ECSE 415 – Computer Vision
Final project

Yuxiang Ma 260714506
Fandi Yi 260722217
Menghan Gao 260711270
Tingzhe Wang 260687154
Kexin Xu 260717893

Table of Content

1. Introduction

In this project, we will develop a facial recognition system. Specifically, the team will explore the use of the popular quantization feature package and k-nearest neighbors as a classifier to classify invisible face images. After collecting the data and training the photo collection, the resulting face recognition system will be used to tag team members in team photos. All the pictures taken will be compared with the PCA-based method by the bag of words method.

2. Acquiring Appropriate Datasets

2.1. Training Images

To be able to detect different face images, we need a large database with enough face feature clusters from seventy-five png photos of group members. This large collection of data could help to reduce the final detection error. All five members in our group acquire to take five upper-body photos in five rotation (-45°, -30°, 0, 30°, 45°) and separately in three scales, small, normal and large. To rule down non-face elements quickly and focus more on the face with more specialized detectors, every candidate stands in front of a monochrome colored wall.

Fandi:

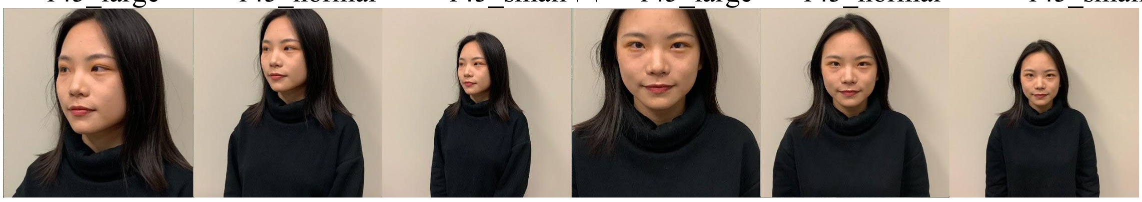
l30_large l30_normal l30_small 0_large 0_normal 0_small

r30_large r30_normal r30_small l45_large large_normal large_small

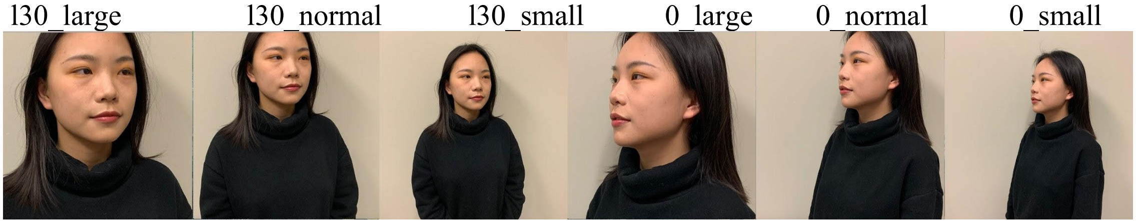
r45_large r45_normal r45_small

Menghan:

r45_large r45_normal r45_small



l30_large l30_normal l30_small 0_large 0_normal 0_small

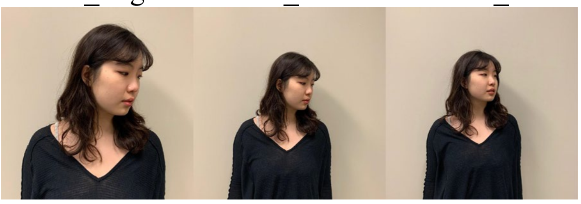


r30_large r30_normal r30_small l45_large large_normal large_small

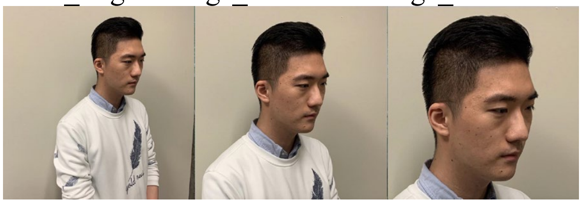
Sherry:

l30_large l30_normal l30_small 0_large 0_normal 0_small

r30_large r30_normal r30_small l45_large large_normal large_small



Tingzhe:



r45_large r45_normal r45_small r45_large r45_normal r45_small

l30_large l30_normal l30_small 0_large 0_normal 0_small

r30_large r30_normal r30_small l45_large large_normal large_small

Yuxiang:

l30_large l30_normal l30_small 0_large 0_normal 0_small

r30_large r30_normal r30_small l45_large large_normal large_small

r45_large r45_normal r45_small

2.2 Testing Images

For the further testing of our images batch, we make a test database that consists of fourteen photos each person, which are five photos of each subject looking up to different direction (-45°, -30°, 0, 30°, 45°), and five photos of looking down, and four photos of different facial expressions and accessories. This dataset is defined as "unseen" and should be identified correctly to the corresponding figure. To achieve successful face detection, the team manually crops a square around each face from each image into haar feature.

Fandi:

down_0 down_l30 down_l45 down_r30 down_r45 glass happy

hat surprised up_0 up_l30 up_l45 up_r30 up_r45

Menghan:

down_0 down_l30 down_l45 down_r30 down_r45 glass happy

hat surprised up_0 up_l30 up_l45 up_r3 up_r45

Sherry:

Down_0 down_l30 down_l45 down_r30 down_r45 glass happy

Hat surprised up_0 up_l30 up_l45 up_r30 up_r45

Tingzhe:

Down_0 down_l30 down_l45 down_r30 down_r45 glass happy

Hat surprised up_0 up_l30 up_l45 up_r30 up_r45

Yuxiang:

Down_0 down_l30 down_l45 down_r30 down_r45 glass happy

Hat surprised up_0 up_l30 up_l45 up_r30 up_r45

3. Face Recognition

Furthermore, we will build several vocabularies using different methods of keypoint detection and descriptor computation. Before we delineate the experiments, details for the training and testing procedure are described.

3.1 Training: Building Vocabulary

Manually define a square integral image around face from each training photos, which computes the sum of the pixel values above and to the left of (x, y). Automatically limit the minimum detection feature size to be (300, 300) as the Fig.1 shown below.

$sum\ of\ pixel = I(C) + I(A) - I(B) - I(D)$

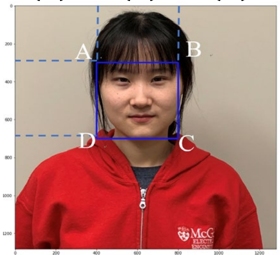


Figure 1. Square for cropping face area

Figure 2. Ten key points images.

Gaussian Mixture Model (GMM) is a collection of K Gaussian distributions that are called as modes of GMM and represent K clusters of data points. We apply *mixture.GaussianMixture* method to obtain those K clusters. Then, classify images by bag-of-words model (BoW) in terms of treating image features as words and computing normalized histogram of words.

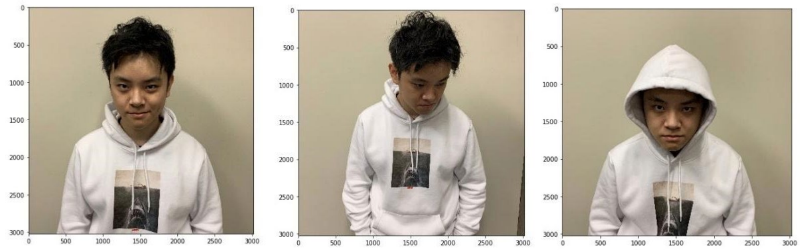


Fig.2 Three images selected for the histogram

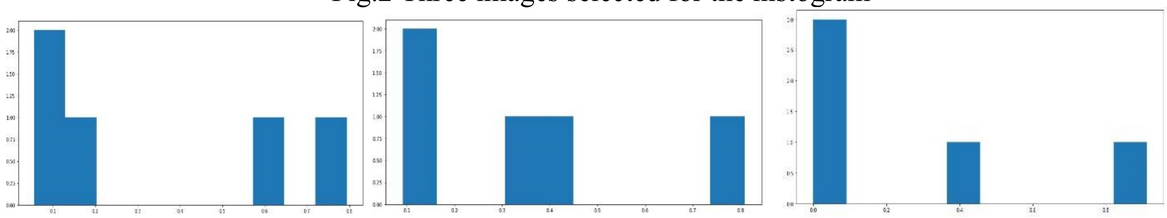


Fig.3 Histograms respect to three pictures above using BoW method

3.2 Testing: Face Recognition

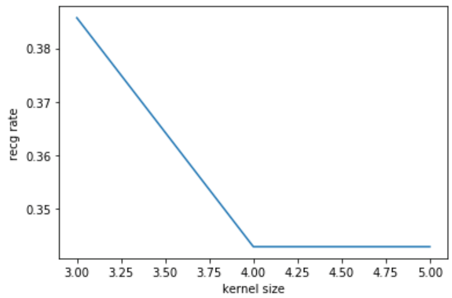
Manually crop a square bounding box around the face in each testing image. Utilize the same method as *Training: Building Vocabulary* to compute normalized histogram of words. Then, find the closest training histogram to one testing photo, so that we can assign different face images to its corresponding normalized histogram.

3.3 Experiments

We perform 3 sets of experiments to find out the best performing keypoint detection and descriptor computation methods respectively on the 3x3, 4x4 and 5x5 cell size of the pictures.

3.3.1

1) For the recognition rate of hog feature, the 3x3 has the of rate 0.38571428571428573, the 4x4 has the rate 0.34285714285714286 and the 5x5 has the rate 0.34285714285714286. The plot is shown below as figure 4. For the confusion matrix of the best performing size which is 3x3 is shown as Fig.5.



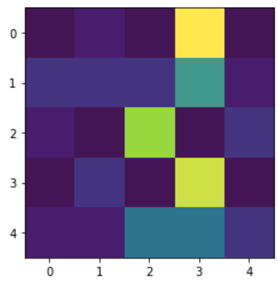
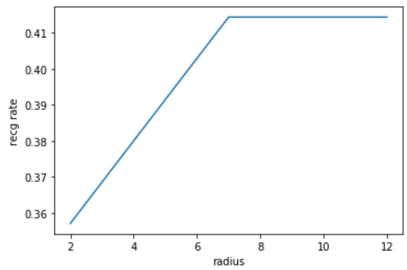


Fig.4 recognition rate Fig.5 3x3 Confusion matrix

2) For the recognition rate of the LBP descriptor, the radius of 2,7,12 has the recognition rate of 0.35714285714285715, 0.4142857142857143 and 0.4142857142857143 respectively. The plot is shown below as Fig.6. For the confusion matrix of the best performing size which is 7 is shown as Fig.7.



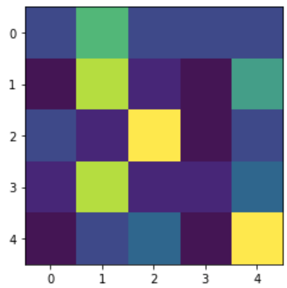
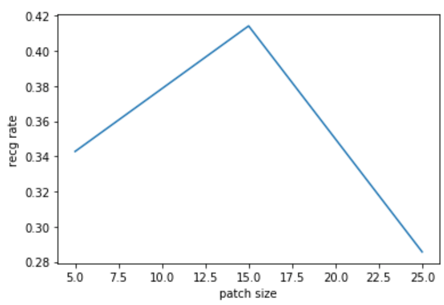


Fig.6 recognition rate Fig.7 Confusion matrix of radius 7

3) The best performing recognition rate for the hog is 3x3 and the best LBP is 7 and 12. They all comparatively have the highest accuracy in their own method, and LBP radius 7 and 12 has higher accuracy than hog 3x3.
Reason: LBP value calculated for this centre pixel and stored in the output 2D array with the same width and height as the input image. LBP labels the pixels of an image by thresholding the neighbourhood of each pixel and considers the result as a binary number. Therefore, the LBP codes can also provide responses salient shapes, which means to monotonically changing intensity functions and edges within the image microstructure. However, for Hog algorithm, we counts occurrences of gradient orientation in localized portions of an image. To conclude, The LBP algorithm has the better contraction and better performance.

3.3.2

1) In this part of the experiment, using SIFT to detect keypoints and the best performing feature detection method on the size of 5x5, 15x15 and 25x25. For the recognition rates 5x5 has the rate of 0.34285714285714286, 15x15 has the rate of 0.4142857142857143 and 25x25 has the rate of 0.2857142857142857. The cell size of the 15x15 has the highest accuracy among the sizes. The plot is shown below as Fig.8. For the confusion matrix of the best performing size which is 15x15 is shown as Fig.9.



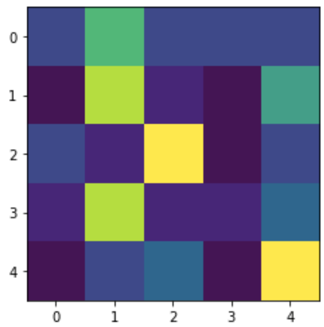


Fig.8 recognition rate Fig.9 Confusion matrix of radius 15x15

3.3.3

1) By using the Harris corner features, the recognition rate and the confusion matrix are shown as below:
It has the recognition rate of 0.2 and confusion matrix of harris as below. By comparing the results of SIFT/SURF and Harris, the average performance of all SIFT/SURF(problem 3.3.2) is better than that of Harris in terms of their recognition rate.

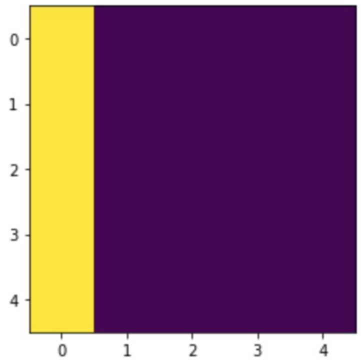


Fig.10 Confusion Matrix of Harris

4. EigenFaces

Principal Component Analysis (PCA) uses orthogonal transformation to convert a set of observed correlated variables into another set of linearly uncorrelated variables, and is regarded as a decorrelation method. Its dimensionality reduction keeps the most important descriptive features of an image into a data set, so that the data is simplified without losing important traits. The eigen representations of those training images are references for classifying testing data, because a test image will be identified by the nearest representation.

For a person Fi, if the training images are Ti1, Ti2, …… Tik then we obtain the set of projections as

$$y_{ik} = \phi_M^T(T_{ik} - \psi)$$

Where $\phi_M^T$is the matrix containing principal eigenvectors and Ψ is the mean image of the training set. When have a face candidate x, we find it's projection yx similar to above and find the MSE as

$$MSE_x = \sum_{i=1}^{k}(y_x - y_{ik})^T(y_x - y_{ik})$$

If a face $\tilde{x}$ is such that MSE_x=min(MSE_x) then we decide that $\tilde{x}$=Ft

The first five eigenfaces are shown below:

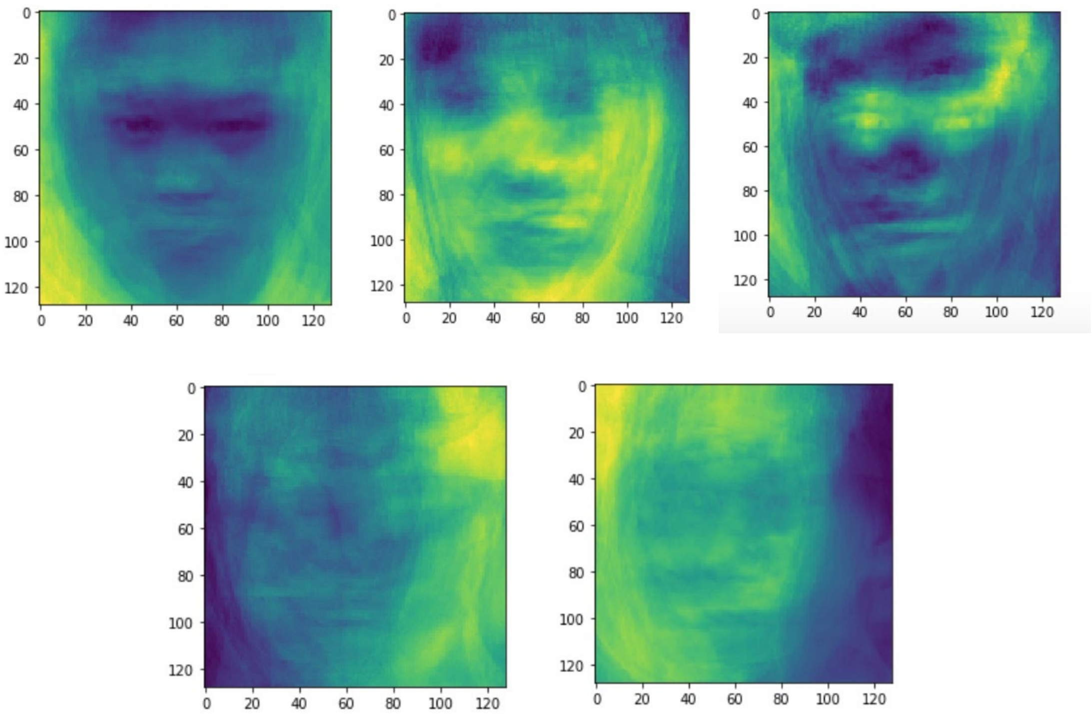


Figure 11. five eigenfaces images

By using the PCA method, the recognition rate and confusion matrix are shown as below:

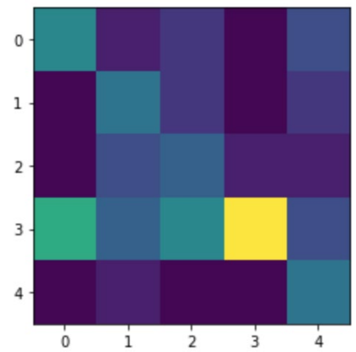


Figure 12. confusion matrix

recognition rate: 0.4714285714285714.

Application of PCA improves the accuracy of image detection to 0.471 and the confusion matrix shows a clear diagonal with high intensity, which all represents progress in face detection. Suppose using BoG model, it will parse a lot of text and those feature set becomes unmanageable, and the marginal benefit of adding another variable diminishes. While PCA is used to remove the least beneficial feature so that we have a smaller dataset, but without losing too much predictive power.

In summary, HOG refers the capture of bounding box fringes, while LBP prefers to color capture, and PCA refers to a specific and basic linear transformation to rebuild a dataset. We think, in the region of face detection, especially for our dataset, detecting address of intensity performs better than detecting unimportant lines and points on the face. Therefore, PCA is an optimal method for face detection, since it solves the least square problem that fits a hyperplane in the face feature space which minimizes the least square error.

5. Face Tagging

Tagged images:

Figure 4. five original group images.

Figure 5. five tagged group images