

PCA and KPCA algorithms for Face Recognition – A Survey

Surabhi M. Dhokai¹, Vaishali B. Vala², Vatsal H. Shah³

¹Department of Information Technology, BVM Engineering College, surabhidhokai@gmail.com

²Department of Information Technology, BVM Engineering College, vaishalivala305@gmail.com

³Assistant Professor, BVM Engineering College, vatsal.shah@bvmengineering.ac.in

Abstract— Face Recognition has emerged as an important alternative to traditional authentication systems as the terror of Cybercrime shadows the network security. This report pins down the basic steps involved in face recognition technique along with its two common challenges. It also surveys the PCA and KPCA algorithm for face recognition.

Keywords - Face Recognition, PCA, eigen faces technique, KPCA, Kernel trick, Curse of dimensionality

I. INTRODUCTION

The innovations in the technology have not only increased the affordability but have also revolutionized the way transactions are carried out over the past few years. However the networked world poses a different form of threat to the world of security: Cybercrime. Due to the limitations of the traditional methods (access codes and passwords) there arises a need for a better, accurate and reliable form of user identification and authentication technology.

Biometrics is nowadays increasingly used for authentication purpose. It utilizes biological properties like: fingerprint, iris recognition, DNA, retina, face recognition etc. which are difficult to forge.

II. FACE RECOGNITION SYSTEMS

The automated face recognition is essentially defined as the process of verifying or determining the identity of the person given an input face image and database of images of the known individuals. It can be achieved by comparing the features of the face (which don't change over time). Face recognition offers following advantages [1] over other biometric techniques:

- 1) Non-intrusive and can be done from a distance without a user being aware about it.
- 2) Hygienic as no physical contact is involved.
- 3) Can be used for surveillance.
- 4) It can use existing hardware infrastructure.

2.1 Process Overview

In general, the automated face recognition system comprises of following steps as shown in Figure 1. [1]

- 1) Acquire: During this step the image containing the face is captured.
- 2) Detect: Face detection segments the face area from the background.
- 3) Align: The face may not be completely perpendicular to the camera and hence difficult to recognize and hence needs to be aligned.
- 4) Extract: This step involves creation of face template, face print using the facial features that is useful for distinguishing between faces of different people.
- 5) Match: During this step the face template or the face print generated for the input image is matched with stored images in the database to generate scores.

6) Report: The matches are then made according to the scores generated.

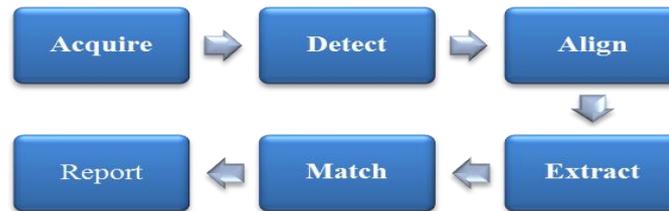


Figure 1. Face Recognition process view

2.2 Challenges

Despite various advantages of face recognition over other biometric system for user identification and authentication and there are certain variant parameters of an image which add complexity to the recognition problem. The two most prominent variant parameters [2, 3] plaguing the face recognition system includes:

- 1) **Illumination problem**: The changes in the appearance induced due to variation in the light can be more striking than the one due to difference in identities. Figure 2. [3] Illustrates the variation in a single image due to different lighting.
- 2) **Pose variations**: The pose variations occur due to variations in the angles of camera during the image acquisition process. Figure 3. [3] Illustrates the variation in single image due to change of pose.



Figure 2. Changes in the single image under different illumination conditions.



Figure 3. Changes in the single image due to pose variations.

III FACE RECOGNITION APPROACHES

There are two predominant approaches to face recognition problem: (i) Feature based (ii) Holistic.

In Feature based approach [4], during the first stage the local features of face like location of eyes, mouth, nose, head outline etc. are derived from the raw intensity data. In the next stage a graph representation of the face using the extracted features is constructed and in the last stage this graph is utilized for match making purpose. Hence the task of face recognition is accomplished. There are various algorithms to implement feature based approach like: Elastic Bunch Graph Matching.

In Holistic approach, [5] a little number of distinguishable features are derived from the pixel information of the input image which are then utilized for uniquely identifying one person from the other. The various algorithms used to implement the holistic approach include: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), etc.

The taxonomy for the face recognition algorithms is shown in Figure 4 .

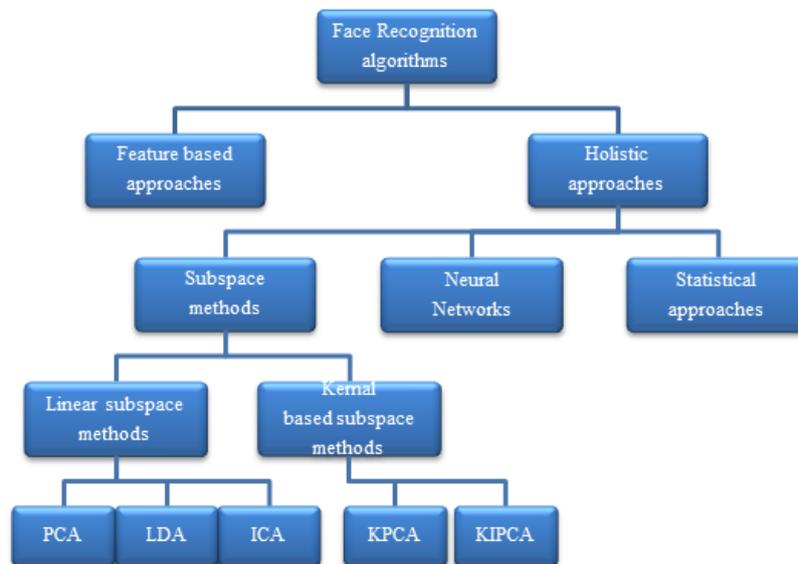


Figure 4. Taxonomy of face recognition algorithms

3.1 Principal Component Analysis (PCA)

An image space can be considered analogous to any other vector space with dimension equal to the number of pixels making up the image and having values equal to pixel value. Thus any image can be considered as a point in the “image space” and hence can be expressed as a vector.

For example: 100x100 image = 10,000 dimensions

Thus, any image in this image space would be a 100x100 vector. Hence, for effective usage large amount of resources both for storage as well as computation would be required. Moreover very few of these high dimensional vectors would be face images and therefore useful. This problem of making high dimensional data to effective use is termed as “curse of dimensionality”. PCA also known as Eigen face method aims to address this problem of “curse of dimensionality” by building a face space which better describes a face.

PCA is statistical dimensionality reduction method where a set of observations of correlated variables are mapped onto a set of linearly uncorrelated variables known as principal components using orthogonal transformations. It is derived from Karhunen-Loeve’s transformation. Hence the number of principal components is less than the number of correlated variables. Thus, PCA maps data from high dimensional space to low dimensional sub-space.

A Pentaland and M. Turk proposed an eigen face method based on PCA in 1991. It is one of the most successful techniques for face recognition. The method uses Principal Component Analysis (PCA) on a set of images to extract principal components or the distinct features from the set of images. Hence using this technique each face image is represented by a single vector known as eigen vector. The set of eigen vectors derived from the covariance matrix of the probability distribution over the high dimensional vector space of face images is known as eigen space. Once eigen space is found then each new face can be constructed using the combination of this eigen space depending upon the nature of weights.

3.1.1 PCA for face recognition

The basic steps involved in the algorithm based on eigen face are as under [1]:

Step 1: Set of face images collectively known as training set is first acquired. The images in the training set should have been taken under same lightning conditions, must be aligned and resampled to common pixel resolution.

Step 2: From this training set a set of M eigen faces are calculated. This is known as face space.

Step 3: Afterwards the weights for each image in the face space is calculated and stored in the vector W.

Step 4: Then the weight for the input image X is calculated and stored in the vector Wx.

Step 5: Afterwards the weight vector Wx is compared with the weights of the images in the face space by calculating the average distance D between weight vectors from W and that of input image Wx.

Step 6: Based on the weight pattern the face is recognized as known or unknown face. If the value of D exceeds some threshold value Θ then weight vector of the input image Wx lies far apart and input image X is not considered a face. Otherwise Wx is looked for a match from the database.

The basic model for the PCA algorithm is as shown below [6]:

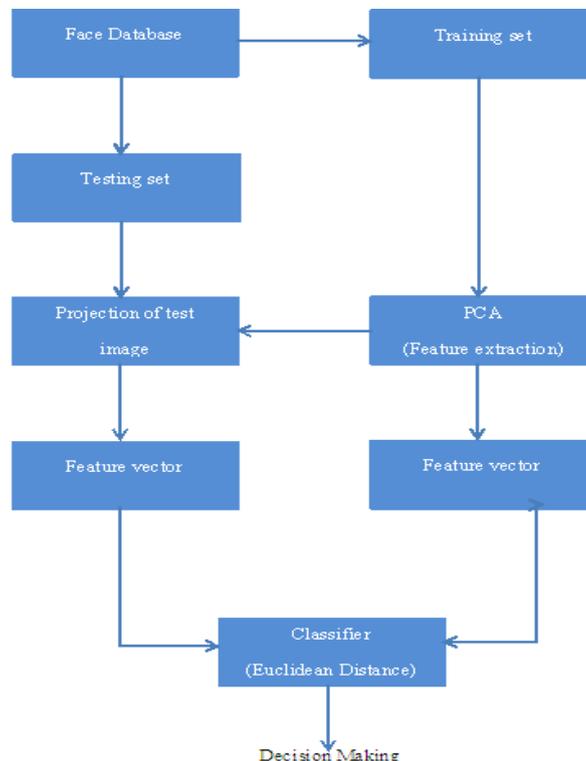


Figure 5. Model for PCA algorithm

3.1.2 Limitations

- 1) The direction with the largest variance is assumed to be of the most interest.
- 2) Dimension reduction can only be achieved if the original variables were correlated. If the original variables were not related, PCA does nothing, except for ordering them according to their variance.
- 3) PCA is not scale variant.
- 4) PCA is based only on the mean vector and the covariance matrix of the data. Some distributions are completely categorized by this, but others are not.
- 5) Only allows linear dimensionality reduction [7].

3.2 Kernel Principal Component Analysis (KPCA)

PCA aims at reducing the dimensionality of the data space by extracting the “right” features or rather the principal structure of the dataset in order to reduce time complexities. The basis of our new coordinate is principal components. In many cases small number of principal components is sufficient to describe most of the structure in the dataset. However, sometimes PCA ignores some of the vectors essential to describe the dataset and hence fails in extracting the high order information of the image. Kernel PCA was therefore proposed to solve this problem. [8]

Kernel PCA is a nonlinear PCA created using the kernel trick. Kernel PCA first maps the original sample images into the higher dimensional feature space using the kernel method. The kernel trick is a way of mapping observations from a general set S into an inner product space V without having to compute the mapping explicitly because the observations will gain meaningful linear structure in V . The trick or the method is used to avoid explicit mapping by utilizing an appropriate kernel function.

The general model for the KPCA algorithm is as shown in Figure 6 [7].

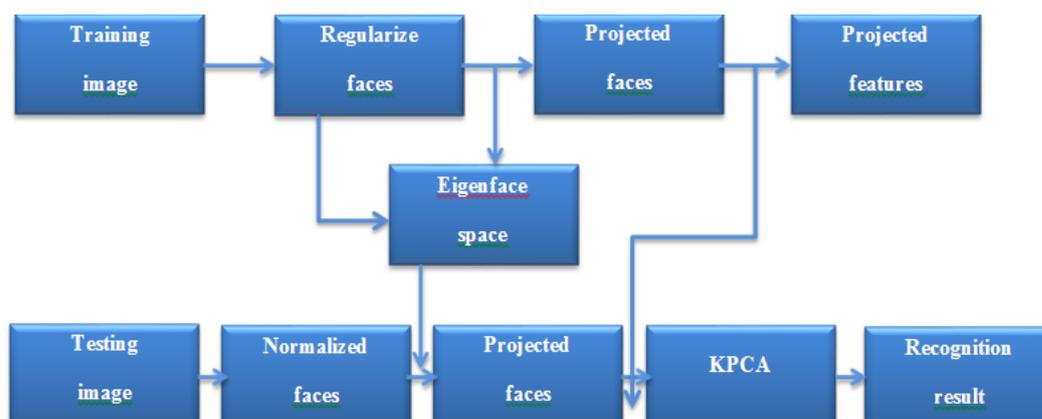


Figure 6. Model for KPCA algorithm

3.2.1 KPCA for face recognition

The steps involved in the face recognition based on KPCA are as follows [8]:

Step 1: The face image $I (M, N)$ is rewritten as a column vector $I (M*N, 1)$, where M and N are number of rows and columns of I , respectively.

Step 2: Then the centralized Kernel matrix of the training examples is constructed.

Step 3: Next the eigenvalues and the eigenvectors of the constructed kernel matrix are calculated.

Step 4: Feature extractors are then acquired by collecting the eigenvectors corresponding to the first several largest eigenvalues in RF.

Step 5: Then the nonlinear features of the training and the test examples are obtained by mapping them onto the feature extractor.

Step 6: Then a classifier such as nearest neighbor algorithm is used to classify the test faces.

3.2.2 Limitations

Even though unlike PCA, KPCA is an effective means of capturing the nonlinear features of the image like the face curves and outperforms other Eigen face based methods for face recognition but the computational time required by KPCA is higher than that required by PCA.

IV. CALCULATION

In the following section we show the calculation of eigen space and kernel matrix essential for the implementation of the algorithms.

4.1 Eigen Space Calculation

The eigen space can be calculated using the following steps:

- 1) The first step is to obtain a set S with M face images. Each image is transformed into a vector of size N and placed into the set.

$$S = \{ I_1, I_2, I_3, I_4, \dots, I_m \}$$

- 2) After you have obtained your set, you will obtain the mean image Ψ .

$$\Psi = \frac{1}{M} \sum_{n=1}^m I_n$$

- 3) Then you will find the difference Φ between the input image and the mean image.

$$\Phi_i = I_i - \Psi$$

- 4) Next we seek a set of M orthonormal vectors, u_n , which best describes the distribution of the data. The k^{th} vector, u_k , is chosen such that

$$\lambda_k = \frac{1}{M} \sum_{n=1}^M (u_k^T \Phi_n)^2$$

$$u_l^T u_k = \delta_{lk} = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{otherwise} \end{cases}$$

Here u_k and λ_k are the eigen vectors and the eigen values of the covariance matrix C .

- 5) We obtain the covariance matrix C in the following manner.

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T$$

$$= AA^T$$

$$A = \{\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_n\}$$

6) A^T

$$L_{mn} = \Phi_m^T \Phi_n$$

7) Once we have found the vectors v_1, u_1

$$u_l = \sum_{k=1}^M v_{lk} \Phi_k \quad l = 1, \dots, M$$

4.2 Construction of Kernel Matrix

Suppose $\Phi(x)$ is some nonlinear transformation from the original D-dimensional feature space to an M-dimensional feature space, where $M \gg D$. Then each data point x_n is projected to a point $\Phi(x_n)$. We can perform traditional PCA after this mapping of data points to new feature space however the computational cost is high and hence kernel methods are used.

In order to derive the kernel matrix following steps need to be followed [9]:

Assume that the projected new features have zero mean:

$$\sum_n \Phi(x_n) = 0$$

The covariance matrix of the projected features (MxM) is calculated by following equation:

$$C = \frac{1}{N} \sum_{n=1}^N \Phi(x_n) \Phi(x_n)^T$$

And its eigenvalues and eigenvectors are given by

$$C v_i = \lambda_i v_i \text{ where } i = 1, 2, \dots, M$$

Hence we have,

$$\frac{1}{N} \sum_{n=1}^N \Phi(x_n) \{\Phi(x_n)^T v_i\} = \lambda_i v_i$$

Therefore

$$v_i = \sum_{n=1}^N \Phi(x_n) a_{in}$$

Hence substituting the value of v_i we have

$$\frac{1}{N} \sum_{n=1}^N \Phi(x_n) \Phi(x_n)^T \sum_{m=1}^N \Phi(x_m) a_{im} = \lambda_i \sum_{n=1}^N \Phi(x_n) a_{in}$$

If the kernel function is defined as

$$k(x_n, x_m) = \Phi(x_n)^T \Phi(x_m)$$

Multiplying either side of above equation by $\Phi(x_i)^T$,

$$\frac{1}{N} \sum_{n=1}^N k(x_n, x_m) \sum_{m=1}^N k(x_n, x_m) a_{im} = \lambda_i \sum_{n=1}^N k(x_n, x_m) a_{in}$$

Or the matrix notation

$$K^2 a_i = \lambda_i N K a_i$$

Where

$$K_{n,m} = k(x_n, x_m)$$

And a_i is the N-dimensional column vector of a_{ni} . a_i can be solved by

$$K a_i = \lambda_i N a_i$$

And the resulting kernel principal components can be calculated using

$$y_i(x) = \Phi(x_n)^T v_i$$

If the projected dataset does not have zero mean then the Gram matrix can be used to substitute the kernel matrix. The gram matrix is given by:

$$\tilde{K} = K - \mathbf{1}_N K - K \mathbf{1}_N + \mathbf{1}_N K \mathbf{1}_N$$

Where $\mathbf{1}_N$ is an NxN matrix with all the elements equal to 1/N.

V. CONCLUSION

Face Recognition has gained hefty attention from researchers over the past few years because of its potential for applications in various domains. There has been rigorous research in progress to develop an effective algorithm for same. In this report PCA and KPCA two most practiced techniques for face recognition are surveyed. PCA though being easy to implement and requiring less computational time as compared to KPCA and various other linear subspace techniques, has the limitation of being scale invariant. Hence is not suitable under all conditions.

KPCA on the other hand successfully extracts the nonlinear features of face but yet has the limitation of requiring high computational time and hence being infeasible for implementation.

Hence, there remains a way for the development of new algorithms as well as improvement of present algorithms for better and effective face recognition.

REFERENCES

- [1] Atul Gupta, Vikas Dewangan and V.V Ravi Prasad, "Facial Recognition – A Primer", Infosys – White Paper, pp.2-4.
- [2] Preeti B. Sharma, Mahesh M. Goyani, "3D Face Recognition Techniques – A Review", International Journal of Engineering Research and Applications(IJERA), pp. 788, 2012.
- [3] Wei-Lun-Chao, "Face Recognition", pp.13.
- [4] Rabia Jafri, Hamid R. Arabnia, "A Survey of Face Recognition Techniques", Journal of Information Processing Systems, Vol 5, No 2, pp. 43, 2009
- [5] Stewart Tseng, "Comparison of Holistic and Feature based approaches for Face Recognition", pp. 13
- [6] Anshuman Prakash, Manoj Kumar Tewari, "Study of different algorithms for Face recognition", pp. 31
- [7] Ritu Upadhyay, Rakesh Kumar Yadav, "Kernel Principle Component Analysis in Face Recognition System: A survey", International Journal of Engineering Research and Applications(IJERA), pp. 351, 2013
- [8] Wangli Hao, Jianwu Li and Xiao Zhang, "Learning KPCA for Face Recognition", 142-143
- [9] Quan Wang, "Kernel Principal Component Analysis and applications in Face recognition and active shape models", pp. 2, 2011