Content based image retrieval system using Eigenfaces

HAMIM SHAIKH

May 28, 2020

Contents

1 Abstract

Face image retrieval system is one of the emerging technologies with various application across many domains.Designing a complete face image retrieval system involves two key components: feature extraction and distance measurement.Understanding the features of faces by using Principal component analysis(PCA) and using an optimal distance metric to find distance between a training image and test image to finally classify a test image are the most important stages of this system.A final evaluation of the model is performed based on a set of evaluation metrics.

2 Introduction

For past few decades face recognition has been widely used in payments where a customer can confirm their payment by their own photo, facial bio-metric is one of most common use to secure mobile devices,Automated criminal identification based on security camera feeds is one of the effective ways of catching criminals, targeted advertisement based on facial features and finally within identifying illness by extracting facial features are few of most important applications of facial recognition systems.
Previous face recognition systems increasingly relied on facial features like eyes, nose, ears which were mapped into mathematical way.Such methodologies are dependent on various factors and known to be vulnerable to inaccurate results. The concept of eigenfaces dates all the way to 1987 by Sirovich and Kirby who were developing a method represent human faces as projected components by using PCA.

A number of research still continues on this topic with number of branches relating to object detection and using convolutional neural network for learning image features is being actively conducted to this date.

In this report we will focus on real life application of eigenfaces where a complete life cycle of image pre-processing, image training and evaluation of model based on testing images is covered. Eigenfaces is one of the oldest and easiest method of face recognition.The method used in this report will show how an information of a facial image is can be retained inform of eigenvectors and used for facial recognition.

3 Database preparation

Preparing a database which stores images(Training and Testing datasets) in a matrix format with their corresponding names in text file.A python object stores image matrix with their corresponding labels.All images are preprocessed based on the requirements given in the task description.

3.1 Training images

Set of Images already provided in the project with additional set of images made based on methodology in Appendix 2. 27 images from Yale dataset with addition 3 images having a total of 30 images

3.2 Testing images

Set of Images that will be used for testing and performance evaluation of the model 18 images from Yale dataset with addition 2 images having a total of 20 images

3.3 Preprocessing

1. Resizing: All input images should have the same shape and size. Also to avoid too much computation time we can lower the size of the image. However, this is actually a trade-off between quality of Images obtained with time

2. Cropping: In face recognition applications,only facial images are required hence, the background will be removed keeping only facial images.

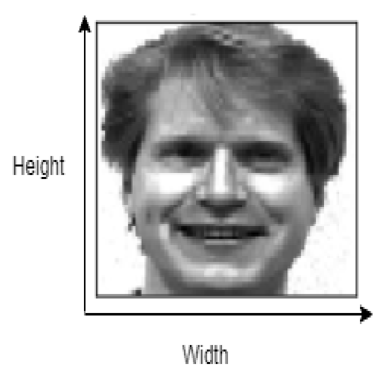
3. Conversion to Gray-scale: To avoid high computational cost of color images the images are converted to gray-scale

4. Normalization: All pixel values are normalized from 0 to 255 to 0 to 1 to prevent any biases happening due to high pixel values.

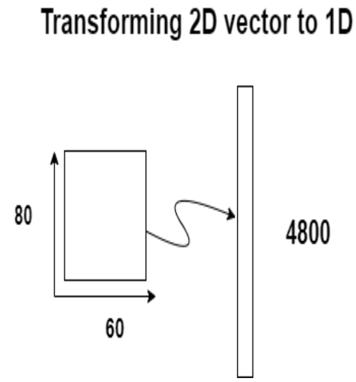
4 Computation of Images using PCA

1. The image is resized based on specific requirements of height and width

Figure 1: Sample Image



2. All the 2 dimensional image matrices are converted to 1 dimensional array



3. Lets consider a set S of all images in train dataset where each value in a set represent an Image.
   $S = \{I_1, I_2, I_3 .... I_i\}$

4. The mean of all images are calculated along a particular axis which gives us mean face which has 4800 values
   $M = \frac{1}{N} \sum_{i=1}^{N} I_i$

5. Deviation matrix is calculated by subtracting every image with its mean face which can also be refered to as face normalization.
   $A_i = I_i - M$

6. Calculating co-variance from deviation matrix .

7. Eigen vectors and eigen values are calculated from the co-variance matrix which is $A^T A$. where λ_i is eigen value and V_i is eigenvector.

8. When a single eigenvector is reshaped to the training image a face is appeared which looks like a combination of different faces hence these eigenvectors are called as eigenfaces in face recognition system.

9. Each facial image in the training data can represented as a weighted linear combination of eigenfaces.

   $w_i = V_i^T A_i$

4.1 Image Recognition

1. Lets consider a set S of all images in test dataset where each value in a set represent an Image.

$S = \{T_1, T_2, T_3....T_i\}$

2. Repeating same preprocessing step which was performed on the training dataset.
3. Normalizing the test faces $A_t = T_i - M$
4. Multiplying the normalized test faces with eigenfaces to obtain weights $w_i = V_i^T A_t$
5. Once the weights are calculated testing and training dataset are projected on eigenspace

4.2 Choosing the number of principal components

After the process of PCA there are number of principal components obtained however, in many scenarios majority most of information can be obtained with limited number of components hence it is also known to be popular dimension reductionality technique.The project requires two scenarios where 15% and 5% of information loss is acceptable.

1. $T_{k1} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} = 0.85$
2. $T_{k2} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} = 0.95$

4.3 Distance calculation

Distance metric is used to measure distances between a test image with the every training image.The one with least distance is our predicted class variable In this report only two distances will be used

1. Euclidiean distance

   $d(p, q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$

2. Using KNN classifier with distance metric of Minkowski and Manhattan distance

4.4 Evaluation

In this report evaluation of the face image retrieval system will be done based on 3 parameters

1. $\textbf{Recall} = \frac{\textbf{number of retrieved relevant items}}{\textbf{the total number of relevant items in database}}$

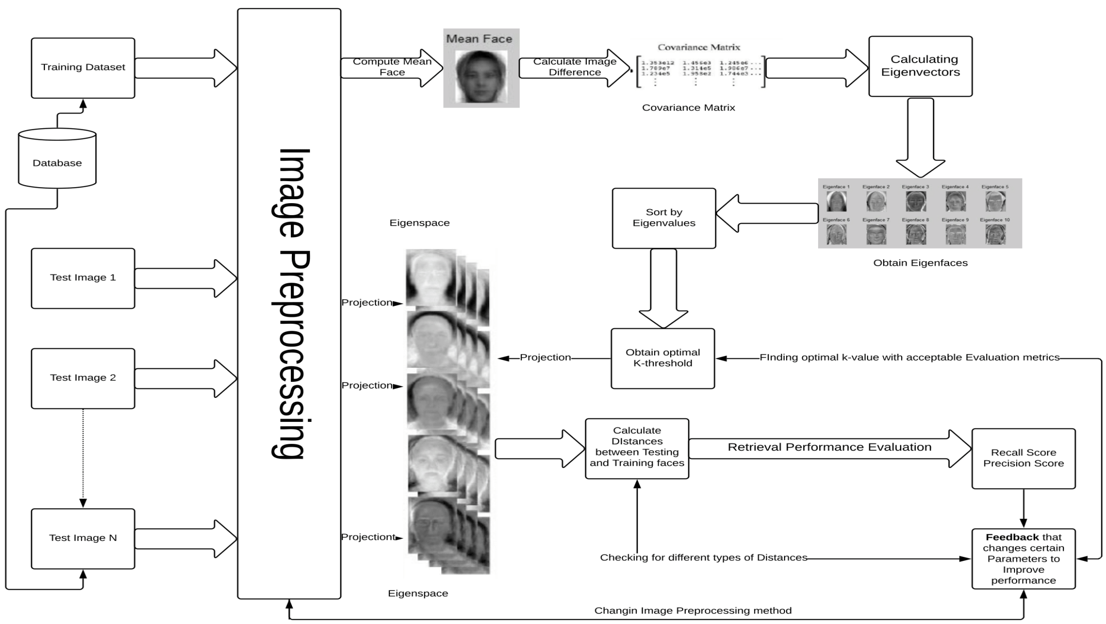
2. **Precision** $= \dfrac{\textbf{number of retrieved relevant item}}{\textbf{the total number of total retrieved items}}$

3. **Accuracy** $= \dfrac{\textbf{number of correct predictions}}{\textbf{the total number of total predictions made}}$

5 Design draft for face image retrieval system

Final design draft with all the components placed

Figure 2: Design Draft

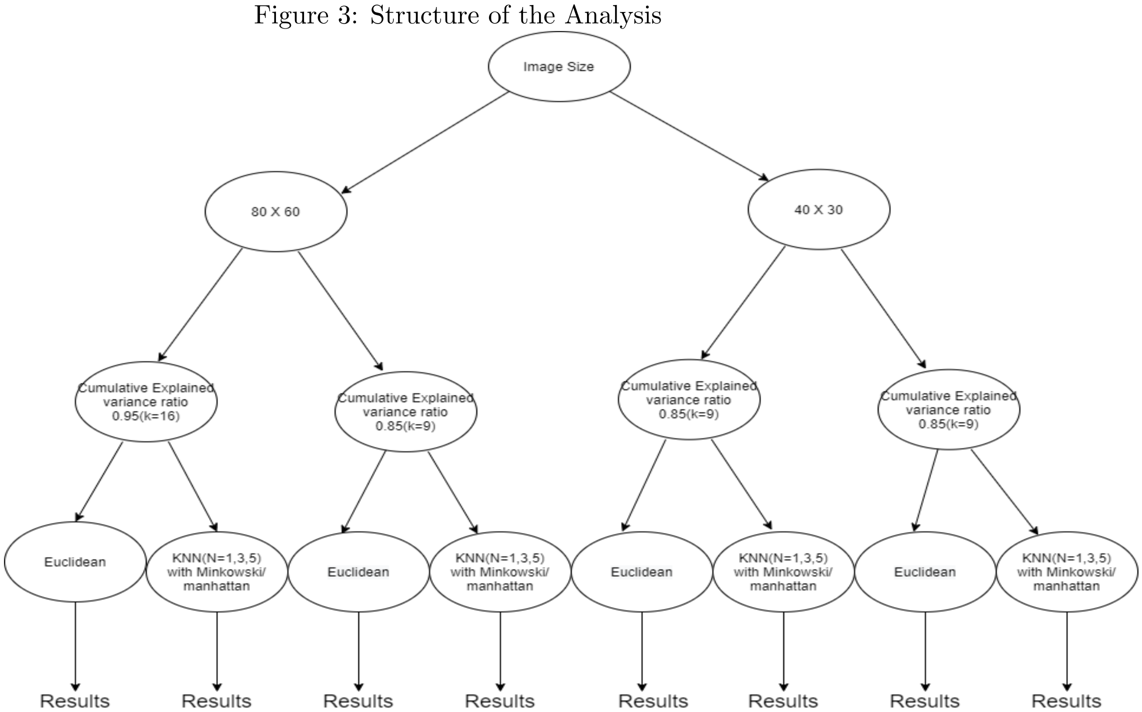


6 Observations

6.1 Structure of Analysis

After performing the all the steps, a number of observations were taken into account.
The complete analysis is divided divided on the basis of image size which is further divided on cumulative explained variance ratio and then finally on distance metrics.
The following diagram illustrates the complete procedure of analysis.

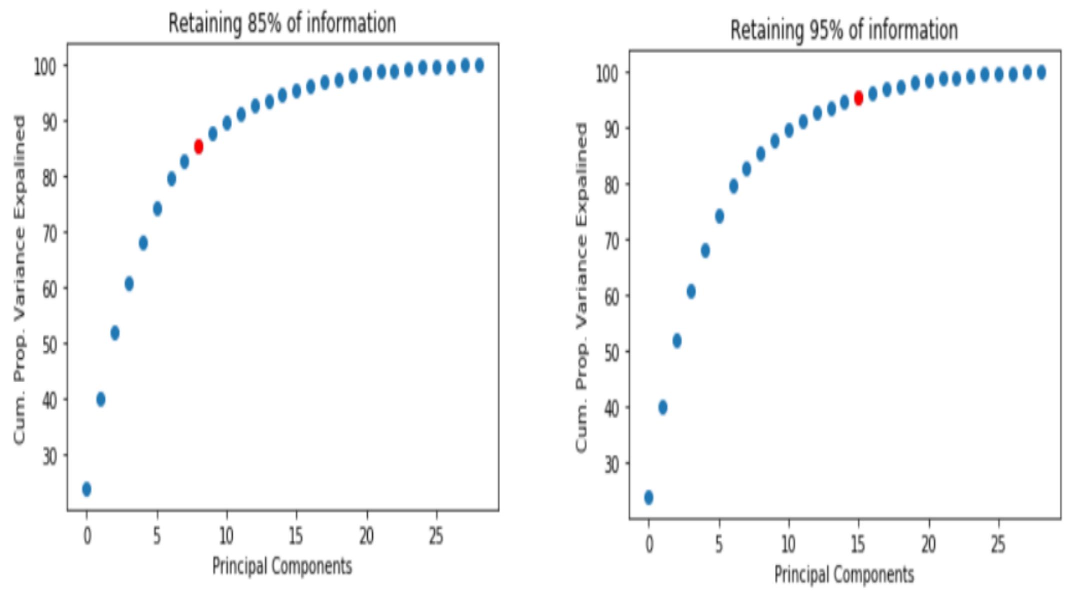
Figure 3: Structure of the Analysis



6.2 Principal component analysis

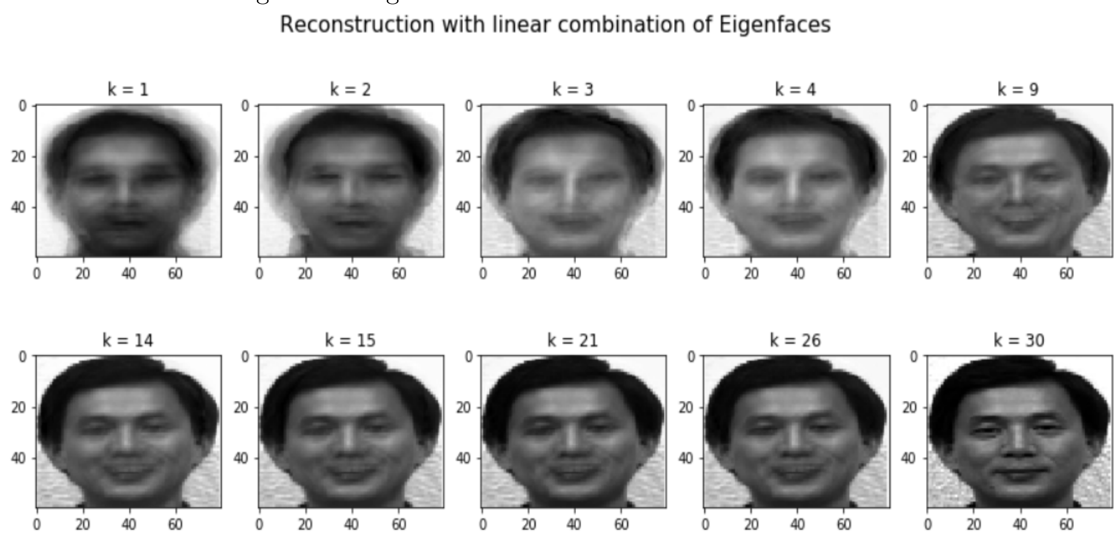
After computing PCA, around 4800 eigenvectors and eigenvalues were obtained for the image size of 80×60 and 1200 eigenvectors and eigenvalues for the image size of 40×30

Figure 4: Explained variance ratio vs Principal components



The optimum number of eigenfaces for with an information loss of 15% and 5% was 9 and 16.
Ideally the number of components should be taken closer to 20 which would capture almost all the information required.The same information could be obtained by reconstructing the face to get an idea an intuitive sense by comparing the reconstructed image with original image.Most of the faces after k=9 look pretty similar to original reconstructed face.

Figure 5: Original face reconstruction



6.3 Observations on evaluation metric

Table 1: Metric evaluation of Image size 40×30

Image Size	Cumulative Explained Variance Ratio	Distance.Metric/KNN Nearest Neighbour	Avg Precision	Avg Recall	Accuracy
40 X 30	0.95	NN classifier =1 ,Manhattan Distance	0.83	0.90	0.90
40 X 30	0.95	NN classifier =3 ,Manhattan Distance	0.82	0.85	0.85
40 X 30	0.95	NN classifier =5 ,Manhattan Distance	0.64	0.65	0.65
40 X 30	0.95	NN classifier =1 ,Minkowski Distance	0.78	0.85	0.85
40 X 30	0.95	NN classifier =3 ,Minkowski Distance	0.82	0.85	0.85
40 X 30	0.95	NN classifier =5 ,Minkowski Distance	0.66	0.70	0.70
40 X 30	0.95	Euclidean Distance	0.78	0.85	0.85
40 X 30	0.85	NN classifier =1 ,Manhattan Distance	0.93	0.90	0.90
40 X 30	0.85	NN classifier =3 ,Manhattan Distance	0.78	0.85	0.85
40 X 30	0.85	NN classifier =5 ,Manhattan Distance	0.65	0.70	0.70
40 X 30	0.85	NN classifier =1 ,Minkowski Distance	0.78	0.85	0.85
40 X 30	0.85	NN classifier =3 ,Minkowski Distance	0.73	0.80	0.80
40 X 30	0.85	NN classifier =5 ,Minkowski Distance	0.73	0.75	0.75
40 X 30	0.85	Euclidean Distance	0.78	0.85	0.85

The highlighted observations shows the best possible combination of Precision,Recall and Accuracy. In this report there more emphasis on taking explained variance ratio of 0.85 as the number of components only 9 with a combination of image size 40×30 since there is no change in the above mentioned scores.The following combination would also be helpful in scenarios where there is limited storage avilability.

The metrics relevant to euclidean distance is almost the same in every scenario signifying that the performance there are potentially less number of parameters that can be tweaked to obtain best possible results.

Table 2: Metric evaluation of Image size 80 × 60

Image Size	Cumulative Explained Variance Ratio	Distance.Metric/KNN Nearest Neighbour	Avg Precision	Avg Recall	Accuracy
80 X 60	0.95	NN classifier =1 ,Manhattan Distance	0.83	0.90	0.90
80 X 60	0.95	NN classifier =3 ,Manhattan Distance	0.78	0.80	0.80
80 X 60	0.95	NN classifier =5 ,Manhattan Distance	0.66	0.70	0.70
80 X 60	0.95	NN classifier =1 ,Minkowski Distance	0.78	0.85	0.85
80 X 60	0.95	NN classifier =3 ,Minkowski Distance	0.82	0.85	0.85
80 X 60	0.95	NN classifier =5 ,Minkowski Distance	0.77	0.75	0.75
80 X 60	0.95	Euclidean Distance	0.78	0.85	0.85
80 X 60	0.85	NN classifier =1 ,Manhattan Distance	0.93	0.90	0.90
80 X 60	0.85	NN classifier =3 ,Manhattan Distance	0.75	0.80	0.80
80 X 60	0.85	NN classifier =5 ,Manhattan Distance	0.65	0.70	0.70
80 X 60	0.85	NN classifier =1 ,Minkowski Distance	0.78	0.85	0.85
80 X 60	0.85	NN classifier =3 ,Minkowski Distance	0.88	0.85	0.85
80 X 60	0.85	NN classifier =5 ,Minkowski Distance	0.73	0.75	0.75
80 X 60	0.85	Euclidean Distance	0.78	0.85	0.85

6.4 Poorest performing class : John

Based on Table 3 it is clear that almost all model perform poorly on this particular class. One possible reason could be that the majority of the faces in

Figure 6: Class John

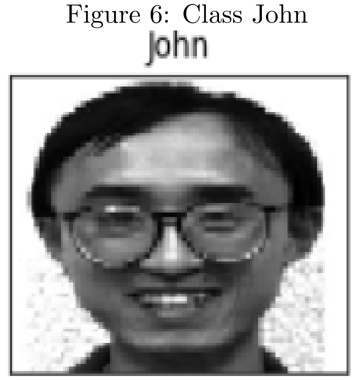


Figure 7: Weights of eigenfaces for Class John Happy

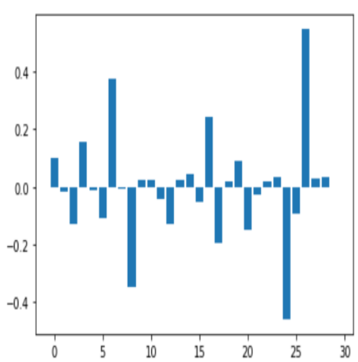


Table 3: Evalution metric of Class John

Subject Name	Model Hyperparameters	Precision	Recall
John	NN classifier =1 ,Manhattan Distance	1	0.500
John	NN classifier =3 ,Manhattan Distance	0	0.000
John	NN classifier =5 ,Manhattan Distance	0	0.000
John	NN classifier =1 ,Minkowski Distance	0	0.000
John	NN classifier =3 ,Minkowski Distance	0	0.000
John	NN classifier =5 ,Minkowski Distance	0	0.000
John	Euclidean Distance	0	0.000

the training dataset have no glasses which could possibly mean that the model requires more dataset for the faces which have glasses. Further, based on figure 9 it is possible that majority of eigenfaces with higher weights belong to eigenfaces after $K = 9$ or $K = 16$ and hence resulting in poor performance at the moment.

6.5 Euclidean Based Distance Metric observation on a class: Hamim

Figure 8: Top 5 Predictions rakings based on Euclidean distance

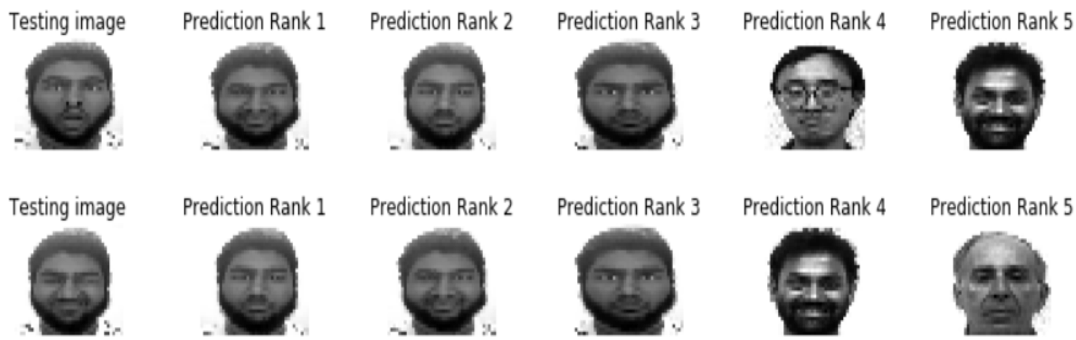


Figure 9: Class based Precision and Recall scores

	precision	recall
Ahmed	1.00	1.00
Anthony	1.00	1.00
Brad	1.00	1.00
Charels	0.50	0.50
Chris	0.67	1.00
Darren	1.00	1.00
Hamim	0.67	1.00
James	1.00	1.00
John	0.00	0.00
Michael	1.00	1.00
accuracy		
macro avg	0.78	0.85

The results comply with observations from Table 2 by having good recall and accuracy but lower precision on euclidean based distance metric .
The lower precision of our euclidean based model could be related to lower precision score of 0.67 within the class **Hamim** as shown in the figure 6.It is potentially caused by limited control of environmental factors like illumination, background and other factors.

Further,Prediction rank 5 of row 1 and prediction rank 4 of row 2 could be expected as one of facial feature of namely beard could have possibly been capture by eigenfaces.

7 Conclusion

Based on the above observations, it is clear that highlighted hyper-parameter with cumulative explained variance ratio of 0.85 and 0.95 with an KNN classifier using K-neighbour value of 1 having a distance metric of Manhattan distance gave best possible results on evaluation metric of precision, recall and accuracy in both image size combinations.However, it is better to pick explained variance ratio of 0.85 with image size of 40×30 for better storage solutions.
Based on recall score, it can be concluded that 90% of total relevant results were correctly classified by the model.
Face image retrieval using eigenface approach is implemented and comparisons among various distance classifiers is done
Based on precision score, it can be concluded 93% proportion of the data points classified by the model were relevant to the existing classes.
To get real scalable solution more data is required to perform better.