Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique

Université de Carthage

Ecole Polytechnique de Tunisie

وزارة التعليم العالي و البحث العلمي

جامعة قرطاج

المدرسة التونسية للتقنيات

Computer Vision PROJECT 3 Report

Face Emotion Recognition

Written by :
BARKOUS Hamdi
DAOUD Aymen

Supervised by :
Dr. Walid BARHOUMI

Contents

1 Abstract

Emotion Recognition is a task to process a human facial expression and classify it into certain emotion categories. Such task typically requires the feature extractor to detect the feature, and the trained classifier produces the label based on the feature. The problem is that the extraction of feature may be distorted by variance of location of object and lighting condition in the image. In this project, we address the solution of the problem by using a deep learning algorithm called Conventional Neural Network (CNN) to address the issues above. By using this algorithm, the feature of image can be extracted without user-defined feature-engineering, and classifier model is integrated with feature extractor to produce the result when input is given. In this way, such method produces a feature-location-invariant image classifier that achieves higher accuracy than traditional linear classifier when the variance such as lighting noise and background environment appears in the input image [1] . The evaluation of the model shows that the accuracy of our lab condition testing data set is 94.63%, and for wild emotion detection it achieves only around 37% accuracy. In addition, we also hope to demonstrate our project shown as figure 1

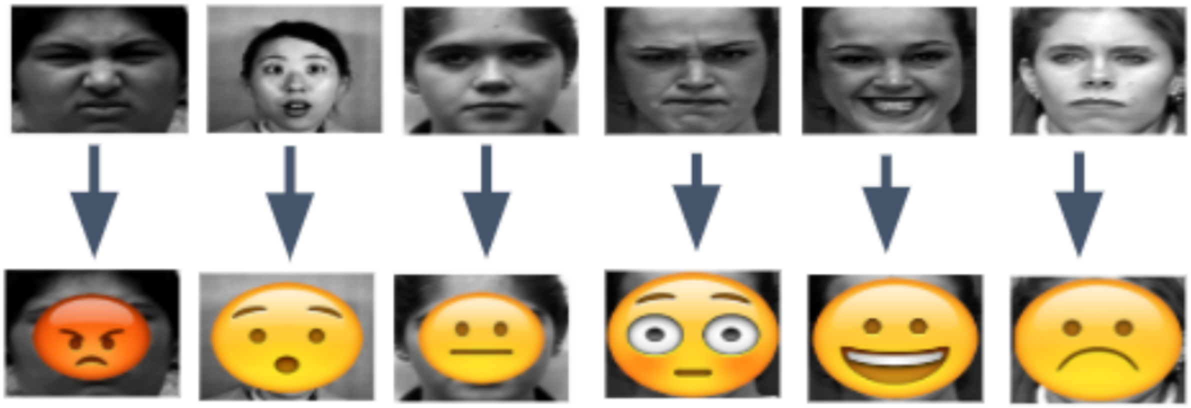


Figure 1: An Example of demonstration that the emojis are imposed on the input images

2 Goals of the Project

The goals of this project are to establish a model that can classify 7 basic emotions: happy, sad, surprise, angry, disgust,neutral, and fear. In addition to this, our project also aims to analyze the results of our model in terms of accuracy for each class. In the future, the model is expected to perform wild emotion recognition that has more complex variance of condition than lab condition images.

3 The General Problem

In real life, people express their emotion on their face to show their psychological activities and attitudes in the interaction with other people. The primary focus of this project is to determine which emotion an input image that contains one facial emotion belongs to. Because human face is complex to interpret, emotion recognition can be specifically divided into classification of basic emotion and classification of compound emotion [8]. For the goals of our project, the essential problem is to focus on the classification of 7 basic emotions(shown at below):

In conclusion, we want to construct a system by which an input image that contains one expression belonging to one of the 7 basic emotions can generate an output that correctly labels the input image.

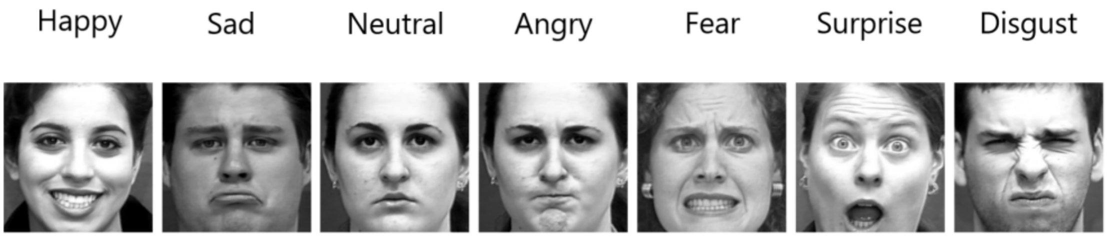


Figure 2: The examples of 7 basic emotions:Happy, Sad, Neutral, Angry, Fear, Surprise, and Disgust

4 Conventional Methods and Their Issues

4.1 A Short Introduction to Conventional Method

Like every other classification problems, the emotion recognition problem requires an algorithm to complete feature extraction and categorical classification .In order to classify an emotion, we need to extract certain feature from data and build an model that can classify the input based on the feature. The procedure can be outlined as following.

4.1.1 Data Pre-processing:

The data pre-processing is to standardize the data. The typical way is to set the mean of the data to 0 and to also divide the data by the standard deviation.

4.1.2 Feature Extraction:

The typical conventional method is to detect the face and extract the Action Units(AU)(shown in figure 3) from the face, and certain emotion contain the combination of AUs code as feature.

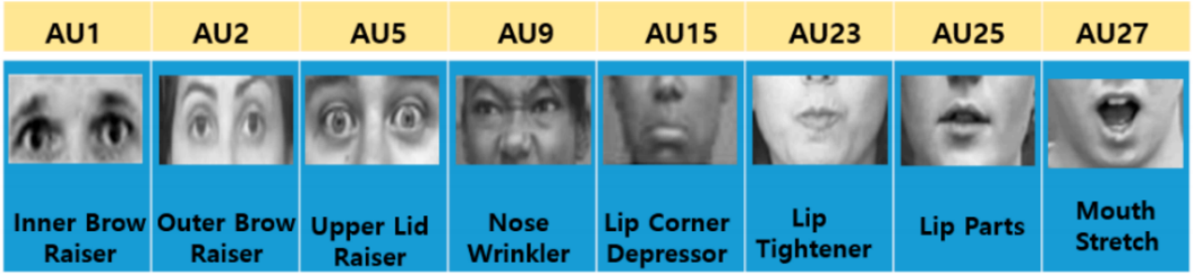


Figure 3: Examples of some Action Units

4.1.3 Model Construction:

The conventional classifier can be either supervised or unsupervised algorithm. A typical example of supervised algorithm is Support Vector Machine, and the examples of unsupervised algorithm include Principle

Component Analysis(PCA) and Linear Discriminant Analysis (LDA

4.2 A Short Introduction to Conventional Method

The issues of Conventional method are :
Variance of Lights Since each image is taken in the completely different background and lighting conditions, the intra-class noise of lights will distort the model to classify the emotion. As results, the same type of emotions may be classified differently because of the effect of lighting noise.
Variance of Location Since the feature is typically extracted by filters such application of Local Binary Pattern the location of the feature, therefore, may affect the functionality of the feature extraction. As results, the AU may be extracted incorrectly if the face has is rotated or is in different part of the image.
These two issues are major problems faced by conventional algorithms.

5 Solution to the Issues of Conventional Method

5.1 Summary of Solution

In order to solve the issues of the conventional methods, the solution is to apply the algorithm of Convolutional Neural Network(CNN) to perform classification of emotion. In contrast to conventional method,the key differences of algorithm are [1] :

5.1.1 Automatic Generation of Feature Extractor:

The feature of images can be captured automatically without users' built feature extractor because the feature extractors are generated in the process of training based on the given ground truth.

5.1.2 Differences of Mathematical Model:

The conventional method is typically performing classification through linear transformation, so they are typically referred as Linear Classifier. In comparison, CNN as well as other Deep Learning algorithm typically combines the linear transformation with nonlinear function such as sigmoid (Logistic Function) and Rectified Linear Unit(ReLU) to distinguish differences in process of classification.

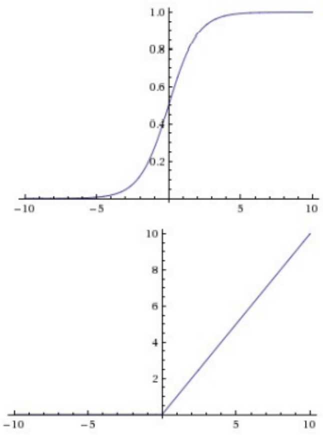


Figure 4: Examples of some Action Units

5.1.3 The Deeper Structure

The conventional method typically performs only one layers of operation: for example SVM only has one set of weights.(shown in equation 1) However, CNN as well as other deep learning algorithm does multiple layers of operation in the process of classification.

$$S = Wxi + b \tag{1}$$

where S is the classification score,W is weights matrix, and b is bias In the following section, we are going to discuss CNN in details and point out its advantages and disadvantages.

5.1.4 Advantages and Disadvantages of CNN

The major advantages of this algorithm are:
A. The feature can be captured regardless of its location
B. The users do not have to design the filters to extract feature as mentioned in summary
C. The negative effects of variance of lights can be reduced because the model is trained to learn the effect of noise.
However, it is apparent that this algorithm has several major issues:
A. The model requires a very large dataset to train because it has to cover the as many situations as possible. However, it is difficult to collect the dataset of emotion.
B. The model takes a very long to train from beginning(2 to 3 weeks)
C. The training requires machine with very good handware, and they are expensive and consumes a large amount of energy.
In the next section, we are going to approach these issues by designing our own solutions.

6 Facial expression classification: our approach

From previous section, we understand CNN does solve the issues of conventional method such as variance of feature location and noise surrounding the face. It, however, has its own requirement in formation of training process. In this session, our own solution will be presented to improve the model to fit the local dataset and constraints.

6.1 Data Augmentation

In this project, data augmentation was used to artificially increase the size of the training dataset by applying random transformations to the images. This can help to improve the performance of the model by making it more robust to variations in the images, such as changes in lighting, rotation, and translation.

The data augmentation was performed using the ImageDataGenerator class from the Keras library. This class allows for easy implementation of a wide range of data augmentation techniques, such as rotation, translation, and flipping.

First, the train and test datasets were reshaped to have the correct format for the data generator. Next, data gen args dictionary was defined with the parameters for the data augmentation, such as rotation range, width shift range, height shift range, shear range, zoom range, horizontal flip, channel shift range, fill mode. The rotation range parameter controls the amount of rotation applied to the images, while the width shift range and height shift range control the amount of translation applied to the images. The shear range parameter controls the amount of shearing applied to the images, while the zoom range parameter controls the amount of zooming applied to the images. The horizontal flip parameter controls whether the images are flipped horizontally, and the channel shift range parameter controls the amount of channel shifting applied to the images. Finally, the fill mode parameter controls the method used to fill in new pixels created by the transformation.

Then, the data generator was fit to the training dataset using the fit() method. After that, random transformations were applied to the training and test datasets using the flow() method, with a batch size equal to the size of the dataset. Finally, the reshape method was applied to the augmented train and test datasets to reshape them to the original format.

Using data augmentation in this way can help to improve the performance of the model by making it more robust to variations in the images, and reducing the risk of overfitting.

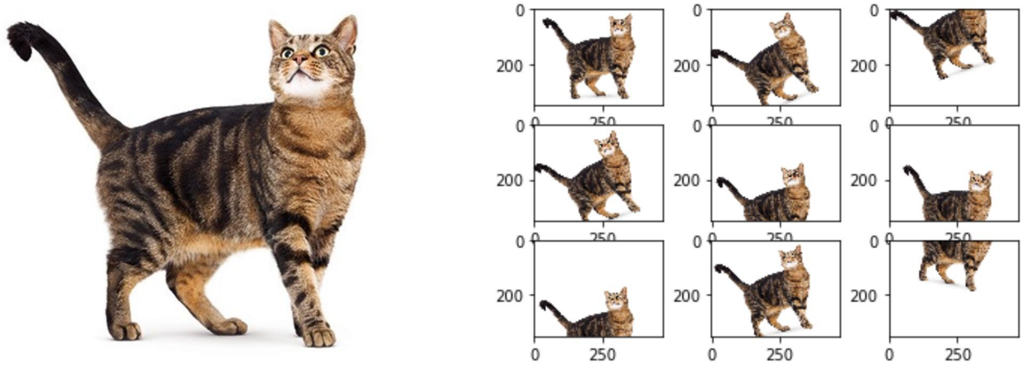


Figure 5: Data Augmentation

6.2 Data Prepocessing: features extraction

In this facial expression recognition project, several feature extraction techniques were used to extract relevant information from the images of faces. These techniques include:

6.2.1 VGGface

VGGface is a pre-trained deep convolutional neural network (CNN) model for facial recognition developed by researchers at the Visual Geometry Group (VGG) at the University of Oxford. The model was trained on a large dataset of over 2.6 million images of faces from over 2,500 different individuals.

The VGGface model is based on the VGG-16 architecture, which is a deep CNN architecture that was originally proposed for image classification tasks. The VGG-16 architecture consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. In VGGface, the last fully connected layers were replaced with a new set of layers that are specifically designed for facial recognition tasks.

The VGGface model is pre-trained on a large dataset of faces, which means that it has already learned to recognize a wide range of facial expressions and features. This pre-training allows the model to extract high-level features from images of faces that are useful for facial recognition tasks. By using the features extracted by the VGGface model as input, the model can learn to recognize different facial expressions more effectively.

VGGface can be fine-tuned to a specific task by training the last fully connected layers on a smaller dataset. This allows the model to learn to recognize specific facial expressions or to adapt to a specific domain.

In this project, the VGGface model was used to extract high-level features from the images of faces. These features represent the overall appearance of the face, including facial features, shapes, and textures. By using VGGface to extract these features, the model can learn to recognize different facial expressions more effectively.

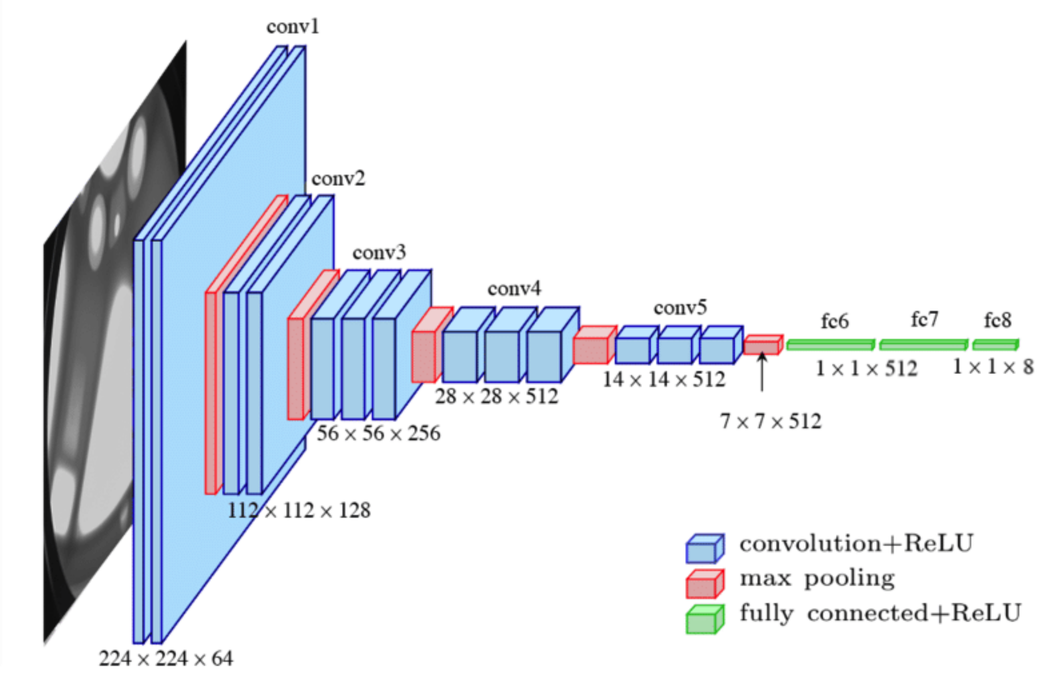


Figure 6: VGGface architecture

6.2.2 Facial landmarks

Facial landmarks are specific points on the face that correspond to important features such as the eyes, nose, and mouth. In this project, facial landmarks were used to extract features from the images of faces. The specific facial landmark detection algorithm used in this project is called the ”shape predictor 68 face landmarks”. This algorithm is a pre-trained model that is based on a technique called facial landmark detection. The model was trained on a dataset of faces and is able to detect 68 specific points on the face that correspond to important facial features. These points are used to define the shape of the face, and the spatial location of these points can be used to extract features that are useful for facial expression recognition.

The facial landmarks features represent the spatial location of important facial features. This information can be useful for recognizing different facial expressions because different facial expressions often correspond to different positions of the facial features. For example, in a happy expression, the corners of the mouth are raised and the eyes are opened wide, while in a sad expression, the corners of the mouth are lowered and the eyes are narrowed.

By using the shape predictor landmarks algorithm to extract facial landmarks, the model can learn to recognize different facial expressions by learning to recognize the spatial location of important facial features. The following figure illustrates a use case of facial landmarks detector:

6.2.3 Local Binary patterns and Histogram of Oriented Gradients

Local Binary patterns (LBPs) are a texture-based feature extraction technique that is commonly used in computer vision. LBPs work by comparing the intensity of a pixel to its surrounding pixels and encoding the result in a binary code. The resulting code, called the LBP value, is then used as a feature to represent the texture of the image. In this project, LBPs were used to extract features from the images of faces. The LBPs features represent the texture of the face, which can be useful for recognizing different facial expressions. Different facial expressions can have different textures, for example a smiling face has wrinkles around the

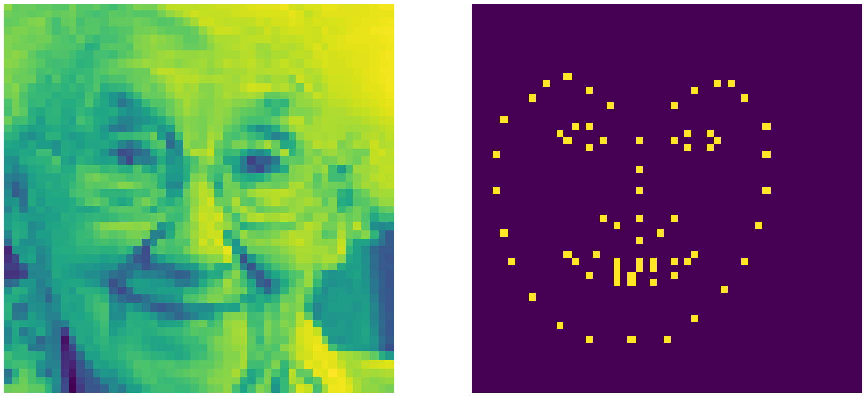


Figure 7: Facial Landmarks Detection

eyes, while a neutral face does not. By using LBPs to extract these features, the model can learn to recognize different facial expressions by learning to recognize the texture of the face.

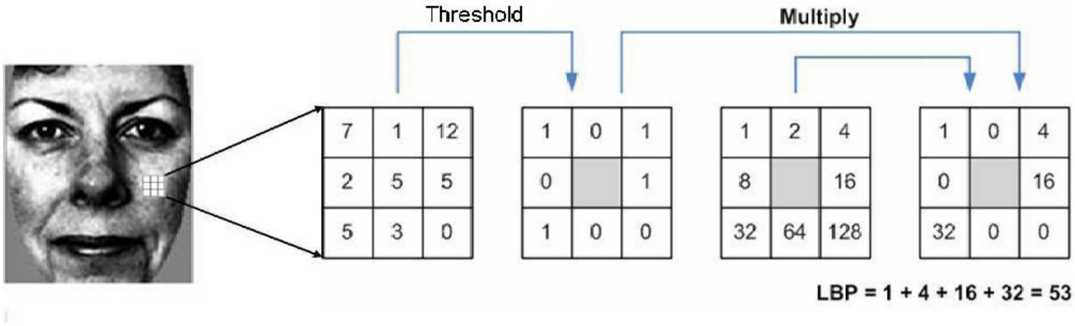


Fig. 1. Example of an LBP calculation

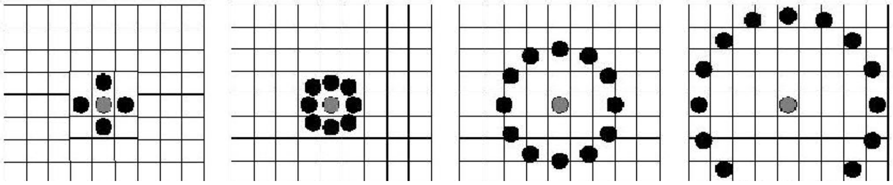


Figure 8: LBPs

Histogram of Oriented Gradients(HOGs) are a feature extraction technique that is commonly used in computer vision. HOGs work by computing the gradient of the image in different orientations and then constructing a histogram of these gradients. The resulting histogram, called the HOG descriptor, is then used as a feature to represent the shape of the image. In this project, HOGs were used to extract features from the images of faces. The HOGs features represent the edges and shapes of the face, which can be useful for recognizing different facial expressions. Different facial expressions can have different shapes and edges, for example a smiling face has a curved shape of the mouth, while a neutral face does not. By using HOGs to extract these features, the model can learn to recognize different facial expressions by learning to recognize the shape and edges of the face.

By using LBPs and HOGs to extract features from the images of faces, the model can learn to recognize

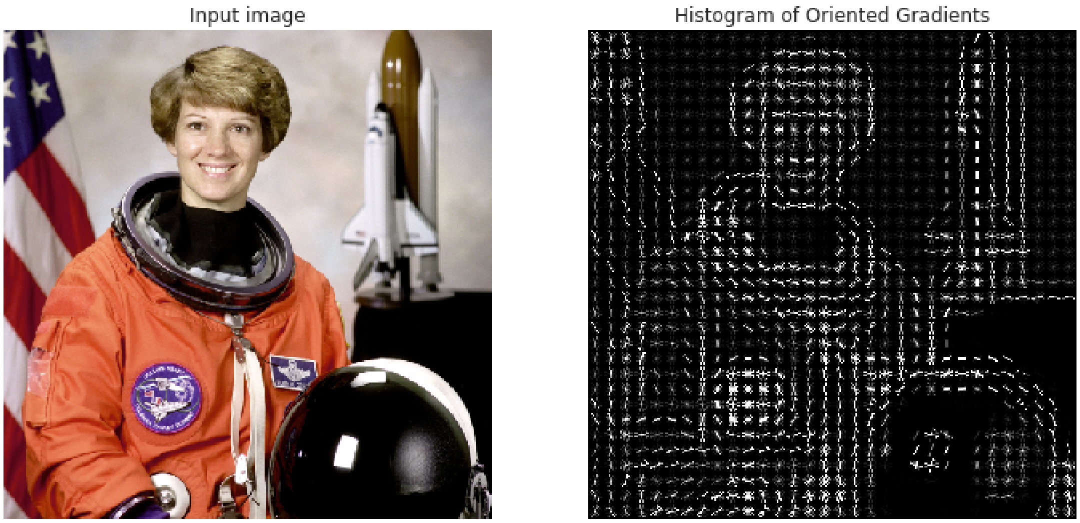


Figure 9: HOGs

different facial expressions by learning to recognize the texture and shape of the face. These features can be used as input to the model, along with the other features extracted by VGGface and facial landmarks, to improve the performance of the model by providing it with multiple sources of information to learn from.

6.3 model architecture

The model architecture is responsible for recognizing different facial expressions based on the features extracted in the first part. This is done using a combination of convolutional neural networks (CNNs) and fully connected layers. The model architecture includes 4 CNN layers for original images, 2 CNN layers for the landmarks, the outputs are concatenated along with the other features(Hogs, LBPs and the output predictions of the VGGface). After that it passes through 2 fully connected layers.

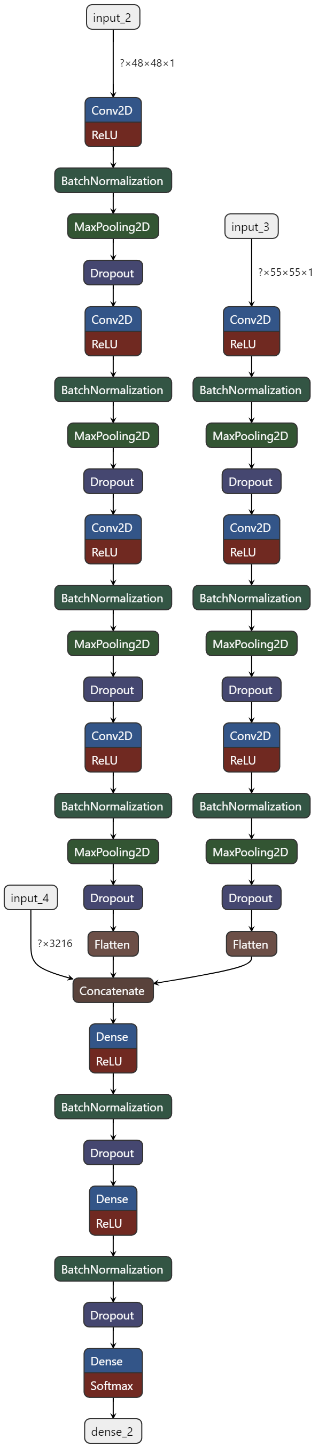


Figure 10: Model Architecture

6.4 Evaluation and Prediction

The optimization algorithm used in the training process was Adam, which is a variant of Stochastic Gradient Descent that uses adaptive learning rates. This allows the algorithm to adjust the learning rate for each parameter individually, which can lead to faster convergence.

The loss function used in this project was sparse categorical cross-entropy. This loss function is well suited for multi-class classification problems with integer labels, and it measures the dissimilarity between the predicted probability distribution and the true distribution. The lower the value of the loss function, the better the model is at predicting the correct class.

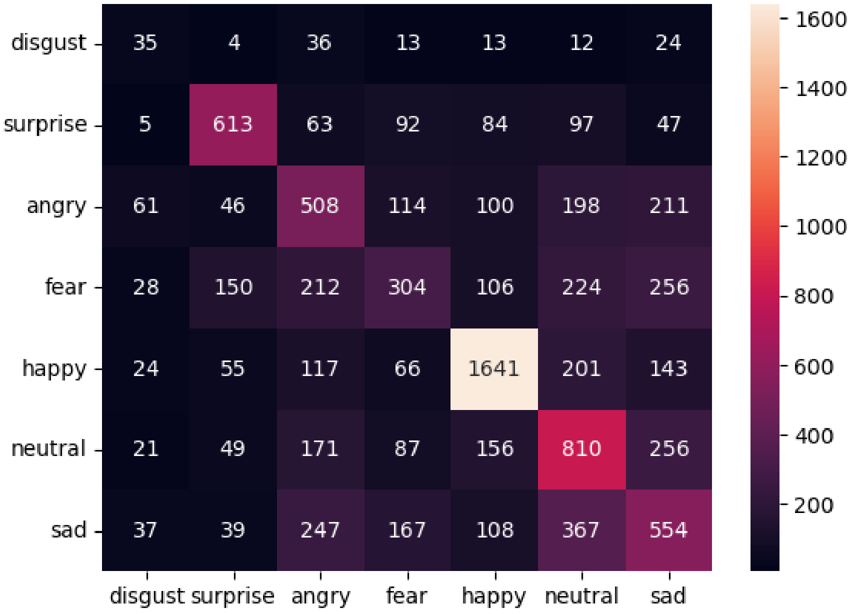
The number of epochs is the number of times the model is trained on the entire dataset. In this project, the model was trained for 10 epochs. This number was chosen through trial and error.

To evaluate the performance of the model, several metrics were used, including sparse categorical accuracy and sparse categorical cross-entropy. The sparse categorical accuracy is the percentage of correct predictions made by the model on the test dataset. The sparse categorical cross-entropy is the loss function used in the training process.

In addition, the confusion matrix was used to evaluate the model's performance, it shows the number of correct and incorrect predictions made by the model for each class. The confusion matrix provides a detailed view of the model's performance and allows us to see which classes the model struggles with.

7 Results

7.1 confusion matrix



	disgust	surprise	angry	fear	happy	neutral	sad
disgust	35	4	36	13	13	12	24
surprise	5	613	63	92	84	97	47
angry	61	46	508	114	100	198	211
fear	28	150	212	304	106	224	256
happy	24	55	117	66	1641	201	143
neutral	21	49	171	87	156	810	256
sad	37	39	247	167	108	367	554

Figure 11: confusion matrix

7.2 precision recall curve

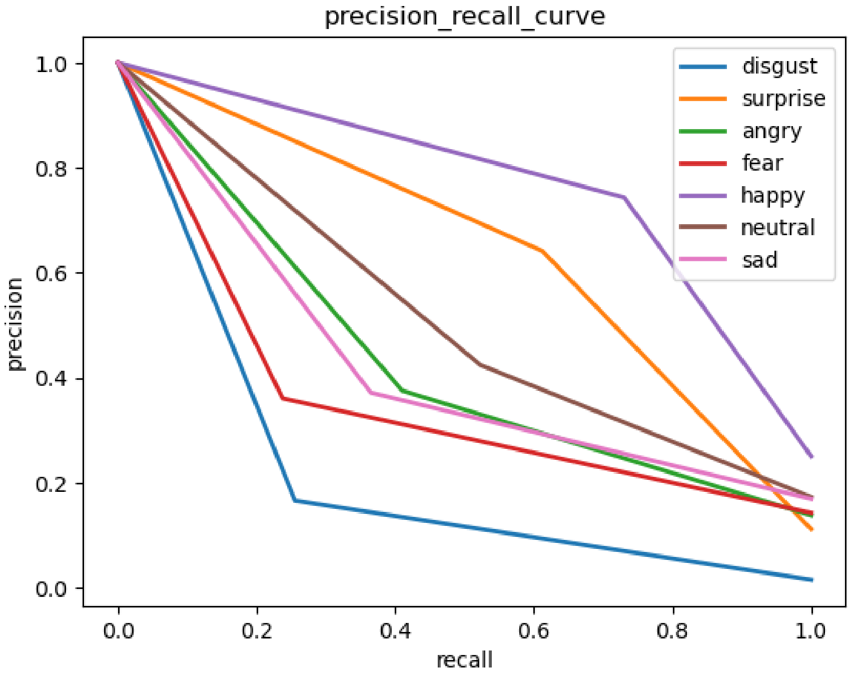


Figure 12: precision recall curve

7.3 ROC curve

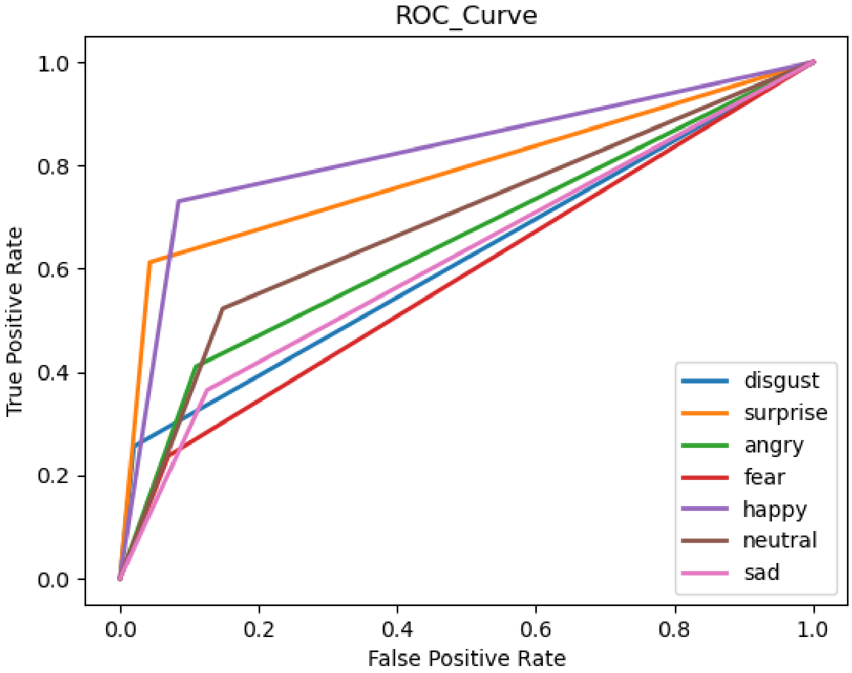


Figure 13: ROC curve

7.4 interpretations

The results of the facial expression classification model show that while it was able to achieve a high overall accuracy, it struggled to accurately predict certain minority classes. This can be seen in the confusion matrix, where certain classes have a higher number of misclassifications than others.

One possible explanation for this phenomenon is that the model is encountering a class imbalance problem. This occurs when the dataset has a disproportionate number of examples for each class, leading to the model being more likely to predict the majority class. In this case, it appears that the model is having difficulty distinguishing between certain minority classes, leading to a higher number of misclassifications.

Another possible explanation is that the model is not able to extract enough relevant features from the images of the minority classes. This may be due to the minority classes being visually distinct from the majority classes, making it more challenging for the model to extract features that can accurately differentiate them.

8 Video Tester

The trained deep neural network was extracted and used in realtime session.We used a laptop computer with a web cam in resolution 1,920 × 1,080 equipped with CPU Intel Core i5-6300U at 2.4 GHz.Only the face was sent to the our trained network as the background was cropped out. The neural network classified the image and sent the result back to the application that displayed it as a label on the screen.

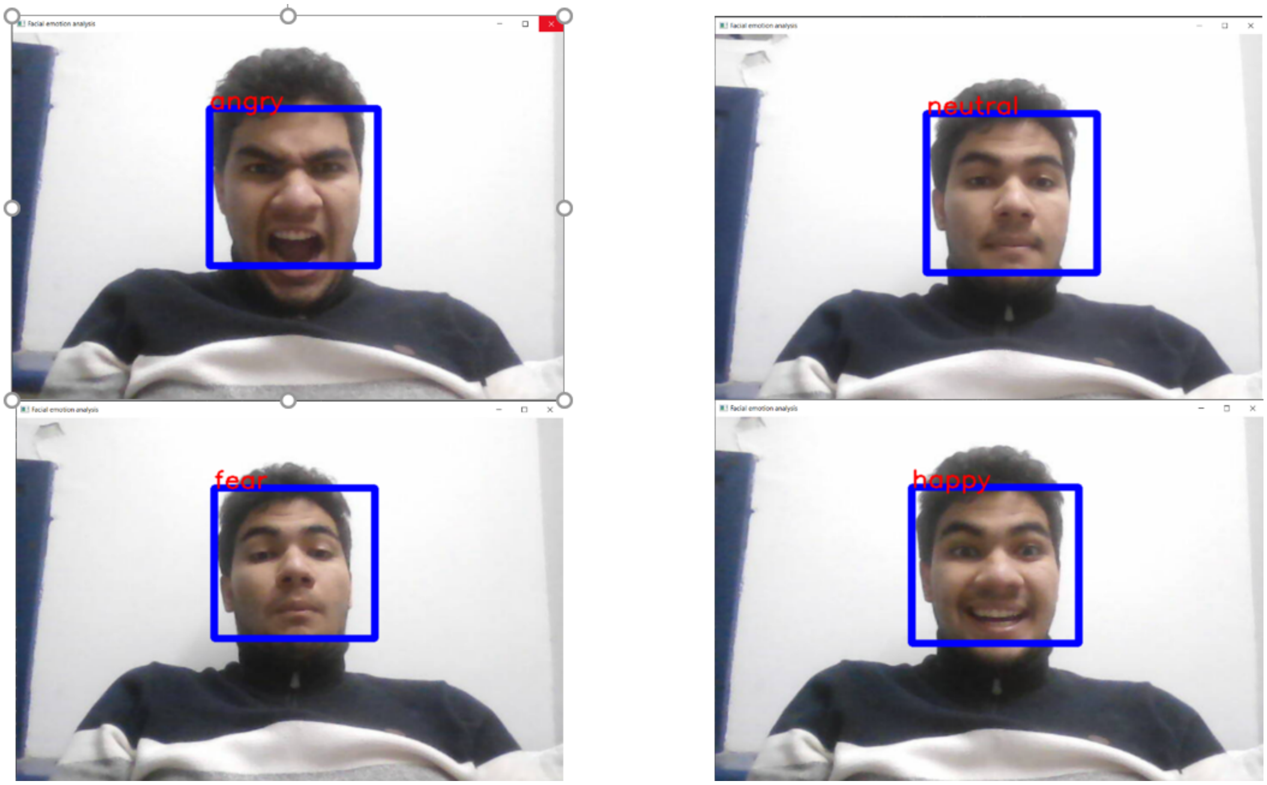


Figure 14: Examples of real-time expression detection

9 Conclusion

The purpose of this project is to develop a real-time facial emotion recognition algorithm that detects and classifies human emotions.

Our future goal is to integrate the recognition algorithm into a system of effective pedagogical agents that will respond to the students' detected emotions using different types of emotional intelligence. Our experiments show that overall test accuracy is sufficient for a practical use and we hope that the entire system will be able to enhance learning.

There are several possible avenues for future work. While our preliminary results show satisfactory precision on our tested data, it would be interesting to actually validate our system in a real-world scenario. We conducted a preliminary user study in which we asked 10 people to make certain facial expression and we validated the detection. However, this approach did not provide satisfactory results, because we did not find a way to verify that the people were actually in the desired emotional state and their expressions were genuine - some participants started to laugh each time the system detected emotion they were not expecting. Emotional state is a complicated. Happy people cannot force themselves to make sad faces and some of the expressions were difficult to achieve even for real actors. So while validation of our detector remains a future work, another future work is increasing the precision of the detection by expanding the training data set and tuning the parameters of the deep neural network.