Optik 127 (2016) 3935–3944

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

Overview of principal component analysis algorithm

Lingjun Li [a,b], Shigang Liu [b,*], Yali Peng [a,b], Zengguo Sun [b]

[a] Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, China
[b] School of Computer Science, Shaanxi Normal University, Xi'an 710119, China

ARTICLE INFO

Article history:
Received 15 October 2015
Accepted 4 January 2016

Keywords:
Face recognition
PCA algorithm
Extended-PCA algorithms
Recognition rate

ABSTRACT

Principal component analysis (PCA) algorithm has been extensively employed in face recognition. However, existing PCA algorithms have some limitations in face recognition. In order to overcome these limitations, researchers proposed some extended-PCA algorithms. In this paper, we present algorithm implementation of the original PCA algorithm and main extended-PCA algorithms including two-dimensional PCA (2DPCA), 2DPCA-based feature fusion approach, the kernel PCA (KPCA), the modular PCA, improved KPCA (IKPCA), efficient sparse KPCA (ESKPCA) and incremental PCA (IPCA). 2DPCA directly computes the projection of a matrix onto a transforming axis. 2DPCA-based feature fusion approach combines the features generated from the two schemes of 2DPCA. KPCA can perform well in extracting features from samples whose components have nonlinear relations. The modular PCA approach divides the original face image into sub-images and applies the original PCA approach to each of these sub-images. The IKPCA algorithm improves KPCA for more efficient feature extraction. The efficient sparse KPCA (ESKPCA) improves the computational efficiency of the previous sparse KPCA methods on large-scale training sample sets. Incremental PCA (IPCA) overcomes the limitation which is hard to scale up the developed systems. In order to compare the recognition rate of these algorithms in face recognition, a series of experiments are performed on three face image databases: ORL, Yale and NIR face databases.

1. Introduction

Face recognition has become a research hotspot in computer vision and pattern recognition communities. And because of high user acceptability, it has a wide range of application prospects [1–3]. Face recognition includes the following steps: facial image acquisition, pretreatment of facial images, faces detection, face representation, and face verification, etc. Due to the high dimension of the original image, directly using the original image for face recognition will increase the computational complexity. So it is necessary to reduce the dimension of the original image. The main purpose of face representation is to extract low dimensional features from the original image.

There have been numerous feature extraction methods of face images. One of them, principal component analysis (PCA) is a very effective feature extraction method in face representation. It not only can effectively reduce the dimension of the face image, but also can retain the main information of original face images. PCA is also referred to as the Karhunen–Loève transformation [4], and was introduced into face processing by Kirby and Sirovich [5]. Since the eigenfaces [6] method for face recognition was presented by Turk and Pentland [7] in 1991, PCA has already successfully been implemented in face recognition [8,9]. PCA is very effective for finding features to reduce the dimension of data [10–15]. PCA has become one of the most effective methods in face recognition [16–22] and has been widely investigated. However some shortcomings of the PCA algorithm have been revealed in face recognition. It is necessary for the PCA algorithm to previously transform the 2D face image matrices into 1D image vectors, which usually lead to inaccurate evaluation of the covariance matrix due to its high dimensional image vector space. Moreover the implementation of PCA on face images is usually very time-consuming [23,24]. In order to overcome these limitations, researchers have proposed some extended-PCA algorithms. The kernel-based PCA (KPCA) algorithm [25–35] is a nonlinear form of principal component analysis. KPCA can perform well in extracting features from samples whose components have nonlinear relations. The modular PCA approach [36–40] divides the original face image into sub-images and applies the original PCA approach to each of these sub-images. The complex-PCA [41] is another extension of the PCA, which uses a complex vector to represent information from the sample of one subject. The matrix-based complex PCA (MCPCA) [42] uses two matrices to represent two different biometric traits of one subject. Yang et al. [23] proposed the two-dimensional

* Corresponding author. Tel.: +86 2985310163.
E-mail address: shgliu@snnu.edu.cn (S. Liu).

http://dx.doi.org/10.1016/j.ijleo.2016.01.033

principal component analysis (2DPCA) for face recognition. Xu et al. [24] proposed 2DPCA-based feature fusion approach for face recognition. 2DPCA [43,44] and 2DPCA-based feature fusion approach [45–47] are based on 2D image matrices, and directly extract features from image matrices. Ren et al. [48] proposed the incremental learning algorithm of bidirectional principal component analysis (IBDPCA). And the class-augmented PCA [49,50] has recently been proposed. Xu et al. [51] proposed the IKPCA algorithm to improve KPCA for more efficient feature extraction. Incremental PCA (IPCA) [52,53] overcomes the limitation which is hard to scale up the developed systems. Xu et al. proposed an efficient sparse KPCA (ESKPCA) [54] to improve the computational efficiency of the previous sparse KPCA methods on large-scale training sample sets. Zhao et al. [55] proposed the SVD updating-based incremental PCA (SVDU-IPCA) which has been attached much attention, etc. Among the extensions of the original PCA algorithm, the complex-PCA and matrix-based complex PCA are applicable to personal authentication with bimodal approaches. The bimodal approaches combine two biometric traits to perform personal authentication, and they can produce better results than single biometrics using a single trait alone. PCA not only is applied in image classification such as face recognition, it also is applicable for other various problems such as analysis of gene data [56–58], video analysis [59] and machine learning tasks [60].

In this paper, we present algorithm implementation of the original PCA and main extended-PCA algorithms. To compare recognition rates of these algorithms in face recognition, a series of experiments are performed on three face image databases: ORL, Yale and NIR face databases. The remainder of the paper is organized as follows. In Section 2, the PCA algorithm and main extended-PCA algorithms are described. In Section 3, we describe some experiments. Finally, Section 4 offers our conclusion.

2. Review of various algorithms

2.1. Principal component analysis (PCA)

Due to the similarity of the overall configuration of the face, the distribution of a face image is not random. The PCA algorithm regards a training image as a random vector with a certain distribution. The PCA algorithm can obtain the principal components of a face image, and it uses the principal components of the face image to express the face image and achieves the purpose of dimension reduction. When PCA is applied to face recognition, the PCA algorithm first transforms all 2D face images into 1D image vectors, and then calculates the covariance matrix of all the training samples. It calculates the eigenvalues and eigenvectors of the covariance matrix and projects each sample (including training samples and test samples) onto eigenvectors which have the first n maximum eigenvalues [61]. Finally, the classification of all the test samples is performed by using a classifier [62,63]. The original PCA algorithm is generally implemented on image data as follows.

Let samples of the training sample set $\{x_i^{(j)}\}$ belong to c categories $\omega_1, \omega_2, \ldots, \omega_c$ respectively, where $i = 1, 2, \ldots, c$, $j = 1, 2, \ldots, n_i$; n_i denotes the number of samples in the ith class. $x_i^{(j)}$ denotes the jth sample in the ith class. Within-class scatter matrix S_w, between-class scatter matrix S_b and the total scatter matrix S_t are, respectively, defined as

$$S_w = \sum_{i=1}^{c} P(\omega_i) 1/n_i \sum_{j=1}^{n_i} (x_i^{(j)} - \overline{x_i})(x_i^{(j)} - \overline{x_i})^T \quad (1)$$

$$S_b = \sum_{i=1}^{c} P(\omega_i)(x_i - \overline{x})(x_i - \overline{x})^T \quad (2)$$

$$S_t = S_w + S_b = \sum_{i=1}^{c} P(\omega_i) 1/n_i \sum_{j=1}^{n_i} (x_i^{(j)} - \overline{x})(x_i^{(j)} - \overline{x})^T \quad (3)$$

where $\overline{x_i} = 1/n_i \sum_{j=1}^{n_i} x_i^{(j)}$ is the mean vector of all the samples in the ith class; $\overline{x} = \sum_{i=1}^{c} P(\omega_i)\overline{x_i}$ is the mean vector of all samples; $P(\omega_i)$ is the prior probability of the ith class.

Let $P(\omega_i) = n_i/N$, $\overline{x}$ can be expressed by $\overline{x} = 1/N \sum_{i=1}^{c} \sum_{j=1}^{n_i} x_i^{(j)}$. Because $P(\omega_i) = n_i/N$, the total scatter matrix S_t can also be expressed by

$$S_t = 1/N \sum_{i=1}^{c} \sum_{j=1}^{n_i} (x_i^{(j)} - \overline{x})(x_i^{(j)} - \overline{x})^T = 1/N \overline{X}\overline{X}^T \quad (4)$$

where the matrix $\overline{X} = (x_1^{(1)} - \overline{x}, \ldots, x_1^{n_1} - \overline{x}, \ldots, x_c^{(1)} - \overline{x}, \ldots, x_c^{(n_c)} - \overline{x})$, $N \left(N = \sum_{i=1}^{c} n_i\right)$ is the total number of samples.

Let $Q = 1/N\overline{X}^T\overline{X}$, then the dimension of Q is $N \times N$. According to the singular value decomposition theorem, the first n eigenvalues $(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n)$ of Q are equal to the first n eigenvalues of S_t. Firstly, we calculate the first n eigenvalues $(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n)$ of matrix Q and it is the corresponding eigenvectors $(u_1 \geq u_2 \geq \cdots \geq u_n)$, then according to the formula (5), we can obtain the first n maximum eigenvalues and the corresponding standard orthogonal eigenvectors $(v_1 \geq v_2 \geq \cdots \geq v_n)$ of matrix S_t.

$$v_k = \frac{1}{\sqrt{\lambda_k}} \overline{X} u_k, \quad k = 1, \ldots, n \quad (5)$$

For any x_i sample, the principal component of x_i is

$$y_i = (v_1, v_2, \ldots, v_n)^T x_i \quad (6)$$

Finally, a nearest neighbor classifier is used for classification.

2.2. The two-dimensional principal component analysis (2DPCA)

In face recognition, the original PCA needs to convert the 2D face image matrix into 1D image vector, so the disadvantages are obvious. Firstly, when a face image matrix is transformed into a vector, the resultant vectors of faces usually lead to a high dimensional vector space [23], and the amount of computation will be greatly increased. So the implementation of PCA is usually very time-consuming [24]. Secondly, it usually inaccurately evaluates the covariance matrix due to its high dimensional vector space. In contrast, in the 2DPCA algorithm, the image matrix does not need to be transformed into a vector, and the covariance matrix can be directly obtained from the original image matrices. Therefore, the calculation amount of the 2DPCA algorithm is far less than that of the original PCA algorithm, and this algorithm can significantly improve the extraction speed of image features. Finally, the recognition rate of 2DPCA may be higher than that of PCA [23]. The specific description of the 2DPCA algorithm is as follows.

Supposing the jth training image is denoted by an $m \times n$ matrix $A_i^{(j)}$. We can replace x by A in formulas (1)–(3), and define the within-class scatter matrix G_w, between-class scatter matrix G_b and total scatter matrix G_t as follows.