Facial Recognition via Eigenfaces and K-means Clustering for Soft Biometrics

Kumar Rohit Malhotra

***Abstract*—Facial recognition is a key aspect of biometric identification. Facial recognition via eigenfaces is one of the widely used methods for this purpose. Even before recognizing a face, it can be classified on the basis of its gender, race, etc. which is known as soft biometric classification. This paper talks about how eigenfaces can be used to implement facial recognition, and how K-means clustering can be useful in soft biometrics.**

***Keywords*—PCA, K-means Clustering, Biometric recognition, Soft biometrics.**

I. Introduction

With the advent of technology, it was important to restrict access to critical information only to the authorized personnel. With due course of time, the need of access control spread to other fields as well, may that be the access to a computer system, or the access to your house. As it was realized that token-based methods like passwords were not enough for implementing the restrictions, especially in critical cases, there was a need for more advanced methods of authentication. Biometric identification is a major realization of this idea, and among its various forms, facial recognition is a key area of authentication.

One of the original and widely used methods of facial recognition is using eigen faces. Eigen faces are obtained by Principal Component Analysis(PCA) of the images. PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. In terms of facial recognition, PCA is used to get the eigen faces or eigen vectors representing the directions for the most variant features, and identify the person on the basis of the similarity between the projections from the stored images and the image to be authenticated. The basic idea of finding the similarity is representing every image as a linear combination of weights(or eigen coefficients) and eigen faces, and finding the distance between the face images using weight vectors, expecting the matching faces to have least distance between each other as compared to their distance from non-matching faces.

When performing large-scale facial recognition, a way to improve performance is to reduce the search space for matches by first performing soft biometric classification. This classification may be on the basis of gender, ethnicity, age, etc. Once soft biometric classification is done, the the person to be authenticated can be compared against the images only from their respective classes for quicker results.

In this paper, we explain how to implement Principal Component Analysis for facial recognition using eigen faces. We begin by getting the PCA projections of the gallery images, the images that are already stored in the database. Then we compare the probe images, the images to be authenticated, by representing each one of them as a weight vector and comparing them with the weight vectors of images from the gallery. This experiment is done for different number of principal components. We compare the results obtained with the recognition performed after PCA projection to the result obtained with the recognition done without PCA projection. Then, we perform k-means clustering on our gallery and probe images on the basis of gender, and observe the performance of the clustering algorithm for various number of principal components by using an internal and an external validity criteria.

This paper was submitted on Nov 28, 2016.

Kumar Rohit Malhotra is with the Department of Computer and Information Science and Engineering at the University of Florida, Gainesville, FL 32611, USA (e-mail: krohitm@ufl.edu)

II. Facial Recognition

We have data in two sets, the gallery set and the probe set. The gallery set has 100 images – one for each individual. This image set is equivalent to the facial images stored in the database. The probe set has 200 images – two for each individual, with the two images varying slightly in their facial expressions. Having two facial images with different facial expressions for each individual helps in identifying the flexibility of the recognition system. The gallery set shall be used for training the recognition system, which will be learning the PCA basis to be then used to project images as features to be used for recognition. The probes set is used for testing the recognition system, by finding the distance of projected probe images from the projected gallery images using the eigen coefficients.

A. PCA projection of data

We begin by finding the principal components of the data. We have 100 images in the gallery set. Each of our image has a dimension 50 x 50. We first reshape each image in the gallery set to a 2500 x 1 column vector, and combine these column vectors to get a 2500 x 100 matrix. We obtain the mean of the face images, and subtract it from the matrix, thereby resulting in a normalized data matrix A. We then get the covariance matrix for the normalized matrix using $S = AA^T$, and find the eigen vectors and corresponding eigen values of this covariance matrix. These eigen vectors are the Principal Components of the facial images. The eigen vectors comprise an uncorrelated orthogonal basis set. High

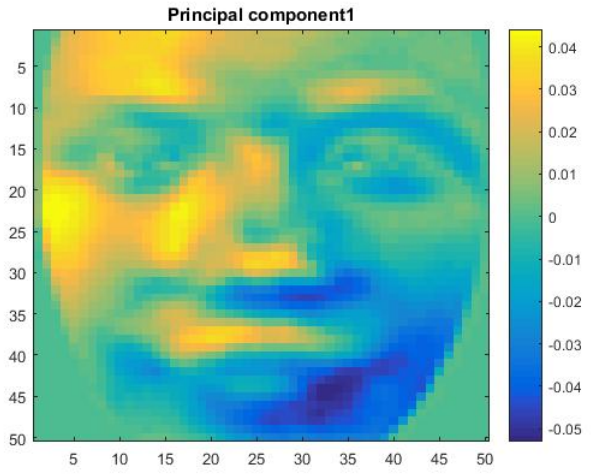

Fig. 1: Principal Component 1

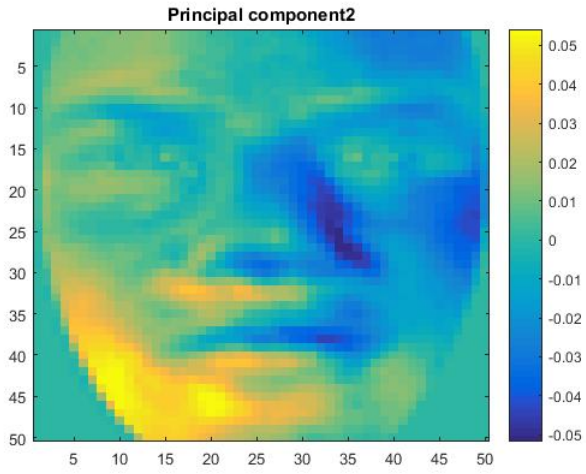

Fig. 2: Principal Component 2

eigen values correspond to high variance in the direction of the corresponding principal component. Since the number of samples is less than the features, the matrix $S = AA^T$ won't be a full rank and hence most of the eigen values would be zero, which will anyways be of no use as we would be taking only the eigen vectors with high variances. Moreover, it results in higher computation cost. Instead we can find the covariance matrix $S = A^TA$, and then from the obtained eigen vectors from this covariance matrix, find the most important eigen vectors using the formula $u = Av$, where A is the normalized matrix and v is the matrix of eigen vectors obtained previously. The first three principal components are displayed as images in Fig. 1., Fig. 2. and Fig. 3.

The first three principal components depict the directions of the highest variance in the data. However, in terms of facial recognition, high variance may be observed because of noisy factors such as high luminance. As such, the the first principal component may be representing the most noise in the data.

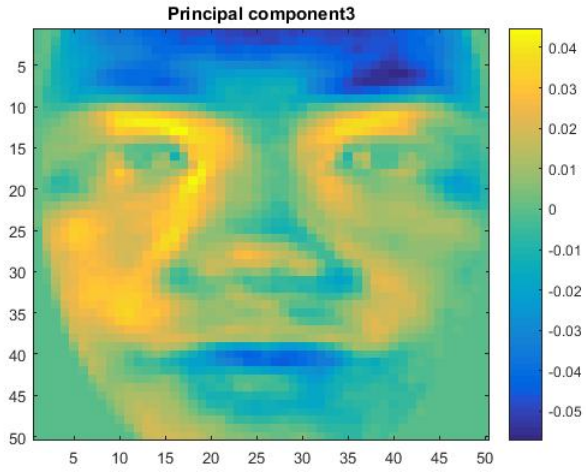

Fig. 3: Principal Component 3

B. Recognition Performance

From the principal components obtained above, we find the weight vectors for the images. First we normalize the gallery set, as we did before finding the eigen vectors. Then we project the gallery image vectors onto a certain number of principal components only, which will give us the weight vectors, using equation 1:

$$\omega_j = {u_j}^T \Phi_i \tag{1}$$

where Φ_i is the normalized i^{th} image vector, ${u_j}^T$ is the j^{th} eigen vector, and ω_j is the j^{th} weight or eigen coefficient of the i^{th} image. While doing so, we take the principal components in order of decreasing variance, represented by their corresponding eigen values. Each weight vector will be a column vector, with each such vector representing the projection of an image onto the selected principal components. Thus, we get the PCA projections of the gallery set. Any image can be represented as a linear combination of its weights and eigen vectors. In the same way, we get the PCA projections of the normalized probes set, for the same number of principal components.

Next step is finding the distance between the gallery images and the probe images. We measure this distance as the euclidean distance in our case. Euclidean distance between any two points (u,v) in an n dimensional space can be calculated by using equation 2.

$$d(u,v) = d(v,u) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2} \tag{2}$$

TABLE I: Recognition Performance for various number of principal components after ignoring various numbers of principal components

	0	1	2	3	4	5	6	7	8
10	56	52.5	49	48	52.5	54.5	55	53.5	56.5
20	65	63	60.5	60	67	69	67.5	65.5	70.5
30	68.5	68	63	65.5	72	72.5	72	72.5	72
40	71	69	67	68.5	75.5	75.5	74.5	74	72
50	72	71	67.5	70	76.5	77	76	74.5	75
60	73	73	68.5	72.5	79.5	78	76.5	76	76.5
70	74.5	74	70	72.5	80	79.5	78.5	77.5	77
80	75	74.5	70.5	73	80	80.5	80	79	78.5
90	76	75.5	71	73.5	80.5	81	82.5	81	81
100	76.5	75.5	71.5	74	80.5	81	82	81.5	81.5

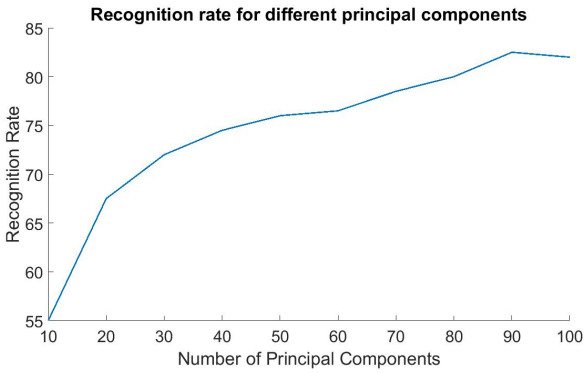

Fig. 4: Recognition Rate for different number of principal components

For every image in the probe set, we measure the euclidean distance of the image from every image in the gallery set, using their weight vectors. If the distance between a probe image is minimum from its corresponding gallery image as compared to its distance from other gallery images, we call it a match; otherwise if the distance of a probe image is minimum from some non-corresponding gallery image as compared to its corresponding gallery image, we call it a mismatch.

This experiment is repeated for different number of principal components. As the principal components representing highest variance may depict noise, we ignore the first few principal components when calculating the weight vectors for the gallery images and the probe images. The choice of how many principal components should be ignored is made heuristically by ignoring different number of high variance principal components. Table 1 shows the recognition performance when i principal components with the highest variance were ignored and then next j principal components with the highest variance were selected, the columns' headings denoting i and rows' index denoting j .

A few observations can be made from this table. We can see that the recognition performance monotonically increases with the increasing number of principal components, with being the lowest when only ten principal components are selected. The reason for this is that when we are projecting the 2500 features of an image onto the principal components and selecting only 10 of them, even though we are selecting the principal components representing high variance, we are ignoring too much information, in terms of features, which separates the various images. As we continue using more and more principal components, we are comparing the images on the basis of more features, which consequently is giving us more information about each image.

Another observation that can be made is that after a certain number of principal components, increase in principal components doesn't improve the performance of the system any further. In fact, for the highest accuracy obtained i.e. 82.5% in case of ignoring first 6 principal components, the performance slightly deteriorates after going from 90 principal components to 100 principal components. This is because principal components represent the directions with the highest variance and hence most of the required information is obtained from the projections on the principal components with the highest variance itself. The principal components after a certain number don't give much information on the variance of data and thus projections on those components won't help in differentiating the images any more than what has already been achieved. As such, after a certain number of principal components, the recognition performance becomes stable and doesn't improve any further.

Another observation that can be made is that better performance is seen when the first few principal components are ignored. This is because in terms of facial recognition, the principal components with the highest variance may be representing noisy features. Fig. 4. shows the recognition rate as a function of the number of principal components used in case of the highest accuracy obtained, which was in the case of ignoring the first six principal components with high variance. The comparison of performance seen by dropping various number of highest variance principal components can be seen in Table 1 too.

Using Euclidean distance as the distance measure and the original images as feature vectors, the recognition performance was found to be 76.5% while the maximum accuracy found by using the PCA projection of the data was 82% obtained by ignoring first 6 principal components with the highest variance and then using the next 90 principal components with the highest variance.

III. Soft Biometrics

In our next step, we perform k-means clustering on our data to find how well separable it is after PCA projection. For this, we use both the gallery and the probes set. After clustering the images, we evaluate the clustering on the basis of an internal validity criteria and an external validity criteria. In our case, we have used Dunn's Index for internal validity and Pairwise F measure for external validity criteria. We have a gender data set having the gender of each individual, which we'll use as labels when evaluating external validity.

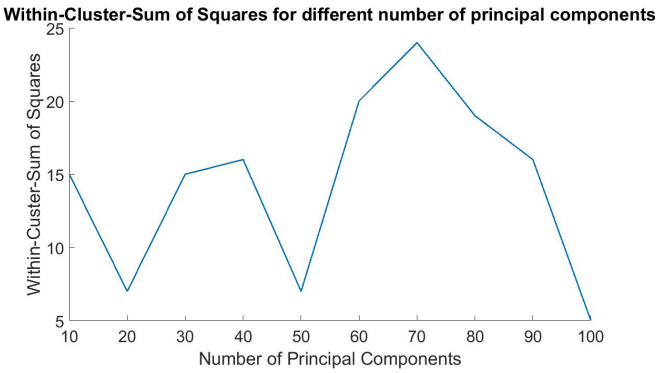

Fig. 5: Within-Cluster Sum of Squares for different number of principal components

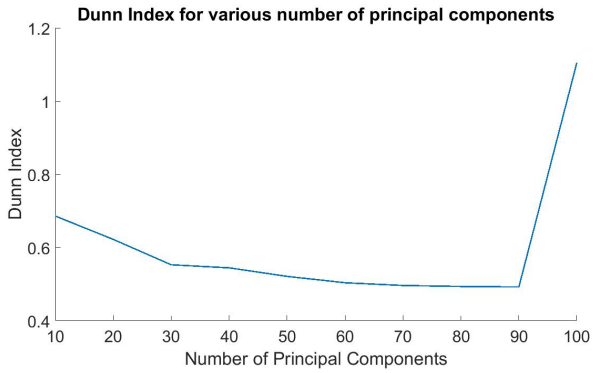

Fig. 6: Dunn's Index for various number of principal components

A. Clustering

We perform k-means clustering on the weight vectors that we obtained earlier, by ignoring the first 6 principal components. Given a set of observations $(x_1, x_2, ..., x_n)$, where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k ($\leq n$) sets S = $S_1, S_2, ..., S_k$ so as to minimize the within-cluster sum of squares (WCSS) (sum of distance functions of each point in the cluster to the K center). In other words, its objective is to find:

$$arg \min_{s} \sum_{i=1}^{k} \sum_{x \in S_i} \| x - \mu_i \| \tag{3}$$

where μ_i is the mean of points in S_i. This is achieved by randomly initializing K centroids in the neighborhood of the data, and alternatively updating the membership matrix of the K clusters such that the WCSS is minimum, and updating the new K centroids. For initialization of K centroids, we have implemented the Forgy method, which randomly chooses k observations from the data set and uses them as initial means.

In our case, since we are forming the clusters to demonstrate clustering for the genders, we take K to be 2. While assigning the membership to each image, it is ensured that the image from one cluster does not belong to the other. In case of ties, membership is assigned arbitrarily. These updates are done until the clusters become stable. We perform this experiment for various number of principal components and obtain the WCSS as shown in Fig. 5. The minimum WCSS obtained according to this graph is when 100 principal components are used.

B. Evaluation of Clustering

We first evaluate the clustering using an internal criteria Dunn's Index[1]. Let S and T be two non-empty subsets of R^N. Then the diameter Δ of S and set distance δ between S and T are:

$$\Delta(S) = \max_{\overline{x}, \overline{y} \in S} d(\overline{x}, \overline{y}) \tag{4}$$

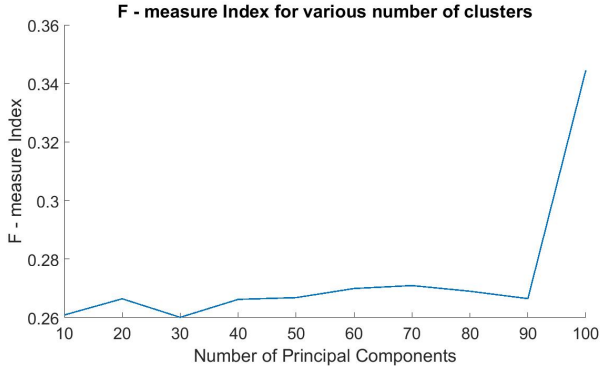

Fig. 7: Pairwise F measure for various number of principal components

and

$$\delta(S, T) = \min_{\overline{x} \in S, \overline{y} \in T} d(\overline{x}, \overline{y}) \tag{5}$$

where $d(\overline{x}, \overline{y})$ is the distance between points $\overline{x}$ and $\overline{y}$. For any partition, Dunn defined the following index:

$$Dunn = \min_{1 \leq i \leq K} \{ \min_{1 \leq j \leq K, j \neq i} \{ \frac{\delta(C_i, C_j)}{\max_{1 \leq k \leq K} \Delta(C_k)} \} \} \tag{6}$$

Large values of Dunn correspond to good clusters. We calculate the Dunn's index for various number of principal components, always taking K = 2. Fig. 6. shows the graph for Dunn's index for various numbers of principal components. As per the figure, the best value of Dunn's index is obtained for 100 principal components.

We also evaluate the clustering on the basis of an external criteria Pairwise F-measure[2]. To evaluate the clustering results, precision, recall, and F-measure were calculated over pairs of points. Precision is calculated as the fraction of pairs correctly put in the same cluster, recall is the fraction of actual pairs that were identified, and F-measure is the harmonic mean of precision and recall. Fig. 7. shows the graph for F-measure index for various number of principal components. The best clusters are obtained when 100 principal components are used.

TABLE II: Average WCSS, Dunn's Index and F measure for various number of principal components

Principal Components	WCSS	Dunn's Index	Pairwise F-measure
10	10.3	0.5917	0.2747
20	9.3	0.6815	0.2813
30	15.3	0.6028	0.2711
40	12.9	0.5649	0.2721
50	12.6	0.5489	0.2666
60	10.1	0.6582	0.2786
70	15.5	0.54	0.2634
80	13.1	0.5354	0.266
90	16.6	0.5411	0.2633
100	10.2	0.5379	0.2667

The three graphs obtained in Fig. 5, 6 and 7 are for a particular set of experiments. For another set of experiments, the results may be different. This is because k means clustering is a heuristic algorithm and there is no guarantee that it will converge to the global optimum. As such, the results may depend on the initial clusters which directly depends on the initial means. In some cases, the external and internal validity criteria may not be agreeing upon the same clusters to be the best. Hence, in order to ensure the quality of clusters, we should experiment multiple times with different initial means. An average of WCSS, Dunn's index and Pairwise F-measure over fifteen different sets of experiments with different initial means is given in table 2. The graphs for these 3 averages are shown in Fig. 8, 9 and 10 respectively.

From all three of these graphs, we can see that the best clustering is obtained for 20 principal components, looking at either of the parameters viz. WCSS, internal criteria and external criteria.

Another observation that can be made here is that the trends in clustering don't tell anything directly about the trends in identification. In terms of recognition, we observed higher performance for increasing number of principal components. No such trend is seen in the clustering of the data. For the number of principal components giving the best k-means clustering, the performance of the recognition system was not up to the mark as compared to its best performance. It can be seen that for most of the part, the quality of clusters decreases, with a sudden spike for 60 principal components.

Looking at the second best results of clustering, which is obtained for 60 principal components, we can see that the performance of the recognition is quite near the best performance for these many number of clusters. Because of this, even though there are no matching trends between the recognition system and the clustering, the two systems are giving a good performance for the same number of principal components i.e. 60 in our case.

There probably shouldn't be such a spike as we continue with increasing the number of principal components. This is because as we will increase the number of principal components beyond 100 in our case, we have seen earlier that the principal components after that don't represent much variance.

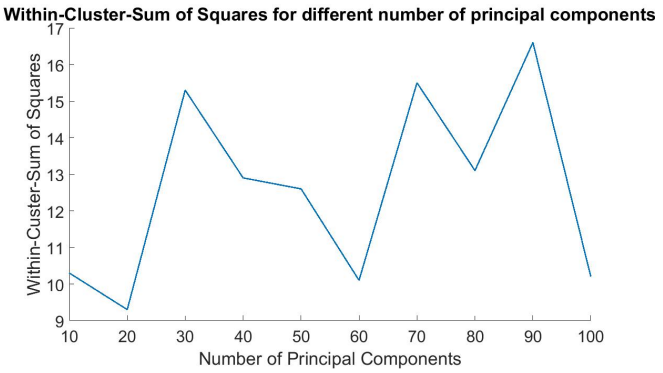

Fig. 8: Average Within-Cluster Sum of Squares for various number of principal components

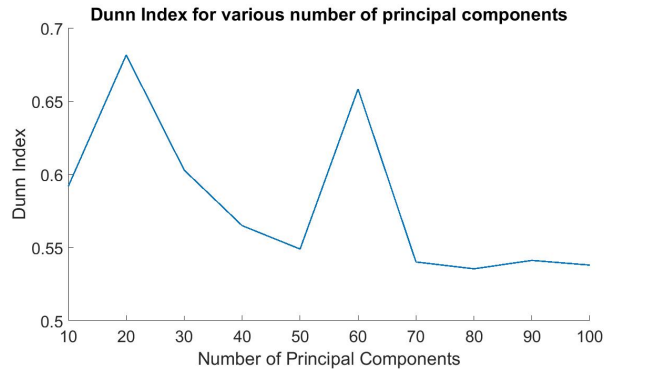

Fig. 9: Average Dunn's Index for various number of principal components

Hence, in terms of clustering also, those principal components shouldn't be playing much role in changing the variance of the projected data. So, the clusters won't change much for increasing principal components.

IV. Conclusion

We can draw a few conclusions from our experiments. We saw that the principal components with the highest variance don't give the same advantage in facial recognition as it may do in other fields. Discarding a few principal components with high variance may result in better performance, as it may help in removing the noise. Additionally, the performance increases with increasing number of principal components, and becomes stable after a certain number. We saw that the recognition done by PCA projection may give even better results than direct recognition. We also saw that clustering, in the field of soft biometrics may help in classifying the data on the basis of a trait, such as gender, which may help in easier identification, if we allow some compromise in the performance.

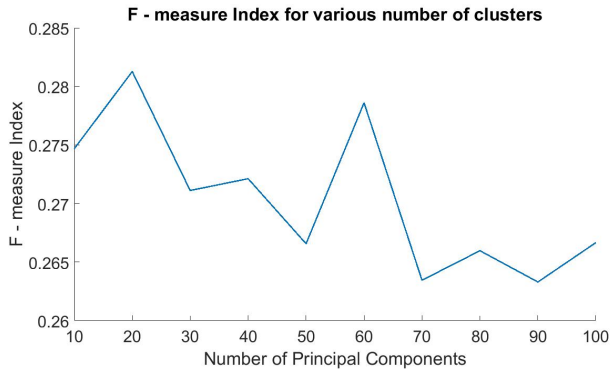

Fig. 10: Average Pairwise F measure for various number of principal components

REFERENCES

[1] Dunn, J. C. A Fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J. Cyber. 3, 1973, pp. 32-57.

[2] A Banerjee et al. Model-based Overlapping Clustering. KDD05, August 2124, 2005, Chicago, Illinois, USA. Copyright 2005 ACM 1-59593-135-X/05/0008