Study of Similarity Measures with Linear Discriminant Analysis for Face Recognition

Mohamed A. El-Sayed[1,2], Kadry Hamed[3,4]

[1]Department of Mathematics, Faculty of Science, Fayoum University, Al Fayoum, Egypt
[2]Department of CS, Computers & IT College, Taif University, Taif, KSA
[3]Department of CS, Faculty of Computers and Information, Minia University, Minya, Egypt
[4]Department of CS, College of Computing and Information Technology, Shaqra University, Shaqra, KSA
Email: mas06@fayoum.edu.eg, kadryha1@yahoo.com

Abstract

Face recognition systems have been in the active research in the area of image processing for quite a long time. Evaluating the face recognition system was carried out with various types of algorithms used for extracting the features, their classification and matching. Similarity measure or distance measure is also an important factor in assessing the quality of a face recognition system. There are various distance measures in literature which are widely used in this area. In this work, a new class of similarity measure based on the Lp metric between fuzzy sets is proposed which gives better results when compared to the existing distance measures in the area with Linear Discriminant Analysis (LDA). The result points to a positive direction that with the existing feature extraction methods itself the results can be improved if the similarity measure in the matching part is efficient.

Keywords

Fuzzy Sets, Similarity Measure, Distance Measure, LDA, Face Recognition

1. Introduction

The concept of similarity is fundamentally important in almost every scientific field. For example, in mathematics, geometric methods for assessing similarity are used in studies of congruence and homothetic as well as in allied fields such as trigonometry. Face recognition has been studied extensively for more than 40 years. Now it is one of the most imperative subtopics in the domain of face research [1]-[4]. Face recognition is a technology which recognizes the human by his/her face image. Face recognition has gained much attention in recent years

and has become one of the most successful applications of image analysis and understanding.

A fundamental challenge in face recognition lies in determining which steps are important in the recognition of faces. Several studies [5]-[7] have indicated the significance of certain steps in this regard, particularly pre-processing and feature extraction. Measuring similarity or distance between two data points is a core requirement for several data mining and knowledge discovery tasks that involve distance computation. For continuous data sets, the Minkowski Distance [8] is a general method used to compute distance between two multivariate points. In particular, the Minkowski Distance of order 1 (Manhattan) and order 2 (Euclidean) are the two most widely used distance measures for continuous data. The simplest way to address similarity between two categorical attributes is to assign a similarity of 1 if the values are identical and a similarity of 0 if the values are not identical. For two multivariate categorical data points, the similarity between them will be directly proportional to the number of attributes in which they match. Various similarity measure functions are enumerated in the literature [9] [10].

In this paper, we propose a novel approach to derive the similarity between two images using new similarity measure, by representing each numerical value of their feature vectors as a fuzzy set, instead of a single value. This representation takes into account the uncertainty presents in the extraction process of features and consequently, increases the precision rate in the image retrieval process. The results obtained by the proposed approach present higher performance than the traditional ones.

The rest of this paper is organized as follows: Section 2 presents the mathematical foundations similarity measures on digital images. Section 3 describes the various similarity measures and Section 4 describes about the proposed novel similarity measure with its properties which can be used for effective face recognition. In Section 5, Linear Discriminant Analysis is explained briefly. Experimental details and results on real data are outlined in Section 6, and finally, the conclusions are given in Section 7.

2. Similarity Measure

The similarity of two images is obtained by computing the similarity (or dissimilarity) between their feature vectors [11]. Several measures have been proposed to measure the similarity between fuzzy sets or images [12]-[16]. There is no generic method for selecting a suitable similarity measure or a distance measure. However, a prior information and statistics of features can be used in selection or to establish a new measure. Van der Weken and *et al*. [17] gave an overview of similarity measures, originally introduced to express the degree of comparison between fuzzy sets, which can be applied to images. These similarity measures are all pixel-based, and have therefore not always satisfactory results. To cope with this drawback, in [18] they proposed similarity measures based on neighborhoods, so that the relevant structures of the images are observed better.

Some of the distance or similarity functions frequently used for information retrieval from the databases are listed below [19].

2.1. Euclidean Distance

The Euclidean distance between two points p and q is the length of the line segment connecting them $\left(\overline{pq} \right)$. In Cartesian coordinates, if $p = (p_1, p_2, \cdots, p_n)$ and $q = (q_1, q_2, \cdots, q_n)$ are two points in Euclidean n-space, then the distance from p to q, or from q to p is given by:

$$d(p,q) = d(q,p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \tag{1}$$

2.2. Squared Euclidean Distance

The standard Euclidean distance can be squared in order to place progressively greater weight on objects that are farther apart. In this case, the equation becomes:

$$d^2(p,q) = (q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2 \tag{2}$$

2.3. City Block Distance

The City block distance between two points a, b with k dimensions is calculated as follows:

$$d(a,b) = \sum_{i=1}^{k} |a_i - b_i| \tag{3}$$

The City block distance is always greater than or equal to zero. The measurement would be zero for identical points and high for points that show little similarity.

2.4. Minkowski Distance

The Minkowski distance of order p between two points $P = (x_1, x_2, \cdots, x_n)$ and $Q = (y_1, y_2, \cdots, y_n) \in \mathfrak{R}^n$ is defined as:

$$d(x, y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} \tag{4}$$

2.5. Mahalanobis Distance

Formally, the Mahalanobis distance of a multivariate vector $x = (x_1, x_2, \cdots, x_N)^{\mathrm{T}}$ from a group of values with mean $\mu = (\mu_1, \mu_2, \cdots, \mu_N)^{\mathrm{T}}$ and covariance matrix S is defined as:

$$D_M(x) = \sqrt{(x - \mu)^{\mathrm{T}} S^{-1} (x - \mu)} \tag{5}$$

2.6. Chebyshev Distance

The Chebyshev distance between two vectors or points p and q, with standard coordinates p_i and q_i, respectively, is

$$D_{\mathrm{Chebyshev}}(p, q) = \max_i (|p_i - q_i|) \tag{6}$$

2.7. Cosine Correlation

The *Cosine correlation* between two points a and b, with k dimensions is calculated as

$$D(a,b) = \frac{\sum_{j=1}^{k} a_j \times b_j}{norm(a) \times norm(b)} \tag{7}$$

where $norm(a) = \sqrt{\sum_{j=1}^{k} a_j^2}$.

The cosine correlation ranges from +1 to −1 where +1 is the highest correlation. Complete opposite points have correlation −1.

2.8. Canberra Distance

The Canberra distance, d^{CAD}, between two vectors p, q in an n-dimensional real vector space is given as follows:

$$d^{CAD}(p,q) = \sum_{j=1}^{n} \frac{|p_j - q_j|}{|p_j| + |q_j|} \tag{8}$$

where $p = (p_1, p_2, \cdots, p_n)$ and $q = (q_1, q_2, \cdots, q_n)$ are vectors.

3. Proposed Similarity Measure

Similarity measure is defined as the distance between various data points. The performance of many algorithms depends upon selecting a good distance function over input data set. While, similarity is a amount that reflects the strength of relationship between two data items, dissimilarity deals with the measurement of divergence between two data items [20] [21].

3.1. Properties of Similarity Measure

The theory of fuzzy sets $F(X)$ was proposed by Zadeh [21]. A fuzzy set A in a universe $X = \{x_1, x_2, \cdots, x_n\}$ is characterized by a mapping $\chi_A : X \to [0,1]$, which associates with every element x in X by a degree of membership $\chi_A(x)$ in the fuzzy set A. In the following, let $a = \{a_1, a_2, \cdots, a_n\}$ and $b = \{b_1, b_2, \cdots, b_n\}$ be the vector representation of the fuzzy sets A and B respectively, where a_i and b_i are membership values $\chi_A(x_i)$ and $\chi_B(x_j)$ with respect to x_i and x_j $(i, j = 0, 1, 2, \cdots, n)$ respectively. Furthermore, suppose $F(X)$ be the class of all fuzzy sets of X, $A^c \in F(X)$ is the complement of $A \in F(X)$. There is no unique definition for the similarity measure, but the most common used definition is the following [22] [23].

Definition 3.1. A mapping $S : FS(X) \times FS(X) \to [0,1]$, is said to be a measure between Fuzzy Sets $A \in FS(X), B \in FS(X)$ if $S(A, B)$ satisfies the following properties

(SP_1): $0 \le S(A, B) \le 1$, for all $A, B \in F(x)$ is characterized by a mapping $\chi_A : X \to [0,1]$,

(SP_2): $S(A, B) = 1$, iff $A = B$

(SP_3): $S(A, B) = S(B, A)$, $A, B \in FS(X)$

(SP_4): If $A \subseteq B \subseteq C$ for all $A, B, C \in F(x)$ then $S(A, B) \ge S(A, C)$ and $S(B, C) \ge S(A, C)$.

Definition 3.2. If $S(A, B)$ is a similarity measure which defined as above, then

$$d(A, B) = 1 - S(A, B) \tag{9}$$

is a distance measure between A and B.

Definition 3.3. (Minkowski distance): The distance between two sets A and B (L_p metric), is given by:

$$L_p = \left[\sum_{i=1}^{n} |a_i - b_i|^p \right]^{1/p}, \quad p \ge 1 \tag{10}$$

is a distance measure between A and B.

Family of distance based similarity measures presented by Sanitini in the following definition [23]:

Definition 3.4. Let d_r is the distance between two fuzzy sets A and B, then

$$S_r(A, B) = 1 - d_r(A, B), \quad r = 1, 2, \cdots, \infty \tag{11}$$

are similarity measures.

Based on this definition a class of similarity measures can be given in the following.

Definition 3.5. Let $A \in FS(X), B \in FS(X)$ be fuzzy sets, then

$$S_p(A, B) = 1 - \left(\sum_{i=1}^{n} |a_i - b_i|^p \right)^{1/p}, \quad p \ge 1 \tag{12}$$

are similarity measures.

3.2. Formulation of the Proposed Similarity

In this section, we introduce the new classes of similarity measure $S_{i,j}(A, B)$ based on L_p metrics. To prove the proposed new classes of similarity measure, the following lemma will be needed:

Lemma 3.1 Let $p, q \in [0,1]$ and $p \ge q$ then

$$\frac{2p}{1+p} \ge \frac{2q}{1+q} \tag{13}$$

Theorem 3.1 Let $S_i(A, B)$ is a similarity measure between $A \in F(x)$ and $B \in F(x)$, then

$$S_{ii}(A, B) = \frac{2S_i(A, B)}{1 + S_i(A, B)} \tag{14}$$

is a similarity measure. Generally, for $\dfrac{2S(A,B)}{1+S(A,B)}$, the n_{th}-formula is $\dfrac{2^{n}S(A,B)}{1+(2^{n}-1)S(A,B)}$ and for any finite integer n, is defined a class of similarity measures.

Proof: For the first partition of this theorem, it is quite easy to verify that the conditions (SP_{1})-(SP_{3}) are satisfied and for (SP_{4}), the above lemma 3.1 can be used to prove. For the second partition of the theorem, the induction is used as the following:

For $n=1$ its clear. For $n=k$, we have $\dfrac{2^{k}S(A,B)}{1+(2^{k}-1)S(A,B)}$. To prove the case when $n=k+1$, it is enough to note that:

$$\frac{2^{k+1}S(A,B)}{1+(2^{k+1}-1)S(A,B)}=2\left(\frac{2^{k}S(A,B)}{1+(2^{k}-1)S(A,B)}\right)\Bigg/\left(1+\frac{2^{k}S(A,B)}{1+(2^{k}-1)S(A,B)}\right)$$

where the left hand side is similarity measure as in the first partition. So the theory was done.

Theorem 4.2 Let $S(A,B)$ be a similarity measure between $A\in F(x)$ and $B\in F(x)$, then

$$\frac{1-S(A,B)}{1+S(A,B)} \tag{15}$$

is dissimilarity measure. Generally, the n_{th}-formula for $\dfrac{1-S(A,B)}{1+S(A,B)}$ is $\dfrac{1-S(A,B)}{1+(2^{n}-1)S(A,B)}$, for any finite integer n, is defined a class of dissimilarity measures.

Proof: It is noted that: $\dfrac{1-S(A,B)}{1+S(A,B)}=1-\dfrac{2S(A,B)}{1+S(A,B)}$. But from Theorem 4.1, $\dfrac{2S(A,B)}{1+S(A,B)}$ is similarity measure. So, $1-\dfrac{2S(A,B)}{1+S(A,B)}$ is dissimilarity measure.

Similarly, $\dfrac{1-S(A,B)}{1+(2^{n}-1)S(A,B)}=1-\dfrac{2^{n}S(A,B)}{1+(2^{n}-1)S(A,B)}$ is dissimilarity measure.

Example: For low dimensional p, such that $p=\{1,2,3,4\}$, we describe a several similarity measures which are a combination of the similarity measures, S_{1},S_{2},S_{3} and S_{4} according to the above relation in theorem 4.1.

$$S_{1}=1-\sum_{i=1}^{n}|a_{i}-b_{i}|, \quad S_{2}=1-\sqrt{\sum_{i=1}^{n}(a_{i}-b_{i})^{2}}, \quad S_{3}=1-\left(\sum_{i=1}^{n}|a_{i}-b_{i}|^{3}\right)^{1/3}, \quad \text{and} \quad S_{4}=1-\left(\sum_{i=1}^{n}(a_{i}-b_{i})^{4}\right)^{1/4}.$$

From the above similarity measures, S_{1},S_{2},S_{3} and S_{4}, the classes using the relation in Equation (14) can be driven:

$$S_{11}(A,B)=\frac{2S_{1}(A,B)}{1+S_{1}(A,B)}, \quad S_{22}(A,B)=\frac{2S_{2}(A,B)}{1+S_{2}(A,B)}, \quad S_{33}(A,B)=\frac{2S_{3}(A,B)}{1+S_{3}(A,B)}, \quad \text{and} \quad S_{44}(A,B)=\frac{2S_{4}(A,B)}{1+S_{4}(A,B)}.$$

It can extend to the following classes (practically these classes are satisfies all previous conditions of similarity measures):

$$S_{12}(A,B)=\frac{2S_{1}(A,B)}{1+S_{2}(A,B)}, \quad S_{13}(A,B)=\frac{2S_{1}(A,B)}{1+S_{3}(A,B)}, \quad S_{23}(A,B)=\frac{2S_{2}(A,B)}{1+S_{3}(A,B)} \quad\ldots\ldots\text{etc.}$$

4. Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) [24] finds the vectors in the underlying space that best discriminate among classes. For all samples of all classes the between-class scatter matrix S_{B} and the within-class scatter matrix S_{W} are defined by:

$$S_B = \sum_{i=1}^{c} M_i (x_i - \mu)(x_i - \mu)^{\mathrm{T}} \tag{16}$$

$$S_W = \sum_{i=1}^{c} M_i (x_i - \mu)(x_i - \mu)^{\mathrm{T}} \tag{17}$$

where M_i is the number of training samples in class i, c is the number of distinct classes, μ_i is the mean vector of samples belonging to class i and X_i represents the set of samples belonging to class i with x_k being the k-th image of that class. S_W represents the scatter of features around the mean of each face class and S_B represents the scatter of features around the overall mean for all face classes. For more details about LDA, see [25].

LDA uses the class information and finds a set of vectors that maximize the between-class scatter while minimizing the within-class scatter. Beveridge *et al.* [26] claim in their tests that LDA performed uniformly worse than PCA. Martinez [27] states that LDA is better for some tasks, and Navarrete *et al.* [28] claim that LDA outperforms PCA on all tasks in their tests (for more than two samples per class in training phase). The **Figure 1** shows the flow of a general subspace face recognition system.

5. Experimental Design

In this section, two sets of databases which are available in the internet are used. First one called ORL database of face [29] and the second one called the Georgia tech face database [30]. ORL database of face images contains slight variations in illumination, facial expression (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). It is of 400 images, corresponding to 40 subjects (namely, 10 images for each class). Each image has the size of 112×92 pixels with 256 gray levels. The Georgia tech database contains face images which are created and maintained by Georgia institute of technology for the face related research. The faces size average in these images is 150×150 pixels. The images are stored in 50 directories s1, ..., s50. For each directory, there are 15 images 01.jpg, ..., 15.jpg corresponding to one person in the database. The images are resized into 200×200 (which are face centric) for our purpose and converted into grey scale image. For our experiment, 40 specimen folders and 10 images from each specimen are considered.

5.1. Initial Steps

The data of the image folder are unpacked as follows: for each image, the intensity values of all pixels are read continuously starting from pixel position (1,1) and going column width until a new row, then the pixels values of this row are read and finally all the values are made into a row vector. All images are unpacked in this fashion gives a data matrix X of $400 \times 10{,}304$ dimensions. (*i.e.* 400 images and each image having $112 \times 92 = 10{,}304$ pixels). Hence, each image is considered a vector of 10,304 dimensional space. In the case of Georgia Tech database, the data matrix will be $400 \times 40{,}000$ and each image can be considered a vector in 40,000 dimensional space.
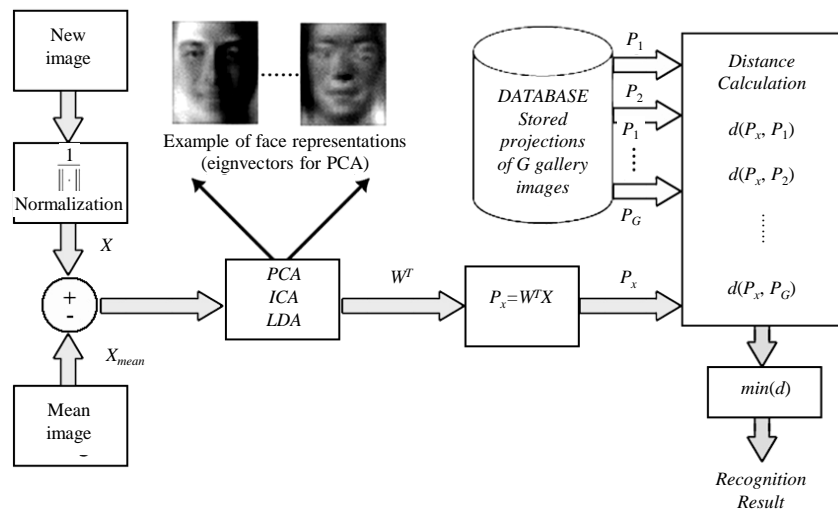
Figure 1. A general subspace face recognition system.

For statistical analysis, the pixels (the data matrix columns) are considered as random variables and the different images (the data matrix rows) are considered as samples of these variables. It is expected that the variables will be highly correlated. This is because most of pixels which form the background will have the same intensity in the image. And thus, they would be highly correlated. Similarly, a lot of other pixels would be correlated. So, there is a lot of redundancy in the data.

5.2. Algorithm in Details

For a given face image, correctly identify the individual is the main objective. To this end, we use the Linear discriminant analysis (LDA) assuming multivariate normality of features given groups and common covariance.

The individuals based on their face images are classified by using Linear Discriminant Analysis (LDA). Each 10,304 (Consider the dataset used is ORL to show the calculations) pixel intensity vector considers a vector of features and each individual considers a class that the features are classified into. Even with the assumptions already made, LDA cannot be performed on the raw pixel intensity values (features) as the covariance matrix of large dimensions ($10,304 \times 10,304$). Computing the covariance matrix and its inverse are both computationally not feasible.

To overcome this problem, Principal Component Analysis (PCA) is performed on the pixel intensity data. But, to get the principal components of X which is $400 \times 10,304$, the covariance matrix which is $10,304 \times 10,304$ dimensions should be obtained and then its eigenvectors are found. So, the problem still remains. Nevertheless, we have to note that the rank of X is only 400, which means there will only be 400 eigenvectors that have non-zero eigen values.

A simple way of finding these eigenvectors is proposed by (Turk and Pentland [31]). If v is an eigenvector of XX^{T} then an eigen vector of $X^{\mathrm{T}}X$ is $X^{\mathrm{T}}v$. Mathematically, if v is an eigenvector of XX^{T} then

$$XX^{\mathrm{T}}v = \lambda v \tag{18}$$

Multiplying both sides by X^{T} we get

$$\left(X^{\mathrm{T}}X\right)X^{\mathrm{T}}v = \lambda\left(X^{\mathrm{T}}v\right) \tag{19}$$

Hence $X^{\mathrm{T}}v$ is an eigenvector of $X^{\mathrm{T}}X$.

Note that X, as used here, is a mean centered. The mean of X (beforemean centering) *i.e.* the mean of all faces is called the *average face*. The average face is shown in **Figure 2**.

Also, note that the eigen values not change *i.e.* the eigenvalues of XX^{T} are the same as those of XX^{T}. The calculated eigenvectors by this way are normalized. The order of the covariance matrix from which eigenvectors have to be estimated is reduced from $10,304 \times 10,304$ to 400×400 (*i.e.* from square of pixels to square of number of images). The eigenvectors and eigen values are estimated using MATLAB. The larger an eigen value the more important the eigenvector, in the sense that the variance of the data set in that direction is more compared to the variance in the direction of eigenvectors with lesser eigen values.

These eigenvectors are called eigen faces [31]. The components of each eigenvector can be regarded as a weight of the corresponding pixel in forming the total image. The components of the eigenvectors calculated, generally do not lie between 0 - 255, which is required to visualize an image. Hence, the eigenvectors are transformed so that the components lie between the 0 - 255 range, for visualization. These eigen faces can be regarded as the faces that make up all the faces in the database, *i.e.* a linear combination of these faces can be used to represent any face in the data set. Also, at least one face can be represented by a linear combination of all the eigenfaces. The first few eigenfaces (after transformation) are shown in **Figure 3**.

Figure 2. The average face.

Figure 3. Eigen faces.

The eigen faces represent vectors in the 10304 dimensional space spanned by the variables (pixels). If the new vectors are considered to be new axes that represent the data, then these new axes will be the principal components we are seeking for. They are just a rotation of the original axes to more meaningful directions. To illustrate how principal components are closer to normality, the normal probability plots some of the first few principal components are given. To check the multivariate normality of the principal components, the uni_variate normality of individual Principal Components (PC's) is checked.

5.3. Projections (Reconstruction of an Image)

The projection of a vector *v* onto a vector *u* is given by:

$$\left(Proj \right)_v^u = \frac{\langle v,u \rangle}{\langle u,u \rangle} u \tag{20}$$

If *v* is any image (after subtracting the *average face*) and *u* is any eigen face, then the above formula gives the projection of the face *v* on the eigen face *u*. Any image can be projected onto each of the first few eigen faces. All the projected vectors are added to get a final image (reconstructed). After this, the average face has to be added again to get the actual projected face onto the lower dimensional space spanned by the eigen faces. For example, a face and its projection of 5, 20, 50, 150, 400 dimensions (8 faces) are shown respectively as **Figure 4**.

Any image in the database can be completely reconstructed by using all the eigen faces. The last image above is a complete reconstruction of the original image.

6. Performing LDA and Experimental Results

With the above basic frame work and with the additional assumption of equivalence of the covariance matrices for each group, LDA for classification can be used. The assumption of equivalence of covariance matrices is quite strong. No statistical test has been performed to check this assumption, as most of these tests are not robust enough (e.g. Box's test). Instead, two-fold cross validation has been performed to check the error rate on the final classification and the assumptions are validated based on these results. The covariance matrix has been estimated using a spooled estimator. The prior probabilities have been considered to be equal. With this spooled estimator and the means of the individual groups, the σ_y's and β_y's for each group have been calculated by

$$\alpha_y = \left(\frac{-1}{2} \right) \times \mu_y' \overset{-1}{\sum} \mu_y, \quad \beta_y = \overset{-1}{\sum} \mu_y \tag{21}$$

The classification rule (Bayes's minimum cost rule) using equal losses then becomes,

$$\delta(x) = \arg\min\left(\sum_{y=1}^{J} I(j \neq y) P(y|x) \right) = \arg\min\left(\sum_{y \neq j} P(y|x) \right)$$

$$= \arg\max\left(P(j|x) \right) = \arg\max\left(\alpha_j + \beta_j' x \right) \tag{22}$$

where *x* is a feature vector and *J* is the total number of classes (equal to 40). The obtained classification depends on the number of principal components.

We can evaluate error rates by means of a training sample (to construct the discrimination surf surface) and a test sample. An optimistic error rate is obtained by reclassifying the design set: this is known as the apparent error rate APER. So that APER is equal to proportion of items in the training set that are misclassified. If an independent test sample is used for classifying, we arrive at the true error rate AER.

Figure 4. Projected faces.

Table 1. Cross validated error rates.

No. of PC's	23	60	100
Error rate %	11	10.25	11.5

Table 2. Performance metrics for the ORL and Georgia-tech face databases using different distance measures.

Performance metrics	Database	Cosine	Euclidean	City block distance	Mahalanobis	Proposed
Rank one recognition rate equals (%)	ORL	86.07	83.21	81.43	83.57	90.00
	Georgia	86.79	86.79	86.79	87.14	91.28
The equal error rate equals (%)	ORL	4.28	6.78	7.85	4.64	3.12
	Georgia	8.58	8.56	9.64	7.86	8.21
Minimal half total error rate equals (%)	ORL	4.09	6.32	7.23	4.13	2.47
	Georgia	6.54	5.80	7.69	5.71	5.95
Verification rate at 1% FAR equals (%)	ORL	90.00	84.29	82.86	88.57	93.75
	Georgia	87.14	87.50	85.36	88.21	87.18
Verification rate at 0.1% FAR equals (%)	ORL	76.43	65.71	65.71	72.14	86.88
	Georgia	85.36	82.86	81.07	86.07	81.54
Verification rate at 0.01% FAR equals (in %)	ORL	64.29	45.36	43.21	53.93	81.25
	Georgia	80.71	75.00	76.07	81.07	81.54

The number of principal components to be used for classification is chosen so that the APER (Apparent Error Rate) is almost 0%. Using 23 principal components, it was found that the APER is 0.02%. The APER is 0% when 60 or more principal components are considered. A closer estimate of the AER (actual error rate) can be calculated using *two-fold cross validation*. These were calculated based on the confusion matrices. The confusion matrices are not shown here as they are of 40×40 dimensions. Instead the final cross validated error rates are shown.

Table 1 shows the number of principal components used and the AER.

It can be seen that the error rate does not decrease much with the increase in the number of principal components. In fact, when 100 PC's are used, the error rate is more than that when only 23 PC's are used. This can be attributed to the larger rounding errors when dealing with inverses of large dimension covariance matrices. When using 100 PC's the spooled covariance matrix becomes 100×100. Because of the error rate does not decrease much with increasing the number of PC's, so 23 PC's can be considered to be optimal. The state of 23 PC's explaining the variance of the data set up to 75%. The APER as the AER is quite small, so the assumptions of multivariate normality of features that given a group and the equivalence of covariance matrices are not unreasonable.

In the stage of classification, various similarity measures for the recognition purpose have been analyzed. Euclidean, City block, Mahanalobis, and Cosine similarity measure are used together with the proposed distance measure. In the proposed similarity measures, Equation (9) is used for the experimental purpose. The results are

divided into two sets. Various performance metrics for the two data sets and the graphical representation or various plots that show the performance and error details are explained as the following:

In the performance metrics, six measures are used. They are Rank one recognition rate, equal error rate, minimal half total error rate, verification rate at 1%, 0.1% and 0.01% of False Acceptance Rate (FAR). For all the five distance measures, these metrics are calculated to both datasets. It is mentioned in **Table 2**. The performance metrics show that the proposed distance outperforms all the other four measures. An average of 90.6% of recognition rate is achieved which far better than the other measures.

7. Conclusion

Analysis of various similarity measures was carried out in the context of face recognition. Feature extraction was done with the renowned Linear Discriminant Analysis method. A new similarity measure based on fuzzy Minkowski's theorem is introduced in this work. Various similarity measures such as cosine similarity, Euclidean distance, City Block distance and Mahanalobis distance measures were compared with the proposed similarity measure. This novel similarity measure outperformed all the other distance measures. These results provide us the strong guideline in developing future face recognition using an efficient similarity measure. Also, we can use this measure as an important factor for evaluating a face recognition system on tested datasets.

References

[1] Zhao, W., Chellappa, R., Rosenfeld, A. and Phillips, P.J. (2003) Face Recognition: A Literature Survey. *ACM Computing Surveys*, **35**, 399-458. http://dx.doi.org/10.1145/954339.954342

[2] Tolba, A.S., El-Baz, A.H. and El-Harby, A.A. (2005) Face Recognition: A Literature Review. *International Journal of Signal Processing*, **2**.

[3] Chellappa, R., Wilson, C.L. and Sirohey, C. (1995) Human and Machine Recognition of Faces: A Survey. *Proceedings of the IEEE*, **83**, 705-740. http://dx.doi.org/10.1109/5.381842

[4] Yang, M.-H., Kriegman, D.J. and Ahuja, N. (2002) Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**. http://dx.doi.org/10.1109/34.982883

[5] Tolba, A.S., El-Baz, A.H. and El-Harby, A.A. (2006) Face Recognition: A Literature Review. *International Journal of Information and Communication Engineering*, **2**, 88-103.

[6] Zhao, W.-Y., Chellappa, R., Jonathon Phillips, P. and Rosenfeld, A. (2003) Face Recognition: A Literature Survey. *ACM Computing Surveys*, **35**, 399-458. http://dx.doi.org/10.1145/954339.954342

[7] Philips, P.J., Moon, H., Rauss, P. and Rizvi, S.A. (1997) The FERET Evaluation Methodology for Face Recognition Algorithms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 137-143.

[8] Voitsekhovskii, M.I. (2001) Minkowski Inequality. Hazewinkel, Michiel, Encyclopedia of Mathematics, Springer.

[9] Cha, S.-H. (2007) Comprehensive Survey on Distance/Similarity Measures between Probability Density Function. *International Journal of Mathematical Models and Methods in Applied Sciences*, **1**.

[10] Randall, W.D. and Martinez, T.R. (1997) Improved Heterogeneous Distance Functions. *Journal of Artificial Intelligence Research*, **6**, 1-34.

[11] Hjaltason, G.R. and Samet, H. (2003) Index-Driven Similarity Search in Metric Spaces. Survey Article, ACM Transaction on Database Systems.

[12] Kwon, S.H. (2001) A Similarity Measure of Fuzzy Sets. *Journal of Fuzzy Logic and Intelligent Systems*, **11**, 270-274.

[13] Liu, X. (1992) Entropy, Distance Measure and Similarity Measure of Fuzzy Sets and Their Relations. *Fuzzy Sets and Systems*, **52**, 305-318. http://dx.doi.org/10.1016/0165-0114(92)90239-Z

[14] Wang, X., DeBaets, B. and Kerre, E.E. (1995) A Comparative Study of Similarity Measures. *Fuzzy Sets and Systems*, **73**, 259-268. http://dx.doi.org/10.1016/0165-0114(94)00308-T

[15] Chen, S.M. (1995) Measures of Similarity between Vague Sets. *Fuzzy Sets and Systems*, **74**, 217-223. http://dx.doi.org/10.1016/0165-0114(94)00339-9

[16] Wang, W.J. (1997) New Similarity Measure on Fuzzy Sets and on Elements. *Fuzzy Sets and Systems*, **85**, 305-309. http://dx.doi.org/10.1016/0165-0114(95)00365-7

[17] Van der Weken, D., Nachtegael, M. and Kerre, E.E. (2002) An Overview of Similarity Measures for Images. *Proceeding of ICASSP* 2002 (*IEEE International Conference on Acoustics*, *Speech and Signal processing*), Orland, 13-17 May 2002, 3317-3320.

[18] Van der Weken, D., Nachtegael, M. and Kerre, E.E. (2004) Using Similarity Measures and Homogeneity for the Com-

parison of Images. *Image and Vision Computing*, **22**, 695-702. http://dx.doi.org/10.1016/j.imavis.2004.03.002

[19] Zadeh, L.A. (1965) Fuzzy Sets. *Information and Control*, **8**, 338-353.
http://dx.doi.org/10.1016/S0019-9958(65)90241-X

[20] Huang, A. (2008) Similarity Measures for Text Document Clustering. *NZCSRSC* 2008, Christchurch, 14-18 April 2008.

[21] Taghva, K. and Veni, R. (2010) Effects of Similarity Metrics on Document Clustering. 2010 7*th International Conference on Information Technology*, Las Vegas, 12-14 April 2010, 222-226. http://dx.doi.org/10.1109/itng.2010.65

[22] Omran, S. and Hassaballah, M. (2006) A New Class of Similarity Measures for Fuzzy Sets. *International Journal of Fuzzy Logic and Intelligent Systems*, **6**, 100-104. http://dx.doi.org/10.5391/IJFIS.2006.6.2.100

[23] Wang, W. and Xin, X. (2005) Distance Measure between Intuitionistic Fuzzy Sets. *Pattern Recognition Letters*, **26**, 2063-2069. http://dx.doi.org/10.1016/j.patrec.2005.03.018

[24] Mika, S., Ratsch, G., Weston, J., Scholkopf, B. and Muller, K. (1999) Fisher Discriminant Analysis with Kernels. *IEEE Conference on Neural Networks for Signal Processing IX*, Madison, 23-25 August 1999, 41-48. http://dx.doi.org/10.1109/nnsp.1999.788121

[25] Zhao, W., Chellappa, R. and Phillips, J. (1999) Subspace Linear Discriminant Analysis for Face Recognition. Technical Report, CFAR-TR99-914, University of Maryland.

[26] Beveridge, J.R., She, K., Draper, B. and Givens, G.H. (2001) A Nonparametric Statistical Comparison of Principal Component and Linear Discriminant Subspaces for Face Recognition. *Proceedings of the* 2001 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, 8-14 December 2001, 535-542. http://dx.doi.org/10.1109/cvpr.2001.990520

[27] Martinez, A. and Kak, A. (2001) PCA versus LDA. *IEEE Transactions on Pattern Analysis Machine Intelligence*, **23**, 228-233. http://dx.doi.org/10.1109/34.908974

[28] Navarrete, P. and Ruiz-del-Solar, J. (2002) Analysis and Comparison of Eigen Space Based Face Recognition Approaches. *International Journal on Pattern Recognition and Artificial Intelligence*, **16**, 817-830. http://dx.doi.org/10.1142/S0218001402002003

[29] The ORL Database of Faces. http://www.face-rec.org/databases/

[30] Nefian, A.V., Khosravi, M. and Hayes III, M.H. (1997) Real-Time Human Face Detection from Uncontrolled Environments. SPIE Visual Communications on Image Processing.

[31] Turk, M. and Pentland, A. (1991) Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, **3**, 71-86. http://dx.doi.org/10.1162/jocn.1991.3.1.71