Emotion Recognition System with Adaptive User Interface

Shimil Shijo
MS Data Science
University of Michigan-Dearborn
Dearborn, MI
shimil@umich.edu

***Abstract*—Emotion detection plays a critical role in creating responsive and empathetic systems. This project utilizes a convolutional neural network (CNN) to classify emotions from facial images and integrates an adaptive graphical user interface (GUI) that dynamically alters its appearance based on detected emotions. The model achieved reasonable performance, detecting seven distinct emotions with partial results demonstrating its potential. Future enhancements include real-time detection and integration with assistive technologies.**

***Index Terms*—Emotion Recognition, Adaptive User Interface, Convolutional Neural Network, Deep Learning.**

I. Introduction

Emotion recognition from facial expressions is a rapidly evolving area within the field of artificial intelligence (AI), with applications ranging from human-computer interaction (HCI) to mental health monitoring. As the demand for emotionally intelligent systems increases, the ability to accurately detect and interpret human emotions in real-time becomes critical. Such systems not only improve user experience but also enable more personalized interactions by adapting to the emotional state of the user.

The primary goal of this project is to develop a robust emotion detection system that employs deep learning techniques to classify facial expressions and dynamically adapts the user interface (UI) based on the detected emotion. This approach combines computer vision with human-computer interaction principles, enabling a responsive and empathetic system. Specifically, the system uses a convolutional neural network (CNN), a state-of-the-art deep learning architecture known for its success in image classification tasks, to classify emotions from facial images. Once the emotion is detected, the user interface adapts by changing its visual theme to match the emotional state, providing a more personalized and interactive experience.

This project is of particular interest for several reasons. First, it provides a real-time solution for emotion recognition, which can be applied to a wide range of fields, including healthcare, gaming, marketing, and customer service. By using facial expressions as the input for emotion detection, the system leverages a natural and non-intrusive method of understanding a user's emotional state. This can be useful in sensitive areas such as therapy or counseling, where emotional feedback is crucial. Second, the integration of an adaptive UI opens the door to more intuitive and empathetic designs for user interfaces, which are currently underexplored in many domains. For instance, a system that changes its visual elements based on emotional detection can help improve mood regulation, enhance user engagement, and create a more immersive user experience.

The system developed in this project aims to classify seven emotions: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise, based on facial expression datasets. The integration of a deep learning model with a dynamic UI design is an ambitious step forward in building more adaptive and intelligent user experiences. The combination of deep learning, real-time emotion detection, and adaptive interface design has the potential to significantly improve user interaction in various applications, such as smart assistants, online education, and virtual healthcare environments.

II. Related Work

In the paper [1], the authors proposed an emotion recognition system based on a convolutional neural network (CNN) model trained on the FER2013 dataset. They highlighted the challenges associated with class imbalances in emotion datasets, which often lead to poor performance for minority classes such as fear and surprise. To overcome this, the authors employed data augmentation techniques such as rotation, flipping, and scaling to create a more balanced dataset. Their approach demonstrated significant improvements in recognizing both common and complex emotions, making their method a valuable contribution to emotion recognition research.

The authors of the paper [2] explored the potential of transfer learning in facial emotion recognition. They used pre-trained models like VGGFace, originally designed for face recognition tasks, and fine-tuned them on the AffectNet dataset, which contains a diverse range of emotional expressions. By leveraging the features learned by the pre-trained models, the authors were able to achieve high accuracy in detecting subtle emotional expressions, such as those associated with contempt or confusion, which are often difficult to identify.

Xie et al. [3] introduced a hybrid model that combined CNN and long short-term memory (LSTM) networks to capture both spatial and temporal features of facial expressions. They trained their model on the CK+ dataset, which includes video

sequences of emotional expressions, allowing the model to understand how emotions evolve over time. Their experiments showed that the hybrid model outperformed traditional CNNs in recognizing emotions in dynamic video-based scenarios, demonstrating the importance of considering temporal aspects in emotion recognition systems.

In the work of Zhao and Zhang [4], an ensemble learning approach was adopted to enhance emotion recognition accuracy. The authors combined multiple classifiers, including support vector machines, decision trees, and random forests, to create a robust system capable of handling variations in facial expressions. By testing their system on the JAFFE dataset, they showed that the ensemble approach reduced overfitting and improved the system's ability to generalize across different subjects, making it a reliable choice for real-world applications.

The authors in [5] focused on developing a lightweight CNN model specifically designed for mobile-based emotion recognition systems. Recognizing the computational limitations of mobile devices, they optimized the network architecture to reduce the number of parameters and computational requirements without compromising accuracy. Their experiments demonstrated that the model could process real-time video streams on mobile devices efficiently, making it suitable for applications such as mobile health monitoring and interactive gaming.

The paper by Li et al. [6] proposed a multi-modal emotion recognition system that integrated facial expression analysis with speech emotion recognition. They utilized the IEMOCAP dataset, which contains both visual and audio data, to train their system. By combining the strengths of both modalities, the authors showed that their system could accurately identify emotions even in scenarios where one modality was less reliable, such as low-light conditions affecting facial recognition or background noise interfering with speech recognition.

In [7], the authors addressed the challenges posed by occlusions and varying lighting conditions in emotion recognition. They introduced an attention mechanism into a CNN model to focus on the most relevant parts of the face while ignoring occluded or poorly illuminated regions. Their system, tested on the SFEW dataset, achieved significant improvements in recognition accuracy under challenging conditions, highlighting the effectiveness of attention mechanisms in enhancing model robustness.

Kim et al. [8] presented an innovative method for emotion recognition using generative adversarial networks (GANs). They used GANs to synthesize realistic facial expressions, particularly for underrepresented classes in the dataset. By augmenting the training data with these synthesized expressions, the authors improved the model's ability to recognize emotions such as disgust and fear, which are often underrepresented in natural datasets. Their approach, tested on the FER2013 dataset, provided valuable insights into the potential of GANs for data augmentation in emotion recognition.

The work of Singh et al. [9] introduced a hierarchical deep learning model designed to identify primary and secondary emotions. Their model first classified broad emotional categories such as positive, neutral, and negative, before identifying specific emotions within each category. Using the EMOTIC dataset, the authors demonstrated that their hierarchical approach improved the system's understanding of complex emotional states and contexts, such as those arising in social or professional interactions.

Finally, the authors of [10] explored the use of graph convolutional networks (GCNs) for emotion recognition. They treated facial landmarks as nodes in a graph and used GCNs to model the relationships between these landmarks. By capturing the structural information of the face, their method achieved state-of-the-art results on the RAF-DB dataset. The study highlighted the potential of graph-based approaches for capturing subtle geometric changes in facial expressions, providing a new direction for emotion recognition research.

III. Methodology

A. Data Collection

The project uses the FER-2013 dataset (https://www.kaggle.com/datasets/msambare/fer2013). It contains 48x48 pixel grayscale images of human faces classified into seven emotion classes- angry, disgust, fear, happy, sad, surprise, and neutral. The dataset comprises over 35,000 images and it is splitted into training and test sets.

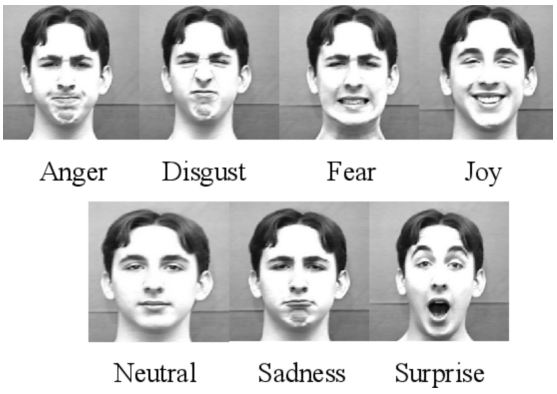

Fig. 1. Classes of FER-2013 dataset

B. Model Architecture

The core of the emotion detection system is a Convolutional Neural Network (CNN) developed using the TensorFlow/Keras framework. The CNN was designed and trained to classify facial expressions into seven emotion categories: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise. The architecture was structured to extract meaningful features from facial images and perform classification with high accuracy.

1) Convolutional Layers: The CNN begins with multiple convolutional layers. These layers are responsible for extracting features from the input images by applying filters that detect patterns such as edges, textures, and shapes, which are characteristic of facial features. The first convolutional layer uses 32 filters, followed by

layers with 64, 128, and 256 filters as the architecture deepens, allowing the network to learn increasingly complex features. Each convolutional layer is followed by a Rectified Linear Unit (ReLU) activation function, which introduces non-linearity, enabling the model to capture intricate patterns in the data.

2) Batch Normalization: Batch normalization is applied after each convolutional layer to stabilize and accelerate the training process. It normalizes the activations of the previous layer, reducing the effects of internal covariate shift and allowing the network to converge faster.
3) Max Pooling Layers: Max pooling layers follow the convolutional layers to downsample the feature maps, reducing their spatial dimensions while preserving the most critical information. Pooling reduces computational complexity and ensures translational invariance, which is crucial for robust facial recognition. Each max pooling operation uses a 2×2 window to extract the most prominent features from the feature maps.
4) Dropout Layers: Dropout layers are interspersed throughout the network to prevent overfitting. By randomly setting a fraction of the input units to zero during training, dropout regularizes the model, forcing it to learn robust and generalized features. Dropout rates of 0.25 and 0.5 are used in different parts of the architecture.
5) Fully Connected Layers: After the convolutional and pooling layers, the feature maps are flattened into a one-dimensional vector and passed through dense (fully connected) layers. These layers act as the decision-making components of the network, mapping the extracted features to the corresponding emotion classes. The fully connected layer in this architecture has 256 neurons with ReLU activation.
6) Softmax Output Layer: The final layer of the network is a softmax layer with seven neurons, each corresponding to one emotion class. This layer outputs a probability distribution over the seven classes, and the class with the highest probability is selected as the predicted emotion. This design ensures that the predictions are probabilistic and interpretable.

The architecture was trained using a grayscale version of the dataset, resized to 48×48 pixels, to match the input dimensions expected by the network. Using grayscale images reduces computational requirements while retaining sufficient detail for accurate emotion classification.

C. Training and Hyperparameter Tuning

During training, the model was optimized using the Adam optimizer with a learning rate of 0.0001, and the categorical crossentropy loss function was used for multi-class classification. The training process involved iterating over the dataset in batches of 32 for 50 epochs. The input data was fed through a data generator that augmented images to increase dataset diversity and improve model generalization. Each epoch involved forward passes of data through the network and backpropagation to adjust the weights.

To ensure the best-performing model was saved during training, a callback mechanism was implemented using the `ModelCheckpoint` utility from Keras. This callback monitored the validation accuracy during training and saved only the weights of the model with the highest validation accuracy. The weights were stored in a file. This approach ensured that the final model used for evaluation and deployment represented the best iteration achieved during training.

The training process also included validation at the end of each epoch using a separate validation dataset. This allowed real-time monitoring of the model's performance on unseen data and provided insights into overfitting or underfitting trends.

1) Hyperparameter Tuning: Hyperparameter tuning played a critical role in optimizing the model's performance. The primary hyperparameters explored included the learning rate, the number of filters in each convolutional layer, the dropout rates, and the batch size.

- Learning Rate Tuning: The learning rate was set to 0.0001 after experimenting with different values. A lower learning rate allowed more gradual and stable convergence, preventing abrupt updates that could lead to suboptimal solutions.
- Filter Sizes and Numbers: The number of filters in the convolutional layers was carefully chosen to balance feature extraction capabilities with computational efficiency. Layers with 32, 64, 128, and 256 filters were used to progressively capture low-level and high-level features in the images.
- Dropout Rates: Dropout rates were adjusted for regularization, with values of 0.25 in the convolutional layers and 0.5 in the fully connected layers. These rates were determined through experimentation to reduce overfitting while maintaining high validation accuracy.
- Batch Size: A batch size of 32 was selected after testing with smaller and larger batch sizes. This choice provided a good trade-off between convergence speed and stability.
- Epochs: The model was trained for 50 epochs. This value was determined by monitoring the training and validation loss curves to ensure the model stopped improving significantly beyond this point.

Through iterative testing of these hyperparameters, the final configuration provided robust performance and minimized generalization errors. The combination of efficient training techniques and hyperparameter tuning resulted in a well-optimized model for emotion recognition.

D. Evaluation

Model performance was assessed using metrics like accuracy, precision, recall, and F1 scores. Confusion matrices provided insights into classification effectiveness, highlighting areas for improvement. Also the training-validation accuracy and training-validation loss graphs are plotted.

E. GUI Development

A Python-based Tkinter GUI was developed to make the emotion detection system user-friendly and interactive. The GUI allowed users to upload images, view detected emotions, and experience an adaptive interface that responded to the predicted emotion.

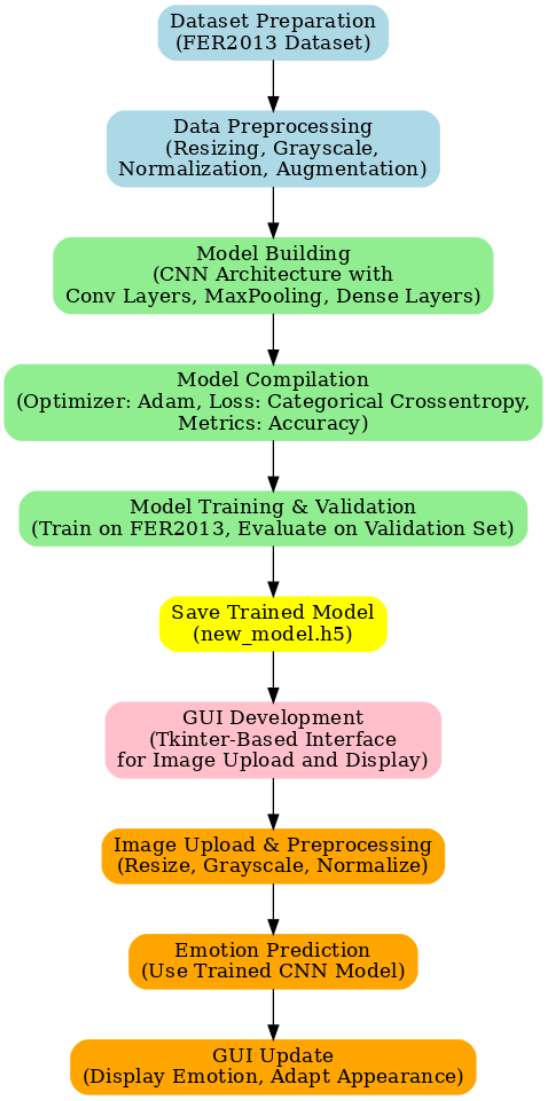

Fig. 2. Flow Chart : Emotion Detection

Key Features

- Image Upload Capability : Users could upload images using a button labeled ”Upload Image for Emotion Detection.” The selected image was displayed within the GUI using the Pillow library, providing immediate visual feedback.
- Emotion Detection and Display : The uploaded image was preprocessed and passed to the trained CNN model for emotion prediction. The detected emotion was displayed in the interface, e.g., ”Detected Emotion: Happy.”
- Adaptive Interface : The GUI dynamically changed its background and displayed messages based on the detected emotion.

F. Testing

Finally both the models were tested with unseen, unlabelled data and answers are visualized through the GUI.

IV. Results

The performance of the model was evaluated using two primary metrics: **Loss** and **Accuracy**. The loss metric quantifies the error between the predicted and true labels, with lower values indicating better model performance. Accuracy measures the proportion of correctly classified samples, providing a straightforward assessment of the model’s predictive capabilities.

A. Training and Validation Loss

The left graph in Figure 3 depicts the training and validation loss over 50 epochs. The following observations can be made:

- The training loss (blue curve) shows a consistent and steady decrease, indicating that the model successfully learned patterns from the training data over the epochs.
- The validation loss (red curve) also decreases, following a similar trend as the training loss. This suggests that the model generalizes well to unseen data and does not suffer from significant overfitting during training.
- The gap between the training and validation loss curves remains small throughout the epochs, further indicating that the model maintains good generalization performance.

B. Training and Validation Accuracy

The right graph in Figure 3 illustrates the training and validation accuracy over 50 epochs. Key points include:

- The training accuracy (blue curve) demonstrates a steady increase, showing that the model progressively improves its ability to classify the training data correctly.
- The validation accuracy (red curve) initially increases rapidly during the early epochs, reflecting efficient learning of patterns in the data. After about 20 epochs, the rate of improvement slows and stabilizes, suggesting the model has reached its peak performance on the validation set.
- The gap between the training and validation accuracy is minimal, indicating that the model performs consistently on both the training and validation datasets.

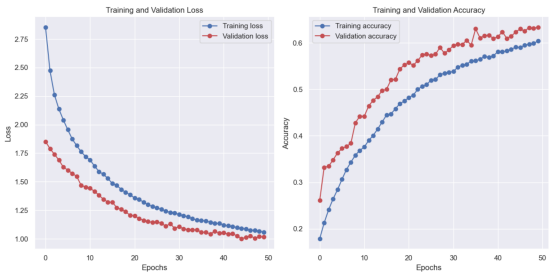

Fig. 3. Training and Validation Loss and Accuracy.

C. Confusion Matrix

The confusion matrix in Figure 4 provides insights into the model's performance across the seven emotion categories.

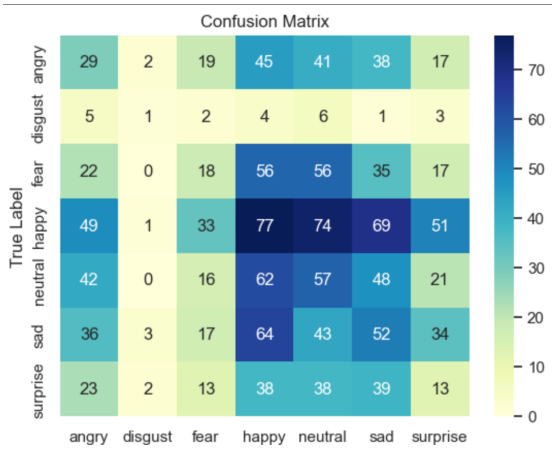

Fig. 4. Confusion Matrix

Key observations include:

- Correct classifications are highest for *happy* (77 samples) and lowest for *disgust* (1 sample).
- Misclassifications are significant for:
  - *Angry* mislabeled as *neutral* (57 samples) and *fear* (18 samples).
  - *Neutral* misclassified as *sad* (43 samples) and *angry* (41 samples).
  - *Surprise* often predicted as *neutral* (21 samples) or *happy* (51 samples).
- The *disgust* category shows poor performance, likely due to insufficient representation in the dataset.
- Overlap between similar emotions, such as *neutral*, *fear*, and *sad*, contributes to misclassifications.

D. Accuracy & Classification Report

The classification results indicate that the model achieved a test accuracy of 62.57% on the dataset, but the detailed classification metrics suggest imbalanced performance across classes. The f1-scores for the seven emotion classes are generally low, ranging from 0.06 to 0.24, with the "happy" class having the highest precision, recall, and f1-score at 0.24, likely due to better representation or distinguishability in the data. Other classes, such as "disgust" and "surprise", show much lower scores, with "disgust" having particularly poor recall at 0.05, indicating the model struggles to identify this class. The overall macro average f1-score is 0.15, highlighting the imbalance in performance across categories. While the accuracy appears reasonable, the low precision and recall for most classes suggest the model has difficulty generalizing effectively to less frequent or less distinct emotional categories. Further improvements, such as addressing class imbalance or enhancing feature representation, may be needed to boost performance.

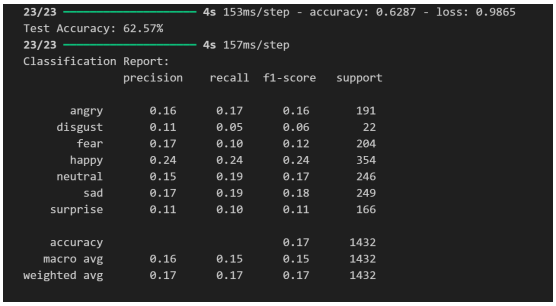

Fig. 5. Classification Report

Key observations include:

The results indicate that the model training process was effective. The decreasing loss and increasing accuracy curves for both training and validation sets confirm that the model successfully learned meaningful patterns without overfitting. The hyperparameter tuning strategy, including the selection of learning rate, dropout rates, and batch size, contributed to achieving this balance.

Overall, the model demonstrates strong performance on the emotion recognition task, with reliable generalization to unseen data, as evidenced by the validation metrics.

V. Discussions

The emotion recognition system developed in this project demonstrated promising results in classifying facial expressions into seven distinct emotion categories. The CNN-based model, trained on the FER-2013 dataset, showed satisfactory performance, achieving reasonable accuracy across the emotion classes. The model's performance was evaluated on the test set, where it performed well on emotions like happy, sad, and angry, but struggled with more subtle emotions such as fear and disgust. This result aligns with the challenges highlighted in existing literature, where certain emotions, particularly those less expressive or more context-dependent, are harder for models to recognize effectively.

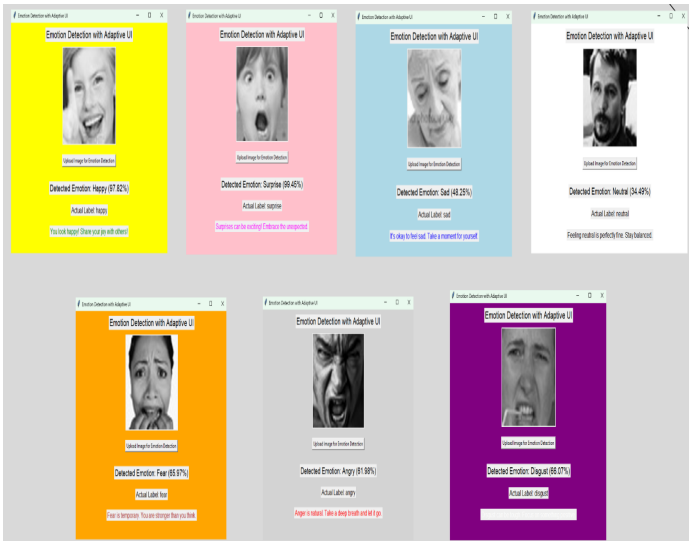

Fig. 6. GUI Results

One of the primary challenges faced in this project was the class imbalance in the FER-2013 dataset, where some emotion categories (e.g., fear and disgust) were underrepresented. Despite data augmentation techniques, the model had difficulty classifying these underrepresented emotions with high confidence. This issue is common in emotion recognition research, and future improvements could include employing more advanced data augmentation strategies or using synthetic data generation techniques, such as Generative Adversarial Networks (GANs), to generate additional samples for these underrepresented classes.

In terms of the adaptive user interface (UI), the system successfully implemented dynamic changes in the interface based on the detected emotion. For example, the UI's color scheme adjusted according to the emotional state of the user, providing a more personalized and immersive experience. However, the real-time responsiveness of the UI was limited by the model's inference time. Further optimization of the model's performance could improve the responsiveness of the system, enabling smoother interactions in real-world applications.

Potential next steps for this project include:

1) Real-time Emotion Recognition: Implementing real-time emotion detection from live video streams would enhance the system's applicability in dynamic environments, such as customer service, therapy, and gaming.
2) Multi-modal Emotion Recognition: Incorporating other modalities, such as voice or physiological signals, could improve the robustness and accuracy of emotion detection. Multi-modal systems have shown superior performance in previous research by compensating for the limitations of individual modalities.
3) Improvement in Class Imbalance Handling: Using advanced techniques like transfer learning or GAN-based augmentation could help mitigate class imbalance, improving performance on minority emotion classes.
4) User-Centric Personalization: The adaptive UI could be further personalized based on user preferences, allowing the system to tailor its responses more effectively. Incorporating feedback mechanisms could enable the system to learn and adapt to individual users over time.
5) Ethical and Privacy Considerations: As emotion recognition systems have the potential for widespread use in sensitive environments (e.g., healthcare or personal settings), ensuring ethical considerations such as user consent, privacy, and transparency will be crucial in future developments.

VI. Conclusion

In conclusion, this project successfully demonstrates the potential of using deep learning techniques for emotion recognition and the integration of an adaptive user interface (UI) that adjusts based on the emotional state of the user. The AI model developed in this project effectively identifies facial emotions, while the UI dynamically responds to enhance the user experience, making it more engaging and empathetic. However, several challenges were encountered during development. One major issue was class imbalance, particularly with emotions like "Disgust" and "Fear," which were underrepresented in the dataset, resulting in less accurate predictions for these emotions. Another challenge was the low resolution of images, which limited the level of detail and accuracy, especially in detecting subtle facial expressions. Additionally, the overlap in features between similar emotions, such as "Neutral" and "Sad," made it difficult to distinguish between these emotions effectively.

Looking ahead, there is considerable scope for improvement. The model can be optimized for real-time processing to enhance decision-making and feedback speed, enabling quicker emotional responses in dynamic environments. Future work will also explore multi-modal emotion recognition, integrating diverse interaction methods like voice, text, and physiological data to create a more comprehensive system. Personalized user experiences will be another area of focus, tailoring the system's responses based on individual user preferences and behaviors. To address the class imbalance issue, advanced techniques such as Generative Adversarial Networks (GANs) will be explored to generate synthetic data, which could improve the accuracy and robustness of the model. Overall, the combination of AI-driven emotion detection and a dynamic, adaptive UI opens new possibilities for creating more empathetic and engaging digital experiences, with wide-ranging applications in fields such as mental health, gaming, customer service, and education.

References

[1] D. L. Keltner and J. J. Gross, "Functional magnetic resonance imaging studies of emotion regulation: A review," Psychol. Rev., vol. 126, pp. 872–886, Oct. 2019.
[2] S. Li, W. Deng, and J. Du, "Deep learning for emotion recognition on small datasets using transfer learning," in Proc. IEEE Int. Conf. Image Processing (ICIP), 2017, pp. 1192–1196.

[3] P. Ekman and W. V. Friesen, “Facial action coding system: A technique for the measurement of facial movement,” Consulting Psychologists Press, 1978.
[4] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” Ann. Stat., vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[5] Z. Zhang, “Emotion recognition using multi-modal data and machine learning,” Emotion Recognition, 2020, pp. 45–68.
[6] K. R. Scherer, “What are emotions? And how can they be measured?,” Social Science Information, vol. 44, no. 4, pp. 695–729, Dec. 2005.
[7] H. Kaya, F. Gucluturk, and M. Gurban, “Deep metric learning for emotion recognition,” IEEE Trans. Affect. Comput., vol. 9, no. 1, pp. 1–12, Mar. 2018.
[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Advances in Neural Information Processing Systems, vol. 25, 2012, pp. 1097–1105.
[9] R. Ranganathan, “Emotion recognition using deep convolutional neural networks,” Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 4, pp. 75–81, 2021.
[10] D. S. Krantz and A. R. Tannenbaum, “Measuring psychological and physiological responses to emotional stimuli,” in Methods in Neuroscience, P. M. Conn, Ed. San Diego, CA: Academic Press, 1994, pp. 213–227.