# Facial Expression Detection Using CNN Model

Justin Javier

*Department of Science, University of Western Ontario*

jjavier3@uwo.ca

***Abstract*— This project aims to develop a Convolutional Neural Network (CNN) model for facial emotion detection. The dataset utilized is the FER2013 dataset from Kaggle, consisting of approximately 40,000 48x48 grayscale images of seven categories: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise. The project involves data collection, preprocessing, model construction, training, evaluation, and testing. Through the implementation of CNN architecture and training techniques, the model endeavors to accurately classify facial expressions. The evaluation involves analyzing the model's performance metrics and examining its capability to generalize to unseen data. The findings provide insights into the challenges and opportunities in facial emotion detection using deep learning techniques.**

## I. Introduction

Facial emotion detection has significant applications in various fields, including human-computer interaction, psychological research, and affective computing. However, achieving accurate emotion recognition from facial expressions poses challenges due to the complexity and variability of human emotions. This project addresses the need for robust and efficient facial emotion detection systems by leveraging deep learning techniques. By training a CNN model on a diverse dataset of facial images, this project aims to develop a reliable system capable of recognizing different emotional states accurately. The outcomes of this research contribute to advancements in emotion recognition technology, with implications for improving human-machine interaction and the partial subjectivity of facial expressions.

## II. Background and Related Work

Previous research in facial emotion detection has explored various methodologies, including traditional machine learning algorithms and deep learning architectures. Early approaches relied on handcrafted features and classifiers to recognize facial expressions. However, with the advent of deep learning, CNNs have emerged as powerful tools for image analysis tasks, including emotion recognition. Recent studies have demonstrated the effectiveness of CNNs in automatically learning discriminative features from raw pixel data, leading to improved performance in facial emotion detection. The FER2013 dataset has been widely used in related research, providing a benchmark for evaluating emotion recognition models. While significant progress has been made in this field, challenges such as dataset bias, class imbalance, and subjective labeling remain areas of ongoing investigation.

## III. Methods

### *A. Research Objectives*

- To develop a CNN model for facial emotion detection.
- To train the model on the FER2013 dataset.
- To evaluate the model's performance metrics, including accuracy and loss.
- To analyze the model's ability to generalize to unseen data.

### *B. Research Methodology*

#### *1) Data Collection and Preprocessing:*

The data collection process involves obtaining the FER2013 dataset, which contains grayscale images of facial expressions labeled with seven emotion categories. The ImageDataGenerator from the Keras library was utilized to preprocess the images for training and testing. This included operations such as rescaling pixel values to the range [0, 1], rotations, shears, zooms, and a horizontal flip for data augmentation. The flow_from_directory method was used to load the images from the specified training/testing data directories and organize them into batches of 64. This preprocessing step ensures that the images are appropriately prepared for input into the CNN model, enhancing its ability to learn meaningful features from the data.

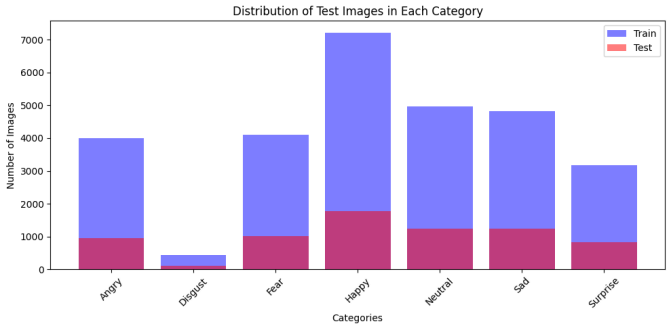



Fig. 1 Distribution of the images in the training/test data for each category. Note the large amount of Happy images, and the low amount of Disgust images

Fig. 1 shows the data we collected and the distribution of images in their respective categories. Even before training the model, the data provided suggests that our model may have a harder time recognizing disgust images, and may overfit features by incorrectly labeling them Happy. For a better performing model with higher accuracy, it would be more ideal to use a dataset with a relatively even distribution.

### *2) Model Construction:*

The model construction phase involves designing the architecture of the CNN model using Keras layers. We constructed a sequential model consisting of convolutional layers, max-pooling layers, dropout layers, and dense layers. Each convolutional layer extracts features from the input images, while max-pooling layers reduce spatial dimensions to capture essential information. Dropout layers mitigate overfitting by randomly dropping units during training. The final dense layer with a softmax activation function outputs probabilities for each emotion category. This sequential architecture is a standard approach for building CNN models for image classification tasks. For better model performance, we could try adding more convolutional and max-pooling layers. In Fig. 2, the overall layout can be visualized:

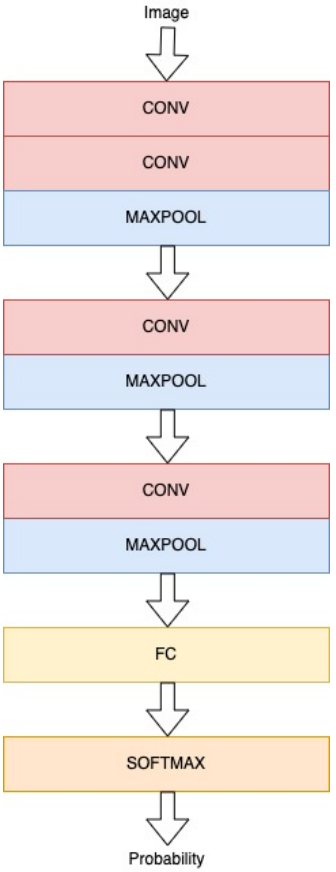



Fig. 2 Architecture of our chosen model. It takes an image, applies multiple convolutional layers to it, and max-pools to capture important features.

### *3) Model Training:*

The model training process involves feeding the preprocessed data into the constructed CNN model and optimizing its parameters to minimize the loss function. The model was compiled using the adam optimizer and categorical cross entropy loss function. Additionally, we used 30 epochs for training the model. With more time and space, a larger epoch number could be used. During training, the model iteratively updates its weights based on the gradient descent algorithm, aiming to improve its performance on the training data. Early stopping was implemented as a callback to prevent overfitting by monitoring the validation accuracy and stopping training if it does not improve for a specified number of epochs (patience=5). In our case, since the epoch amount is low, there was no need for early stopping. This training methodology

ensures that the model learns to generalize well to unseen data and mitigates the risk of overfitting.

### *4) Model Evaluation:*

The evaluation process involves assessing the performance of the trained model using validation data and analyzing its accuracy and loss metrics. The model was trained using the fit method and monitored for both training and validation accuracy over the specified number of epochs. Fig. 3 below illustrates the progression during training.

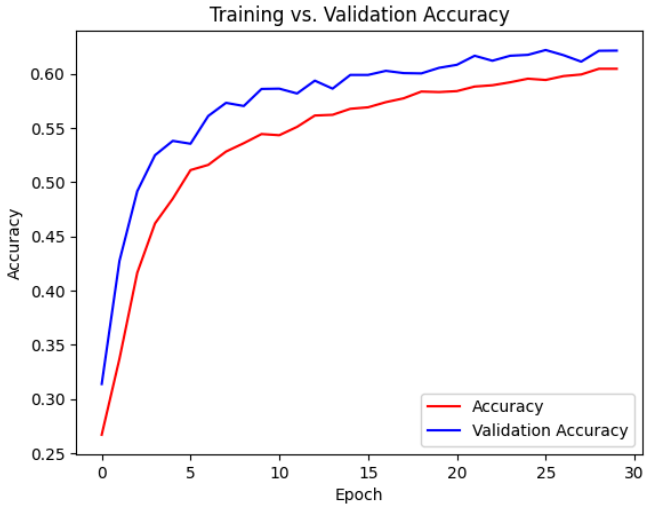



Fig. 3 Training vs. Validation Accuracy during model training.

Additionally, we want to evaluate the model's classification performance on the validation set. To this, we can use a confusion matrix to easily visualize its strengths and weaknesses.

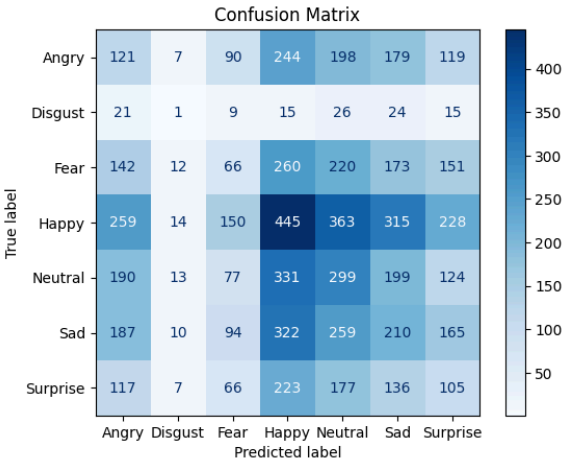
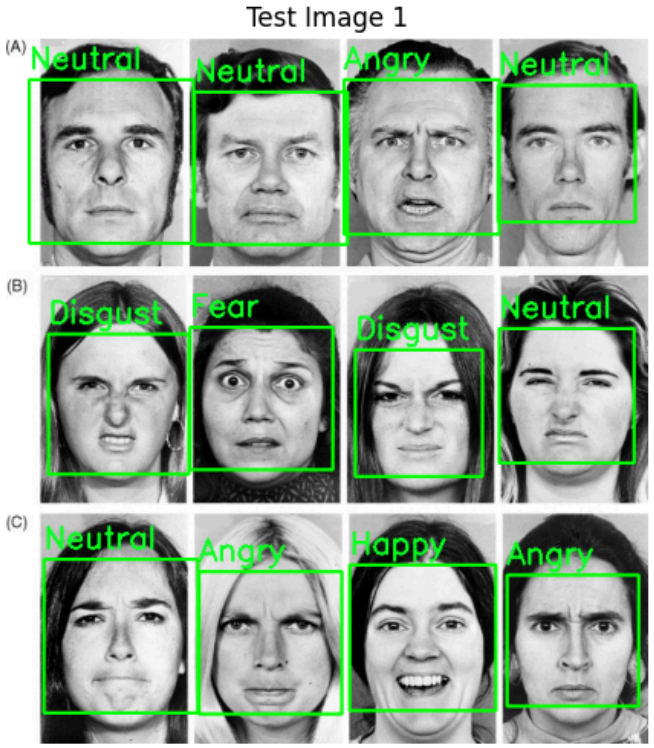



Fig. 4 Confusion matrix showing our results when applying the test set to our model. Disgust performs poorly as expected, and Happy overfits many other categories' images.

This evaluation methodology provides insights into how well the model distinguishes between different emotion categories and identifies areas where it may require further refinement.

### *5) Testing:*

The testing process involves assessing the model's performance on external images to evaluate its real-world applicability. We implemented code to preprocess and classify facial expressions in test images using the trained model, and the Haar Cascades algorithm for face detection in random test images. The model was applied to predict emotion labels for detected faces in our new test images. Though we had no ground truth labels for these images pulled from the Internet, there are cases where a facial expression is clearly classified incorrectly, or is too ambiguous to hard label. Regardless, this testing methodology provides a practical assessment of the model's effectiveness and identifies potential areas for improvement in future iterations.



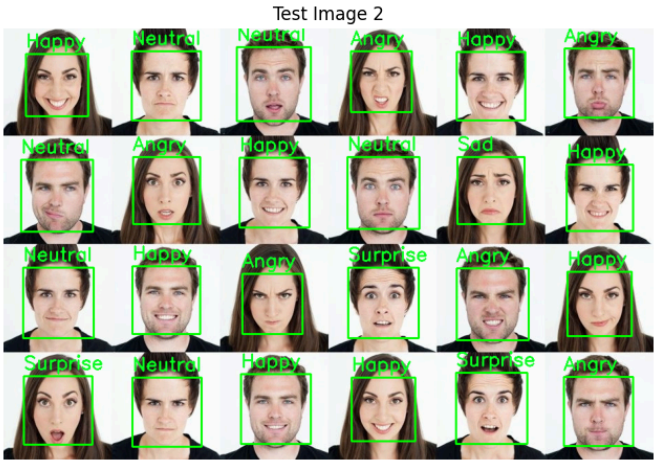



Fig. 5 & 6 Sample internet images fed to our model. The majority are labelled correctly, but there are a few faces with expressions that can be debated over. Some of the faces are labelled incorrectly.

## IV. Results

### *1) Training and Validation Accuracy:*

During the training process, the CNN model achieved a training accuracy of 60.47% and a loss of 1.0447, while the validation accuracy reached 62.15% with a loss of 1.0121. Fig. 3 illustrates the progression of training and validation accuracy over the 30 epochs, showcasing the model's learning behavior over time. These metrics indicate the model's ability to learn and generalize from the training data while maintaining acceptable performance on unseen validation data.

### *2) Confusion Matrix Analysis:*

The confusion matrix, depicted in Fig. 4, provides a comprehensive overview of the model's classification performance on the validation set. It highlights the distribution of predicted labels compared to the ground truth labels across different emotion categories. Notably, the model demonstrates challenges in accurately classifying images in certain categories, such as Disgust, where performance is notably poor. This analysis enables the identification of areas where the model may require further refinement or additional training data to improve its classification accuracy.

### *3) Testing on External Images:*

The model's real-world applicability was assessed by testing it on external images obtained from the internet. By preprocessing and classifying facial expressions in these test images, the model's ability to generalize to unseen data was evaluated. Despite the absence of ground truth labels for these images, visual inspection revealed instances where the model's predictions aligned with human judgment, as well as cases of misclassification or ambiguity. This testing methodology provides practical insights into the model's performance and highlights areas for potential improvement in future iterations.

Overall, the findings from the project contribute to our understanding of facial emotion detection using CNNs and provide a basis for further research and development in this domain.

## V. Conclusion

This project underscores the potential of convolutional neural networks (CNNs) in the domain of facial emotion detection. While the achieved validation accuracy of approximately 62% signifies a commendable performance, it also highlights areas for further refinement and optimization. The findings affirm the importance of preprocessing techniques, model architecture, and training methodologies in enhancing the accuracy and robustness of emotion recognition systems. By leveraging advancements in deep learning and computer vision, future research endeavors can strive to push the boundaries of facial emotion detection, ultimately leading to more accurate and reliable systems.

Looking ahead, several avenues for future work emerge from this project. Firstly, addressing dataset biases and augmenting the training data with a more diverse range of facial expressions could significantly improve the model's generalization capabilities. Additionally, exploring advanced model architectures, such as attention mechanisms or multi-task learning frameworks, may yield enhancements in emotion recognition accuracy and efficiency. Furthermore, efforts to enhance the interpretability of CNN models through techniques like attention visualization or feature attribution methods can provide deeper insights into the decision-making processes of these models. Finally, the application of facial emotion detection technology in real-world scenarios, such as healthcare, education, and human-computer

interaction, warrants further investigation to assess its practical utility and impact on societal well-being. Through collaborative interdisciplinary research efforts, the field of facial emotion detection can continue to evolve and address the complex challenges inherent in understanding human emotions through computational means.