International Journal of Bio-Science and Bio-Technology
Vol.8, No.2 (2016), pp.197-206
http://dx.doi.org/10.14257/ijbsbt.2016.8.2.18

Performance Analysis of Pixel based Face Recognition Methods

Unsoo Jang[1] and Eui Chul Lee[2,*]

[1]*Department of Computer Science, Graduate School, Sangmyung University, Seoul, Republic of Korea*
[2]*Department of Computer Science, Sangmyung University, Seoul, Republic of Korea*
[*]*Corresponding author: eclee@smu.ac.kr*

Abstract

From the recent diversity of face recognition applications, it can be concluded that studies on face recognition are being actively conducted. However, high-performance algorithms sometimes cannot be adopted because of environmental reasons such as pose variation of targets or lighting conditions at the outdoor CCTV. Further, a low-resolution CCTV image cannot be used for an appearance-based high-performance method. Therefore, we consider pixel-based face recognition methods as CCTV applications. Before applying the methods, a Retinex-based illumination normalization method is used. The methods are applied with various facial poses, and three kinds of face recognition methods are comparatively analyzed in this paper. The experimental results showed that linear discriminant analysis exhibited the best performance as the matching method taking into consideration various facial poses.

Keywords*: Principal components analysis, linear discriminant analysis, local binary patterns, Illumination normalization, face recognition*

1. Introduction

Recently, improvement in face recognition performance has been accelerating owing to advances by leading technology companies such as Facebook and Google. In particular, deep-learning-based face recognition techniques have seen dramatic developments in the biometric industry. DeepFace, developed by Facebook, is an example of a structure that utilizes deep learning [1]. Many security and social network applications use face recognition techniques [2], such as the tagging system of Facebook in Fig 1. However, this high-performance algorithm is difficult to use with low-resolution images characterized by various facial poses and lighting conditions. Therefore, we look at the traditional face recognition methods.

Traditional face recognition methods can be classified into three types. First, appearance-based face recognition methods use global feature extracted from the face region. Principal components analysis (PCA) and linear discriminant analysis (LDA) can be categorized as appearance-based methods [3]. Second, texture-based face recognition methods use textural characteristics extracted from the local face region. In texture-based methods, local binary pattern (LBP) is generally used [4]. Third, geometry-based face recognition uses the positions of feature points such as the eyes, nose, and mouth. For example, the active appearance model is regarded as a geometry-based method [5].

ISSN: 2233-7849 IJBSBT

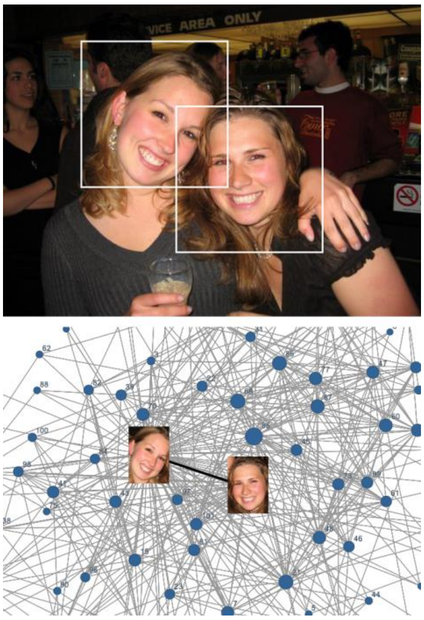


Figure 1. A Tagged Photo from the Facebook Social Network (Up), and the Visualization Illustrating the Social Connections among these Two Individuals and their Friends (Down) [1]

Especially in criminal situation monitoring and recognition, continuous checking and wide-range monitoring are needed. Therefore, recent face recognition algorithms are frequently applied to CCTV systems to implement intelligent CCTVs. Because the face image in the typical CCTV camera has a low resolution, geometry-based methods cannot be considered in this situation. Consequently, appearance- and texture-based face recognition methods can be used as CCTV applications.

In this paper, pixel-based face recognition methods such as PCA, LDA, and LBP are comparatively analyzed in terms of recognition accuracy. Here, facial pose variation is considered by multi-frame enrollment of a single subject.

2. Used Methods

Three recognition accuracies of face recognition methods are analyzed. Before face recognition, an illumination normalization algorithm is applied to eliminate lighting variation. The illumination normalization algorithm will be described in the next subsection and the three face recognition methods in the following subsections.

2.1. Illumination Normalization

The Retinex-based illumination normalization method normalizes light conditions in the original image by using light condition elements extracted through the blur image of the original image [6].

First, the blur image is extracted from the original image by using an average convolution filter. The blur image includes the light condition elements of the original image. Next, the illumination removing image is calculated from the difference between the original image and the blur image. Finally, the illumination normalizing image is formed by normalizing the histogram of the illumination removing image. Fig 2 contains visualizations and examples of the aforementioned algorithm.

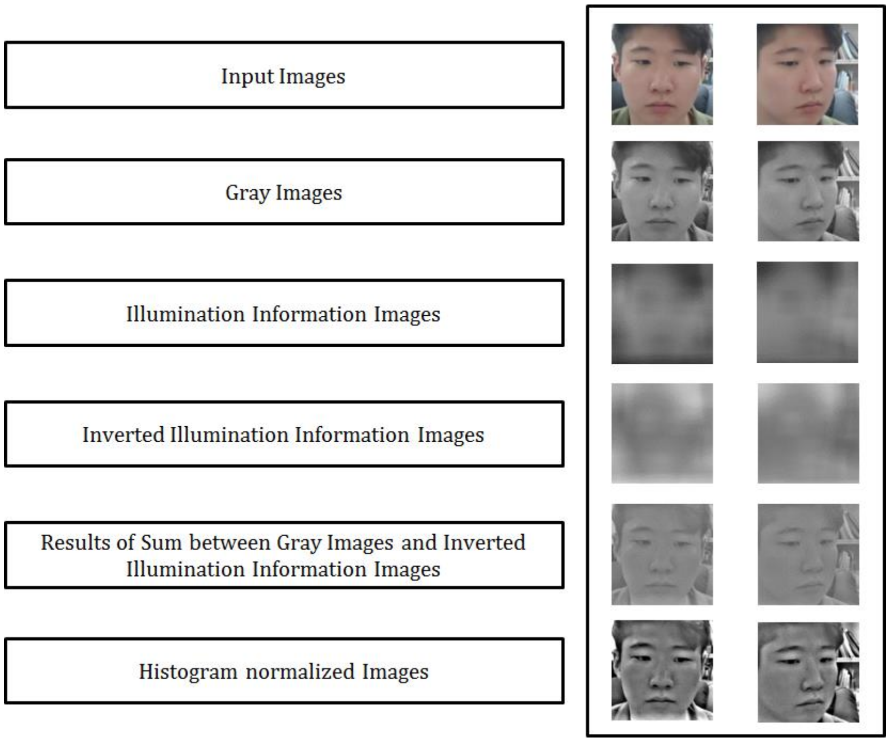


Figure 2. Visualizations (left) and Examples (right) of the Illumination Normalization Algorithm

2.2. PCA Face Recognition Method

PCA uses an eigenvector calculated from the enrolled face images. Because the eigenvector is acquired by analyzing the principal components of the input data, unique eigenvectors can be selected to represent dimension-reduced face information. PCA is regarded as a linear transformation from high-dimensional data to a low-dimensional sub-space [3]. In addition, PCA approximates vectors by calculating a basis in an appropriate lower-dimensional space. Eigenvectors can be represented as an image, as shown in Fig 3, by allocating pixel values at each dimension that are called eigenfaces.

Figure 3. Eigenfaces Calculated by PCA

The following is a description of the PCA algorithm:

Step 1: Data setting

First, let X be a random vector with observation$x_i \in R^d$.

$$X = \{x_1, x_2, \cdots, x_n\}$$

Where

X: A set of face images

x: Each face image

n: The number of face images

Step 2: Compute the mean

After forming set X, compute the mean μ using the following equation:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Step 3: Compute the difference between the original image and the average image

Save the difference ϕ_i between the original image δ_i and the average image μ_i:

$$\phi_i = \delta_i - \mu_i$$

Step 4: Compute the covariance matrix

Compute the covariance matrix C using the following equation:

$$C = \frac{1}{n}\sum_{i=1}^{n} \phi_i \phi_i^T$$

Step 5: Compute the eigenvalues and eigenvectors of the covariance matrix

In this step, compute the eigenvalues λ_i and eigenvectors v_i of covariance matrix C.

$$Cv_i = \lambda_i v_i$$

Step 6: Select the feature vectors

As the eigenvalue of the eigenvector is larger, it represents the better features of the face. Therefore, finding the eigenvector v\` having the maximum eigenvalue $\lambda$\` is necessary. On the other hand, as the eigenvalue of the eigenvector is smaller, it represents the less features of the face. Therefore, removing the eigenvectors having small eigenvalues is necessary.

After obtaining the eigenvalue v\` in the aforementioned manner, training sets are created from the eigenvalue $v$\`. New input images are also classified using the created training sets.

2.3. LDA Face Recognition Method

LDA is similar to PCA in terms of the use of linear transformations and dimension reduction. LDA is especially used to classify input data. For the classification, both the within-class scatter of each class and the between-class scatter are calculated. Because the purpose of LDA is classification into different classes, the representative vectors are determined by maximizing between-class scatter and minimizing within-class scatter. Consequently, LDA classifies unknown class data on the basis of the aforementioned representative vectors. Using a similar method as for eigenfaces, the vectors can be represented by images (Fisherfaces) as shown in Fig 4.

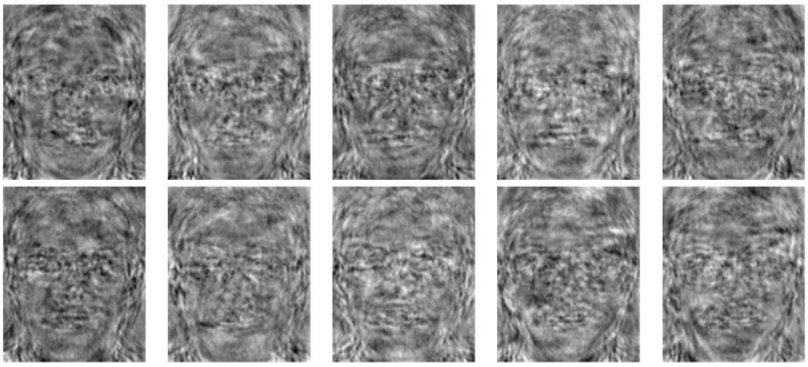


Figure 4. Fisherface Calculated by LDA

A description of the LDA algorithm follows:

Step 1: Make dataset

Let X be a random vector with samples drawn from c classes, and let each class have n data.

$$X = \{X_1, X_2, \cdots, X_c\}$$
$$X_i = \{x_1, x_2, \cdots, x_n\}$$

Where

X: Dataset of c classes

X_i: i class including n data

Step 2: Calculate the total mean and the means of classes $i \in \{1, 2, \cdots, c\}$

The means μ_i of X_i are calculated, $i \in \{1, 2, \cdots, c\}$:

$$\mu_i = \frac{1}{|X_i|} \sum_{x_j \in X_i} x_j$$

where $|X_i|$ is the data size of i class, and the total mean μ of the entire data x_j is calculated for $j \in \{1, 2, \cdots, N\}$:

$$\mu = \frac{1}{N} \sum_{j=1}^{N} x_j$$

Where

N: Data size of the entire data

Step 4: Calculate two scatter matrices

The within-class scatter matrices S_W are calculated with the following equation:

$$S_W = \sum_{i=1}^{c} \sum_{x_j \in X_i} (x_j - \mu_i)(x_j - \mu_i)^T$$

where c is the size of classes included in set X, and the between-classes scatter matrices S_B are calculated with the following equation:

$$S_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T$$

Step 5: Look for a projection

Fisher's classic algorithm now looks for a projection W that maximizes the class separability criterion by using the following equation:

$$W_{opt} = \mathrm{argmax}_{\mathrm{W}} \frac{|W^T S_B W|}{|W^T S_W W|}$$

A solution to this optimization problem is given by solving the general eigenvalue problem [7].

$$\mathrm{S_B} v_i = \lambda_i S_W v_i$$

$$\mathrm{S_W^{-1}} S_B v_i = \lambda_i v_i$$

Step 6: Classify input images

By comparing the projection of the input image with the projection of the training images, the similarity between an input image and each class is calculated. The input images are thus classified into similar classes.

2.4. LBP Face Recognition Method

LBP is a simple and strong algorithm for local feature descriptors. LBP was originally designed for texture representation. Regional characteristics of a certain size are represented by a decimal number. Decimal numbers of an entire image are then used to compare different images. Fig 5 shows examples of LBP numbers represented in image format.

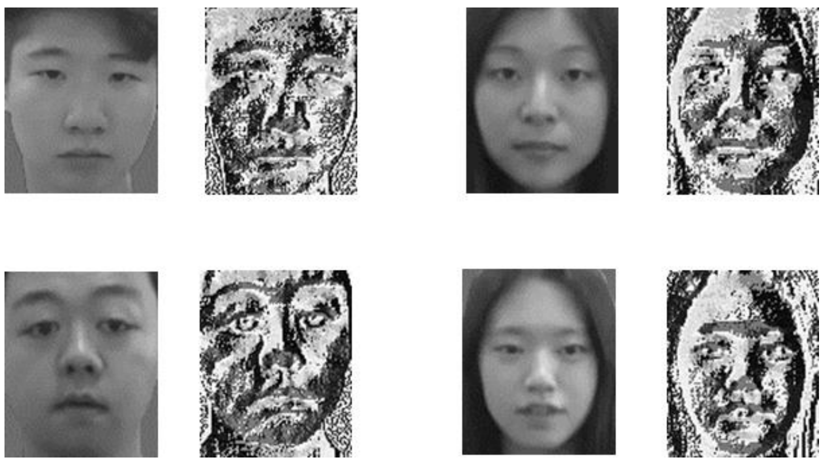


Figure 5. LBP Images Generated from Original Face Image (left: Original, Right: LBP Image)

For calculating LBP numbers, the following method is applied to each pixel of the entire image. First, a 3×3 area around one pixel is binarized with the threshold extracted from the center pixel. Next, the binarized eight pixel values are represented by a single decimal or binary value [8]. Fig 6 shows a visualization of this method.

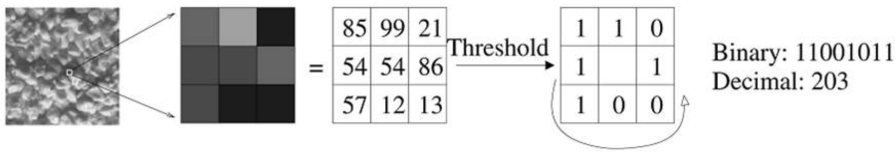


Figure 6. Explanation of How to Get LBP Code [8]

The following is the formula representing the aforementioned method:

$$\mathrm{LBP}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c)$$

Where
(x_c, y_c): The center pixel
i_c: The value of center pixel
$i_p (p \in \{1, 2, \cdots, P\})$: P neighborhood pixels of center pixel
The function s is defined as follows:

$$s(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \; else \end{cases}$$

3. Experimental Result

To perform experiments, 20-s facial videos (640 × 480, 30 frames/s) were captured. Without illuminative variation, five subjects' videos were captured in which pose variations (front, left, right, up, and down) were included as shown in Fig 7.

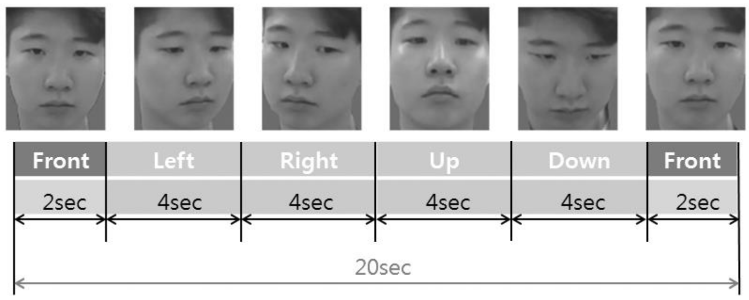


Figure 7. Composition of 20-s Experimental Video

In the captured video, 600 frames were used for face recognition. Failure frames of face detection were excluded, and 10 frames of each video were used as enrollment images. Here, the face region was cropped and resized at 92 × 112 pixels. To perform the experiment taking facial pose variation into consideration, three types of enrollment were organized as shown in Fig 8. Type #1, as in Fig 8(a), has a large amount of facial pose variation. On the other hand, type #3, as in Fig 8(c), has no facial pose variation. Type #2, as in Fig 8(b), has a medium variation between type #1 and type #3.

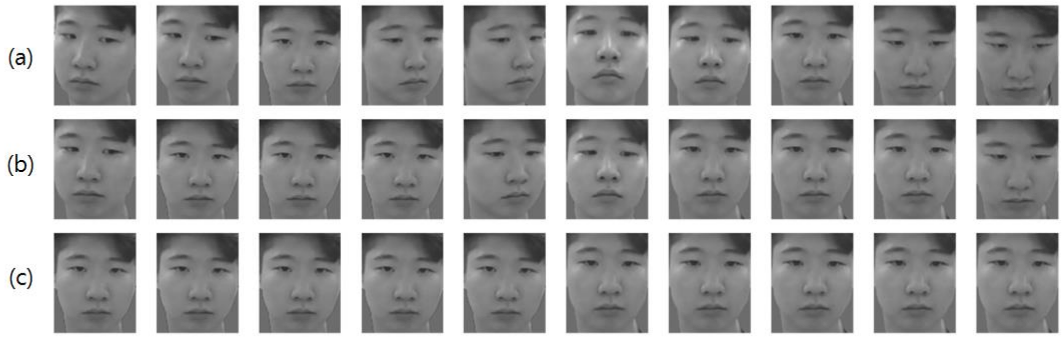


Figure 8. Three Type of Enrollment. (a) Type #1: Large Facial Pose Variation, (b) type #2: Medium Facial Pose Variation, (c) Type #3: No Pose Variation

First, Table 1 shows the recognition accuracies in terms of the enrollment types (#1, #2, and #3) and the recognition method (PCA, LDA, and LBP). The accuracy is calculated by dividing the number of genuine recognized frames by the total number of face detected ones.

Table 1. Recognition Accuracies in Terms of Enrollment Types and Recognition Methods

Types \ Method	PCA	LDA	LBP
#1	84.76%	93.53%	83.18%
#2	81.77%	63.02%	84.08%
#3	63.99%	66.06%	64.07%

It is seen that the LDA-based face recognition method showed the best performance when the type #1 enrollment criterion was used. Because LDA trained enrolled face images in terms of classification, we concluded that the method showed the best recognition accuracy. Because input data has facial pose variation, the type #2 or #3 enrollment criterion is inadequate. Therefore, only the type #1 result is regarded as reliable. PCA showed a similar recognition performance as LBP.

4. Conclusion

In this study, three generally used appearance-based face recognition methods—PCA, LDA, and LBP—were comparatively analyzed in terms of recognition accuracy. In the experiment, facial pose variation was considered on the basis of three types of enrollment criteria. The results show that LDA exhibited the best recognition performance compared with PCA and LBP. In particular, PCA showed a similar performance as LBP. In future, we will apply the type #1 enrollment criterion and the LDA recognition method to CCTV, after which the field test will be performed.

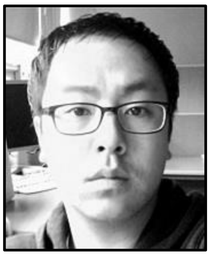
Acknowledgement

This work was supported by the ICT R&D program of MSIP/IITP. [R0126-15-1045 , The development of technology for social life logging based on analyzing social emotion and intelligence of convergence contents]

References

[1] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf, "Deepface: Closing the Gap to Human-Level Performance in Face Verification", Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Conference on, **(2014)**, pp.1701-1708.
[2] Z. Stone, T. Zickler, and T. Darrell, "Autotagging Facebook: Social Network Context Improves Photo Annotation", Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society Conference on **(2008)**, pp.1-8.
[3] W. Zhao, A. Krishnaswamy, R. Chellappa, D. L. Swets, and J. Weng, "Discriminant Analysis of Principal Components for Face Recognition", Face Recognition, Springer Berlin Heidelberg **(1998)**, pp.73-85.
[4] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face Recognition with Local Binary Patterns", Computer Vision – ECCV 2004, Springer Berlin Heidelberg **(2004)**, pp.469-481.
[5] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Face Recognition using Active Appearance Models", in Computer Vision – ECCV 1998, Springer Berlin Heidelberg, **(1998)**, pp.581-595.
[6] H. G. Kim, E. J. Lee, G. J. Yoon, S. D. Yang, E. C. Lee, and S. M. Yoon, "Illumination Normalization for SIFT Based Finger Vein Authentication", in Advances in Visual Computing, Springer Berlin Heidelberg **(2012)**, pp.21-30.
[7] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection", IEEE Trans. Pattern Anal. Mach. Intell. **(1997)**, pp.711-720.
[8] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition", IEEE Trans. Pattern Anal. Mach. Intell, **(2006)**, pp. 2037-2041.

Authors



Eui Chul Lee. He received a BS degree in software from Sangmyung University, Seoul, South Korea, in 2005. He received the MS and PhD degrees in computer science from Sangmyung University in 2007 and 2010, respectively. He was Researcher at the National Institute for Mathematical Sciences from March 2010 to February 2012. Since March 2012, he has been Assistant Professor in the Department of Computer Science at Sangmyung University, Seoul, Korea. His research interests include computer vision, image processing, pattern recognition, brain computer interface (BCI), and human computer interface (HCI).

Unsoo Jang. He received a BS degree from the Department of Computer Science, Sangmyung University, Seoul, South Korea, in 2014. He has been an MS course student in the Department of Computer Science, Sangmyung University, since September 2014. His research interests include pattern recognition, human identification, image analysis, and biosignal analysis.

