An Efficient LDA Algorithm for Face Recognition

Jie Yang, Hua Yu, William Kunz
School of Computer Science
Interactive Systems Laboratories
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

It has been demonstrated that the Linear Discriminant Analysis (LDA) approach outperforms the Principal Component Analysis (PCA) approach in face recognition tasks. Due to the high dimensionality of a image space, many LDA based approaches, however, first use the PCA to project an image into a lower dimensional space or so-called face space, and then perform the LDA to maximize the discriminatory power. In this paper, we propose a new, unified LDA/PCA algorithm for face recognition. The new algorithm maximizes the LDA criterion directly without a separate PCA step. This eliminates the possibility of losing discriminative information due to a separate PCA step. We discuss the connection between the new algorithm and the traditional PCA+LDA approach. We also prove that the new algorithm is equivalent to the eigenface (PCA) approach in a special case, where each person has only one sample in the training set. The feasibility of the new algorithm has been demonstrated by experimental results.

1. Introduction

Automatic face recognition has been an active research area in the last decade. The progress in this area can be found in review papers [11, 2] and proceedings of last four international conferences on face and gesture recognition. Among various approaches, techniques based on Principal Components Analysis (PCA) [7, 12], popularly called *eigenfaces* [15, 10], have played a fundamental role in dimensionality reduction and demonstrated excellent performance. PCA based approaches typically include two phases: training and classification. In the training phase, a eigen-space is established from the training samples using the principal components analysis method. The training face images are then mapped onto the eigen-space. In the classification phase, the input face image is projected to the same eigen-space and classified by an appropriate method. Many

different methods have been used for face recognition, such as the Euclidean distance [15], Bayesian [9] and Linear Discriminant Analysis (LDA) [14, 1, 3, 19, 8].

Unlike the PCA which encodes information in an orthogonal linear space, the LDA encodes discriminatory information in a linear separable space of which bases are not necessarily orthogonal. Researchers have demonstrated that the LDA based algorithms outperform the PCA algorithm for many different tasks [1, 19]. However, the standard LDA algorithm has difficulty processing high dimensional image data. PCA is often used for projecting an image into a lower dimensional space or so-called face space, and then LDA is performed to maximize the discriminatory power. In those approaches, PCA plays a role of dimensionality reduction and form a PCA subspace. The relevant information might be lost due to inappropriate choice of dimensionality in the PCA step [20]. However, LDA can be used not only for classification, but also for dimensionality reduction. For example, the LDA has been widely used for dimensionality reduction in speech recognition [6]. LDA algorithm offers many advantages in other pattern recognition tasks, and we would like to make use of these features with respect to face recognition as well. In this paper, we propose a unified LDA/PCA algorithm for face recognition. The new algorithm maximizes the LDA criterion directly without a separate PCA step. This eliminates the possibility of losing discriminative information due to a separate PCA step. We also discuss the connection between the new algorithm and the traditional PCA+LDA approach, and prove that the new algorithm is equivalent to the eigenface (PCA) approach in the special case where each person has only one sample in the training set. The feasibility of the new algorithm has been demonstrated by experimental results.

The remainder of the paper is structured as follows. Section 2 reviews LDA algorithms for face recognition and presents a new LDA algorithm. Section 3 comments the new algorithm. Section 4 describes the datasets and experimental results. Section 5 summarizes the paper.

2. A Direct LDA Algorithm

2.1 LDA for Face Recognition

The basic idea of LDA is to find a linear transformation such that feature clusters are most separable after the transformation. This can be achieved through scatter matrix analysis [4]. For an M-class problem, the between- and within-class scatter matrices S_b and S_w are defined as:

$$S_b = S_b = \sum_{i=1}^{M} Pr(C_i)(\mu_i - \mu)(\mu_i - \mu)^T = \Phi_b \Phi_b^T, \quad (1)$$

$$S_w = \sum_{i=1}^{M} Pr(C_i)\Sigma_i = \Phi_w \Phi_w^T, \quad (2)$$

where $Pr(C_i)$ is the prior probability of class C_i and usually is assigned to $\frac{1}{M}$ with the assumption of equal priors; μ is overall mean vector; Σ_i is the average scatter of the sample vectors of different classes C_i around their representative mean vector μ_i:

$$\Sigma_i = E[(x - \mu_i)(x - \mu_i)^T | C = C_i].$$

The class separability can be measured by a certain criterion. A commonly used one is the ratio of the determinant of the between-class scatter matrix of the projected samples to the within-class scatter matrix of the projected samples:

$$\mathcal{J}(A) = \arg\max_A \frac{|A S_b A^T|}{|A S_w A^T|} \quad (3)$$

where A is an $m \times n$ matrix with $(m \leq n)$. A solution to the optimization problem of Equation (3) is to solve the generalized eigen value problem [17]:

$$S_b A^* = \lambda S_w A^* . \quad (4)$$

For classification, the linear discriminant functions are:

$$D_i(X) = A^{*T}(X - \mu_i), \ i = 1, 2, \cdots, m. \quad (5)$$

A solution to Equation (4) is to compute the inverse of S_w and solve a eigen problem for matrix $S_w^{-1} S_b$ [17]. But this method is numerically unstable because it involves the direct inversion of a likely high-dimensional matrix. The most frequently used LDA algorithm in practice is based on simultaneous diagonalization [4]. The basic idea of the algorithm is to find a matrix A that can simultaneously diagonalize both S_w and S_b, i.e.,

$$A S_w A^T = I, \ A S_b A^T = \Lambda, \quad (6)$$

where Λ is a diagonal matrix with diagonal elements sorted in a decreasing order. If we want to reduce dimension of the matrix from n to m, we can simply use first m rows of A as the transformation matrix, which corresponds to the largest m eigen values of Λ. The simultaneous diagonalization algorithm also involves inversion of matrix. To our knowledge, most algorithms require that the within-class scatter matrix be S_w non-singular, because the algorithms diagonalize S_w first. Such a procedure breaks down when the within-class scatter matrix S_w becomes singular. This can happen when the number of training samples is smaller than the dimension of the sample vector. This is the case for most face recognition tasks. For example, a small size of image of 64x64 turns into a 4096-dimensional vector when vectorized. The solution to this problem is to perform two projections [14, 1, 3, 20]:

1. Perform PCA to project the n-dimensional image space onto a lower dimensional sub-space;

2. Perform discriminant projection using LDA.

The PCA step helps to remove null spaces from both S_b and S_w. However, this step potentially loses useful information. In fact, the null space of S_w contains the most discriminant information when the projection of S_b is not zero in that direction. Consider an extreme case where each class has only one sample, we can maximize S_b subject to the constraint that $S_w = 0$. The solution A^* is the set of $n \times m$ matrices with orthonormal columns contained in the kernel of S_w [1]. Therefore, we should not simply discard the null space of S_w. In the following subsection, we present a direct LDA algorithm that can keep the null space of S_w.

2.2 A New LDA Algorithm

The null space of S_w may contains useful information if the projection of S_b is not zero in that direction. But the null space of S_b can be safely discarded. To our knowledge, almost all the LDA algorithms diagonalize S_w first. This results in the requirement of S_w non-singular because the procedure involves inversion. However, the simultaneous diagonalization algorithm can start from either matrix of two symmetric matrices. In other words, we can diagonalize S_b first instead of S_w. If we begin diagonalization from S_b, we need to keep S_b non-singular. It will not lose any useful information if we remove the null space from S_b. An efficient way to remove the null space can come from the following lemma:

Lemma 1: $C = LL^T$, where C is an $n \times n$ matrix, L is $n \times m, m < n$. Mapping $x \rightarrow Lx$ is a one-to-one mapping of eigenvectors of $L^T L$ onto those of LL^T.

Proof: If $L^T L x = \lambda x$, multiplying both side by L, $LL^T Lx = \lambda Lx$. Thus Lx is an eigenvector (times a scalar) of LL^T. It's easy to verify all eigenvectors remain orthogonal to each other after the mapping.

In fact, Lemma 1 has been widely used for face recognition, especially "eigenface" approach [15]. We can use this lemma to remove the null space from S_b efficiently. In the direct LDA algorithm presented below, we still use Fisher's criterion Equation (3). We modified the traditional simultaneous diagonalization procedure to obtain an exact LDA solution without a separate dimensionality reduction step.

Direct LDA Algorithm for for Face Recognition

1. Remove the null space from S_b and diagonalize S_b.

   Do an eigen-analysis of $\Phi_b^T \Phi_b$ (an $M \times M$ matrix). Sort eigenvectors in decreasing order of the corresponding eigenvalues. Map each eigenvector x of $\Phi_b^T \Phi_b$ onto $v = \Phi_w x$, which is the eigenvector of S_b. Normalize the v's and write them down side by side to get V, such that

   $$V^T S_b V = \Lambda, \tag{7}$$

   where $V^T V = I$, Λ is diagonal matrix sorted in decreasing order. Discard those with eigenvalues sufficient close to 0 (below ϵ). Let Y be the first m columns of V, we have

   $$Y^T S_b Y = D_b, \tag{8}$$

2. Diagonalize S_w Let $Z = Y D_b^{-\frac{1}{2}}$, we have

   $$(Y D_b^{-\frac{1}{2}})^T S_b (Y D_b^{-\frac{1}{2}}) = Z^T S_b Z = I. \tag{9}$$

   Diagonalize $Z^T S_w Z$ by eigen analysis:

   $$U^T Z^T S_w Z U = D_w, \tag{10}$$

   where $U^T U = I$, D_w may have $0s$ in its diagonal. Again, we can utilize Lemma 1 to compute eigenvalues, i.e.,

   $$Z^T S_w Z = Z^T \Phi_w \Phi_w^T Z = (\Phi_w^T Z)^T \Phi_w^T Z . \tag{11}$$

   Since the objective is to maximize the ratio of between-scatter against within-scatter, those eigenvectors corresponding to the smallest eigenvalues of D_w are the most discriminative dimensions. We can optionally pick only the most discriminative several dimensions. In fact, we can sort the diagonal elements of D_w in a decreasing order and discard some eigenvectors with large eigenvalues.

3. The LDA transformation is:

   $$A = (ZU)^T . \tag{12}$$

   Matrix A diagonalizes both the numerator and the denominator of Fisher's criterion:

   $$A S_w A^T = D_w, \ A S_b A^T = I.$$

4. Finally, we can sphere the data into a more spherical shape, which is done with the transformation:

   $$X^* \leftarrow D_w^{-\frac{1}{2}} A X. \tag{13}$$

3 Discussion

Some comments are in order:

1. Although LDA has demonstrated good performance in face recognition tasks, traditional LDA algorithms have problems handling a degenerated S_w. Some of the most discriminant dimensions are potentially lost by removing the null space of S_w. In fact, the full rank requirement of S_w can be transferred to S_b when applying simultaneous diagonalization procedure. It will not lose any useful information by removing the null space from S_b.

2. The first step of the new algorithm has a dual purpose: dimensionality reduction and sub-space mapping. This step, in fact, directly projects raw image data onto the face sub-space, provided that sample images are aligned. Therefore, we can safely remove the null space, which makes no contribution to face recognition.

3. The new algorithm keeps the most discriminant projection direction embedded in the null space of S_w. The algorithm can take advantage of all useful information inside and outside of S_w's null space.

4. The new algorithm is an "unified" algorithm with some previous face recognition algorithms if we modify the Fisher's criterion. In fact, there are other variants of Fisher's criterion [4], for example,

   $$\arg \max_A \frac{|A S_t A^T|}{|A S_w A^T|}, \tag{14}$$

   where $S_t = S_b + S_w$ is the total scatter matrix. If we use the criterion (Equation (14)), the first step of the new algorithm performs PCA function exactly as other "PCA + LDA" algorithms did in a separated procedure. In an extreme case where each class has only one sample, the new algorithm will get the same result as a PCA algorithm. Therefore, the new algorithm is a "unified PCA + LDA" algorithm.

4 Experimental Results

Many researchers have demonstrated that LDA outperforms PCA for many different tasks. We have the proposed algorithm using human face images from Olivetti-Oracle

Research Lab (ORL) [16]. The ORL dataset consists of 400 frontal faces: 10 tightly-cropped images of 40 individuals with variations in pose, illumination, facial expression (open/closed eyes, smiling/not smiling) and accessories (glasses/no glasses). The size of each image is 92x112 pixels, with 256 grey levels per pixel. Figure 1 shows 10 randomly selected from the dataset.

Figure 1. Sample images from the ORL dataset

We have performed many different experiments to study the new algorithm. Without any preprocessing step, the best recognition rate for the new algorithm is 95%, which is compatible to the best result obtained by other researchers on the same test set using different algorithms. We describe two experiments below. In the first experiment, We tested the recognition accuracy vs. dimensionality reduction. Five randomly-selected images for each individual in the dataset were placed in the training set, and the remaining images were used for testing. Ten runs for each of five randomly selected images were performed with different, random partitions between training and testing images, and the results were averaged. Figure 4 shows the results of recognition accuracy vs. dimensionality reduction. The recognition accuracy approaches the best result with 40 dimensions. There is no significant improvement if more dimensions are used.
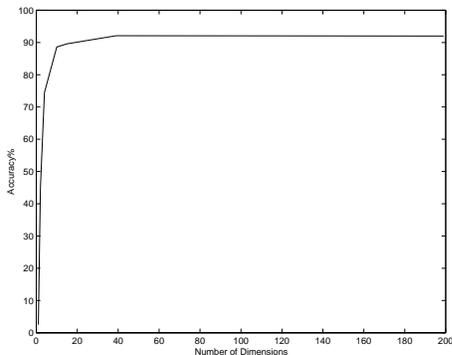

Figure 2. Recognition accuracy vs. dimensionality reduction

Then we fixed the number of dimension to 39 and tested recognition accuracy vs. number of training samples. We varied the number of training samples from 1 to 9. When we tested n samples ($n = 1, 2, \cdots, 9$), we reduced the dimension to 39 dimensions. n randomly-selected images for each individual in the dataset were placed in the training set, and the remaining images were used for testing. Ten runs for each of n samples were performed with different, random partitions between training and testing images, and the results were averaged. Figure 4 shows the results.
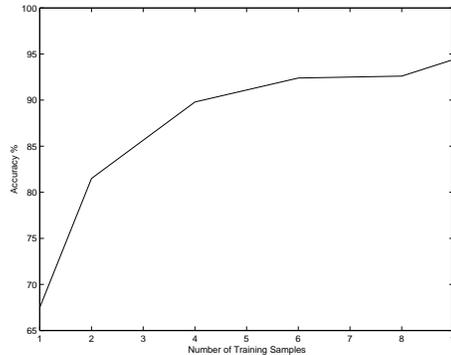

Figure 3. Recognition rate vs. number of training samples

5 A Real-time System

We have implemented a real-time face recognition system. The system inputs the image from a video camera. The system consists of two parts: a real-time face tracker and a face recognition system. These two parts exchange information through a socket as shown in Figure 4. The reason behind the decision to separate face tracking from face recognition is to allow the two units to work at different rates. It is essential that the tracking unit continually updates the location of a person face for coherency reasons. If a large amount of time were to lapse before the tracking program is allowed to track a persons face, the face is likely to be in a very different location. The tracking algorithm is much more efficient and accurate if the subject's face in the current frame is relatively close to the position where the subject's face was found in the previous frame. This requires the tracking unit to work at a speed of many frames per second. Face recognition, on the other hand, trades off accuracy for speed. At many frames per second, accuracy of recognition is too degraded to be useful. However, it is not necessary that recognition occurs near as frequently as tracking; therefore, we have separated the units to allow the tracking unit and the recognition unit to work at speeds optimal for their purpose. This has the added advantage of

allowing one machine to perform the tracking and image acquisition, while another computer performs recognition on the face. This means that a remote computer or computers that are connected to cameras can track the subject(s), and relay the facial image of the subject to a control system that contains the large database of facial images for facial recognition. Then, through socket communication, the recognizer can relay the recognition results to whatever computer/program desires the information. This centralizes the recognition process, eliminating the need for duplicated copies of the facial database, and allows for a distributed tracking system. On the other hand, both programs can coexist and run simultaneously on the same system.
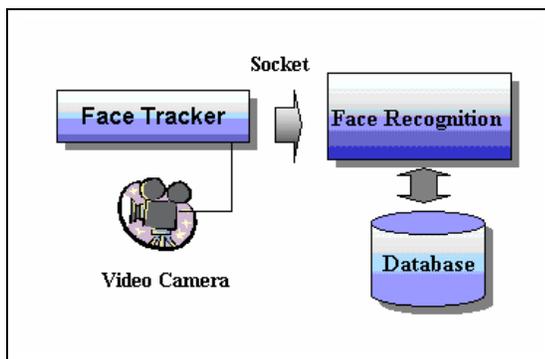

Figure 4. Architecture of the face recognition system

5.1 Face Extraction

Facial region extraction is a prerequisite for face recognition. From our experience, the quality of face extraction effects face recognition rate directly. For face extraction, we use the real-time face tracking technology developed at the Interactive Systems Lab at Carnegie Mellon University [18]. It uses an adaptive skin-color model to extract what is in high likelihood a face from an image. This color model can be initialized to a person's skin color for more accurate tracking, though not a necessary procedure for accurate tracking. The program also monitors motion in the image, because motion likely indicates a moving subject. Using the information extracted from the motion, it looks at the current frame, the previous frame, and the current color model to choose the most likely face region. It then actively updates the color model to make it a better match with the current image information. This insures accurate tracking even when the subject steps into various lighting regions.

From the extracted face, the system can further find various features in a subject's face [13]. It is capable of correctly

Figure 5. An example of face tracker

identifying the location of the subject's eyes, the subject's nostrils, and the corners of the subject's mouth as shown in Figure 5. For face extraction purposes, the system focuses on the location of the eyes, and uses particularly the distance between the eyes to establish a bounding box for the face. Using the eyes as an anchor establishes a consistent method for face extraction. Accurate recognition requires that many distinctive features of the face be visible, including the subject's hair and head contour, which means the bounding box must be large enough to include the features. On the other hand, it is undesirable to include the background in the facial image because a variable background adversely affects recognition, which means the bounding box should be small enough to exclude any background information. In addition, recognition methods that use principle component analysis require that distinguishing features be found in roughly the same place for accurate recognition. This means that if a subject's nose is found in the middle of the training image, then the subject's nose needs to be in the middle of the test image for accurate identification. Thus, using the eyes as an anchor, distinguishing characteristics are consistently found in relatively the same location, and the bounding box can be calculated to fit well around the subject. It was empirically determined that a satisfactory size for the bounding box is four times the distance between the eyes in height and three times the distance between the eyes in width centered around the point between the eyes. Though it should be kept in mind that the actual calculations of the bounding box are not near as important as it is to keep the relative distances of the bounding box consistent. Accurate recognition requires consistency. It should be noted that if the subject rotates his/her head with respect to the camera, the dimensions of the box are distorted relative to the face. As long as the recognition portion is trained with images of this rotation, this would not present a problem.

5.2 Face Recognition

We have developed a general recognition module, which acts as a platform to allow many different methods of face recognition to be 'plugged-in' and run. This platform facilitates rapid development and testing for different recognition algorithms. The module has separated basic functions such as input and output from recognition algorithms. The module starts by loading a database of face images. These images are used for training the various recognition methods. The training method is determined by the recognition algorithm. The module then allows a user to run the various recognition methods on a recognition set. This recognition set can be a set of different image files stored in another database, or live image data that comes from the face-tracking module. Running recognition on stored images allows for controlled experimentation on the strengths and weaknesses of the various recognition methods. Running recognition on a live image is useful for identifying people that are currently in the range of the camera and relaying that information on to the various programs that need that information.

We have currently implemented many different face recognition algorithms, such PCA [15], PCA + LDA [1], DSW [5], and direct LDA.

6 Conclusions

We have proposed a direct LDA algorithm for face recognition. By transferring the full rank requirement from S_w to S_b, the algorithm has avoided losing the most discriminant dimensions because of removing the null space of S_w. The new algorithm has unified PCA/LDA algorithm by naturally combining the PCA technique into eigen analysis of LDA. We have developed a real-time face recognition system by combining face tracking and face technologies together. The system has provided a platform for developing new face recognition algorithms. We are currently working on improving face recognition accuracy by a sequence of video images.

Acknowledgements

References

[1] P. Belhumeur, J. Hespanha, and D. Kriegman. Using discriminant eigenfeatures for image retrieval. *PAMI*, 19(7):711–720, 1997.

[2] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5):705–740, 1995.

[3] K. Etemad and R. Chellappa. Discriminant analysis for recognition of human face images. *Journal of the Optical Society of America A*, 14(8):1724–1733, 1997.

[4] K. Fukunaga. *Introduction to Statistical Pattern Recognition (Second Edition)*. New York: Academic Press, 1990.

[5] R. Gross, , J. Yang, and A. Waibel. Face recognition in a meeting room. In *Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'2000)*, Grenoble, France, 2000.

[6] R. Haeb-Umbach and H. Ney. Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *Proc. ICASSP92*, pages 1/13–1/16, 1992.

[7] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. *PAMI*, 12(1):103–108, January 1990.

[8] C. Liu and H. Wechsler. Enhanced fisher linear discriminant models for face recognition. In *Proceedings of Fourteenth International Conference on Pattern Recognition*, volume 2, pages 1368–1372, Brisbane, Qld., Australia, 1998.

[9] B. Moghaddam and A. Pentland. Probabilistic visual learning for object recognition. *PAMI*, 19(7):696–710, 1997.

[10] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proceedings of the 1994 Conference on Computer Vision and Pattern Recognition*, pages 84–91, Seattle, WA, 1994. IEEE Computer Society.

[11] A. Samal and P. Iyengar. Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognition*, 25(1):65–77, 1992.

[12] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, 4(3):519–524, March 1987.

[13] R. Stiefelhagen, J. Yang, and A. Waibel. A model-based gaze tracking system. *International Journal on Artificial Intelligence Tools*, 6(2):193–209, 1997.

[14] D. Swets and J. Weng. Using discriminant eigenfeatures for image retrieval. *PAMI*, 18(8):831–836, August 1996.

[15] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):72–86, 1991.

[16] ORL web site:. *http://www.cam-orl.co.uk*.

[17] S. S. Wilks. *Mathematical Statistics*. New York: Wiley, 1962.

[18] J. Yang and A. Waibel. A real-time face tracker. In *Proceedings of WACV'96*, pages 142–147, Sarasota, Florida, 1996.

[19] W. Zhao, R. Chellappa, and N. Nandhakumar. Empirical performance analysis of linear discriminant classifiers. In *Proceedings of the 1998 Conference on Computer Vision and Pattern Recognition*, pages 164–169, Santa Barbara, CA, 1998.

[20] W. Zhao, R. Chellappa, and P. Philips. *Subspace Linear Discriminant Analysis for Face Recognition*. Technical Report CAR-TR-914. Center for Automation Research, University of Maryland, 1999.