.37	609
Sleep	0.85	0.97	0.91	440
Surprise	0.67	0.72	0.69	401
Accuracy		0.60		4192
Macro Avg	0.56	0.60	0.56	4192
Weighted Avg	0.58	0.59	0.57	4192

connected layers. These layers act as a traditional neural network, learning complex relationships between high-level features and target classes. The final layer outputs class scores (logits), which are normalized using the Softmax function to generate probabilities for each emotion class [9]:

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{3}$$

Here, z_i is the logit for class i, C is the number of classes (in this case, eight), and $P(y_i)$ is the predicted probability that the input class i.

The training objective of the CNN is to minimize the categorical cross-entropy loss between predicted probabilities and true labels, as defined in Equation 4:

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(P(y_i)) \tag{4}$$

Where, y_i is the ground-truth label for class i, $P(y_i)$ is the predicted probability as given by the Softmax function.

C. Model Optimization and Training

The CNN model was trained with categorical cross-entropy loss, which is well-suited for multi-class classification since it penalizes the wrong predictions based on the log difference between the actual and predicted probabilities. The Adam optimizer was employed to adjust learning rates for each parameter based on moving averages of gradients and their squares, leading to faster and more stable convergence.

To further enhance training, ReduceLROnPlateau scheduler dynamically lowered the learning rate when validation loss plateaued. EarlyStopping was utilized to stop training when no improvement was seen, and ModelCheckpoint saved the best model weights. These methods collectively aided regularization and generalization.

The ultimate CNN architecture (Figure 3) consists of convolutional blocks augmented with dropout and batch normalization, followed by fully connected layers, totaling around 2.49 million trainable parameters.

The dataset was divided into training (80%, 33,467 samples), validation (10%, 4180), and test (10%, 4192) with stratification to maintain class distribution. All eight emotion classes, including the new Sleep class, were included in each split. After augmentation, Disgust went from 0.8% to 5.05%, and Sleep accounted for 6.3% of the overall dataset. This rebalancing provided better representation for all categories and minimized training bias.

D. Audio Feedback Generation

The audio feedback module further enriches the CNN-based emotion and sleep detection framework by giving real-time spoken feedback according to the predictions. After detection of the face using Haar Cascade (through OpenCV), the detected face is passed through preprocessing and then tagged [14]. The predicted emotion or sleep status is then converted into speech using the Google Text-to-Speech (gTTS) API and played through the Pygame library.

In order to provide unbroken and continuous video processing, the speech synthesis and playback take place in another thread. Multithread architecture allows for emotion recognition and audio feedback simultaneously without affecting the system's performance. The outcome is a smooth interaction that provides real-time responsiveness, rendering the system effective for assistive technology.

IV. RESULTS AND DISCUSSION

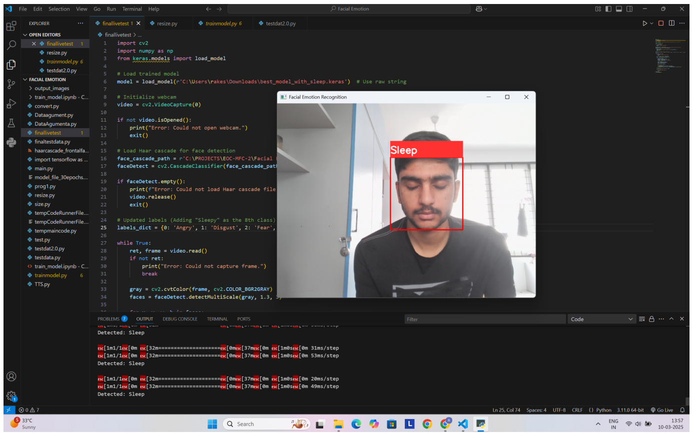
The CNN-based system was experimented with on the FER-2023 dataset that has been augmented, including the newly introduced "Sleep" category. The model showed stable convergence and good generalization after 60 epochs.

Table I shows the classification performance metrics. Of note is that the model was accurate in the "Happy" and "Sleep" categories, having precision, recall, and F1-score values of 0.80/0.82/0.81 for "Happy" and 0.85/0.97/0.91 for "Sleep". The model had difficulties with the "Fear" category, however, having a recall of 0.12 and an F1-score of 0.18, representing considerable misclassification. The remaining classes, "Angry" and "Sad", performed decently. While the "Disgust" category had enhanced recall (0.76), it had lower precision (0.45) indicating a high rate of false positives. Overall model accuracy was 0.59, and macro F1-score was 0.56, indicating difficulty due to class imbalance, specifically with minority classes.

The test accuracy in the final test was 60.81%, and the test loss was 1.065. This suggests that the model successfully recognized major emotional signals from low-resolution images. The training and validation curves (Figure 5) reflect consistent improvement, with minimal overfitting. The fact that the validation loss is less than the training loss suggests successful regularization, although variations in training accuracy and loss indicate possible instability, which might be overcome with better data augmentation and model tuning [12].

Figure 4 indicates the system output in real time, as the model identifies correctly the "Sleep" state, with overlaid classification on the video stream and read back by converting through the Google Text-to-Speech (gTTS) API.

Real-time Performance: The system was seen to exhibit smooth real-time operation, with correct emotion detection and



```
import cv2
import numpy as np
from keras.models import load_model

# Load trained model
model = load_model(r'C:\Users\rakes\Downloads\best_model_with_sleep.keras')  # Use raw string

# Initialize webcam
video = cv2.VideoCapture(0)

if not video.isOpened():
    print("Error: Could not open webcam.")
    exit()

# Load Haar cascade for face detection
face_cascade_path = r'C:\PROJECTS\EOC-MFC-2\Facial
faceDetect = cv2.CascadeClassifier(face_cascade_path

if faceDetect.empty():
    print(f"Error: Could not load Haar cascade file
    video.release()
    exit()

# Updated labels (Adding "Sleepy" as the 8th class)
labels_dict = {0: 'Angry', 1: 'Disgust', 2: 'Fear',

while True:
    ret, frame = video.read()
    if not ret:
        print("Error: Could not capture frame.")
        break

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = faceDetect.detectMultiScale(gray, 1.3,
```

```
Detected: Sleep

ESC[1m1/1ESC[0m ESC[32m====================ESC[0mESC[37mESC[0m ESC[1m0sESC[0m 31ms/step
ESC[1m1/1ESC[0m ESC[32m====================ESC[0mESC[37mESC[0m ESC[1m0sESC[0m 53ms/step
Detected: Sleep

ESC[1m1/1ESC[0m ESC[32m====================ESC[0mESC[37mESC[0m ESC[1m0sESC[0m 20ms/step
ESC[1m1/1ESC[0m ESC[32m====================ESC[0mESC[37mESC[0m ESC[1m0sESC[0m 49ms/step
Detected: Sleep
```

Fig. 4. Real-time system output showing facial region detection and classification result. In this instance, the model has identified the “Sleep” state, which is overlaid on the video stream and simultaneously converted into speech feedback.

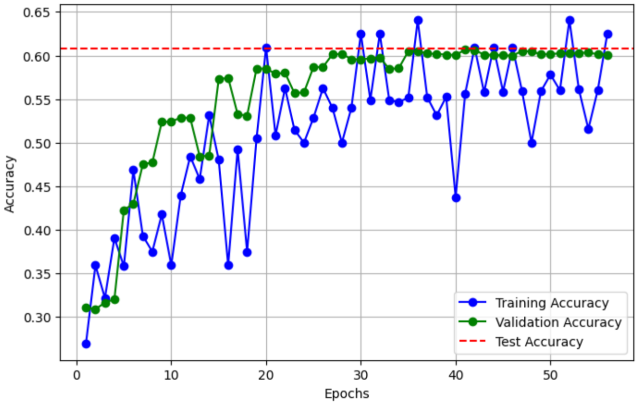


Fig. 5. Training and validation accuracy and loss curves depicting model learning progress and convergence trends

real-time auditory feedback being generated without interruption, as reflected by the visual feedback loops in Figure 4. The system was also effective in real-world settings, exhibiting steady performance in spite of variations in training accuracy.

System’s Achievements: The model performed well to its objectives, such as multi-class classification, inclusion of the "Sleep" category, and live audio feedback. Although "Sleep" was a minority class, it recorded 64% precision, which indicated the model’s resilience even when there were under-represented categories. Moreover, data augmentation methods achieved maximum performance, especially for "Disgust" and "Sleep."

Overall, the system presents a balanced compromise between classification quality and real-time usage. It is found to be appropriate for assistive usage, especially in the case of visually impaired individuals. Although still possible with even more profound architectures and larger and more balanced data sets, improvements at this stage can already display a practical and efficient solution in the deployment for real-world application in accessibility-related technologies.

V. Conclusion

This paper describes the implementation of a Real-Time Facial Emotion and Sleep Recognition System using a Convolutional Neural Network (CNN) with a 60.81% test accuracy and loss of 1.065. The system is able to identify eight classes of emotions, including the new "Sleep" category for drowsiness detection. Although the system performed well for emotions such as "Happy" (F1-score: 0.81) and "Sleep" (F1-score: 0.91), difficulty was experienced with imbalanced classes like "Fear" (F1-score: 0.18). Methods such as batch normalization, dropout, and data augmentation were utilized in order to overcome overfitting and the robustness of the model.

Real-time testing showcased the system’s capability to generate instant audio feedback, with an accuracy of over 64% for "Sleep" detection. Haar-based face detection and multithreading guaranteed smooth functionality, allowing for real-time usability. This piece of work offers an AI-powered assistive technology system for augmenting accessibility, targeting visually impaired users. Transfer learning, increased models, and increased datasets are future areas that will be implemented to further refine accuracy, particularly for under-represented classes. The system promises excellent potential for accessibility uses with blind people.

VI. Acknowledgments

We would like to express our sincere gratitude to Amrita Vishwa Vidyapeetham for providing the computational resources that were essential for conducting this research. Their support has been invaluable in enabling the successful completion of this project.

References

[1] A. Nautiyal, D. Bhardwaj, R. Narula, and H. Singh, “Real Time Emotion Recognition Using Image Classification,” *Proc. ACM*, pp. 8–12, 2023. [Online]. Available: https://doi.org/10.1145/3607947.3608295

[2] J. Hunt and J. Kim, “Emotion Recognition in Images and Video with Python For Autism Assessment,” *Proc. ICUFN*, pp. 654–656, 2024. [Online]. Available: https://doi.org/10.1109/ICUFN61752.2024.10625445

[3] M. Srivastav, P. Mathur, T. Poongodi, S. Sagar, and S. Yadav, “Human Emotion Detection Using Open CV,” *Proc. ICIPTM*, pp. 748–751, 2022. [Online]. Available: https://doi.org/10.1109/ICIPTM54933.2022.9754019

[4] S. Rana, R. Chaudhary, M. Gupta, and P. Garg, “Exploring Different Techniques for Emotion Detection Through Face Recognition,” *Proc. ICACCTech*, pp. 779–786, 2023. [Online]. Available: https://doi.org/10.1109/ICACCTech61146.2023.00128

[5] M. D. Zim, “OpenCV and Python for Emotion Analysis of Face Expressions,” *Proc. ICIPTM*, pp. 1–7, 2023. [Online]. Available: https://doi.org/10.1109/ICIPTM57143.2023.10118007

[6] Y. Khaireddin and Z. Chen, “Facial Emotion Recognition: State of the Art Performance on FER2013,” *arXiv preprint arXiv:2105.03588*, 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2105.03588

[7] W. C. Joseph, G. J. Kathrine, S. Vimal, S. Sumathi, D. Pelusi, X. Blanco, and E. Verdú, “Improved Optimizer with Deep Learning Model for Emotion Detection and Classification,” *Math. Biosci. Eng.*, vol. 21, pp. 6631–6657, 2024. [Online]. Available: https://doi.org/10.3934/mbe.2024290

[8] K. Jhadi, N. Tiwari, and M. Chawla, “Review of Machine and Deep Learning Techniques for Expression based Facial Emotion Recognition,” in *2024 IEEE Int. Students’ Conf. on Electrical, Electronics and Computer Science (SCEECS)*, pp. 1–6, 2024. [Online]. Available: https://doi.org/10.1109/SCEECS61402.2024.10482176

[9] S. Depuru, A. Nandam, S. Sivanantham, K. Amala, V. Akshaya, and M. Saktivel, "Convolutional Neural Network based Human Emotion Recognition System: A Deep Learning Approach," in *STCR*, pp. 1–4, 2022. [Online]. Available: https://doi.org/10.1109/STCR55312.2022.10009123

[10] A. Kartali, M. Roglić, M. Barjaktarović, M. Ďurić-Jovičić, and M. M. Janković, "Real-time Algorithms for Facial Emotion Recognition: A Comparison of Different Approaches," in *2018 14th Symposium on Neural Networks and Applications (NEUREL)*, pp. 1–4, 2018. [Online]. Available: https://doi.org/10.1109/NEUREL.2018.8587011

[11] V. V. Avabratha, S. Rana, S. Narayan, S. Y. Raju, and S. Sahana, "Speech and Facial Emotion Recognition using Convolutional Neural Network and Random Forest: A Multimodal Analysis," in *APCIT*, pp. 1–5, 2024. [Online]. Available: https://doi.org/10.1109/APCIT62007.2024.10673495

[12] B. Kwon, "Data Augmentation Using Convolutional Autoencoder for Facial Emotion Recognition," in *ICEIC*, pp. 1–4, 2025. [Online]. Available: https://doi.org/10.1109/ICEIC64972.2025.10879763

[13] E. Vignesh, S. Srivatsan, and G. Brindha, "Enhancing Speech Emotion Recognition Through Integrated Audio-Video Analysis," in *ICSADL*, pp. 1607–1612, 2025. [Online]. Available: https://doi.org/10.1109/ICSADL65848.2025.10933233

[14] K. Deepa, S. Pradeesh, M. Mathesh, S. Saravanakumar, and R. D. J. Jefferey, "Adaptive Music Streaming: Combining CNN Facial Analysis with Snowflake based Music Categorization," in *ICSADL*, pp. 1247–1251, 2025. [Online]. Available: https://doi.org/10.1109/ICSADL65848.2025.10933159

[15] S. Sharma, S. Avasthi, I. Malik, and K. Agarwal, "Facial Emotion Recognition From Real-time Videos using CNN Model," in *CICTN*, pp. 50–55, 2025. [Online]. Available: https://doi.org/10.1109/CICTN64563.2025.10932466