Optik 140 (2017) 853–859

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

Original research article

Weighted sparse representation based on virtual test samples for face recognition

Ningbo Zhu, Shuoxuan Chen *

College of Information Science and Engineering, Hunan University, Changsha, China

ARTICLE INFO

Article history:
Received 3 December 2016
Accepted 4 May 2017

Keywords:
Face recognition
Sparse representation
Virtual sample
Automatic weight assignment

ABSTRACT

Face recognition with a very limited or even one training sample per subject is a very difficult task and it seems very challengeable to arise the accuracy of face recognition in such a condition. In this paper, we propose a novel weighted sparse representation method based on virtual test samples for face recognition. The presented method includes three steps. Firstly, generating virtual test samples for original test samples, and computing the distance between the test sample and each training sample to build a weighted training set. Secondly, representing the test sample over the weighted training set. Finally, computing the weight of each test sample and then conducting classification. The use of virtual samples of each individual allows us to get more distinguishing features and to obtain facial variations information from the external data. The used weight plays a role in enhancing the importance of these training images closer to a query image in representing this query image. An important advantage of the proposed approach is that the weight of each test sample is dynamically computed, instead of manual setting. Extensive experiments on YALE, AR and FERET face databases indicate that the proposed approach outperforms the other methods used in competition.

1. Introduction

As a very active and important topic in computer vision, face recognition (FR) has a large amounts of applications, containing access control, social network, human-computer interface, criminal investigation, etc [1–3]. During the past two decades, thousands of algorithms and methods have been proposed for FR [4–14]. Recent years, sparse representation based classification (SRC) [15] has been widely applied in face recognition due to its excellent performance. SRC [15] used a linear combination of all the training images to represent the query face image, and then to compute the residual of each class in representing the query image, finally to classify the query image according to the residual of all classes. This method boosts the study of sparsity based pattern classification [16–22]. Gao et al. [17] combined the Gaussian kernel function with sparse coding for FR, while Yang et al. [18] used the image local Gabor feature for SRC with a related Gabor occlusion set to deal with the occluded face images. Yang and Zhang [19] proposed a robust sparse coding to solve different types of outliers (e.g., expression, occlusion, pose angle, etc.).

Sparse representation based face recognition methods can achieve interesting results when each class provides enough training samples. However, in some practical applications, there is just a very limited, or even a single training sample for

* Corresponding author.
E-mail address: csx@hnu.edu.cn (S. Chen).

http://dx.doi.org/10.1016/j.ijleo.2017.05.032

each individual. The recognition accuracy of SRC declined sharply with the training samples per person decreasing. The main reason of this phenomenon is that we can't be good enough to predict facial variations information of the query image from the limited training samples. Thus, many literatures [29–31] used virtual samples to improve the recognition accuracy when each individual just provided very limited training images. Tang and Zhu [31] expanded the training set by adding random noise to the original training images, while Xu et al. [29] exploited 'symmetrical face' to build a symmetric training set, then used the original training set and symmetric training set to perform FR. Although the approaches in [29,31] notly uprated the performance of FR compared with the original SRC method, they didn't emphasize the distinctiveness of different training samples when building classifiers.

In the field of pattern recognition, different training samples usually make their own unique contributions to building classifiers. Some scholars suggested that we should assign different weights to different samples according to the distribution of the sample when building classifiers. We noticed that these training images which are nearer to a query image are more important than others in representing this query image [23–28]. Fan et al. [28] constructed a weighted training set to represent and classify the test sample. Xu and Zhang [27] used a two-stage test sample classification algorithm for FR. This method in [27] firstly finds out *M* training images which are much closer to the coming query image from the training set, then exploits the selected *M* training images to represent and perform classification for the query image. The algorithm in [27] shows excellent performance in FR, but leaves a question that how many nearest neighbors should be selected to represent and perform classification for the query image which results in the best classification.

In this paper, we propose a weighted sparse representation based on virtual test samples for face recognition. The use of virtual samples of each individual allows us to get more distinguishing features and to obtain facial variations information from the external data. The presented scheme contains three main steps. In the first place, it produces virtual test images for original test image, and computes the weight of each training image. Then new training set is built. Next, this method represents the test samples via new training sets and computes the residuals of each class. The last step of this method computes the weight of each test sample and then conducts classification. Extensive experiments indicate that our method is better than the other competing approaches. It is notable that the proposed method adopts weighted score level fusion to automatically and dynamically set a weight to each test image and is very easy to be used to practical applications. The proposed idea and scheme to determine the weight also seems to be helpful to improving other approaches.

The organization of the rest parts follows: Section 2 represents the proposed scheme. Section 3 analyzes the method. Section 4 performs the experiments and Section 5 concludes this paper.

2. The weighted sparse representation method based on virtual test samples

In this section, the weighted sparse representation method based on virtual test samples (WSRVTS) will be formally introduced in detail. The WSRVTS includes three main steps. Firstly, WSRVTS generates virtual test samples for original test sample, and computes the weight of each training sample. Then new training set is built. Secondly, WSRVTS represents the test samples and computes the residuals of each class. Finally, WSRVTS computes the weight of each test sample and then conducts classification.

2.1. Generating virtual test samples and producing new training sets

One can view the face image as a two-dimensional data matrix $R^{r\times c}$, where r and c are the row and the column of the face image, respectively. For any coming test sample, denote the original test sample by $y_1 = y$. We use the symmetry transformation [29] to produce $y_3$two virtual test images y_2 and Virtual test sample, and original test sample satisfy the following relationships:

$$y_2(i,j) = \begin{cases} y_1(i,j),\ i=1,2,\ldots,r \quad j=1,2,\ldots,c/2 \\ y_1(i,col-j+1),\ i=1,2,\ldots,r \quad j=c/2+1,c/2+2,\ldots,c \end{cases} \tag{1}$$

$$y_3(i,j) = \begin{cases} y_1(i,col-j+1),\ i=1,2,\ldots,r \quad j=1,2,\ldots,c/2 \\ y_1(i,j),\ i=1,2,\ldots,r \quad j=c/2+1,c/2+2,\ldots,c \end{cases} \tag{2}$$

where i and j are coordinates of X-axis and Y-axis, respectively.

Denote the training set by $X = \left[x^1, x^2, \ldots, x^n\right]$, where n denotes the size of training images and x^i is the i$^{\text{th}}$ training image in training set *X*. Next, we computes the distance between each query image and each training image by Eq. (3):

$$d(x^i, y_j) = \exp(-||x^i - y_j||_2^2/2\sigma^2),\ j=1,2,3 \quad i=1,2,\ldots,n \tag{3}$$

where σ, the Gaussian kernel width, is simply set as the average Euclidean distance [28] of all the training images in training set X in our experiment. From Eq. (3), we acquire that the value of the Gaussian kernel distance [30] between any two images ranges from 0 to 1, so we set this value as the weight of the training image. That is, for any training sample $x^i \in R^m$, its weight is denoted by w^i and equals to $d\left(x^i, y_j\right)$. Thus, we get the new training set by computing the weight of each training image.

$$X_j = [w^1x^1, w^2x^2, \ldots, w^nx^n] \tag{4}$$