# Facial Emotion Recognition

Team - 32 (Oreo)

April 2019

## 1 Introduction

### 1.1 Problem Statement

Implement an efficient and straightforward facial emotion recognition algorithm to reduce the problem of inter-class pixel mismatch during classification.

### 1.2 Motivation

The number of methods for feature extraction and classification that have been proposed lack template matching methods that allow to achieve high recognition accuracy with minimum computational cost. The algorithm proposed in the paper can be implemented using small computational capacity devices keeping facial emotion recognition operation fast and accurate.

## 2 Literature Survey

We referred the following paper:-
1) **Facial Emotion Recognition using Min-Max Similarity Classifier** by Olga Krestinskaya and Alex Pappachen James.
To solve the problem of Facial Emotion Recognition we used Min-Max Similarity Classifier as proposed in the above paper. The paper states that the methodology used performs better than other methods like Convolutional Neural Network, KNN etc in terms of computational cost and simplicity.

## 3 Methodology

### 3.1 Input

Training data contains 213 images of 10 female faces comprising 6 basic facial expressions (Angry, Disgust, Fear, Happy, Sad, Surprise) and neutral faces. The original images from have the size of 256 * 256 pixels and in our experiments they are cropped to a size of 140 * 114 pixels retaining only the relevant information of the face area.

## 3.2 Method

The algorithm is divided in three steps:-

### 3.2.1 Preprocessing

Illumination variability introduces inter-class feature mismatch which results in inaccurate detection of emotion discriminating features from the face images. Therefore, image preprocessing is essential to normalize the images and reduce the inter-class feature mismatch. The input image is represented as x(i, j), and y(i, j) is the normalized output image, where i and j are the row and column number of the processed image. $\mu$ is the local mean and $\sigma$ is the local standard deviation computed over a window of N * N size.
The normalised image is calculated pixel by pixel using the below formula

$$y(i,j) = \frac{x(i,j) - \mu(i,j)}{6\sigma(i,j)}$$

where

$$\mu(i,j) = \frac{1}{N^2} \sum_{k=-a}^{a} \sum_{h=-a}^{a} x(k+i, h+j)$$

and

$$\sigma(i,j) = \sqrt{\frac{1}{N^2} \sum_{k=-a}^{a} \sum_{h=-a}^{a} [x(k+i, h+j) - \mu(i,j)]^2}$$

Here a = (N-1)/2

### 3.2.2 Feature Detection

The feature parts useful for the facial emotion recognition are eyes, eyebrows, cheeks and mouth regions. Feature detection is performed by calculating local standard deviation of normalized image using a window of M * M size.
The feature extracted image is given by the matrix w which is calculated as follows:-

$$w(i,j) = \sqrt{\frac{1}{M^2} \sum_{k=-b}^{b} \sum_{h=-b}^{b} [y(k+i, h+j) - \mu'(i,j)]^2}$$

The mean of the normalized image y(i, j) and can be calculated by

$$\mu'(i,j) = \frac{1}{M^2} \sum_{k=-b}^{b} \sum_{h=-b}^{b} y(k+i, h+j)$$

### 3.2.3 Emotion Classification

For the recognition stage, we propose a Min-Max similarity metric in a Nearest Neighbor classifier framework. This method is based on the principle that the ratio of the minimum difference to the maximum difference of two pixels will produce a unity output for equal pixels and an output less than unity for unequal pixels. The algorithm parameter trainlen refers to the number of train images, N corresponds to the normalization window size, and M indicates the feature detection window size. Each cropped image is of m * n pixel dimension. The parameter train is a feature array of trainlen*(m * n) size, where each row corresponds to the processed train images. After normalization and feature detection, test images are stored in to a vector test of 1*(m * n) size. A single test image is compared pixel-wise with processed train images of all the classes in the feature array using the proposed Min-Max classifier:

$$s(i,j) = [\frac{min[train(i,j), test(1,j)]}{max[train(i,j), test(1,j)]}]^{\alpha},$$

where a parameter $\alpha$ controls the power of exponential to suppress the outlier similarities.
After Min-Max classification, a column vector z of trainlen * 1 size containing the weights obtained after comparing the test image with each of the trainlen number of train images is calculated using

$$z(i) = \sum_{j=1}^{m \times n} s(i,j)$$

The classification output out shown is the maximum value of z corresponding to the train image that shows the maximum match. The recognized emotion class is the class of the matched train image.

$$out = max(z(i))$$

# 4 Task Assignment

## 4.1 Normalisation & Feature Extraction

Pranjali - 50%
Ekta - 50%

## 4.2 Classification

Kratika - 50%
Akanksha - 50%

# 5 Results

Validation Results:-

```
Pranjalis-MBP:SMAI-Project pranjali$ python3 Project_Test.py
Max_Similarity  14733.640808041111 found with train image YM.HA3.54.tiff
Emotion for the image YM.HA2.53.tiffis :HAPPY
Max_Similarity  14721.975166446606 found with train image NA.DI3.216.tiff
Emotion for the image NA.DI2.215.tiffis :DISGUSTED
Max_Similarity  14648.44216986167 found with train image TM.AN1.190.tiff
Emotion for the image TM.AN2.191.tiffis :ANGRY
Max_Similarity  14990.7195626367 found with train image KL.AN2.168.tiff
Emotion for the image KL.AN1.167.tiffis :ANGRY
Max_Similarity  15386.021763189197 found with train image UY.SA3.142.tiff
Emotion for the image UY.SA2.141.tiffis :SAD
Max_Similarity  15080.167358506365 found with train image NA.HA3.204.tiff
Emotion for the image NA.HA2.203.tiffis :HAPPY
Max_Similarity  14894.26148384222 found with train image KM.SA3.11.tiff
Emotion for the image KM.SA2.10.tiffis :SAD
Max_Similarity  15460.058489182662 found with train image KR.DI2.87.tiff
Emotion for the image KR.DI3.88.tiffis :DISGUSTED
Max_Similarity  14626.269759624025 found with train image YM.SU1.58.tiff
Emotion for the image YM.SU3.60.tiffis :SURPRISED
Max_Similarity  15548.487209245592 found with train image NM.NE3.94.tiff
Emotion for the image NM.NE2.93.tiffis :NEUTRAL
Max_Similarity  15067.357598086262 found with train image UY.SU1.143.tiff
Emotion for the image UY.SU2.144.tiffis :SURPRISED
Max_Similarity  15213.883155955755 found with train image UY.FE3.154.tiff
Emotion for the image UY.FE2.153.tiffis :FEAR
Max_Similarity  14386.069980227348 found with train image YM.NE3.51.tiff
Emotion for the image YM.NE1.49.tiffis :NEUTRAL
Max_Similarity  14771.343976287018 found with train image KL.FE1.174.tiff
Emotion for the image KL.FE3.176.tiffis :FEAR

 Validation Testing results:
Correct_Predictions:  14
Wrong_Predictions:  0
Accuracy:  100.0
```

Test Results:-

```
 Unknown Testing results:
Max_Similarity  13931.543275037271 found with train image NM.SA2.99.tiff

Result for image AK.SU1.11.tiff
Predicted Emotion:  SAD
Actual Emotion:  SURPRISED
2019-04-29 22:09:23.878 Python[18040:1517317] IMKClient Stall detected, *please Report* your user scenario attaching a spindump (o
r sysdiagnose) that captures the problem - (imkxpc_attributesForCharacterIndex:reply:) block performed very slowly (17.57 secs).
Max_Similarity  13771.798788598131 found with train image KL.AN2.168.tiff

Result for image PR.AN1.12.tiff
Predicted Emotion:  ANGRY
Actual Emotion:  ANGRY
Max_Similarity  13620.590128835272 found with train image KL.AN2.168.tiff

Result for image EK.HA1.14.tiff
Predicted Emotion:  ANGRY
Actual Emotion:  HAPPY
Max_Similarity  13502.100510954126 found with train image NM.NE3.94.tiff

Result for image PR.NE1.13.tiff
Predicted Emotion:  NEUTRAL
Actual Emotion:  NEUTRAL
Max_Similarity  12942.502794955719 found with train image NM.AN3.106.tiff

Result for image KR.SU1.15.tiff
Predicted Emotion:  ANGRY
Actual Emotion:  SURPRISED
```

# 6 Conclusion

In this project, we have attempted to implement the approach to improve the performance of emotion recognition task using template matching method. As the referenced paper dictates, we have demonstrated that the pixel normalization and feature extraction based on local mean and standard deviation followed up by the Mix-Max similarity classification can result in the improvement of overall classification rates. The capability of the algorithm to suppress feature outliers and remove intensity offsets results in the increase of emotion recognition accuracy. Low computational complexity is a noticeable benefit of the proposed algorithm that implies the reduction of computational time and required memory space.