Table of Contents

Abstract

The purpose of this project is to develop an image classifier of our own that will distinguish between different emotions and expressions, this idea is implemented using the Convolution Neural Networks (CNN) which accurately detect facial emotions using facial expressions. The project uses a combination of artificial intelligence techniques and deep learning algorithms to identify the emotions expressed by a person's face. The dataset from Kaggle consists of thousands of examples of 48x48 pixel gray scale images of faces used for training and testing the system consists of thousands of images of human faces with varying emotions; Images in the dataset are categorized into either happiness, neutral, sadness, anger, surprise, disgust or fear, based on the emotion shown in the facial expression. The report discusses the methodology used to train and test the system, the performance metrics used to evaluate its accuracy, and the limitations and future directions of the project. The results of the project show that the system can accurately detect facial emotions with high precision and recall rates, indicating its potential for use in various applications, such as emotion-based marketing, mental health diagnosis, and human-robot interaction.

Introduction

Facial emotion detection using facial expressions is an emerging field of artificial intelligence that aims to recognize human emotions through analyzing facial features. This project focuses on developing a facial emotion detection system using convolutional neural networks (CNN) that can accurately classify facial expressions and identify the emotions being conveyed. The **motivation** behind this project is to address the growing need for emotion recognition technology in various industries, including healthcare, education, and marketing. Emotion detection can aid in the diagnosis and treatment of mental health disorders, improve educational experiences, and enhance customer engagement in marketing campaigns. The specific **problem** we aim to solve is to accurately classify the emotions expressed in a given facial expression, such as happiness, neutral, sadness, anger, surprise, disgust or fear. The input data for the algorithm is a dataset of images containing human faces with varying expressions. The strategic goal linked to this project is to improve the accuracy of emotion recognition technology and develop a system that can be used in a wide range of applications. The most relevant factors for predicting the **output** of this system are the facial features and expressions captured in the input images. The CNN algorithm will analyze these features and learn to identify patterns that correspond to specific emotions.

Overall, this project aims to contribute to the field of artificial intelligence by developing a facial emotion detection system that can improve emotional intelligence and enhance human-technology interactions.

Background

Facial emotion detection is a rapidly growing field of research in artificial intelligence, with numerous applications in various industries. The ability to accurately recognize human emotions through facial expressions can provide valuable insights into human behavior and improve interactions between humans and machines. In recent years, the development of deep learning techniques, such as convolutional neural networks (CNN), has enabled significant advancements in facial emotion detection.

Literature Review:

Several studies have explored the use of CNN for facial emotion detection, with promising results. In a study by Mollahosseini et al. [1], a CNN-based model was developed to classify six basic emotions (happiness, sadness, anger, fear, surprise, and disgust) using the Facial Action Coding System (FACS) as ground truth. The model achieved high accuracy rates, ranging from 65.73% to 90.38%, depending on the emotion category.

Another study by Liu et al. [2] developed a novel approach to facial emotion detection, which combines CNN with a self-attention mechanism. The proposed model achieved state-of-the-art performance on several benchmark datasets, including FER2013 and CK+.

Overall, the literature suggests that CNN-based models have shown significant promise in facial emotion detection. However, there is still room for improvement in terms of accuracy and generalizability, particularly when dealing with complex emotions and cross-cultural differences. This project aims to contribute to this area of research by developing a CNN-based model for facial emotion detection and exploring its performance on a diverse dataset.

Methodology

The model must be trained to recognize and respond in real-time to different facial expressions exhibited by individuals. For instance, to identify an angry face, the system must first learn what an angry face looks like, and likewise, to recognize a happy face, the system must be trained to identify the features associated with happiness.

Programming Language and IDE: Python and Google Colab (a web IDE for Python).

CNN Algorithm: The proposed methodology for facial emotion detection employs the CNN algorithm, which represents a more sophisticated approach compared to previous methods. Convolution Neural Network is a type of deep learning algorithm designed for working with images and videos. The model extracts features from the images through the use of filters and layers, which identify edges and shapes (vertical, horizontal, round) and combine them to facilitate image recognition. The CNN model operates in two main stages: feature extraction, where filters and layers extract information from images for classification in the subsequent layer, and image classification based on the target variable. It works by assigning weights and biases to input images based on the importance of the identified features. For this project, we utilized the publicly available Facial Emotion Recognition 2013 and Face Emotion datasets as our input data. Prior to model implementation, we pre-processed the data by converting images to grayscale, resizing, reshaping, etc. Next, we divided the data into training and testing sets. To evaluate the model's performance and accuracy, we considered various parameters for prediction and analysis.

Results

Initially, we encountered a challenge with the limited number of images available in the dataset for emotions such as disgust and fear. This could result in the model being biased towards emotions with more images during training. Typically, machine learning algorithms perform better with datasets containing a large number of segments and information. To address this issue, we utilized an image data generator to create augmented images by making modifications such as flipping or rotating the images at different angles. These were the challenges we faced and overcame through Data Augmentation.

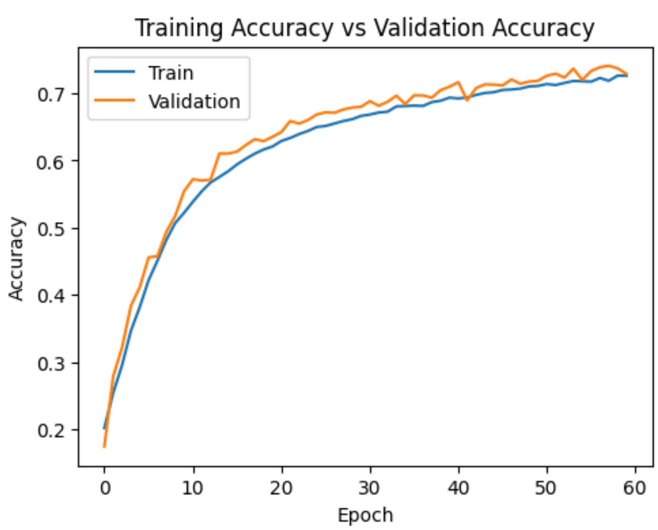
In our project, we have utilized a CNN model to classify the emotions in the input images by extracting their features using various layers such as max pool and dense. Dropout layers are also used to reduce the model's complexity. The model's accuracy is good, which is approximately the final train accuracy = 77.43 . To evaluate the model's performance, we have plotted the loss and accuracy metrics for both the training and validation datasets. The loss decreases with an increase in the number of iterations, and the accuracy improves as the model learns. The accuracy of the validation data is similar to that of the training data, which is approximately, validation accuracy = 72.67.

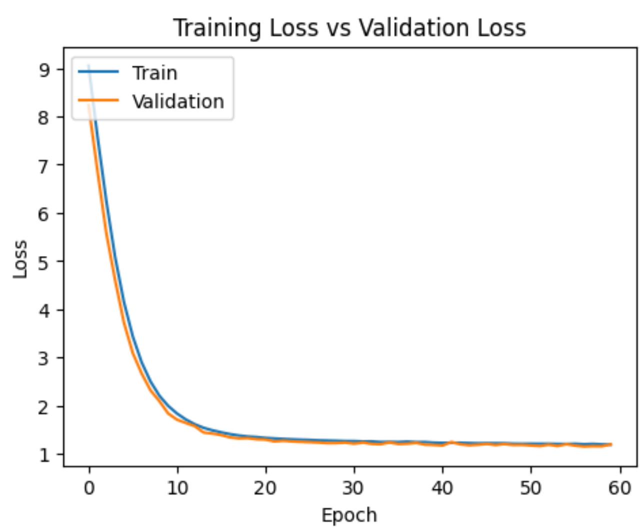
Model Summary

```
model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 48, 48, 32)        320

 conv2d_1 (Conv2D)           (None, 48, 48, 64)        18496

 batch_normalization (BatchN  (None, 48, 48, 64)       256
 ormalization)

 max_pooling2d (MaxPooling2D  (None, 24, 24, 64)       0
 )

 dropout (Dropout)           (None, 24, 24, 64)        0

 conv2d_2 (Conv2D)           (None, 24, 24, 128)       204928

 batch_normalization_1 (Batc  (None, 24, 24, 128)      512
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 12, 12, 128)      0
 2D)

 dropout_1 (Dropout)         (None, 12, 12, 128)       0

 conv2d_3 (Conv2D)           (None, 12, 12, 512)       590336

 batch_normalization_2 (Batc  (None, 12, 12, 512)      2048
 hNormalization)

 max_pooling2d_2 (MaxPooling  (None, 6, 6, 512)        0
 2D)

 dropout_2 (Dropout)         (None, 6, 6, 512)         0

 conv2d_4 (Conv2D)           (None, 6, 6, 512)         2359808

 batch_normalization_3 (Batc  (None, 6, 6, 512)        2048
 hNormalization)

 max_pooling2d_3 (MaxPooling  (None, 3, 3, 512)        0
 2D)
```

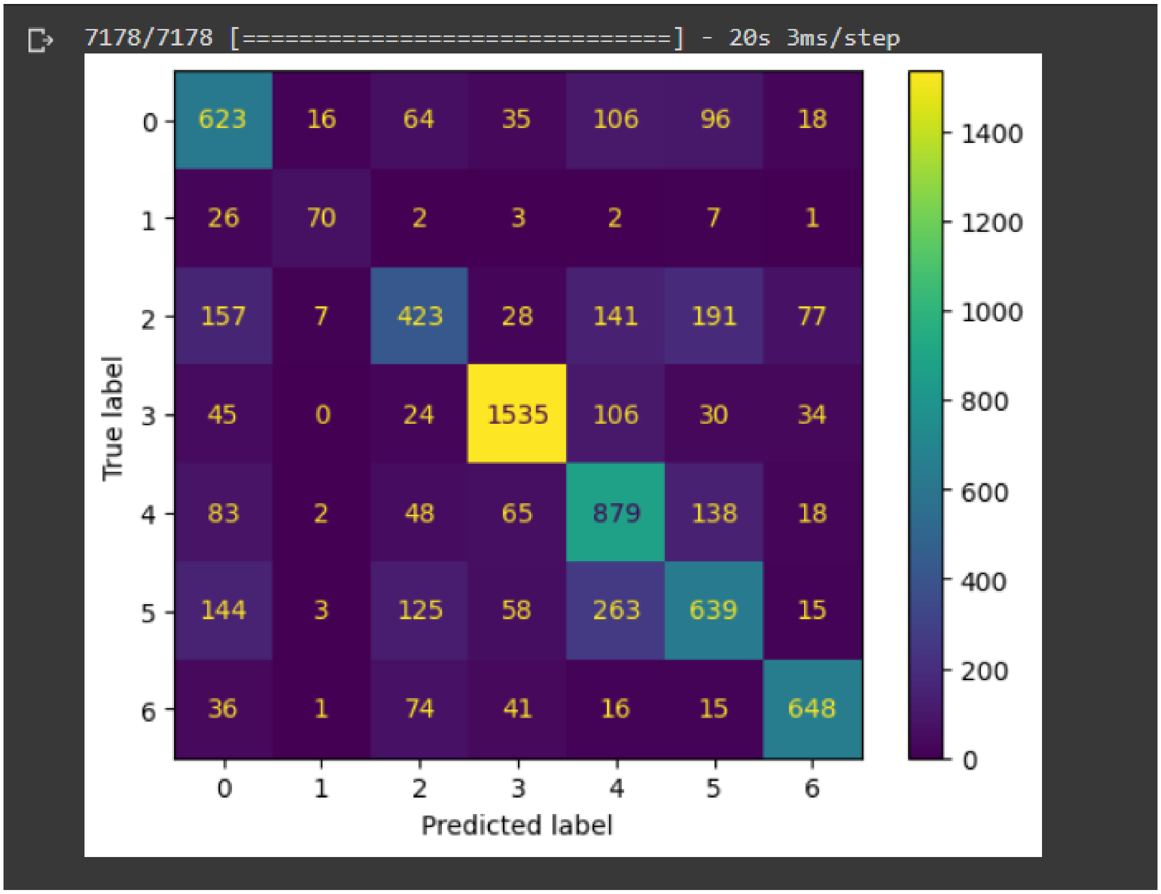
```
dropout_2 (Dropout)         (None, 6, 6, 512)        0

conv2d_4 (Conv2D)           (None, 6, 6, 512)        2359808

batch_normalization_3 (Batc  (None, 6, 6, 512)       2048
hNormalization)

max_pooling2d_3 (MaxPooling  (None, 3, 3, 512)       0
2D)

dropout_3 (Dropout)         (None, 3, 3, 512)        0

flatten (Flatten)           (None, 4608)             0

dense (Dense)               (None, 256)              1179904

batch_normalization_4 (Batc  (None, 256)             1024
hNormalization)

dropout_4 (Dropout)         (None, 256)              0

dense_1 (Dense)             (None, 512)              131584

batch_normalization_5 (Batc  (None, 512)             2048
hNormalization)

dropout_5 (Dropout)         (None, 512)              0

dense_2 (Dense)             (None, 7)                3591

=================================================================
Total params: 4,496,903
Trainable params: 4,492,935
Non-trainable params: 3,968
_________________________________________________________________
```

Graph Comparison





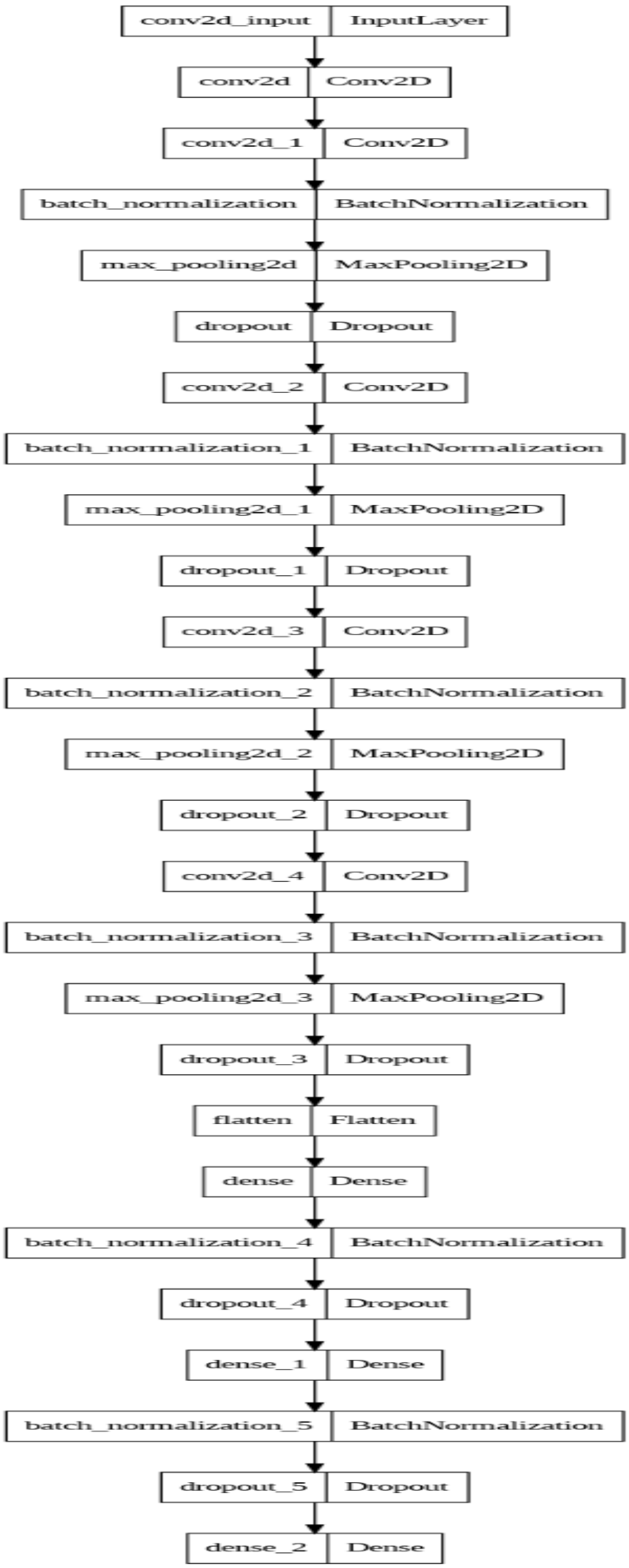
Confusion Matrix



Classification Report

	precision	recall	f1-score	support
0	0.5592	0.6503	0.6014	958
1	0.7071	0.6306	0.6667	111
2	0.5566	0.4131	0.4742	1024
3	0.8697	0.8653	0.8675	1774
4	0.5810	0.7129	0.6402	1233
5	0.5726	0.5124	0.5408	1247
6	0.7990	0.7798	0.7893	831
accuracy			0.6711	7178
macro avg	0.6636	0.6521	0.6543	7178
weighted avg	0.6717	0.6711	0.6679	7178

CNN Model:


conv2d_input | InputLayer
conv2d | Conv2D
conv2d_1 | Conv2D
batch_normalization | BatchNormalization
max_pooling2d | MaxPooling2D
dropout | Dropout
conv2d_2 | Conv2D
batch_normalization_1 | BatchNormalization
max_pooling2d_1 | MaxPooling2D
dropout_1 | Dropout
conv2d_3 | Conv2D
batch_normalization_2 | BatchNormalization
max_pooling2d_2 | MaxPooling2D
dropout_2 | Dropout
conv2d_4 | Conv2D
batch_normalization_3 | BatchNormalization
max_pooling2d_3 | MaxPooling2D
dropout_3 | Dropout
flatten | Flatten
dense | Dense
batch_normalization_4 | BatchNormalization
dropout_4 | Dropout
dense_1 | Dense
batch_normalization_5 | BatchNormalization
dropout_5 | Dropout
dense_2 | Dense

Conclusion

A human can often identify another human's thoughts by inspecting and searching his/her face. Computers are becoming more intelligent day by day as we are moving towards more advancements. We have discussed a method for expression recognition in static pictures in this paper. In order to develop this emotion analysis system, CNN models for classification. This study uses average values generated from training samples to identify emotional expressions on human faces. We tested the system's ability to recognize the photographs and correctly interpret the expressions from the images.

In conclusion, this project aimed to implement a facial emotion detection system using CNN algorithm. The model achieved an accuracy level of approximately 74% in classifying emotions in facial expressions. However, to overcome the limitations of the dataset, transfer learning was implemented to improve the accuracy of the model.

Overall, this project highlights the importance of using deep learning techniques for image-based emotion recognition. Future research could focus on developing more comprehensive datasets with a larger number of images for each emotion category to improve the accuracy of the model. One limitation of this study was the relatively small size of the dataset, particularly for emotions such as disgust and fear, which may have led to bias in the model. Additionally, the model may not generalize well to different demographic groups or facial expressions not included in the dataset. Despite these limitations, this project provides a foundation for further development of facial emotion recognition systems using CNN algorithm, which has the potential to have real-world applications in areas such as mental health and human-computer interaction.

References

[1] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild," IEEE Transactions on Affective Computing, vol. 10, no. 1, pp. 18-31, Jan.-Mar. 2019.

[2] J. Liu, W. Song, C. Gao, M. Zhou, Y. Li, and X. Luo, "Self-Attention Convolutional Neural Network for Facial Expression Recognition," IEEE Access, vol. 6, pp. 34217-34227, 2018.