

Research Article

Sparse Graph Embedding Based on the Fuzzy Set for Image Classification

Minghua Wan ^{1,2,3} Mengting Ge ^{1,2} Tianming Zhan ^{1,2} Zhangjing Yang ^{1,2}
Hao Zheng ³ and Guowei Yang ^{1,2,4}

¹School of Information Engineering, Nanjing Audit University, Nanjing 211815, China

²Jiangsu Key Laboratory of Auditing Information Engineering, Nanjing Audit University, Nanjing 211815, China

³Key Laboratory of Intelligent Information Processing, Nanjing Xiaozhuang University, Nanjing 211171, China

⁴School of Electronic Information, Qingdao University, Qingdao 266071, China

Correspondence should be addressed to Guowei Yang; ygw_ustb@163.com

Received 8 November 2020; Revised 12 December 2020; Accepted 4 January 2021; Published 16 January 2021

Academic Editor: Jianxin Li

Copyright © 2021 Minghua Wan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, many face image feature extraction and dimensional reduction algorithms have been proposed for linear and nonlinear data, such as local-based graph embedding algorithms or fuzzy set algorithms. However, the aforementioned algorithms are not very effective for face images because they are always affected by overlaps (outliers) and sparsity points in the database. To solve the problems, a new and effective dimensional reduction method for face recognition is proposed—sparse graph embedding with the fuzzy set for image classification. The aim of this algorithm is to construct two new fuzzy Laplacian scattering matrices by using the local graph embedding and fuzzy k -nearest neighbor. Finally, the optimal discriminative sparse projection matrix is obtained by adding elastic network regression. Experimental results and analysis indicate that the proposed algorithm is more effective than other algorithms in the UCI wine dataset and ORL, Yale, and AR standard face databases.

1. Introduction

Machine learning has become one of the main research tools for small sample learning in complex networks. Using the ability of small sample and the knowledge learned, it is helpful to improve the model of key participant discovery in reality and effectively find the key participants under different objective criteria. There are more and more manifold learning algorithms with local preserving ability, which have become the active research fields in machine learning and pattern recognition [1–6]. Among them, the classical algorithms such as Laplacian eigenmaps (LE) [7, 8], isometric feature mapping (ISOMAP) [9], local linear embedding (LLE) [10, 11], and local tangent space alignment (LTSA) [12] are widely applied in machine learning and pattern recognition. LLE algorithm creates local optimal coordinates with least construction cost to map them to global coordinates. However, LLE algorithm lacks the generalization ability when applied to new face

samples. Therefore, neighborhood preserving embedding (NPE) [13] was proposed by He et al., which maintained the optimal local neighborhood reconstruction relationship on the data. Another representative learning method of local preserving projection (LPP) [14, 15] is an unsupervised algorithm derived from LE, where local information can be found and retained in an embedding space. Recently, in order to unify the manifold learning framework, linear graph embedding (LGE) algorithm was proposed by Yan et al. [16].

However, in the real world, the face image is always affected by various expressions, posture, and illumination variation, so LGE and its extended algorithms cannot effectively solve the above problems [17–19]. By using the fuzzy k -nearest neighbor of fuzzy set [20] technology, a lot of research studies have been completed on fuzzy image filtration, fuzzy edge detection, and fuzzy image segmentation. Its ultimate goal is to solve the uncertainty in pattern recognition and machine learning [21–24].

However, all the above methods are difficult to get a reasonable explanation, that is, feature extraction and classification; feature plays an important role in high-dimensional data [25–27]. Many sparse subspace learning methods have aroused great research interest [28–30] in recent years. In recent years, the more effective sparse feature extraction methods include the least absolute shrinkage and selection operator (LASSO) [31], LARS [32], and elastic net [33]. Under the L_1 -norm, these methods can make the corresponding coefficient of the partial characteristics shrink to zero [24–38].

In order to solve the above problems, we study the LGE algorithm based on subspace learning, elastic network regression, and fuzzy set theory, namely, sparse graph embedding with fuzzy set for image classification. In the proposed method, fuzzy k -nearest neighbor (FKNN) and sparse learning are used to improve the discrimination ability of the method in a low-dimensional space. Firstly, the value of the membership matrix is calculated by FKNN, and then the value is introduced into the Laplacian scatter matrix of within-class and between-class, respectively.

The novelty of our algorithm includes the following perspectives:

- (1) In this algorithm, manifold learning method and fuzzy set are used to keep the original neighbor relations and the intrinsic geometry structure for overlapped (outliers) data points. At the same time, the algorithm uses FKNN to construct two new fuzzy graphs, which has stronger recognition ability compared with other graph-based algorithms.
- (2) The proposed algorithm has a lower sensitivity to sparsity points in the data through elastic network regression which enhances the robustness of our algorithm. The sparse projections learned by the proposed algorithm indicate physical interpretation in real-world applications.

The rest of this paper is planned as follows: the idea of linear methods is described in Section 2. The algorithm innovation and implementation are reviewed in Section 3. The experimental results of the proposed algorithm verify its discrimination and effectiveness on four different standard databases in Section 4. The conclusions are summarized in Section 5.

2. Outlines of FKNN, LLE, and Lasso Regression Net

Firstly, we consider N set samples $X = \{x_1, x_2, \dots, x_N\}$ of linear transformation mapping original space $x_i \in R^D$ into feature space $Y = \{y_1, y_2, \dots, y_N\} \in R^d$, where $n > d$. The new eigenvector $y_i \in R^d$ is from the linear transformation formula:

$$y_i = P^T x_i, \quad i = 1, \dots, N, \quad (1)$$

where $P \in R^{n \times d}$ is the linear transformation projection matrix. The algorithms of FKNN, LLE, and lasso regression network are reviewed, respectively.

2.1. Fuzzy k -Nearest Neighbor (FKNN). In the FKNN algorithm, the value of the membership degree is generated by the class assignment of the fuzzy k -nearest neighbor and the corresponding degrees of class membership. The following five steps comprise the FKNN algorithm:

- Step 1: the value of the Euclidean distance matrix, which is calculated between eigenvectors using the training sample set.
- Step 2: the value of the diagonal element is set to infinity by using the Euclidean distance matrix.
- Step 3: the value of the distance matrix in the ascending order (process each column separately).
- Step 4: the class corresponding to the set pattern, which is in the nearest neighborhood of the pattern sample (because k neighborhoods are concerned, this returns a list of k integers).
- Step 5: the expression proposed in [20], which is used to calculate the type i membership of mode j , which is

$$u_{ij} = \begin{cases} 0.51 + 0.49 \times \left(\frac{n_{ij}}{K}\right), & \text{if } j^{\text{th}} \text{ sample} \in i^{\text{th}} \text{ class,} \\ 0.49 \times \left(\frac{n_{ij}}{K}\right), & \text{otherwise,} \end{cases} \quad (2)$$

where n_{ij} is the j^{th} sample (patterns) inclusion of the i^{th} class. For example, if n_{ij} has neighbors belonging to the equal pattern, the member value will remain close to 0.51. Otherwise, it will remain close to 0.49. Generally, u_{ij} contains the following important properties:

$$\sum_{i=1}^c u_{ij} = 1, \quad 0 < \sum_{j=1}^N u_{ij} < N. \quad (3)$$

Therefore, U can be obtained by the FKNN algorithm:

$$U = [u_{ij}], \quad i = 1, 2, \dots, c, \quad j = 1, 2, \dots, N, \quad (4)$$

where c is the total class number and N is all sample numbers in the i^{th} class.

2.2. Locally Linear Embedding. LLE is a classic method of manifold learning methods, which can better maintain the original manifold structure after dimensionality reduction. There are three main steps to the algorithm.

The first step is to calculate the k -nearest neighbor relation of each data point x_i by Euclidean distance.

The second step is to count the coefficient of $W = [w_{ij}]_{N \times N}$ by minimizing the reconstruction error. We can get the reconstruction weight matrix W :

$$\min J_L(W) = \sum_{i=1}^N \left\| x_i - \sum_{j=1}^N w_{ij} x_j \right\|^2. \quad (5)$$

The reconstruction error x_i is transformed into the following equation:

$$\begin{aligned} \xi_i &= \left\| x_i - \sum_{j=1}^N w_{ij} x_j \right\|^2 \\ &= \left\| \sum_{j=1}^N w_{ij} (x_i - x_j) \right\|^2 \\ &= \sum_{j=1}^N w_{ij} (x_i - x_j) \sum_{t=1}^N w_{it} (x_i - x_t) \\ &= \sum_{j=1}^N \sum_{t=1}^N w_{ij} w_{it} G_{jt}^i, \end{aligned} \quad (6)$$

where $G_{jt}^i = (x_i - x_j)^T (x_i - x_t)$ is the local Gram matrix. Optimal coefficient with $\sum_{j=1}^N w_{ij} = 1$ constraint is obtained using the least square problem to solve

$$w_{ij} = \frac{\sum_{t=1}^N (G^i)_{jt}^{-1}}{\sum_{p=1}^N \sum_{q=1}^N (G^i)_{pq}^{-1}}. \quad (7)$$

After iterating steps 1 and 2 for N times, we get the reconstruction weight coefficient matrix $W = [w_{ij}]_{N \times N}$.

The third step is to use weight matrix W to reconstruct y_i in equation (7).

By minimizing the reconstruction error function, the inherent geometric features of the embedded data can be maintained. Considering the transformation $y_i = P^T x_i$, the objective function is simplified as follows:

$$\min J_L(P) = \sum_{i=1}^N \left\| y_i - \sum_{j=1}^N w_{ij} y_j \right\|^2 = \text{tr}\{P^T X M X^T P\}, \quad (8)$$

where $M = (I - W)^T (I - W)$.

2.3. Lasso Regression Network. In this study, we use the lasso algorithm to obtain the optimal solution for sparse samples. In this algorithm, a penalty function is constructed to obtain the biased estimation of complex linear data processing. Using the lasso interpretation model, the absolute value of the regression coefficient minimizes the sum of the squares of the residuals so that the regression coefficient is exactly equal to 0.

Suppose we have the value of $X^i = (x_{i1}, \dots, x_{ip})^T$ and the value of $(X^i, y_i), i = 1, 2, \dots, N$ in the i^{th} observation variable.

$$y_i = \alpha_i + \sum_{j=1}^p (\beta_j x_{ij} + \varepsilon_i), \quad (9)$$

where x_i and y_i are variables and p is the number of elements in x_i .

In order to further elaborate and solve this problem, it is assumed that the response variables y_i are independent of the given observation conditions.

In general, the regression structure is as follows:

$$\begin{aligned} \frac{1}{N} \sum_j x_{ij} &= 0, \\ \frac{1}{N} \sum_j x_{ij}^2 &= 0. \end{aligned} \quad (10)$$

The available lasso estimate is

$$\begin{aligned} (\hat{\alpha}, \hat{\beta}) &= \arg \min \left\{ \sum_i \left(y_i - \alpha_i - \sum_j \beta_j x_{ij} \right)^2 \right\}, \\ \text{s.t. } \sum_j |\beta_j| &\leq j. \end{aligned} \quad (11)$$

Here, $t \geq 0$ is a harmonic parameter, and $\alpha: \hat{\alpha} = \bar{y}$ is obtained for all t . Let $t_0 = \sum_j |\beta_j|, t \leq t_0$, and the number of regression coefficients will decrease to 0. If the harmonic parameter t is controlled, the regression coefficient can be reduced.

3. Sparse Graph Embedding with Fuzzy Set

3.1. The Basic Idea of the Proposed Algorithm. Firstly, two new fuzzy graphs are constructed to keep the same data points together and different data points separate by the FKNN algorithm. Secondly, in order to avoid singularity, the algorithm seeks the difference between the fuzzy scatter matrix of within-class and between-class by the MMC [39] framework. Finally, the optimal mapping projection matrix is acquired by the lasso regression network.

3.2. Fuzzy Within-Class Scatter Matrix with Locality Preserving. Firstly, we minimize the fuzzy within-class scatter matrix while maintaining the locality preserving of each data point x_i and reconstruct the data points by using the linear coefficients of other data points. The algorithm is divided into four steps:

- (1) The K_c -nearest neighbors of x_i are obtained by Euclidean distance.
- (2) The reconstruction weight matrix $W = [w_{kj}]_{N \times N}$ is minimized by

$$\min J_L(W) = \sum_{i=1}^N \left\| x_i - \sum_{j=1}^{K_c} w_{ij} x_j \right\|^2, \quad (12)$$

where $w_{kj} = 0$. If x_k is not the neighbor of x_j , the row sum of W is 1: $\sum_{j=1}^{K_c} w_{kj} = 1$.

We get $W = [w_{kj}]_{N \times N}$ after repeating steps 1 and 2 for N times.

- (3) To calculate the fuzzy weight of affinity, we can specify the new weight matrix W_{ij}^G :

$$W_{ij}^G = u_{ij}^G \cdot * W, \quad (13)$$

where $u_{ij}^G = \begin{cases} 0.51 + 0.49 \times (n_{ij}/K_c) \\ 0.49 \times (n_{ij}/K_c) \end{cases}$ and “*” denotes the matrix element-wise multiplication.

- (4) The reconstruction error function is used to reconstruct weight matrix W^G expressed as y_j :

$$\min J_L(Y) = \sum_{i=1}^N \left\| y_i - \sum_{j=1}^N W_{ij}^G y_j \right\|^2, \quad (14)$$

where y_j is the neighbor data point of y_i and y_i is the output of x_i .

- The objective function can be obtained by using equation (1) as follows:

$$\begin{aligned} J_L(V) &= \sum_{i=1}^N \left\| y_i - \sum_{j=1}^N W_{ij}^G y_j \right\|^2 \\ &= \text{tr} \left\{ \sum_{i=1}^N \left(y_i - \sum_{j=1}^N W_{ij}^G y_j \right) \left(y_i - \sum_{j=1}^N W_{ij}^G y_j \right)^T \right\} \\ &= \text{tr} \left\{ \sum_{i=1}^N \left(y_i - \sum_{j=1}^N W_{ij}^G y_j \right) \left(y_i - \sum_{j=1}^N W_{ij}^G y_j \right)^T \right\} \\ &= \text{tr} \left\{ Y \left(I - (W^G)^T \right) \left(I - (W^G)^T \right)^T Y^T \right\} \\ &= \text{tr} \left\{ Y \left(I - W^G \right)^T \left(I - W^G \right) Y^T \right\} \\ &= \text{tr} \left\{ P^T X M X^T P \right\}, \end{aligned} \quad (15)$$

where matrix $M = (I - W^G)^T (I - W^G)$.

3.3. Fuzzy Between-Class Scatter Matrix with Locality Preserving. First, if they have different labels, we output the sum of the maximum square distances between the data. Therefore, the following three steps are used to solve the fuzzy between-class scatter matrix with locality preserving:

- (1) The similarity matrix W_{ij}^P is constructed as follows:

$$W_{ij}^P = \begin{cases} 1, & x_i \in N_{K_p}^+(x_j), \\ 0, & \text{otherwise,} \end{cases} \quad (16)$$

where $N_{K_p}^+(x_j)$ indicates that K_p is the nearest neighbor of sample data point X_j in the different class.

- (2) To calculate the fuzzy weight of affinity, we can specify the new weight matrix K_p :

$$W_{ij}^{G'} = u_{ij}^{G'} \cdot * W_{ij}^P, \quad (17)$$

$$\text{where } u_{ij}^{G'} = \begin{cases} 0.51 + 0.49 \times (n_{ij}/K_p) \\ 0.49 \times (n_{ij}/K_p) \end{cases}.$$

- (3) Complete the final embedding:

$$\max J_G(Y) = \sum_{i=1}^N \sum_{j=1}^N \|y_i - y_j\|^2 W_{ij}^{G'}. \quad (18)$$

By considering the mapping relation in equation (1), the objective function can be simplified:

$$\begin{aligned} J_G(V) &= \sum_i \sum_j \|y_i - y_j\|^2 W_{ij}^{G'} \\ &= \sum_i \sum_j \|P^T x_i - P^T x_j\|^2 W_{ij}^{G'} \\ &= 2P^T X (D^P - W^P) X^T P \\ &= 2P^T X L^P X^T P. \end{aligned} \quad (19)$$

3.4. The Criteria of the Proposed Algorithm. Finally, the MMC algorithm framework is used to minimize fuzzy within-class scatter matrix $J_L(V)$ and maximize fuzzy between-class scatter matrix $J_G(V)$ to find the optimal projection matrix P . In fact, we can get such optimal projection matrix P through the following multiobjective optimization problems, namely,

$$\begin{cases} \min & \text{tr}\{P^T X M X^T P\} \\ \max & \text{tr}\{P^T X L^P X^T P\} \\ \text{s.t.} & P^T X X^T P = I. \end{cases} \quad (20)$$

Finally, the above formula can be transformed into a constraint problem:

$$\begin{aligned} \min & \text{tr}\{P^T X (M - \mu L^P) X^T P\} \\ \text{s.t.} & P^T X X^T P = I, \end{aligned} \quad (21)$$

Where $\mu > 0$ is a constant used to balance matrix M and matrix L^P .

The Lagrange multiplier method can be used in equation (21):

$$\begin{aligned} L(P, \lambda) &= \left\{ P^T X (M - \mu L^P) X^T P - \eta (P^T X X^T P - I) \right\} \\ &= 0, \end{aligned} \quad (22)$$

where η is the Lagrangian multiplier parameter. Thus, the above formula can be further simplified to obtain

$$X (M - \mu L^P) X^T P = \lambda_i X X^T P, \quad (23)$$

where P_i is a generalised eigenvector corresponding to a generalised eigenvalue λ_i .

Projection matrix P can be obtained as follows:

$$\begin{cases} X (M - \mu L^P) X^T P = \lambda_i X X^T P, \\ \text{Card}(P) = K, \end{cases} \quad (24)$$

where $\text{Card}(P) = K \leq n$ indicates the quantity of nonzero elements in matrix P . In this way, the optimal sparse projection matrix $P = [p_1, p_2, \dots, p_d]$ can be obtained in equation (24).

At last, we need the following theorem of feature analysis. After that, the learning problem of the sparse projection function is transformed into the regression framework.

Theorem 1. *Let y be the eigenvector of the following formula with eigenvalue λ :*

$$(M - \mu L^P)y = \lambda y. \quad (25)$$

If $X^T p = y$, then v is the eigenvector of the eigenproblem with the same eigenvalue λ in equation (23).

Proof. Since $X^T p = y$, it is easy to prove the following formula according to equation (23):

$$\begin{aligned} X(M - \mu L^P)y &= \lambda Xy \\ \Rightarrow X(M - \mu L^P)X^T p &= \lambda XX^T p. \end{aligned} \quad (26)$$

Therefore, p is the eigenvector of the same eigenvalue of equation (23).

We can solve this problem with ridge regression:

$$P = \arg \min_p \left(\sum_{i=1}^m (p^T x_i - \bar{y}_i)^2 + \alpha \sum_{j=1}^n \bar{v}_j^2 \right), \quad (27)$$

where $\alpha \geq 0$. \bar{p}_j is the j^{th} element of v , and \bar{y}_i is the i^{th} element of y .

However, the ridge penalty of equation (27) does not supply sparse matrix p_i . When using the lasso regression algorithm, add L_1 -norm to v_i to get

$$P = \arg \min \left(\sum_{i=1}^m (p_i^T x_i - y_i)^2 + \beta \sum_{j=1}^n |p_j| \right). \quad (28)$$

So, we integrate the lasso regression and ridge regression [40]:

$$P = \arg \min \left(\sum_{i=1}^m (p_i^T x_i - y_i)^2 + \alpha \sum_{j=1}^n (\bar{p}_j)^2 + \beta \sum_{j=1}^n |p_j| \right). \quad (29)$$

Then, we get the optimal sparse transformation matrix $P = (p_1, p_2, \dots, p_l)$.

4. Experimental Results and Analysis

In order to test and verify the effectiveness of the proposed algorithm, we compared it with PCA [41], LDA [42], MMC [39], LPP [14, 15], LLE [10, 11], SDE [38], and FLGE/MMC [4] on the UCI wine dataset and ORL, Yale, and AR face standard databases, respectively. In our experiments, the values of parameters in our algorithm are given in Table 1. Each experiment runs independently for ten times when different training samples are taken for the training set. In

the experiment, we used Euclidean distance and nearest neighbor classifier to measure and classify, respectively. The four databases' experimental results are completed in the same experimental environment (CPU: P4 4GHz; RAM: 8 GB).

4.1. The Experiments and Interpretations in the UCI Wine Database. Now, we use a wine database from the UCI machine learning library to further validate and interpret the proposed method. There are 3 classes, 13 features, and 178 instances in the wine database. We select 48 samples in each class, and the basic statistics of the variance, mean, and range of the 13 features in the experiment are shown in Table 2.

Both the proposed method and USSL sparse projections are suitable for classification. We learn that there are precise physical explanations, which provide us a deeper understanding of the database. That is to say, the second (malic acid), the fourth (ash alkaline), the fifth (magnesium), the seventh (xanthic acid), the tenth (color intensity), and the thirteenth (proline) are the most important and significant distinguishing features. Moreover, the proposed method consistently outperforms USSL within lower dimension.

4.2. The Experiments in the ORL Database. There are 40 classes in the ORL face database [43]; each class contains 10 different face images for a total of 400 face images, among which facial expression, posture, and illumination vary. The images allow the face to tilt and rotate 20 degrees. In addition, there are some changes as high as about 10% in this scale. Figure 1 shows ten images of one class in the ORL face database. All images were normalized to 56×46 resolution.

In this experiment, the performance of PCA, MMC, LDA, LLE, LPP, SDE, FLGE/MMC, and the new proposed method is tested, respectively. First of all, T images (T ranging from 2 to 6) from each class of the image database are randomly selected as the training sample set. Secondly, the remaining $10 - T$ face images of each class are selected as the testing sample set. Finally, we independently run 50 times on T images of each class.

In the PCA phase of seven algorithms (MMC, LDA, LPP, LLE, SDE, FLGE/MMC, and the proposed algorithm), 95% of the face image energy is retained for computational effectiveness. Table 3 displays the standard deviation, the best average recognition rates (%), and corresponding feature dimension of various algorithms in the ORL face database.

Figure 2 shows the curve when four face images in each class are randomly selected as the training set in the ORL face database. The curve shows the average recognition rate (%) of each algorithm corresponding to different dimensions.

4.3. The Experiments in the Yale Database. There are 15 classes in the Yale face database [44]; each class contains 11 different face images, a total of 165 face images, among which normal, surprised, happy, sad, sleepy, and winking are varied. Figure 3 shows 11 images of one class in the Yale face database.

TABLE 1: Parameter values of our proposed method on different databases.

Dataset	The parameter of our proposed method				Run times
	T	K_C	K_p	K	
UCI wine	48	$T - 1$	2 : 2 : 20	10 : 10 : 100	10
ORL	2 : 1 : 6	$T - 1$	2 : 2 : 20	5 : 5 : 40	50
Yale	2 : 1 : 6	$T - 1$	2 : 2 : 20	5 : 5 : 40	50
AR	2 : 1 : 6	$T - 1$	5 : 5 : 50	10 : 10 : 40	10

T is training images from each class.

TABLE 2: The first sparse projection axis and the corresponding sparse load ($k = 5$) of the proposed method and USSL.

Features	Mean	Variance	Range	USSL	The proposed method
1	13.0	0.7	3.8	0	0
2	2.3	1.2	5.1	0.13	0.28
3	2.4	0.08	1.9	0	0
4	19.5	11.2	19.4	0.09	0.36
5	99.7	204.0	92.0	0.02	0
6	2.3	0.4	2.9	0	0
7	2.0	1.0	4.7	-0.99	-0.95
8	0.4	0.02	0.5	0	0
9	1.6	0.3	3.2	0	0
10	5.1	5.4	11.7	0	0.19
11	1.0	0.05	1.2	0	0
12	2.6	0.5	2.7	0	0
13	746.9	99166.7	1402	0.0104	0.0008



FIGURE 1: Ten images of one class in the ORL face database.

TABLE 3: The standard deviation, the best average recognition rates (%), and corresponding feature dimension of various algorithms in the ORL face database.

Methods	T				
	2	3	4	5	6
PCA	72.25 ± 2.61 (44)	81.62 ± 0.65 (44)	83.98 ± 1.00 (46)	85.41 ± 1.30 (44)	87.41 ± 1.98 (46)
LDA	76.35 ± 1.07 (38)	83.40 ± 1.69 (38)	84.13 ± 2.03 (38)	86.07 ± 1.16 (38)	88.60 ± 0.78 (38)
MMC	75.01 ± 1.77 (40)	83.96 ± 0.48 (38)	85.37 ± 2.48 (40)	87.47 ± 0.85 (38)	90.49 ± 1.17 (36)
LPP	73.52 ± 0.05 (42)	80.28 ± 0.78 (44)	89.18 ± 1.42 (44)	91.13 ± 0.82 (40)	92.12 ± 1.21 (38)
LLE	69.78 ± 0.82 (46)	72.66 ± 0.73 (50)	76.32 ± 1.14 (50)	78.85 ± 1.15 (50)	88.44 ± 1.55 (48)
SDE	77.29 ± 1.20 (28)	81.20 ± 0.27 (28)	87.89 ± 1.39 (28)	89.24 ± 2.41 (26)	90.12 ± 0.68 (30)
FLGE/MMC	79.30 ± 0.83 (46)	88.61 ± 1.79 (42)	93.53 ± 1.82 (50)	95.84 ± 1.23 (48)	97.36 ± 1.24 (50)
The proposed method	91.75 ± 0.45 (26)	93.14 ± 1.08 (26)	95.14 ± 1.22 (30)	96.30 ± 1.24 (26)	98.13 ± 1.23 (24)

In our experiment, all images were normalized to 50×40 resolution in the Yale database. It also retains 95% of face image energy as the same as the ORL face database. First of all, T images (T ranging from 2 to 6) from each class of image database are randomly selected as the training sample set. Secondly, the remaining $11 - T$ face images of each class are selected as the testing sample set. Finally, we independently run 50 times on T images of each class. The curve in Figure 4 shows the average recognition rate (%) of various algorithms under different feature dimensions when six face images of each class are randomly selected as training sets in the Yale face database. Table 4 shows the standard deviation, the best

average recognition rate (%), and the corresponding feature dimensions of various algorithms in the Yale face database, respectively.

4.4. The Experiments in the AR Database. The AR face database [45] has 70 men and 56 women, a total of 126 people. The database consists of more than 4000 color people, including occluded front views, lighting conditions, and facial expressions. In AR database, there are 120 face images, including 65 male and 55 female. These images are divided into two parts (two weeks apart), each

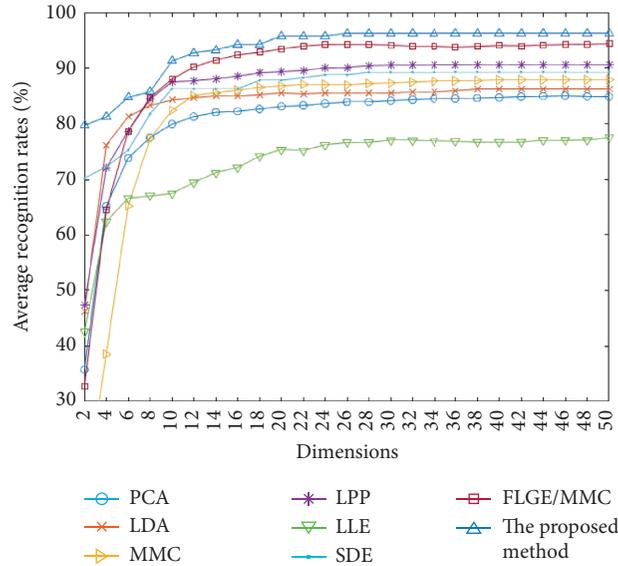


FIGURE 2: The average recognition rates (%) of various algorithms corresponding to various dimensions when four face images of each class are randomly selected as the training set in the ORL face database.



FIGURE 3: Eleven images of one class in the Yale face database.

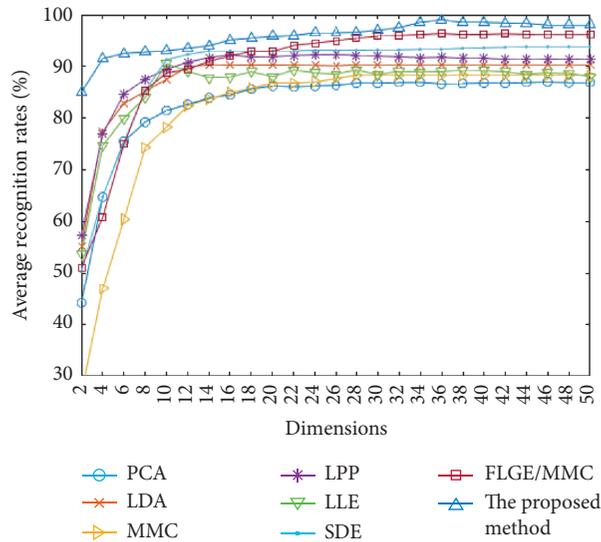


FIGURE 4: The average recognition rates (%) of various algorithms corresponding to various dimensions when six face images of each class are randomly selected as the training set in the Yale face database.

part containing 20 color face images. The face part image of each person is normalized to 50×40 pixels. And Figure 5 displays an example image of one class in the AR face database.

It also retains 95% of face image energy as the same as ORL and Yale face database. First of all, T images (T ranging from 2 to 6) from each class of the image database are

randomly selected as the training sample set. Secondly, the remaining $20 - T$ face images of each class are selected as the testing sample set. Finally, we independently run 10 times on T images of each class. Table 5 shows the standard deviation, the best average recognition rate (%), and the corresponding feature dimensions of various algorithms in the AR face database, respectively.

TABLE 4: The standard deviation, best average recognition rates (%), and corresponding feature dimension of various algorithms in the Yale face database.

Methods	T				
	2	3	4	5	6
PCA	77.29 ± 1.20 (38)	80.20 ± 1.27 (40)	83.99 ± 1.38 (38)	84.72 ± 1.24 (40)	86.21 ± 0.80 (46)
LDA	80.45 ± 1.48 (16)	84.55 ± 1.06 (16)	87.85 ± 0.45 (16)	87.20 ± 1.64 (14)	88.54 ± 0.82 (14)
MMC	80.58 ± 0.71 (12)	82.84 ± 0.88 (12)	85.87 ± 1.12 (12)	86.83 ± 0.37 (12)	87.79 ± 0.50 (14)
LPP	80.40 ± 1.80 (24)	85.79 ± 0.38 (22)	88.00 ± 0.50 (20)	89.57 ± 0.70 (18)	91.05 ± 1.23 (26)
LLE	83.55 ± 1.38 (14)	83.58 ± 0.59 (6)	85.90 ± 0.75 (6)	88.37 ± 1.63 (8)	88.97 ± 1.56 (8)
SDE	85.18 ± 0.70 (48)	86.48 ± 0.82 (46)	87.85 ± 1.20 (46)	90.16 ± 1.45 (48)	92.47 ± 1.32 (46)
FLGE/MMC	90.45 ± 0.75 (46)	91.48 ± 1.01 (46)	93.21 ± 1.24 (44)	95.34 ± 1.21 (42)	96.33 ± 0.21 (46)
The proposed method	91.50 ± 0.55 (36)	92.58 ± 0.17 (36)	95.53 ± 0.86 (38)	97.12 ± 0.37 (38)	98.75 ± 0.35 (36)



FIGURE 5: Twenty images of one person in the AR database. The first and second lines of face images were taken at two-week intervals.

TABLE 5: The standard deviation, best average recognition rates (%), and corresponding feature dimension of various algorithms in the AR face database.

Methods	T				
	2	3	4	5	6
PCA	66.68 ± 1.11 (145)	70.21 ± 1.62 (145)	77.60 ± 1.27 (150)	79.03 ± 0.81 (150)	81.54 ± 1.04 (145)
LDA	71.5 ± 0.71 (110)	75.58 ± 0.76 (115)	82.53 ± 0.81 (120)	87.12 ± 0.33 (115)	87.58 ± 0.75 (120)
MMC	69.39 ± 1.15 (120)	75.71 ± 0.59 (115)	82.98 ± 0.62 (115)	85.27 ± 0.94 (120)	87.86 ± 0.82 (120)
LPP	72.82 ± 0.81 (85)	78.78 ± 1.10 (85)	83.70 ± 0.65 (80)	88.75 ± 0.31 (85)	89.93 ± 0.48 (70)
LLE	70.40 ± 0.89 (125)	74.31 ± 1.16 (145)	83.35 ± 1.49 (140)	84.78 ± 0.76 (150)	86.98 ± 0.40 (145)
SDE	74.90 ± 0.08 (85)	80.10 ± 1.36 (75)	84.35 ± 1.36 (80)	89.80 ± 0.65 (85)	92.50 ± 1.18 (70)
FLGE/MMC	75.09 ± 1.49 (75)	82.42 ± 1.08 (70)	88.23 ± 0.8 (75)	90.17 ± 1.36 (75)	93.38 ± 1.20 (70)
The proposed method	79.31 ± 1.65 (80)	85.79 ± 0.75 (90)	90.86 ± 0.82 (95)	92.74 ± 0.95 (85)	95.82 ± 0.68 (90)

Figure 2 shows the curve when five face images in each class are randomly selected as the training set in the AR face database. The curve shows the average recognition rate (%) of each algorithm corresponding to different dimensions.

4.5. Overall Observations and Discussion. After analyzing the experimental results of the above four databases, the following conclusions can be drawn:

- (1) As can be seen from Table 1, the first six important characteristics (the second, fourth, fifth, seventh, tenth, and thirteenth characteristics) reflect the data distribution. The last characteristic factor plays a decisive role in the data distribution because of its wide range and large variance.
- (2) It can be seen from Tables 3–5 that compared with PCA, MMC, LDA, LLE, LPP, SDE, and FLGE/MMC,

the proposed algorithm obtains higher average recognition rate, which shows that the proposed algorithm can well control the data overlap (outlier) and sparse points. The results also show that the proposed algorithm is more discriminative than other algorithms because the membership degree of the fuzzy neighborhood helps to make the nearest neighbor samples closer to each other in the same class, while repel the far neighbor samples of different classes farther.

- (3) As shown in Figures 2, 4, and 6, under the same experimental conditions, the average recognition rate (%) of the proposed algorithm is better than that of other algorithms. It is further illustrated that the algorithm can obtain the optimal sparse projection discriminant feature because elastic network regression can obtain sparse feature extraction results.

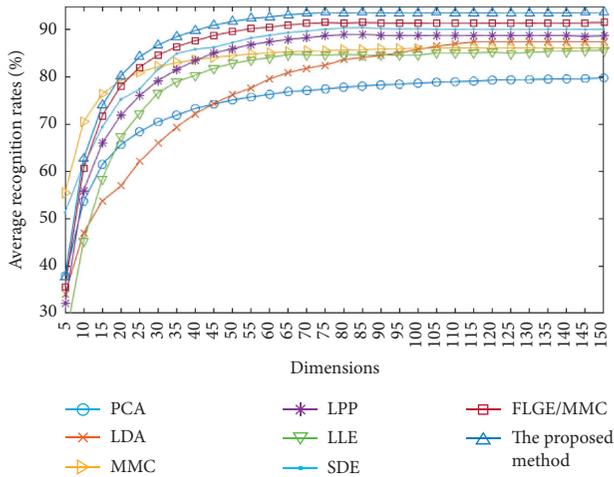


FIGURE 6: The average recognition rates (%) of various algorithms corresponding to various dimensions when five face images of each class are randomly selected as the training set in the AR face database.

- (4) Different from PCA, MMC, and LDA which attempt to preserve the global Euclidean structure, LPP, LLE, SDE, FLGE/MMC, and the proposed algorithm aim to discover the local geometric structure. However, sparse subspace learning methods are superior to the compared methods. The sparse projections learned by the proposed algorithm provide physical interpretation in real-world applications. Therefore, the discriminant knowledge is effectively discovered.

5. Conclusions

This paper proposes a new supervised learning algorithm for feature extraction and face recognition, namely, sparse graph embedding with fuzzy set for image classification. The experimental results show the proposed algorithm is more effective than other techniques. For future work, we will decide optimal parameters such as μ , K_c , and K_p and perform more tests on other types of face data. Specifically, for face recognition, the sparse face subspace shows us an intuitive, semantic, and insightful understanding of the feature extraction. In future work, we will extend the SDE algorithm to kernel and tensor forms via the kernel and tensor methods, respectively.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Science Foundation of China (Grant nos. 61876213, 6177227,

61861033, 61976118, U1831127, and 71972102), the Key R&D Program Science Foundation in Colleges and Universities of Jiangsu Province (Grant nos. 18KJA520005, 19KJA360001, and 20KJA520002), the Natural Science Fund of Jiangsu Province (Grant nos. BK20201397 and BK20171494), the National Key R&D Program (Grant nos. 2017YFC0804002 and 2019YF B1404602), the Natural Science Fund of Jiangxi Province (Grant no. 20202ACBL202007), the Natural Science Foundation of Guangdong Province (Grant no. 2016A030307050), and the Special Foundation of Public Research of Guangdong Province (Grant nos. 2016A020225008 and 2017A040405062).

References

- [1] Z. Fan, D. Zhang, X. Wang, Q. Zhu, and Y. Wang, "Virtual dictionary based kernel sparse representation for face recognition," *Pattern Recognition*, vol. 76, pp. 1–13, 2018.
- [2] S. Sun, N. Akhtar, H. Song, C. Zhang, J. Li, and A. Mian, "Benchmark data and method for real-time people counting in cluttered scenes using depth sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3599–3612, 2019.
- [3] G.-F. Lu, Q.-R. Yu, Y. Wang, and G. Tang, "Hyper-laplacian regularized multi-view subspace clustering with low-rank tensor constraint," *Neural Networks*, vol. 125, pp. 214–223, 2020.
- [4] M. Wan, Z. Lai, G. Yang, Z. Yang, F. Zhang, and H. Zheng, "Local graph embedding based on maximum margin criterion via fuzzy set," *Fuzzy Sets and Systems*, vol. 318, pp. 120–131, 2017.
- [5] Z. Zhang, L. Liu, Y. Luo et al., "Inductive structure consistent hashing via flexible semantic calibration," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2020.
- [6] Z. Zhang, Y. Zhang, G. Liu, J. Tang, S. Yan, and M. Wang, "Joint label prediction based semi-supervised adaptive concept factorization for robust data representation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 5, pp. 952–970, May 2020.
- [7] M. Belkin, P. Niyogi, T. G. Dietterich, S. Becker, and Z. Ghahramani, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *Advances in Neural Information Processing Systems*, vol. 14, pp. 873–878, 2000.
- [8] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [9] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [10] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [11] L. K. Saul and S. T. Roweis, "Think globally, fit locally: unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, pp. 119–155, 2003.
- [12] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM Journal on Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2004.
- [13] X. F. He, D. Cai, S. C. Yan, and H. J. Zhang, "Neighborhood preserving embedding," in *Proceedings of IEEE International*

- Conference on Computer Vision (ICCV)*, pp. 1208–1213, Beijing, China, October 2005.
- [14] X. He and P. Niyogi, "Locality preserving projections,," in *Proceedings of the 17th Annual Conference on Neural Information Processing Systems*, pp. 153–160, Vancouver and Whistler, Canada, December 2003.
 - [15] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.
 - [16] S. Yan, D. Xu, B. Zhang, and H.-J. Zhang, "Graph embedding: a general framework for dimensionality reduction," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, pp. 830–837, 2005.
 - [17] Z. Lai, Y. Chen, J. Wu, W. K. Wong, and F. Shen, "Jointly sparse hashing for image retrieval," *IEEE Transactions on Image Processing*, vol. 27, no. 12, pp. 6147–6158, 2018.
 - [18] S. Xu, R. Zhang, W. Cheng, and J. Xu, "MTLM: a multi-task learning model for travel time estimation," <http://link.springer.com/article/10.1007/s10707-020-00422-x>.
 - [19] G.-S. Xie, Z. Zhang, L. Liu et al., "SRSC: selective, robust, and supervised constrained feature representation for image classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 10, pp. 4290–4302, 2020.
 - [20] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
 - [21] J. C. Bezdek, J. Keller, and R. Krishnapuram, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Kluwer Academic Publishers, Dordrecht, Netherlands, 1999.
 - [22] K. C. Kwak and W. Pedrycz, "Face recognition using a fuzzy Fisher classifier," *Pattern Recognition*, vol. 38, no. 10, pp. 1717–1732, 2005.
 - [23] M. Wan, M. Li, Z. Lai, J. Yin, and Z. Jin, "Feature extraction based on fuzzy class mean embedding (FCME) with its application to face and palm biometrics," *Machine Vision and Applications*, vol. 23, no. 5, pp. 985–997, 2012.
 - [24] M. Wan, G. yang, Z. Lai, and Z. Jin, "Feature extraction based on fuzzy local discriminant embedding with applications to face recognition," *IET Computer Vision*, vol. 5, no. 5, pp. 301–308, 2011.
 - [25] Z. Lai, *Sparse Facial Feature Extraction via Manifold Learning*, Nanjing University of Technology and Engineering, Nanjing, Jiangsu, China, 2011.
 - [26] F. Xia, J. Wang, X. Kong, Z. Wang, J. Li, and C. Liu, "Exploring human mobility patterns in urban scenarios: a trajectory data perspective," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 142–149, 2018.
 - [27] Z. Zhang, F. Li, M. Zhao, L. Zhang, and S. Yan, "Robust neighborhood preserving projection by nuclear/l2, 1-norm regularization for image feature extraction," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1607–1622, April 2017.
 - [28] A. Wagner, J. Wright, A. Ganesh, fnm Zihan Zhou, H. Mobahi, and fnm Yi Ma, "Toward a practical face recognition system: robust alignment and illumination by sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 2, pp. 372–386, 2012.
 - [29] J. Yang, L. Zhang, Y. Xu, and J.-y. Yang, "Beyond sparsity: the role of L1-optimizer in pattern classification," *Pattern Recognition*, vol. 45, no. 3, pp. 1104–1118, 2012.
 - [30] Z. Zhang, S. Yan, and M. Zhao, "Pairwise sparsity preserving embedding for unsupervised subspace learning and classification," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4640–4651, 2013.
 - [31] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
 - [32] B. Efron, T. Hastie, and I. Johnstone, "Least angle regression," *Annals of Statistics*, vol. 32, no. 2, pp. 407–452, 2004.
 - [33] H. Zou and T. Hastie, *Regression Shrinkage and Selection via the Elastic Net, with Applications to Microarra*, University of Minnesota, Minneapolis, MN, USA, 2003.
 - [34] Z. Lai, D. Mo, J. Wen, L. Shen, and W. K. Wong, "Generalized robust regression for jointly sparse subspace learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 3, pp. 756–772, 2019.
 - [35] M. Wan, G. Yang, C. Sun, and M. Liu, "Sparse two-dimensional discriminant locality-preserving projection (S2DDLPP) for feature extraction," *Soft Computing*, vol. 23, no. 14, pp. 5511–5518, 2019.
 - [36] F. Shen, Y. Xu, L. Liu, Y. Yang, Z. Huang, and H. T. Shen, "Unsupervised deep hashing with similarity-adaptive and discrete optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 3034–3044, 2018.
 - [37] M. Wan and Z. Lai, "Feature extraction via sparse difference embedding (SDE)," *Ksii Transactions on Internet & Information System*, vol. 11, no. 7, pp. 3594–3607, 2017.
 - [38] Z. Zhang, F. Li, M. Zhao, L. Zhang, and S. Yan, "Joint low-rank and sparse principal feature coding for enhanced robust representation and visual classification," *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2429–2443, June 2016.
 - [39] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 1157–1165, 2006.
 - [40] D. Cai, X. He, and J. Han, "Spectral regression: a unified approach for sparse subspace learning," in *Proceedings of the Seventh IEEE International Conference on Data Mining (ICDM 2007)*, Omaha, NE, USA, October 2007.
 - [41] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
 - [42] D. L. Swets and J. J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831–836, 1996.
 - [43] AT&T Laboratories Cambridge, "The ORL database of faces,."
 - [44] "Yale database," 1997, <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>>.
 - [45] A. Martinez and R. Benavente, "The AR face database," *CVC Technical Report*, vol. 24, 1998.