Emotion Detection by Facial Recognition using CNN

Meet Vora
meetashishvora@usf.edu

Saurabh Verma
vermas12@usf.edu

Tushar Chotlani
tchotlani@usf.edu

Vinesh Sangoi
sangoiv@usf.edu

I. INTRODUCTION

In contemporary society, understanding and supporting mental health has become imperative. However, accurately discerning an individual's emotional state, particularly in scenarios with limited face-to-face interactions, poses a challenge. To address this, our project aims to develop an emotion detection system utilizing computer vision techniques, specifically targeting expressions like anger, disgust, fear, happiness, sadness, surprise, and neutrality. Leveraging facial recognition technology and deep learning algorithms, this system will analyze facial expressions captured through standard cameras or webcam feeds. The resulting tool will find applications in mental health support platforms, virtual communication tools, and customer feedback analysis systems, facilitating timely interventions, enhancing emotional well-being, and fostering empathetic interactions.

II. DATASET

The FER2013 dataset serves as a fundamental resource for training computer systems to recognize human emotions through facial expressions, a pivotal task in the field of affective computing. Despite extensive research and algorithmic advancements, achieving high accuracy rates remains challenging, with top-performing models reaching approximately 65-75%, comparable to human performance. This dataset comprises nearly 36,000 grayscale images of faces, uniformly resized to 48x48 pixels.
However, the dataset is characterized by an inherent imbalance due to its representation of seven distinct facial expressions. Specifically, it encompasses images categorized as Angry (4,953), Disgust (547), Fear (5,121), Happy (8,989), Sad (6,077), Surprise (4,002), and Neutral (6,198).

Fig. 1. Sample image of each emotion.

III. DATA PREPARATION

1) ***Data Cleaning**:* Data cleaning procedures were rigorously applied to uphold dataset integrity and quality. Initially, acceptable image extensions, namely 'jpeg', 'jpg', and 'png', were defined to serve as criteria for file eligibility. The dataset directory was systematically traversed to explore all files, facilitating thorough inspection. Each file underwent individual examination to assess its suitability for inclusion based on predefined standards. Utilizing the imghdr.what() function, file types were verified to retain only valid image files for further processing. Non-conforming files, identified by extensions not present in the predefined list, were subsequently removed to ensure consistency. Robust error handling mechanisms were implemented to address any exceptions encountered during file processing, with detailed error messages logged for reference. These cleaning procedures were pivotal in enhancing dataset quality, reliability, and usability, ensuring that only valid and conforming image files were retained for subsequent research and analysis purposes.

2) ***Data Analysis:*** For data analysis, a custom function was defined to count the number of files, assumed to be images, within each subdirectory of a given directory. This function iterates over each item in the directory, constructs the full path to the item, and checks if it is a subdirectory. If so, it counts the number of files within that subdirectory and stores the count in a dictionary. Finally, the counts are converted into a DataFrame for easy viewing and analysis, with the set name as the index. The function was applied to both the training and testing directories to generate counts of files within their respective subdirectories. These steps provided valuable insights into the distribution of files across different categories, facilitating further analysis and model development.

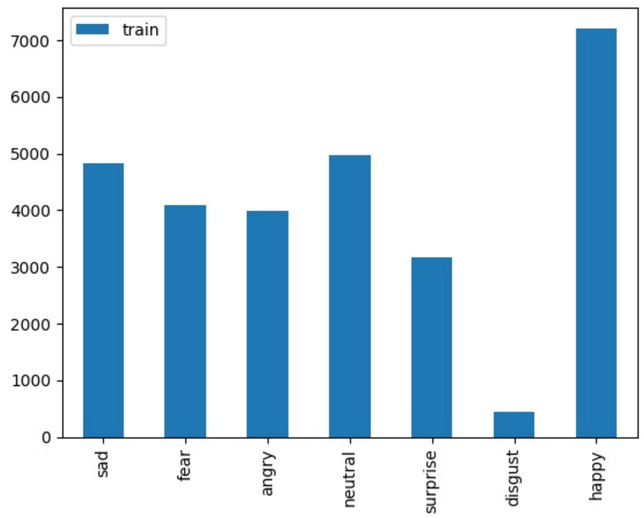

Fig. 2. Data Distribution in train dataset.

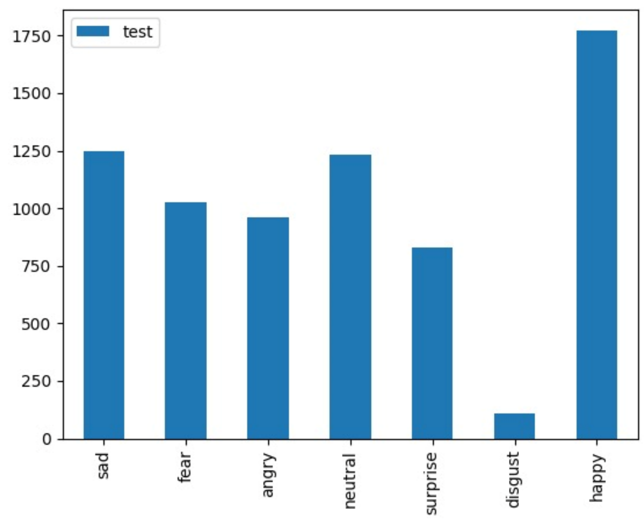

Fig. 3. Data Distribution in test dataset.

IV. MODEL ARCHITECTURE AND TRAINING PROCESS

1) *Model 1: Custom CNN from Scratch*

The custom convolutional neural network (CNN) is designed specifically for the task of facial expression recognition. It starts by initializing image generators for both training and testing datasets. These generators preprocess images, rescaling pixel values, and splitting the data for training and validation. The CNN architecture comprises convolutional layers followed by activation functions, batch normalization, max-pooling, and dropout layers to prevent overfitting. After flattening, dense layers are added, with the final layer outputting probabilities for each emotion class. The model is compiled with appropriate loss functions and metrics, and callbacks are implemented for model checkpointing, early stopping, and learning rate adjustment.

2) *Model 2: Image Augmentation*

For enhanced model generalization, an image augmentation technique is employed. This technique involves generating augmented images from the original dataset by applying random transformations such as rotation, shifting, and flipping. The augmented images are then fed into the same custom CNN architecture as before, with slight modifications to accommodate the augmented input data. The model is trained using the augmented dataset, and its performance is evaluated on both the training and testing sets.

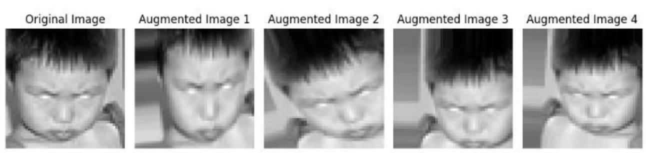

Fig. 4. Image Augmentation.

3) *Model 3: Transfer Learning with VGGNET*

Transfer learning using the VGG16 architecture pretrained on the ImageNet dataset is leveraged for facial expression recognition. The VGG16 base model is loaded and fine-tuned by adding custom dense layers on top. The pretrained layers are frozen, except for the last few, to adapt the model to the specific task. The model is compiled, and callbacks are implemented for training monitoring and optimization. The performance of the transfer learning model is evaluated on both training and testing datasets.

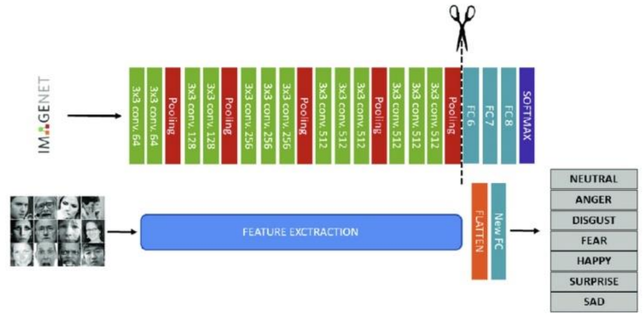

Fig. 5. VGG-16 Deep Convolutional Neural Network architecture.

4) *Model 4: Transfer Learning with ResNet50*

Similarly, transfer learning is applied using the ResNet50 architecture, with adjustments made to accommodate the ResNet50V2 base model. The model is fine-tuned by freezing most of the pretrained layers and adding custom dense layers for emotion classification. Class weights are introduced to handle data imbalances, and callbacks are utilized for model checkpointing and optimization. The transfer learning model's performance is evaluated, and its effectiveness in facial expression recognition is assessed based on training and testing results.

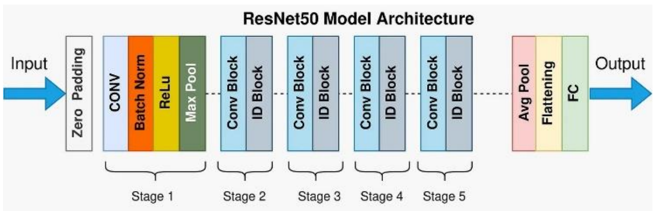

Fig. 6. ResNet50 model architecture.

V. RESULTS

In this research study, we explored multiple approaches to emotion recognition using different neural network architectures and training methodologies. We began by developing a custom CNN from scratch, achieving a final training accuracy of 65.20% and a validation accuracy of 56.60%. To enhance the model's robustness, we then applied image augmentation techniques, resulting in a final training accuracy of 57.21% and a validation accuracy of 57.87%. Next, we employed transfer learning with VGGNet, utilizing pre-trained weights from ImageNet. Despite the promising initial features captured by VGGNet, the model achieved a final training accuracy of 55.42% and a validation accuracy of 55.00%. Lastly, we explored transfer learning with ResNet50, leveraging its deep architecture and residual connections. This approach yielded a final training accuracy of 61.86% and a validation accuracy of 60.74%. Through these experiments, we observed that while the custom CNN showed the highest training accuracy, its validation performance was outperformed by the ResNet50 model, suggesting the potential of deeper architectures and transfer learning for emotion recognition tasks. These findings contribute to advancing the understanding of deep learning techniques for emotion analysis and provide insights into optimizing model architectures for real-world applications.

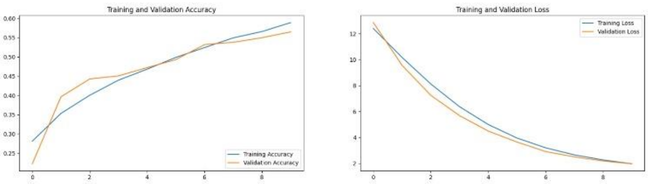

Fig. 7. Results for Custom CNN.

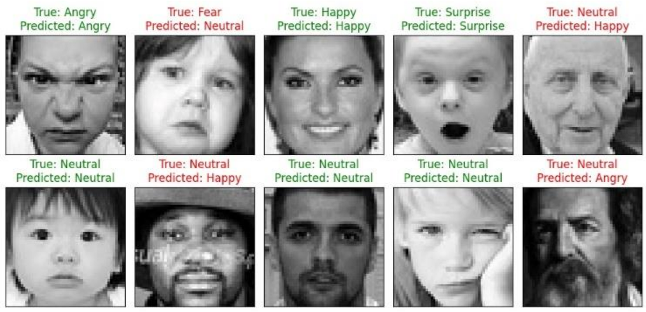

Fig. 8. Model Prediction.

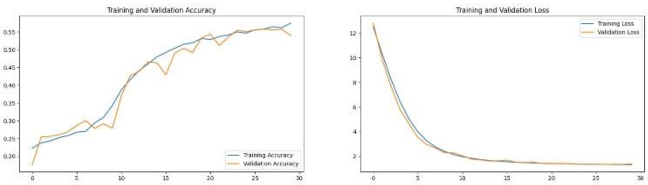

Fig. 9. Results for Image Augmentation

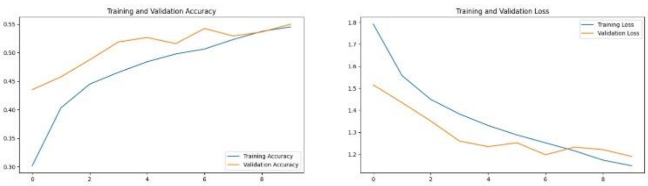

Fig. 10. Results for Transfer Learning with VGGNET.

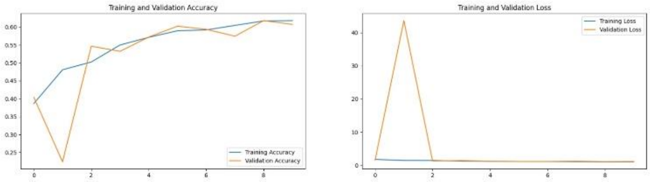

Fig. 11. Results for Transfer Learning with ResNet50

VI. CONCLUSION

In conclusion, our research endeavors aimed at developing an emotion detection system using computer vision techniques, with a focus on facial expressions recognition, carry significant implications for various domains. Through a meticulous exploration of different neural network architectures and training methodologies, we achieved notable results. Our findings indicate that while custom CNNs exhibit high training accuracy, their validation performance may be surpassed by models employing transfer learning with deeper architectures like ResNet50. Furthermore, augmenting data through techniques such as image augmentation enhances model generalization. Despite the challenges posed by imbalanced datasets like FER2013, our study underscores the potential of deep learning in advancing emotion recognition technology. These insights hold promise for applications in mental health support platforms, virtual communication tools, and customer feedback analysis systems, ultimately fostering empathetic interactions and enhancing emotional well-being in contemporary society.

REFERENCES

[1] https://cs230.stanford.edu/projects_winter_2020/reports/32610274

[2] Pramerdorfer, C., Kampel,M.: Facial expression recognitionusing convolutional neural networks: state of the art. Preprint arXiv:1612.02903v1, 2016

[3] Z.Zhang,P.Luo,C.-C.Loy,andX.Tang,"LearningSocialRelation TraitsfromFaceImages,"inProc.IEEEInt.ConferenceonComputer Vision (ICCV), 2015, pp. 3631–3639.

[4] WangJ., and MbuthiaM.,"FaceNet: Facial Expression Recognition Based on Deep Convolutional Neural Network", 2018

[5] QuinnM., SivesindG., andReisG., "Real-time Emotion Recognition From Facial Expressions", 2017

[6] Minaee S., Abdolrashidi A., "Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network", 2019