Emotion Recognition Model Based on Visual Cues and Explainable AI Using Facial Expression Video

18ECP107L- MINOR PROJECT

A PROJECT REPORT

Submitted by

Harish S –RA2111004010419
Harikumaran B –RA2111004010416

Under the guidance of

Dr S Giriprasad

(Assistant Professor, Department of Electronics and Communication Engineering)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

SRM NAGAR, KATTANKULATHUR-603203,

CHENGALPATTU DISTRICT

NOVEMBER 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled **Emotion Recognition Model Based on Visual Cues and Explainable AI Using Facial Expression Video** is the bonafide work of **Harish S [Reg No: RA2111004010419], Harikumaran B [Reg No: RA2111004010416]** who carried out the 18ECP107L-Minor Project work under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr S Giriprasad
GUIDE
Assistant Professor
Dept. of Electronics and Communication Engineering

SIGNATURE

Dr Diwakar R. Marur
PROJECT COORDINATOR
Dept. of Electronics and Communication Engineering

SIGNATURE
PROF.IN.CHARGE/ACADEMIC ADVISOR
Dept. of Electronics and Communication Engineering

ABSTRACT

Emotion recognition plays a crucial role in human-computer interaction, allowing machines to better understand and respond to human emotional states, thus enhancing user experience and engagement. This project aims to develop a sophisticated emotion recognition model using Convolutional Neural Networks (CNNs) within TensorFlow, specifically tailored to process facial images and classify various human emotions accurately. To promote transparency and foster trust in the decision-making process, Explainable AI (XAI) techniques are incorporated, providing detailed insights into how the model arrives at specific predictions. This interpretability is essential, particularly for applications that rely on user trust and understanding. To further enhance the model's real-time functionality and adaptability, the YOLOv10 algorithm is integrated. This addition significantly improves the model's ability to detect and localize faces within dynamic or cluttered environments, ensuring that the system remains effective even when multiple faces or distractions are present. By combining the powerful image classification capabilities of CNNs with the interpretability offered by XAI and the robust detection features of YOLOv10, this system achieves a balance between accuracy and explainability. Designed with versatility in mind, this adaptable model can be deployed across various applications, including emotional monitoring systems, customer experience platforms, educational tools, and mental health assessments, offering valuable insights into human emotions in diverse settings.

ACKNOWLEDGEMENT

We would like to express our deepest gratitude to the entire management of SRM Institute of Science and Technology for providing me with the necessary facilities for the completion of this project.

I wish to express my deep sense of gratitude and sincere thanks to our Professor and Head of the Department Dr. Sangeetha M, for her encouragement, timely help, and advice offered to me.

I am very grateful to my guide **Dr S Giriprasad** Assistant Professor, Department of Electronics and Communication Engineering, who has guided me with inspiring dedication, untiring efforts, and tremendous enthusiasm in making this project successful and presentable.

I would like to express my sincere thanks to the project coordinator **Dr Diwakar R. Marur** for his time and suggestions for the implementation of this project.

I also extend my gratitude and heartful thanks to all the teaching and non-teaching staff of the Electronics and Communications Engineering Department and to my parents and friends, who extended their kind cooperation using valuable suggestions and timely help during this project work.

Harish S

Harikumaran B

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

CNN	Convolutional Neural Network
CPU	Central Processing Unit
FER	Facial Emotion Recognition
GPU	Graphics Processing Unit
Grad CAM	Gradient-Weighted Class Activation Mapping
XAI	Explainable AI
YOLOv10	You Only Look Once

CHAPTER 1

INTRODUCTION

Emotion recognition is vital for enhancing human-computer interaction in our digital world, particularly through Facial Emotion Recognition (FER), which uses facial expressions to interpret emotions like happiness, anger, sadness, surprise, fear, and disgust. FER finds applications in customer service, healthcare, surveillance, and education, enabling machines to better understand and respond to human emotional states.Despite advances, FER systems still face two significant challenges: achieving real-time performance and providing transparency in decision-making. Many models function as "black boxes," offering little insight into how they generate predictions. This opacity can lead to distrust, especially in sensitive fields like mental health, where accurate emotional interpretation is essential. Users and stakeholders need a clearer understanding of these systems to trust their outcomes, making Explainable AI (XAI) crucial.

XAI aims to clarify the internal workings of machine learning models through techniques like feature visualization, decision attribution, and model-agnostic methods. For example, Grad-CAM (Gradient-weighted Class Activation Mapping) highlights areas of an input image that most influence predictions, providing decision attribution and fostering trust. By making FER models more transparent, XAI encourages broader adoption, ensuring that these systems are both interpretable and reliable.This study employs a Convolutional Neural Network (CNN) in TensorFlow to classify emotions. CNNs automatically learn spatial hierarchies, which makes them well-suited for image-based tasks like emotion recognition. By extracting relevant features from facial expressions, CNNs enable accurate classification of emotions.

Additionally, integrating YOLOv10 enhances real-time performance through fast, precise face detection. Known for its ability to handle multiple faces in complex environments, YOLOv10 allows the system to operate efficiently across varied settings. This combination of CNN and YOLOv10 delivers a robust and reliable solution, making it ideal for diverse real-world applications.

CHAPTER 2

LITERATURE SURVEY

This section highlights influential studies that contributed to the development of our emotion recognition model, focusing on facial emotion recognition (FER), real-time systems, and explainable AI.

Sita Rani et al. proposed a machine learning model for recognizing emotional states in children aged 2 to 5, using Principal Component Analysis (PCA) and a Multi-Layer Perceptron (MLP) classifier to analyze a dataset of 273 images across four expressions (happy, sad, neutral, thoughtful). Gradient filtering and Particle Swarm Optimization (PSO) were used for pre-processing, reflecting advanced techniques for understanding child behavior in FER applications.

The use of facial emotion recognition (FER) in real-time systems is investigated by H. Arabian et al., specifically for educating kids with autism spectrum disorder (ASD), who have trouble identifying emotions and facial expressions. This study uses Histogram of Orientated Gradients (HOG) for feature extraction in a machine learning strategy that concentrates on local regions of interest. Three classifiers are used: one employs Support Vector Machine (SVM) classification, while the other two are based on k-Nearest Neighbour. Following training on samples from the Oulu-CASIA database, model performance is evaluated by comparing accuracy on randomly chosen validation sets. This method highlights the potential advantages of localised feature extraction while emphasising the developing techniques in emotion identification.

Sergio Pulido-Castro et al. developed a real-time emotion detection tool for children with ASD, aimed at helping them recognize and mimic emotions. The model used Support Vector Machines (SVM) and Artificial Neural Networks (ANN) to categorize emotions into distinct groups, showing promise for ASD support systems.

In order examine current developments in human emotion detection, Akshita Sharma et al. conducts a comparative analysis of many models, including Convolutional Neural Networks (CNN), Hidden Markov Models (HMM), Eigenfaces, k-means clustering, and Support Vector Machines (SVM). They also summarize various facial emotion identification methods and their uses, looking at the main machine learning approaches used for facial recognition of emotion

and contrasting their benefits, drawbacks, and accuracy.

Anuj Kumar Goel et al. presented an emotion detection model using SVM with OpenCV, analyzing face characteristics like color, shape, and orientation on the FER-2013 dataset. Their model improved accuracy by including additional reference images, illustrating the importance of comprehensive datasets in FER.

Bushra Shaukat Ali et.al study utilizes YOLOv10 technology for efficient object detection in skin cancer identification. By applying preprocessing and augmentation techniques, the pre-trained YOLOv10 model achieved impressive results, including a precision of 1 and a recall of 0.95. The success of YOLOv10 in accurately detecting skin cancer lesions has inspired us to incorporate similar object detection capabilities into our emotion recognition model, leveraging its effectiveness to enhance our approach.

Yan Li et al. demonstrated YOLOv10's robustness in multi-scale pedestrian detection for autonomous driving, emphasizing its reliability in complex environments, which influenced our model's design for real-time efficiency.

To understand SegNet and U-Net, in remote sensing (RS) information building extraction and segmentation, Loghman Moradi et al. look at the use of Explainable Artificial Intelligence (XAI) techniques. There are several main and layer-attribution XAI techniques are statistically assessed using the sensitivity metric. Low sensitivity ratings confirm our results that Deconvolution and Grad-CAM efficiently uncover the fundamental mechanisms of both models and reliably visualise their decision-making processes.

Elias Ennadifi et al. used Grad-CAM for wheat disease detection, highlighting image areas significant for classification and enhancing interpretability. This study further validated our approach to visualizing model decisions.

Uppin Rashmi et al. investigated the BrainCrossFed model for early Alzheimer's detection, integrating federated learning and Grad-CAM for privacy and interpretability. This demonstrated Grad-CAM's value in sensitive applications, reinforcing our commitment to transparent FER.

These studies informed the development of our emotion recognition model, guiding our adoption of YOLO for real-time detection and Grad-CAM for model transparency, creating a balanced and reliable solution for diverse real-world FER applications.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Statement of Problem:

Emotion recognition plays a crucial role in enhancing human-computer interaction, allowing systems to respond adaptively based on user emotions. The problem addressed by this research involves accurately identifying emotions from facial expressions, which is essential for applications in areas such as mental health monitoring, education, and customer service. Traditional systems often face challenges in real-time analysis and accuracy due to variations in facial expressions across individuals. This study proposes a machine learning-based model to improve the reliability and efficiency of emotion recognition.

3.2 Scope for the study:

This study focuses on developing an emotion recognition model using TensorFlow, incorporating Convolutional Neural Networks (CNN) for image processing, and leveraging Explainable AI (XAI) to understand model decisions. It explores the integration of real-time functionality and evaluates the model's accuracy across multiple facial expression datasets. The research is limited to identifying basic emotions like happiness, sadness, anger, surprise, and neutral expressions.

3.3 Objective of the Study:

1. **Develop an Emotion Recognition Model:** Create a robust and accurate model capable of recognizing and classifying human emotions from facial expression videos.
2. **Integrate Explainable AI Techniques:** Implement explainable AI methods to provide insights into the model's decision-making process, ensuring the model's outputs are interpretable.
3. **Incorporate YOLO for Real-Time Detection**: Utilize YOLOv10 for real-time facial detection and tracking within video feeds. YOLO’s high-speed object detection capabilities will enable the model to identify and classify emotions on live video streams, demonstrating practical real-time application potential in dynamic environments.
4. **Evaluate Model Performance**: Assess the model's performance using standard metrics, such as accuracy, precision, recall, and F1-score, and validate its effectiveness across diverse datasets. This evaluation will

confirm the model's reliability and adaptability to different demographic groups and environmental conditions.

5. **Real-World Application:** Demonstrate the model's applicability in real-world scenarios, such as enhancing user experience in human-computer interaction or providing emotional insights in psychological studies

3.4 Realistic Constraints:

This study encounters several constraints:

1. **Computational Constraints**: Real-time emotion recognition requires significant processing power, especially for live detection with YOLOv10.
2. **Data Limitations**: The diversity of emotions in available datasets may limit the generalizability of the model across all demographics.
3. **Accuracy and Reliability**: Achieving high accuracy across different lighting, angles, and facial variations can be challenging.
4. **Ethical Considerations**: Privacy concerns and the responsible use of emotion data are essential constraints to address in deploying emotion recognition systems.

3.5 Engineering Standards:

- Python version 3.12.0
- TensorFlow version 2.13.0
- YOLOv10
- OpenCV version 4.8.0
- NumPy version 1.26.0
- Matplotlib version 3.8.0

CHAPTER 4

DESIGN AND METHODOLOGY

4.1 Theoretical Analysis:

Emotion recognition through image processing relies on the detection of facial features that correspond to distinct emotions. Convolutional Neural Networks (CNN) are chosen for their effectiveness in extracting spatial hierarchies in images, enabling precise analysis of facial regions. YOLOv10 is applied for real-time detection, leveraging its capability for fast object recognition. XAI techniques provide insights into which features the model focuses on during emotion prediction, helping enhance model transparency

4.11 Module:

1. **Data Collection Module**: This module gathers and prepares data by collecting images from well-known facial expression datasets (FER2013) with labelled emotions such as happiness, sadness, anger, and surprise. Preprocessing includes standardizing images by resizing, grayscale conversion, and normalization, ensuring uniformity for improved model accuracy and training efficiency.
2. **Feature Extraction Module**: Using Convolutional Neural Network (CNN) layers, this module extracts essential facial features for emotion recognition. The CNN progressively captures details like eyebrow position, eye shape, and mouth movement, translating these into feature maps that represent emotional cues, crucial for accurate classification.
3. **Real-Time Detection Module**: This module integrates YOLOv10 for high-speed, real-time facial detection in live video feeds. YOLOv10 efficiently detects faces in each frame, enabling the model to analyse and classify emotions continuously as expressions change, a key requirement for interactive applications needing instant feedback.
4. **Interpretability Module**: Grad-CAM (Gradient-weighted Class Activation Mapping) is used in this module to highlight the specific regions of the face that the model focuses on when classifying emotions. Grad-CAM provides a visual heatmap over the input image, indicating which facial areas influenced the prediction, enhancing the transparency and trustworthiness of the model.

4.12 Methodology:

In this research, a deep learning-based model is developed for real-time emotion classification. The model integrates YOLOv10 to detect faces in real time and to extract individual frames from video input for emotion analysis. Convolutional Neural Networks (CNNs), comprising convolution layers, max pooling layers, dense layers, and a softmax classifier, are employed to process these frames for accurate emotion categorization. The model is trained and validated on the FER dataset, with performance assessed using metrics such as Cohen's Kappa score, precision, recall, accuracy, and F1 score. Furthermore, explainable AI techniques, including Grad-CAM, are utilized to create saliency maps that highlight important facial features influencing the model's decisions, enhancing interpretability and transparency.

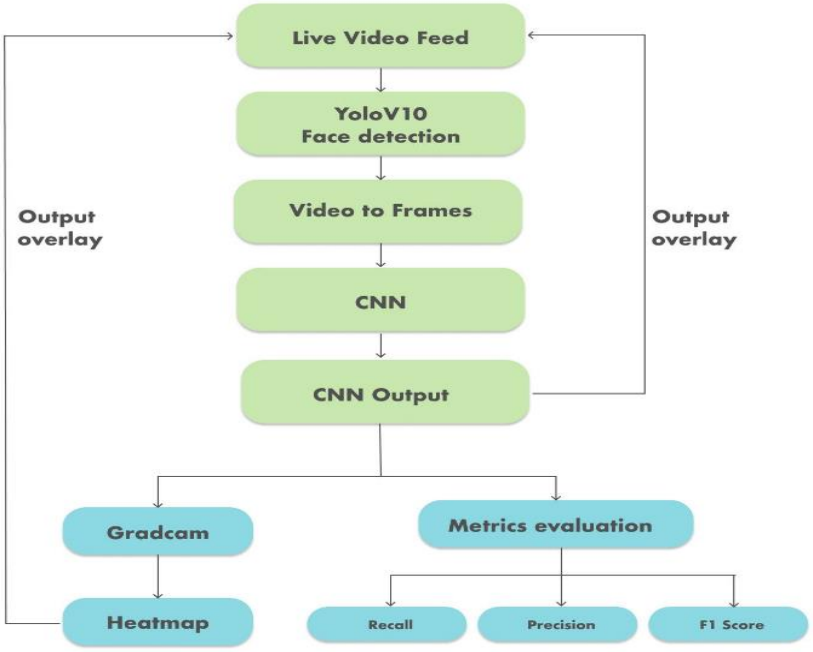

Fig 4.1 Flowchart of the Proposed model

I. Dataset

This project makes use of the FER2013 dataset, which consists of 48x48 pixel greyscale photos classified into seven different emotions: neutral, anger, distaste, fear, happiness, sorrow, and surprise. To make evaluating the model easier, the dataset is separated into training and testing sets.

II. Data Preprocessing

To enhance model performance, several preprocessing steps are applied to the dataset. First, all pixel values are normalized by rescaling them between 0 and

1. For the training set, data augmentation techniques are employed, including random width and height shifts of up to 10%, random zoom adjustments of up to 10%, and random horizontal flips. In contrast, the testing set undergoes only the rescaling process, with no additional augmentation applied.

III. YOLOv10

The YOLOv10 (You Only Look Once version 10) object detection model is employed to detect faces in the input images. Renowned for its high-speed real-time object detection capabilities, YOLOv10 excels at accurately identifying small objects within images. In this project, it identifies bounding boxes around faces, allowing the model to concentrate solely on facial regions for emotion classification. By detecting and isolating facial features, YOLOv10 reduces background noise, thereby enhancing the overall accuracy of emotion recognition. The detected facial regions are subsequently passed to the CNN for further processing.

IV. Convolutional Neural Network

Multiple layers make up the CNN architecture utilized for emotion identification. 48x48x3 image tensors are accepted by the input layer. Following this are three convolutional layers, each of which has max-pooling, batch normalization, and ReLU activation. In particular, a 3x3 kernel size is used by the 32 filters in the first layer, 64 filters in the second layer, and 128 filters in the third layer. Following these layers, a flattening layer creates a 1D feature vector from the 2D feature maps. The network then consists of completely linked layers, with batch normalization following after a dense layer of 256 units and ReLU activation. Dropout layers are included to reduce overfitting, and a second dense layer with 512 units follows, again using ReLU activation. Finally, the output layer is a softmax layer with seven units, each corresponding to one of the emotion classes.

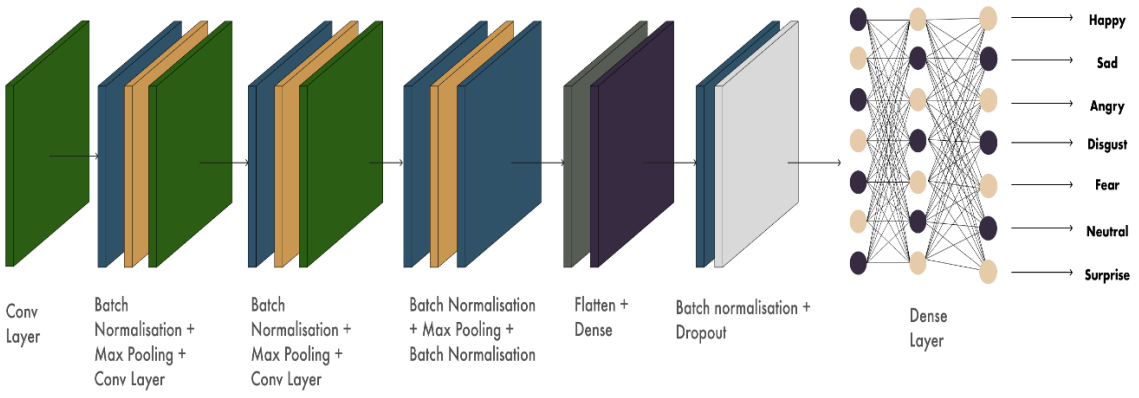

Fig 4.2 Proposed CNN architecture

V. Explainability with GradCam

The framework incorporates GradCAM to improve the comprehension of the model's predictions. The areas of an input picture that the model concentrates on throughout the prediction phase are highlighted in heatmaps produced by GradCAM. This is achieved by superimposing the CNN's convolutional layers' activation maps over the original image, which clearly shows which face characteristics are most important for classifying emotions. GradCAM's visualisation of these regions offers insight into the model's decision-making process and aids in verifying that the right face cues—like the lips and eyes—are affecting the predictions.

The beginning phase involves running a forward pass through the model with an input parameter (x) and a CNN model (f) to determine the type of output (i.e.), the emotion (E) of the model.

$$Y^E = f(x) \tag{4.1}$$

In order to evaluate the gradient up to the last convolutional layer (indexed k), the second phase involves doing a backward pass, commencing with the outcome of the model that corresponds to class E. The gradient map's dimensions (Gij) are c × h × w, where w is its width, h is its height, and c is the number of channels.

$$G_{ij} = \frac{\partial Y^E}{\partial A_{ij}^k} \tag{4.2}$$

The third stage involves averaging the gradients over the width (w) and height (h) dimensions of the activation gradient feature map. There are C elements in resulting vector.

$$\alpha_k^E = \frac{1}{hw} \sum i \sum j \, G_{ij} \tag{4.3}$$

The fourth phase evaluates the activation maps combined in a weighted linear fashion acquired in the forward pass (A^k) with the average aggregated activation gradients (α_k^E) at the identical layer (k). A CAM feature map with dimensions of 1 × h × w is the end product. Here, just the characteristics that have a positive impact on the model for class E are highlighted using a ReLU procedure. Typically, pixels with a negative effect come from different classes. Therefore, the ReLU contributes to the discriminative nature of the Grad CAM class. This is especially helpful when pictures include items from many classes.

$$L_{Grad-Cam}^E = RELU(\sum_k \alpha_k^E B^k) \tag{4.4}$$

In the final phase, the feature map is blended with the output, generating the Grad-CAM visualization. This provides important information about how the model made its decisions and identifies the regions that had the most impact on its predictions.

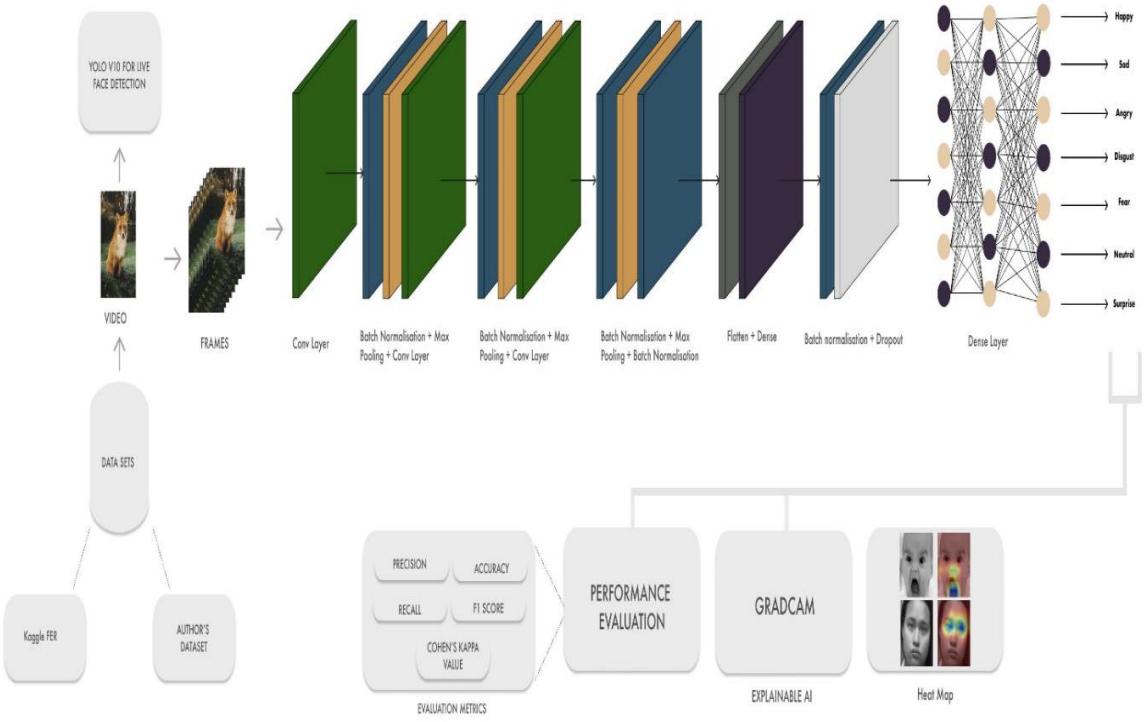

Fig 4.3 Proposed Architecture

VI. Training Procedure

To ensure optimal performance, the model is trained following a carefully configured setup tailored for multi-class emotion classification. The sparse categorical cross-entropy loss function is chosen due to its suitability for handling multi-class classification tasks efficiently, allowing the model to effectively learn distinct emotion classes. Training employs the Adam optimizer, known for its adaptive learning rate adjustments, with an initial learning rate set to 0.001, facilitating a balance between convergence speed and stability. A batch size of 200 is utilized during training to allow the model to process larger data chunks per iteration, improving learning efficiency. For testing, a slightly reduced batch size of 150 is used, enabling finer adjustments in accuracy assessment. To prevent overfitting, an early stopping mechanism is incorporated, monitoring the validation loss over five epochs and halting training if no improvement is observed, thus preserving model generalizability. The entire training process is

conducted across a maximum of 25 epochs, giving the model sufficient exposure to the data to reach its peak performance. This structured approach allows the model to achieve the highest possible accuracy, maximizing its effectiveness in real-time emotion recognition tasks.

VII. Evaluation metrics

Several significant metrics are used to assess the model's performance. A confusion matrix evaluates categorization across all emotion categories, and accuracy is computed on the test dataset. Furthermore, the sensitivity and specificity of the model are shown by measures such as F1 score, Cohen's Kappa, precision, and recall. Grad-CAM visualizations provide heatmaps that show the main regions affecting projections, providing useful explainability to improve interpretability.

4.2 Description of System Environment

The system environment comprises hardware capable of high-speed computation, such as GPUs, to handle the CNN and YOLOv10 algorithms. TensorFlow serves as the primary framework, with additional libraries for image processing and real-time video analysis.

4.3 Design Specifications

4.3.1 Hardware Required:

- **GPU:** Min GTX 1650
- **Camera:** For capturing live video feed in real-time recognition.
- **System Requirements:** AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz

4.3.2 Software Required:

- **TensorFlow**: For building and training the CNN model.
- **OpenCV**: Used for image preprocessing and real-time camera feed handling.
- **YOLOv10**: For efficient real-time facial detection.
- **Scikit learn:** Used for obtaining the evaluation metrics.

CHAPTER 5

RESULTS AND DISCUSSION

5.1Experimental Results

5.1.1 Model Accuracy and Loss Progression

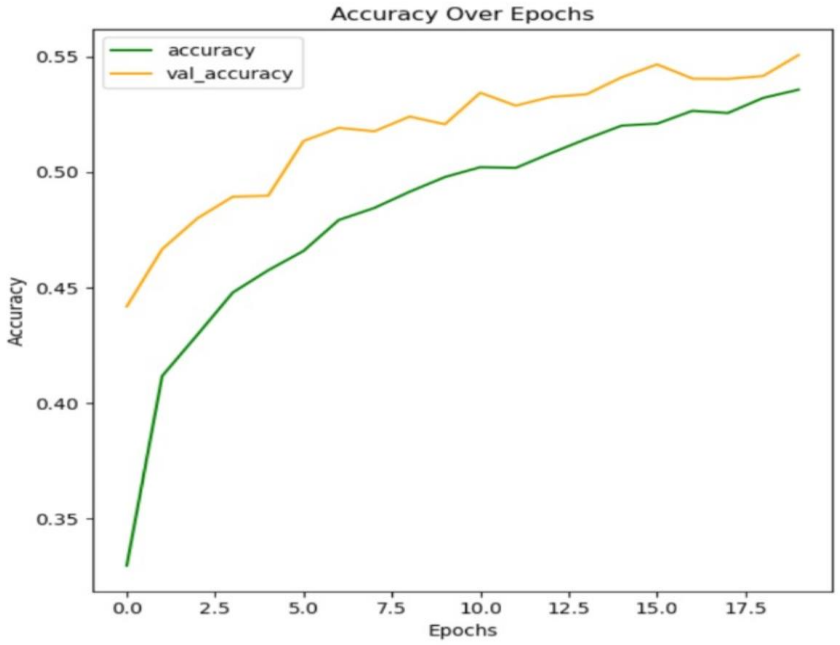

Fig 5.1 Accuracy vs Epochs graph

The model's performance based on accuracy is depicted in Fig. 5.1 by plotting accuracy on the y-axis versus epochs on the x-axis. Here the accuracy of the model is plotted over 20 epochs. Training accuracy is shown by the green line, while validation accuracy is shown by the orange line. The training and validation accuracy increase steadily over the initial epochs, with the validation accuracy remaining consistently higher than the training accuracy throughout. This suggests that the model is learning effectively without significant overfitting, as both accuracies follow a similar upward trend, with only minor fluctuations in the validation accuracy.

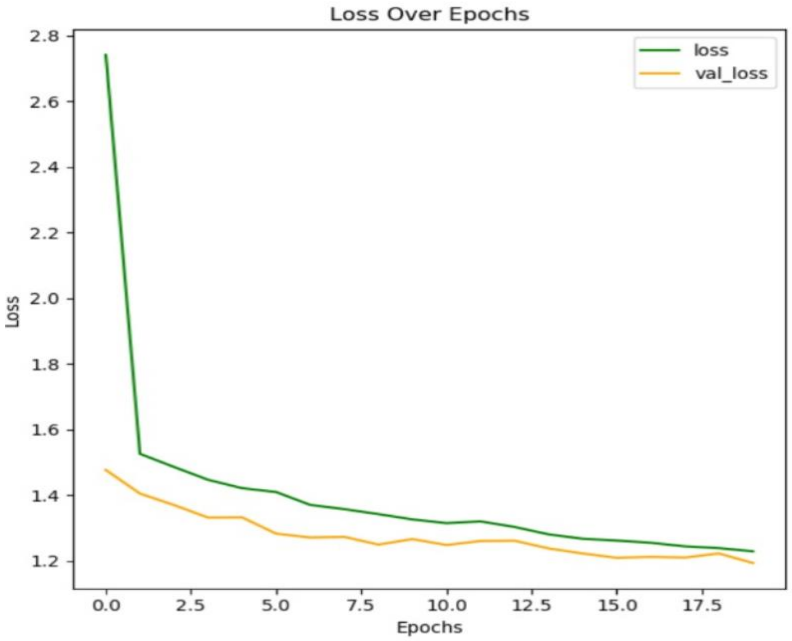

Fig 5.2 Loss vs Epochs Graph

In Fig.5.2 model's training and validation loss across 20 epochs is displayed. In the first few epochs, both the training loss (shown in green) and the validation loss (shown in orange) decline dramatically. The training loss first drops abruptly before progressively reducing. Similar declining trends are seen in the validation loss, which stabilizes after the first few epochs with very slight variations. The model is learning successfully and not overfitting to the training data as both losses reduce without diverging.

5.1.2 Confusion Matrix

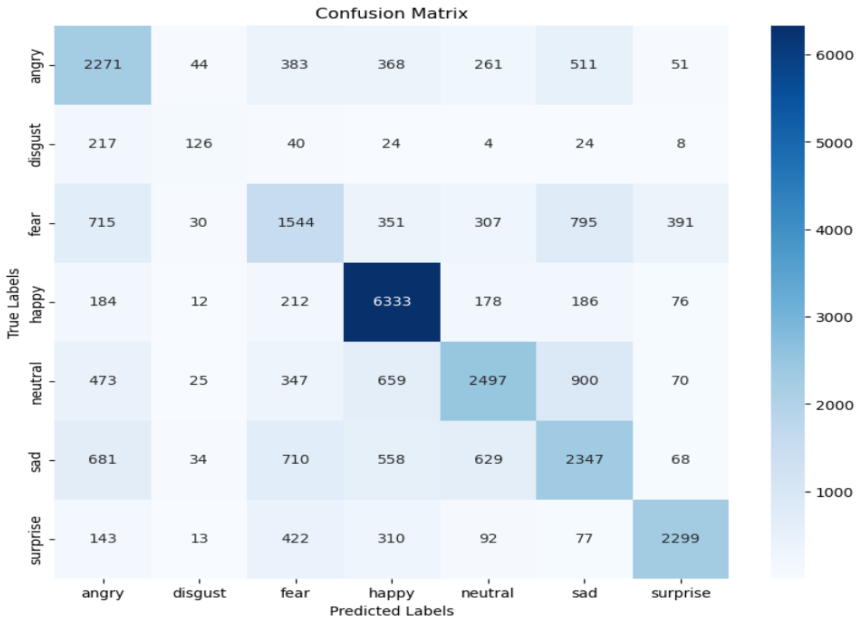

Fig. 5.3 Confusion Matrix of the proposed model

The confusion matrix in Fig 5.3 displays strong accuracy for predicting "happy" (6,333 correct) and "surprise" (2,299 correct), but reveals significant confusion between "angry" and "fear" (715 instances of "fear" misclassified as "angry") and between "neutral" and "sad" (900 "neutral" instances misclassified as "sad," and 629 "sad" misclassified as "neutral"). The model also struggles with "disgust," having only 126 correct predictions and frequent misclassification as "angry" or "fear." Overall, while the model performs well with "happy" and "surprise," it needs improvement in distinguishing between emotions like "fear," "neutral," and "sad."

5.1.3 Precision and Recall

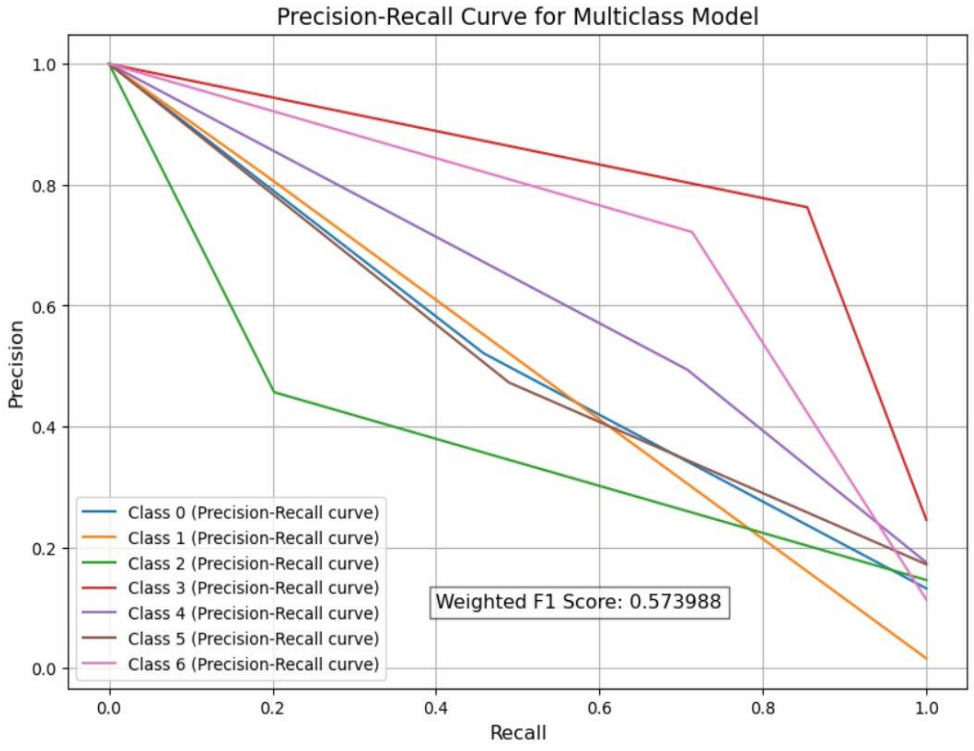

Fig.5.4 Precision-Recall Curve for the proposed model

Fig. 5.4 showcases the Precision-Recall curves for a multiclass classification model, illustrating the balance between precision and recall across seven distinct classes. The model demonstrates strong performance for certain classes, particularly Class 4 (pink) and Class 1 (red), which maintain high precision over a wide range of recall levels, indicating accurate predictions for these categories. While Class 0 (green) and Class 5 show more significant drops in precision and recall, this highlights areas for potential improvement, providing opportunities for fine-tuning to enhance overall model performance The remaining classes exhibit steady, moderate precision-recall trade-offs, showing that the model performs reasonably well across various scenarios. With a weighted F1 score of 0.573988 on the validation set, the model delivers a solid

foundation and shows promise, especially with adjustments that could improve consistency across all classes. The overall performance suggests a strong starting point, with room for refinement to unlock even better results in future iterations.

5.1.4 Cohen Kappa Value:

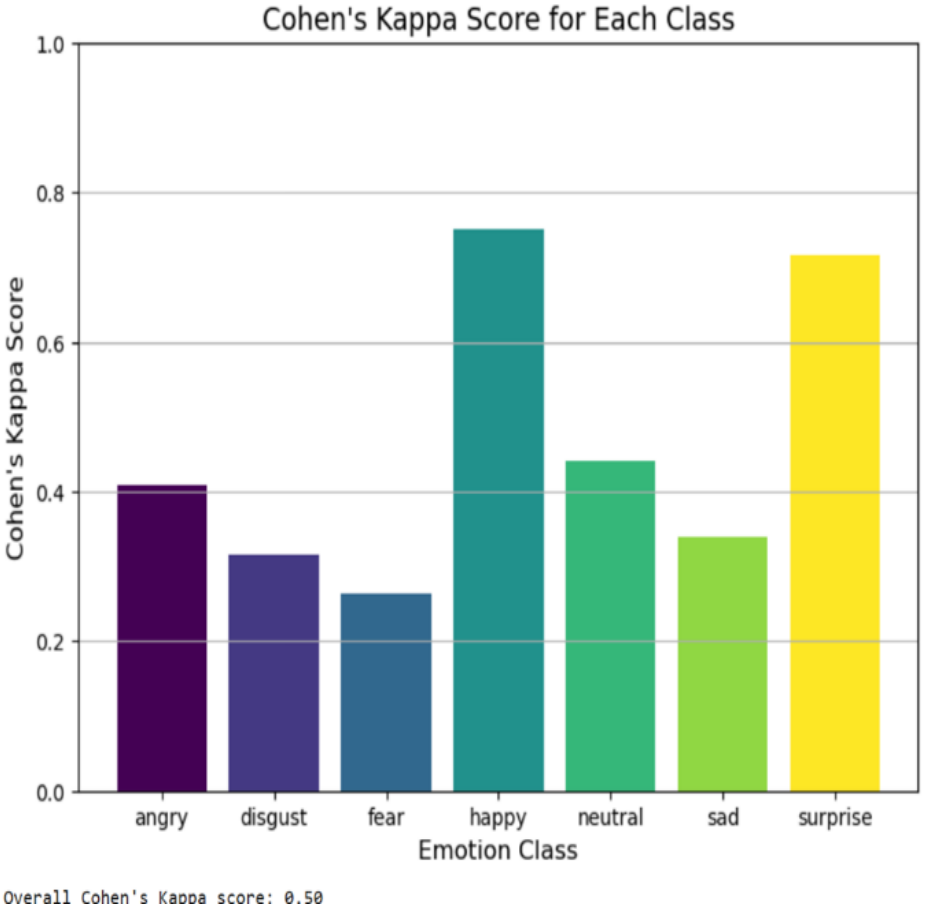

Fig.5.5 Cohen Kappa Score

Fig. 5.5 presents the Cohen's Kappa scores for a multiclass classification model, offering insights into the model's agreement between predicted and true labels for each class. The scores range across classes, with Class 3 and Class 6 exhibiting the highest agreement, achieving Kappa scores close to 0.8, indicating strong consistency in their predictions. Class 4 and Class 5 follow with moderate agreement, while Class 0, Class 1, and Class 2 show lower Kappa scores, with Class 1 and Class 2 having the weakest agreement. The overall Cohen's Kappa score for the model is 0.50, indicating a fair level of agreement across all classes. These results suggest that while the model performs well for certain classes, there is room to improve the consistency of predictions for others. The relatively strong performance in Classes 3 and 6 provides a solid foundation to build upon, and with targeted adjustments, the agreement for underperforming classes could be improved, enhancing the model's overall reliability

S.No	Metrics	Score
1	Precision	0.595951
2	Recall	0.600586
3	F1 Score	0.573988
4	Cohnen Kappa Score	0.50

Table 5.1 Overall Performance metrics of the model

Table 5.1 indicates that the model has a precision of around 59.6%, which is it accurately predicts the positive class roughly 59.6% of the time. This is very important when reducing false positives is crucial. With a recall of around 60.1%, the model detects about 60.1% of true positive instances, which is crucial for lowering false negatives. At around 57.4%, the F1 score strikes a compromise between recall and precision. Finally, a Cohen's Kappa score of 0.50 indicates a considerable degree of agreement between the real labels and the model's predictions, hinting at both places for development and a certain degree of dependability.

5.1.5 Results from Yolov10 and CNN model

Fig .5.6 Emotion Recognition using YOLOv10 and CNN

Fig. 5.6 illustrates a real-time emotion detection system integrating YOLOv10 for precise face identification. The system predicts the emotion as "neutral,"

5.1.6 Grad-CAM visualization

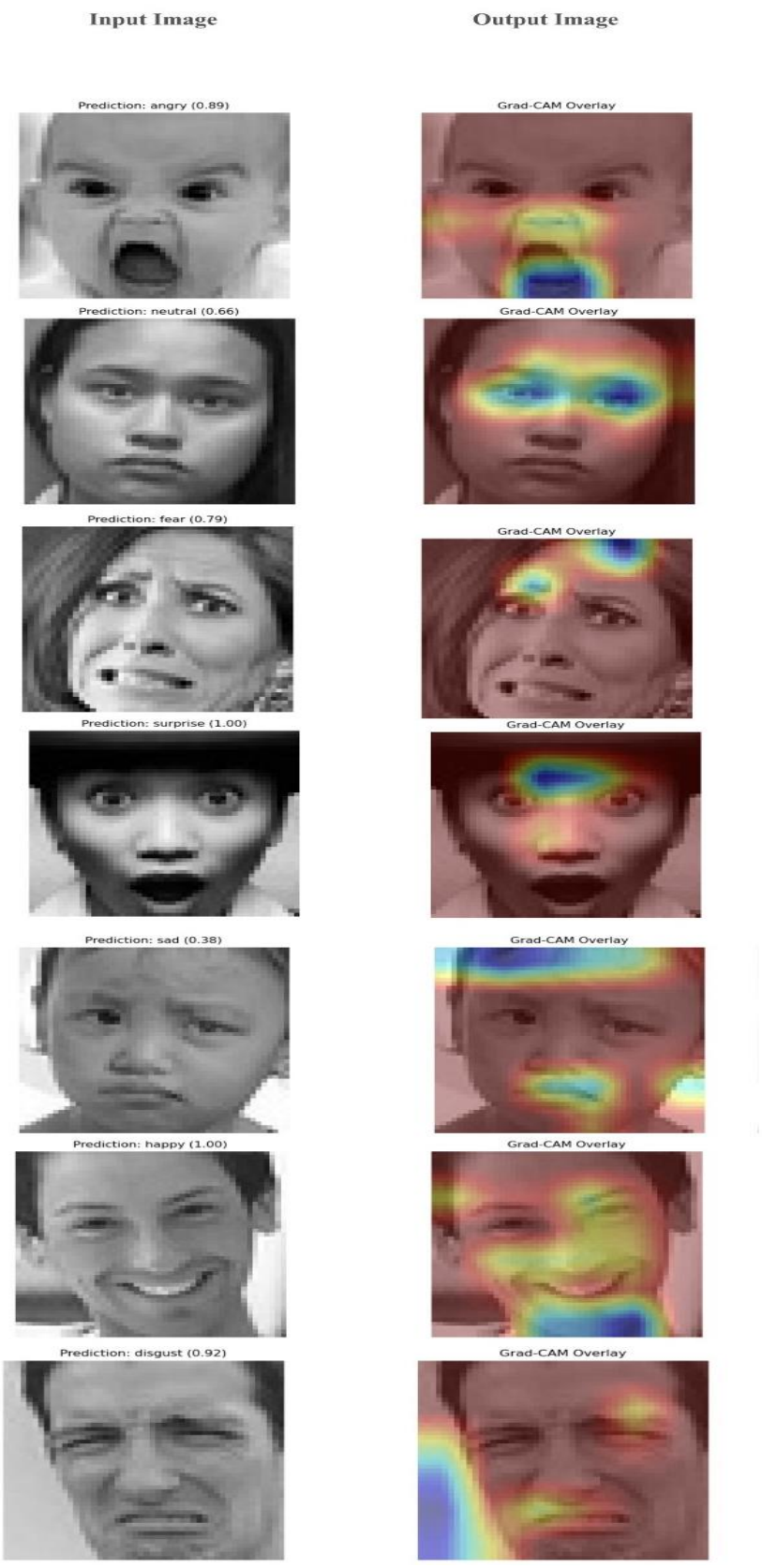

Fig 5.7 Grad-CAM visualization of the output

In Fig 5.7 we can see the model output images which were analyzed by Grad-CAM and the heat maps produced which helps us to understand the areas of focus that were used by the model to make the predictions

5.2 Suggestions and Recommendations

1. **Increase Dataset Diversity**: To enhance the model's generalization, it's recommended to expand the training dataset to include more diverse facial expressions from different age groups, ethnicities, and cultural backgrounds. This reduces biases and improves the model's effectiveness across a wider range of users.
2. **Utilize Advanced Data Augmentation:** Using advanced techniques like Generative Adversarial Networks (GANs) to generate synthetic emotional expressions can significantly expand dataset variety, improving the model's ability to recognize complex emotions.
3. **Implement Ensemble Models:** Combining different models, such as CNNs with RNNs, can improve accuracy by capturing temporal changes in facial expressions, which is particularly useful for video-based emotion detection.
4. **Optimize for Edge and Mobile Devices:** Optimizing the model for mobile devices (e.g., TensorFlow Lite or PyTorch Mobile) can increase speed and accessibility, making the system suitable for real-time applications on wearable devices or mobile apps.
5. **Expand Explainable AI Components:** Exploring additional XAI techniques like SHAP or LIME can enhance interpretability and provide more transparency in how the model identifies emotions, contributing to ethical AI use.
6. **Incorporate Multi-modal Emotion Recognition:** Integrating other modalities like audio analysis or text sentiment analysis can improve accuracy and adaptability, making the system robust across different communication channels.
7. **Explore Lightweight Model Alternatives:** Implementing lightweight models such as MobileNet or EfficientNet can reduce computational demands, enabling real-time emotion recognition on resource-constrained devices**.**

5.3 Future Enhancements

FER will play a key role in real-time applications, such as virtual and augmented reality, and will enable emotionally responsive AI in social robotics and smart devices. In healthcare, FER has the potential to monitor mental health, assist in therapy sessions, and offer real-time emotional feedback. Personalized, emotion-aware smart devices could enhance user experiences, adjusting their interfaces or behavior based on emotional states. Ethical frameworks and privacy protection will become increasingly important as FER technology expands, with a focus on responsible data use and transparency. Explainable AI (XAI) techniques, such as Grad-CAM, will make FER models more interpretable and trustworthy. FER's future applications will also extend to autonomous systems, education, marketing, and security, where emotion-aware technologies can enhance safety, personalized learning, and consumer insights. Data-efficient FER models will enable real-time emotion detection even on low-resource devices, while privacy-preserving machine learning techniques will ensure sensitive facial data remains secure. The further advancement of FER aims to close the emotional space between people and technology, improving responsiveness, empathy, and of interactions

5.4 Conclusion

The model's performance metrics, including precision, recall, F1 score, and Cohen’s Kappa, demonstrate moderate effectiveness in emotion recognition, with notable strengths in identifying distinct emotions such as "happy," "sad," and "angry." However, the model faces challenges in accurately classifying more subtle emotions, particularly "neutral," "fear," and "sad." The integration of YOLOv10 has significantly enhanced the model’s ability to detect and analyze emotions in images, offering high precision in object detection. Additionally, Grad-CAM visualizations provide valuable interpretability by pointing out the areas of the input that are critical in the model’s decision-making process. Overall, the combination of these advanced detection methods and interpretability tools forms a robust foundation for the model. While the system performs well in certain areas, it also reveals opportunities for further refinement, particularly in improving classification accuracy for more nuanced emotions. This framework is well-positioned for continuous improvement in emotion recognition tasks, potentially benefiting human-computer interaction and emotion-aware applications

REFERENCES

[1] S. Rani, P. Bhambri and M. Chauhan, "A Machine Learning Model for Kids' Behavior Analysis from Facial Emotions using Principal Component Analysis," 2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT), Haikou, China, 2021, pp. 522-525, doi: 10.1109/ACAIT53529.2021.9731203

[2] Annamalai, Manikandan & Muthiah, Ponni. (2022). An Early Prediction of Tumor in Heart by Cardiac Masses Classification in Echocardiogram Images Using Robust Back Propagation Neural Network Classifier. Brazilian Archives of Biology and Technology. 65. 10.1590/1678-4324-2022210316.

[3] Manikandan, Annamalai, M,Ponni Bala. (2023). Intracardiac Mass Detection and Classification Using Double Convolutional Neural Network Classifier. Journal of Engineering Research. 11(2A). 272-280. 10. 36909/jer.12237.

[4] A. Sharma, V. Bajaj and J. Arora, "Machine Learning Techniques for Real-Time Emotion Detection from Facial Expressions," 2023 2nd Edition of IEEE Delhi Section Flagship Conference (DELCON), Rajpura, India, 2023, pp. 1-6, doi: 10.1109/DELCON57910.2023.10127369.

[5] A. K. Goel, A. Jain, C. Saini, Ashutosh, R. Das and A. Deep, "Implementation of AI/ML for Human Emotion Detection using Facial Recognition," 2022 IEEE 4th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA), Goa, India, 2022, pp. 511-515, doi: 10.1109/ICCCMLA56841.2022.9989091.

[6] Balamurugan, D. & Seshadri, s.Aravinth & Reddy, P. & Rupani, Ajay & Manikandan, A.. (2022). Multiview Objects Recognition Using Deep Learning-Based Wrap-CNN with Voting Scheme. Neural Processing Letters. 54. 1-27. 10.1007/s11063-021-10679-4.

[7] Y. Li, W. Leong and H. Zhang, "YOLOv10-Based Real-Time Pedestrian Detection for Autonomous Vehicles," *2024 IEEE 8th International Conference on Signal and Image Processing Applications (ICSIPA)*, Kuala Lumpur, Malaysia, 2024, pp. 1-6, doi: 10.1109/ICSIPA62061.2024.10686546.

[8] Sheikdavood K, Surendar P, Manikandan A. Certain Investigation on Latent Fingerprint Improvement through Multi-Scale Patch Based Sparse Representation. Indian Journal of Engineering. 2016; 13(31):59-64

[9] E. Ennadifi, S. Laraba, D. Vincke, B. Mercatoris and B. Gosselin, "Wheat Diseases Classification and Localization Using Convolutional Neural Networks and GradCAM Visualization," *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*, Fez, Morocco, 2020, pp. 1-5, doi: 10.1109/ISCV49265.2020.9204258.

[10] U. Rashmi, B. M. Beena and S. Ambesange, "BrainCrossFed CNN Model for Alzheimer Classification using MRI data and Comparison and Benchmarking proposed model with DINOv2 and ExplainableAI using GradCAM," *2023 International Conference on the Confluence of Advancements in Robotics, Vision and Interdisciplinary Technology Management (IC-RVITM)*, Bangalore, India, 2023, pp. 1-7, doi: 10.1109/IC-RVITM60032.2023.10435182.

APPENDIX

Appendix A: Model Architecture Details

1. **Convolutional Neural Network (CNN) Layers**
    - Overview of CNN layers used for feature extraction.
    - Layer configuration: convolution layers, activation functions (ReLU), pooling layers, and dense layers.
    - Softmax layer for final classification.
2. **YOLOv10 Integration for Face Detection**
    - Description of how YOLOv10 is used for real-time face detection.
    - Configuration and parameters adjusted for optimal performance.
3. **Explainable AI (XAI) with Grad-CAM**
    - Grad-CAM implementation for model interpretability.
    - Example visualizations of saliency maps and explanations.

Appendix B: Data and Preprocessing

1. **Datasets Used**
    - Names and details of datasets (FER2013)
    - Distribution of emotion classes and sample sizes.
2. **Preprocessing Techniques**
    - Steps for image normalization, resizing, and grayscale conversion.
    - Data augmentation methods applied, including rotation, flipping, and scaling.

Appendix C: Hyperparameter Configuration

1. **Training Parameters**
    - Batch size: 200 (training), 150 (testing).
    - Learning rate: 0.001 with Adam optimizer.
    - Epochs: Maximum of 25 with early stopping after 5 epochs if no improvement.

2. **Loss Function**
    - Sparse categorical cross-entropy for multi-class classification.

Appendix D: Performance Metrics and Evaluation

1. **Metrics Used**
    - Accuracy, precision, recall, F1 score, and Cohen's Kappa score.
    - Explanation of each metric's importance in evaluating model performance.
2. **Validation and Testing Approach**
    - Details on training-validation split and testing protocol.
    - Use of cross-validation where applicable.

Appendix E: Software and Libraries

1. **Development Environment**
    - Python version: 3.12.0 (latest).
    - IDEs used (e.g., Jupyter Notebook, PyCharm).
2. **Libraries and Versions**
    - TensorFlow (v2.14.0), OpenCV (v4.8.0), YOLOv10, LIME (v0.2.0.1), SHAP (v0.42.0), NumPy (v1.26.0), Matplotlib (v3.8.0).

Appendix F: Hardware Specifications

1. **System Requirements**
    - **CPU:** AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHzRAM and storage requirements.
    - **GPU:** GTX 1650

Appendix G: Sample Code Snippets

1. Data Loading and Preprocessing

```
# Set up data generators
train_datagen = ImageDataGenerator(
    width_shift_range=0.1,
    height_shift_range=0.1,
    zoom_range=0.1,
    horizontal_flip=True,
    rescale=1./255,

)

validation_datagen = ImageDataGenerator(
    rescale=1./255,

)

# Load datasets
train_dataset = train_datagen.flow_from_directory(
    'C:/Users/hari/OneDrive/Documents/fer/train',
    target_size=(48,48),
    batch_size=200,
    class_mode='sparse',

)

validation_dataset = validation_datagen.flow_from_directory(
    'C:/Users/hari/OneDrive/Documents/fer/test',
    target_size=(48, 48),
    batch_size=150,
    class_mode='sparse',
)
```

2. Model Architecture

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu', input_shape=(48, 48, 3)),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(128, (3, 3), activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(256, activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(len(class_names), activation='softmax')
])
```

3. Grad-CAM Implementation

```
def display_gradcam(img_path, heatmap, alpha=0.4):
    # Load and prepare the original image
    img = cv2.imread(img_path)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img = cv2.resize(img, (48, 48))  # Ensure it matches model input size

    # Rescale the heatmap to range 0-255 and resize it to match the input image size
    heatmap = cv2.resize(heatmap, (48, 48))
    heatmap = np.uint8(255 * heatmap)  # Convert to 8-bit integer format

    # Apply the color map (jet) to create a colored heatmap
    jet = cv2.applyColorMap(heatmap, cv2.COLORMAP_JET)

    # Ensure both images have 3 channels
    if len(img.shape) == 2:  # If the original image is grayscale, convert it to RGB
        img = cv2.cvtColor(img, cv2.COLOR_GRAY2RGB)

    # Overlay the heatmap on the original image
    superimposed_img = cv2.addWeighted(jet, alpha, img, 1 - alpha, 0)

    # Display the image
    plt.imshow(superimposed_img)
    plt.axis('off')
    plt.show()

# Now call the function
display_gradcam(img_path, heatmap)
```