Facial Expression Recognition

4AI3 Project Final Report

Yuelin Wen
Bachelor of Technology – Software Engineering Tech.
McMaster University
Hamilton, Canada
weny36@mcmaster.ca

Ivan Chan
Bachelor of Technology – Software Engineering Tech.
McMaster University
Hamilton, Canada
chani26@mcmaster.ca

Yijing Yang
Bachelor of Technology – Software Engineering Tech.
McMaster University
Hamilton, Canada
yangy492@mcmaster.ca

Abstract - Facial expression recognition is a type of image detection system used to analyze human emotions by classifying facial expressions through learning in deep neural networks. This system uses Convolutional Neural Networks (CNNs) to train the model using pre-collected facial image datasets, leveraging deep learning to extract facial features and accurately identify human expressions.

I. INTRODUCTION

The concept of technology capable of perceiving human emotions, such as smartphones detecting users' feelings, virtual assistants responding to emotional cues, or security systems identifying potential threats through facial expression analysis, was considered highly improbable until recent years. The advancements of facial expression recognition (FER) have transformed this vision into reality. With the use of deep learning and computer vision methods, FER is able to recognize and categorize emotions with astounding precision. This project explores the development of a FER using Convolutional Neural Networks (CNNs) as well as a Multilayer Perceptron Model (MLP), a deep learning architecture well suited for image classification tasks. This report details the implementation of the FER system, including data preprocessing, model architecture, training methodology, and evaluation results. Key techniques such as data normalization, one-hot encoding, and dropout layers are utilized to enhance model performance. Through this study, we gain valuable insights into the challenges of machine learning in emotion detection and the practical applications of deep learning in human-computer interaction.

II. IMPLEMENTATION

A. Data Collection

The data for this project was sourced from the FER2013 (Facial Expression Recognition 2013) dataset, which is a widely used benchmark dataset for training and evaluating facial expression recognition. This dataset contains 35888 grayscale images of humans with 7 different types of facial expressions.

The emotion labels of it can be divided into 7 types: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral. (Figure 1.)

Figure 1: 7 types of different emotions in FER2013

The input consists of a space-separated string of pixel values that represent a 48x48 grayscale facial image. Each value in the string corresponds to the brightness level of a single pixel, ranging from 0 (black) to 255 (white). This format allows the image to be efficiently stored and processed as a one-dimensional sequence, while still representing the two-dimensional structure of the original facial image.

B. Data Process

In the data preprocessing process, we performed four key steps to prepare the dataset for training a facial expression recognition model using deep learning:

Loading data: The dataset is loaded into a pandas DataFrame, allowing efficient data manipulation and analysis. It consists of three columns: emotion, pixels, and usage.

Reshape input image pixel data: Since the image pixel data is stored as a space-separated string, it needs to be converted into a 48×48 matrix for CNN input. This transformation ensures the data is in a format suitable for deep learning models.

One-hot encodes the output data of emotion labels: The emotion labels are initially represented as integers (0-6), each corresponding to a specific facial expression. To enable multi-class classification, we convert these integer labels into one-hot encoded vectors, ensuring the model treats each class independently.

Data normalization: To ensure that the pixel data can be adapted to the stability of the deep learning model, we normalize pixel values from [0,255] into [0,1]. This scaling process helps the neural network converge faster and enhances training performance.

C. Model

Approach 1 - CNN Model

The first approach we choose is CNN (Convolutional Neural Network) as our model. CNN is a type of deep learning model specifically designed for processing image data, making it highly effective for facial expression recognition. By leveraging convolutional operations, CNN can extract and classify features of the image of facial expression.

The layer structure of CNN consists of the input layer, convolutional layers, pooling layer, dropout layer, flatten layer, dense layers, and output layer. Initially, the input image (48 x 48 grayscale) is passed through a convolutional layer, where multiple 3x3 filters extract essential facial features. The max pooling layer then reduces the spatial dimensions, retaining only the most significant features.

To prevent overfitting, a dropout layer is applied, randomly deactivating some neurons during training. The flatten layer converts the 2D feature maps into a 1D feature vector, which is then processed by a fully connected (dense) layer with a ReLU activation function to learn complex patterns. Finally, the softmax output layer assigns probabilities to different facial expressions for classification.

Approach 2 - MLP Model

The second approach we choose is MLP as our model. The structure of MLP consists of an input layer, several fully connected hidden layers with ReLU activation, dropout layers and an output layer. The input to the model is a 1D vector of 2304 features corresponding to a 48x48 grayscale image. The hidden layers transform this input through weighted connections and nonlinear activations to train. The dropout layers are used to prevent overfitting and the final output layer uses the softmax activation function to classify the classification of facial expression.

D. Model Training

In our data training process of both models, 60% of the dataset is processed in training, while 40% is reserved for testing. The training process involves batch processing over 30 epochs, iteratively updating the model parameter to improve its ability to recognize facial expressions accurately. We also perform validation using the test set after each epoch. This will help to monitor performance and prevent overfitting.

III. TEST RESULTS

After testing both models, it is clear that CNN outperforms MLP. The highest accuracy rate for our CNN model is 48.99%, achieved using 21 epochs, while the MLP model, at its highest, only has 35.86% accuracy. While this figure is low, it is significantly better than random guessing, which has an accuracy of only 14.29%, as shown by the model when 0 epochs were used. We attained this accuracy by varying the number of epochs and monitoring how accuracy changed. Additionally, we could have boosted our model's accuracy by increasing the convolutional layers or filters and reducing the number of expressions the model could detect to raise the model's score.

IV. CONCLUSION

Completing this project has taught our group the fundamentals of machine learning, such as collecting and preparing data, training the model, and testing the model. While an accuracy of 48.63% is not ideal for a single image, applying this to a video where multiple photos could be captured and processed by the model would allow the program to select the expression that was predicted the most and provide a corresponding confidence rating.

V. REFERENCES

Moorsy, A. (2018) *Facial expression*, *Kaggle*. Available at: https://www.kaggle.com/datasets/ahmedmoorsy/facial-expression (Accessed: 23 March 2025).

Manav (2025) *Convolutional Neural Networks (CNN) in Deep learning*, *Analytics Vidhya*. Available at: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/ (Accessed: 23 March 2025).