Face Recognition using PCA

MsCV6 - Université de Bourgogne

Peixi LI and Mark Nabil

16/01/2015

1 Normalization

For this project, what was done first is to do the normalization. The face images need to be transformed to a predetermined locations in the 64×64 window, so that they can be operated and calculated properly. The affine transformation is applied in this process. This transformation can be expressed as:

$$f_i^P = Af_i + b \tag{1}$$

where,

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{2}$$

and,

$$b = \begin{bmatrix} b_1 \\ a_2 \end{bmatrix} \tag{3}$$

From above equations it can be seen that for the matrix equation 10 simple equations and 6 variables can be obtained. It can be solved by least squares method.

A vector $\bar{F}$ is defined to store the average locations of the key features of all the faces. After calculating the transformation, it should be applied and compared with this vecotor $\bar{F}$.

For every face image F_i, the transformation should be calculated. And the vector $\bar{F}$ should be optimized by averaging the aligned feature for the F_i. Iterate this process until the minimum error is obtained.

2 Covariance matrix, eigen componnets and Recognizing

For face recognition, there are many faces, and each faces have a lot of pixels. After subtracting the mean face, the data can be expressed in a huge matrix, which is as:

$$D = \begin{bmatrix} I_{1(1,1)} & I_{1(1,2)} & \cdots & I_{1(1,N)} & \cdots & I_{1(M,1)} & I_{1(M,2)} & \cdots & I_{1(M,N)} \\ I_{2(1,1)} & I_{2(1,2)} & \cdots & I_{2(1,N)} & \cdots & I_{2(M,1)} & I_{2(M,2)} & \cdots & I_{2(M,N)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ I_{p(1,1)} & I_{p(1,2)} & \cdots & I_{p(1,N)} & \cdots & I_{p(M,1)} & I_{p(M,2)} & \cdots & I_{p(M,N)} \end{bmatrix} \tag{4}$$

This matrix is the result of the subtraction of the previous one. It can be seen that there are too much date, which is far beyond the size of window used for this processing. So what should be done is to extract the "important" information from this data by calculating the covariance matrix and its eigen components, which represent the "frequency" information from the images. The covariance matrix is expressed as:

$$\Sigma = \frac{1}{p-1} D^T D \tag{5}$$

Solve the eigen components for this covariance matrix. Only the largest eigen components are kept because they represent for the most "important" information. The principal components to images can be calculated by reversing the concatenation operation. This image is called eigenfaces. The dimesion of this principal components matrix is d×k, where d is the number of the elements of the image and k is number of eigen components which are kept. k is much smaller than the previuos d. The image is projected into this space, and only the "high frenquency" and "important" information are kept. The dimension is much lower, which makes the processing more convenient. The projection is expressed as:

$$\phi_i = X_i \phi \tag{6}$$

where ϕ is the principal components matrix.
After the PCA done, calculate the euclidean distance and get the errors.

3 Experiment and results

- The GUI is helped by our senior Mr. Usman.
- One problem: our data set is not reliable.

The results are shown as follows: These results imply that actually our dataset is not reliable. There may be some variations.

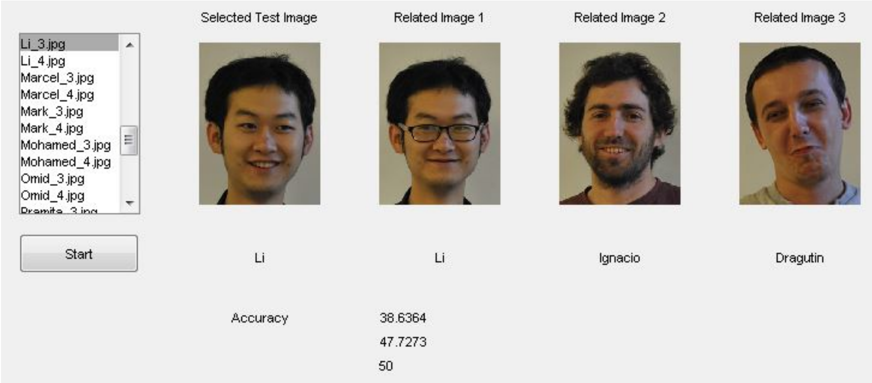

Figure 1: Result 1

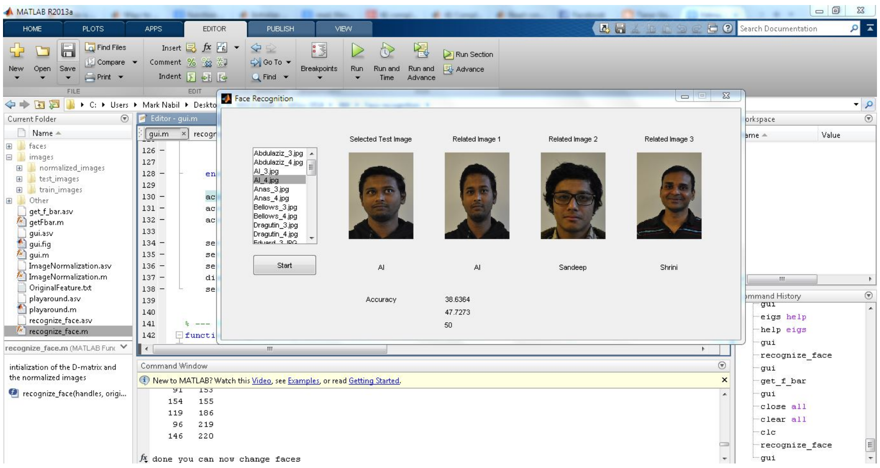

Figure 2: Result 2

4 Conclusion

For this project, what was done first is to normalize the iamges. And secondly the covariance matrix is calculated and eigen components are calculated. We made the projectoin and keep the most important parts. And the results are compared and the faces are recognized.