Kasetsart University Sriracha Campus

Project Report

Facial Expression Recognition Using Convolutional Neural Networks and PyTorch

Jirayu Choonade , Witchayut Withichai , Sasitorn Sangpet , Nalintorn Sopapiriyatorn , Saowaluk Praphasirisulee

Bachelor of Science Program in Digital Science and Technology

Abstract

This project presents the development of a Facial Expression Recognition (FER) system using Convolutional Neural Networks (CNNs) implemented with the PyTorch framework. The system is trained and evaluated using the FER2013 dataset, which consists of grayscale facial images categorized into seven emotional classes. The proposed model employs deep convolutional layers, batch normalization, dropout regularization, and data augmentation techniques to improve classification performance and generalization. Experimental results demonstrate that the proposed system achieves satisfactory accuracy and robustness in recognizing facial expressions.

1. Introduction

Facial expression recognition is an important research area in computer vision and human-computer interaction. It enables machines to interpret human emotions based on facial features and expressions. With the advancement of deep learning, Convolutional Neural Networks have become a dominant approach for image-based recognition tasks. This project focuses on developing a CNN-based FER system using PyTorch and evaluating its performance on a standard benchmark dataset.

2. Problem Statement and Objectives

2.1 Problem Statement

Traditional machine learning approaches often struggle with complex facial variations such as lighting conditions, facial poses, and subtle expression differences. Therefore, a deep learning-based approach is required to automatically extract discriminative features from facial images.

2.2 Objectives

The objectives of this project are as follows:

- To design and implement a CNN model for facial expression recognition
- To train and evaluate the model using the FER2013 dataset
- To analyze the performance of the model using standard evaluation metrics

3. Dataset Description

The FER2013 dataset is a widely used benchmark dataset for facial expression recognition. It contains grayscale facial images with a resolution of 48x48 pixels. The dataset is divided into training and testing subsets, with images categorized into seven emotion classes: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral.

4. System Architecture

4.1 Overall Architecture

The proposed system consists of three main components: data preprocessing, CNN-based feature extraction and classification, and model evaluation. The system is implemented entirely in Python using PyTorch.

4.2 System Flow

The system flow begins with loading and preprocessing the FER2013 dataset. The images are normalized and augmented before being fed into the CNN model. The model is trained using labeled data, validated to monitor performance, and finally evaluated on the test dataset.

5. Model Design

5.1 Convolutional Neural Network Architecture

The CNN model consists of multiple convolutional layers with increasing filter sizes, followed by batch normalization, max-pooling, and dropout layers. A global average pooling layer is used before fully connected layers to reduce overfitting. The final output layer uses a softmax-compatible structure for multi-class classification.

5.2 Regularization Techniques

To improve generalization, dropout layers and batch normalization are applied throughout the network. Label smoothing is also incorporated into the loss function to reduce overconfidence in predictions.

6. Data Preprocessing and Augmentation

Data preprocessing includes grayscale normalization and reshaping of images. Data augmentation techniques such as random horizontal flipping, rotation, affine transformation, and random erasing are applied during training to increase data diversity and improve robustness.

7. Model Training

The model is trained using the AdamW optimizer with weight decay to prevent overfitting. A learning rate scheduler is employed to dynamically adjust the learning rate based on validation loss. Early stopping is applied to prevent excessive training when performance no longer improves.

8. Evaluation Metrics

The performance of the model is evaluated using accuracy, precision, recall, and F1-score with weighted averaging to account for class imbalance. Inference time is also measured to assess computational efficiency.

9. Experimental Results

Experimental results show that the model achieves a test accuracy of approximately 67%. The weighted precision, recall, and F1-score indicate balanced performance across emotion classes. The use of data augmentation and class weighting contributes significantly to improved classification performance.

10. Discussion

The results demonstrate that deep CNNs are effective for facial expression recognition tasks. However, misclassification may still occur due to similarities between certain expressions and dataset limitations. Increasing dataset size and incorporating temporal information may further enhance performance.

11. Limitations and Future Work

The current system is limited to static image-based recognition and does not consider temporal dynamics. Future work may include using video-based FER, transfer learning with pre-trained models, or integrating attention mechanisms.

12. Conclusion

This project successfully demonstrates the application of Convolutional Neural Networks for facial expression recognition using PyTorch. The developed system provides a solid foundation for further research and development in emotion recognition and intelligent systems.