

Article

Weighted Neighborhood Preserving Ensemble Embedding

Sumet Mehta ^{1,2} , Bi-Sheng Zhan ¹ and Xiang-Jun Shen ^{1,*}

¹ School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China; msumet@outlook.com (S.M.); balkela@163.com (B.-S.Z.)

² Department of Electronics and Communication Engineering, JCDM College of Engineering, Haryana 125055, India

* Correspondence: xjshen@ujs.edu.cn; Tel.: +86-0511-8878-0371

Received: 28 December 2018; Accepted: 13 February 2019; Published: 16 February 2019



Abstract: Neighborhood preserving embedding (NPE) is a classical and very promising supervised dimensional reduction (DR) technique based on a linear graph, which preserves the local neighborhood relations of the data points. However, NPE uses the K nearest neighbor (KNN) criteria for constructing an adjacent graph which makes it more sensitive to neighborhood size. In this article, we propose a novel DR method called weighted neighborhood preserving ensemble embedding (WNPEE). Unlike NPE, the proposed WNPEE constructs an ensemble of adjacent graphs with the number of nearest neighbors varying. With this graph ensemble building, WNPEE can obtain the low-dimensional projections with optimal embedded graph pursuing in a joint optimization manner. WNPEE can be applied in many machine learning fields, such as object recognition, data classification, signal processing, text categorization, and various deep learning tasks. Extensive experiments on Olivetti Research Laboratory (ORL), Georgia Tech, Carnegie Mellon University-Pose and Illumination Images (CMU PIE) and Yale, four face databases demonstrate that WNPEE achieves a competitive and better recognition rate than NPE and other comparative DR methods. Additionally, the proposed WNPEE achieves much lower sensitivity to the neighborhood size parameter as compared to the traditional NPE method while preserving more of the local manifold structure of the high-dimensional data.

Keywords: dimensional reduction; graph construction; machine learning; neighborhood preserving embedding; face recognition

1. Introduction

In many machine learning applications, such as data classification [1,2], face recognition [3,4], signal processing [5,6], and text categorization [7,8], the input data is usually high-dimensional which makes the calculations too complex, as well as requiring more computational time. In order to boost the performance and computational efficiency, various dimensionality reduction (DR) techniques have been proposed to preprocess these high-dimensional data. The employment of DR techniques aims at extracting the optimal representative features and transforming the original high-dimensional data into a low-dimensional feature space using different algorithms. According to the general framework these DR methods can be grouped into linear and non-linear methods [9].

Principal component analysis (PCA) [10] and linear discriminant analysis (LDA) [11] are two well-known linear subspace learning DR approaches. Both PCA and LDA assumes that the sample lies on a linear embedded manifold. PCA preserves the global geometrical information whereas LDA tries to preserve the global discriminant information of the high-dimensional data points. As these both aforementioned linear methods are very simple and convenient to use, they still fail to exploit

the essential non-linear structure of the data. In order to overcome the issues in linear DR methods, many manifold learning algorithms have been introduced to discover the non-linear structure of the data. These include isometric feature mapping [12], local linear embedding (LLE) [13], and Laplacian eigenmaps (LE) [14]. However, these algorithms suffer from the out-of-sample problem [15].

To solve the out-of-sample problem in manifold learning algorithms, graph-based embedding methods have been presented to reveal the local manifold structure of the data [16]. Graph-based embedding models preserve the spatial similarities of neighborhoods of the original data points in the form of node pairs by defining each node of graph as a low-dimensional eigenvector using a graph similarity weight matrix. Finally, the optimized projections can be computed by solving the graph-based eigenvector decomposition. Locality preserving projection (LPP) [17] and neighborhood preserving embedding (NPE) [18] are famous graph-based embedding manifold learning DR techniques. Both methods preserve the intrinsic manifold structure of the data without particular conjecture of the data distribution.

Due to the simplicity and effectiveness, NPE has been investigated in many recent studies [19–22]. Although NPE has shown better results than other competitive DR algorithms it is still highly sensitive to the neighborhood size parameter [23,24] as it uses the K nearest neighbor (KNN) criteria for constructing an adjacency graph. In this paper, we propose a novel DR method called weighted neighborhood preserving ensemble embedding (WNPEE). With this approach, our method can preserve more satisfactory local neighborhood structure that illustrates the intrinsic manifold of original high-dimensional data. The key properties of the proposed WNPEE method are summarized below:

- Unlike NPE, the proposed WNPEE constructs an ensemble of adjacent graphs with the number of nearest neighbors varying. With this graph ensemble building, WNPEE can obtain the low-dimensional projections with optimal embedded graphs pursuing in a joint optimization way.
- As rationale behind proposed WNPEE algorithm is similar to NPE; therefore, NPE can be viewed as a special instance of the WNPEE for a fixed parameter of the neighborhood size.
- The WNPEE provides an efficient weighting scheme for the eigenspace which makes it much less sensitive to the parameter of neighborhood size and it can get more stable recognition rate as compared to NPE.

We arrange rest of the paper, subsequent to introduction in Section 1 in the following manner. In Section 2, we present a brief but comprehensive review of state-of-the-art DR methods. The preliminaries of both LPP and NPE are given in Section 3. In Section 4, we describe the proposed WNPEE method with suitable explanation. Experimental results with a comparative discussion of the proposed method along with four comparative DR methods are shown in Section 5. We draw our conclusion and the future directions of research in Section 6.

2. Related Work

Recently, a number of dimensionality reduction methods have been developed with different ideas and methodology. However the aim of these numerous methods is same, that is, to obtain low-dimensional data on which further machine learning tasks can be easily performed with improved computational complexity. The following are some related comparable linear DR techniques used in this article to show the superiority of the proposed WNPEE method.

Linear dimensionality reduction approaches are widely used in face recognition because of their relative simplicity, analytical manipulability, and the direct ability to deal with the testing data points. The most popular linear DR methods include PCA [10], LPP [16], LDA [12], and NPE [18]. In face recognition applications, they are well known as Eigenfaces [11], Fisherfaces [16], Laplacianfaces [25], and NPEfaces [18]. Eigenfaces, which is based on PCA, finds a set of mutually orthogonal basis vectors to characterize the high-dimensional data in a least-squares sense. Different from Eigenfaces, which does not appropriately use label information of the given input data, Fisherfaces spread on LDA to

search the information on which the data points belonging to different classes are expected as distant from each other as possible. However, it assumes that the data points of the same class must be close to each other. Therefore, Laplacianfaces was proposed to find an optimally linear approximation of the eigenfunctions using the Laplace Beltrami operator on the face manifold [25]. The representation map generated by LPP preserves better local information of the essential manifold structure with more discriminating power.

Additionally, to preserve the neighborhood relations of high-dimensional data an unsupervised locally linear embedding (LLE) was firstly introduced by Roweis and Saul [13]. It is based on two linear perceptions: first is that each data point and its neighbors lies on or close to a locally linear cover of a manifold, and second is that the local symmetrical classification in original data space is unchanged in the output data space. LLE is very effective for feature extraction of high-dimensional data, yet it is unable to generate a loading matrix necessary for construction of a prediction model. Therefore, to overcome this problem a linear approximation of the LLE algorithm called neighborhood preserving embedding (NPE) has been developed [18]. NPE possibly outperforms Eigenfaces and Fisherfaces to preserve more Euclidian structure in DR tasks. The following is a brief overview of these related linear dimensionality reduction algorithms.

Let us consider $P = \{p_i \in \mathbb{R}^D\}_{i=1}^N$ a high D -dimensional data with N number of samples. Linear DR mapping is to assign the original high-dimensional data to a low d -dimensional data i.e., $Q = \{q_i \in \mathbb{R}^d\}_{i=1}^N$ ($d \ll D$). If A is a transformation matrix, then linear DR is defined as:

$$q_i = A^T p_i \quad \text{for } i = 1, \dots, N. \quad (1)$$

PCA idea is to find a projection direction that will give maximum variance, i.e., the reconstruction error can be minimal. The objective function of PCA is defined as follows:

$$A_{opt} = \arg \max_A \sum_{i=1}^N (q_i - \bar{q})^2, \quad (2)$$

$$A_{opt} = \arg \max_A A^T C A.$$

where $\bar{q} = \frac{1}{N} \sum q_i$ and C is the data covariance matrix. Finally the eigenvalues and orthogonal set of eigenvectors are computed for this covariance matrix. However, PCA impairs class discrimination as it uses directions which are only efficient for class representation. Thus, LDA, a linear supervised DR method, is a better choice for extracting discriminant features. Let the input data P belong to l classes, then LDA is optimized by the followig objective function:

$$A_{opt} = \arg \max_A \frac{A^T S_B A}{A^T S_W A}, \quad (3)$$

$$S_B = \sum_{i=1}^l N_i (m^{(i)} - m)(m^{(i)} - m)^T, \quad (4)$$

$$S_W = \sum_{i=1}^l \left(\sum_{j=1}^{N_i} (p_j^{(i)} - m^{(i)})(p_j^{(i)} - m^{(i)})^T \right) \quad (5)$$

where m is the total sample mean vector, N_i is the number of samples in the i th class, $m^{(i)}$ is the average vector of the i th class, and $p_j^{(i)}$ is the j th sample in the i th class. Additionally, S_B is the between-class scatter matrix and S_W is the within-class scatter matrix.

3. Preliminaries

In this section we briefly describe the functioning of LPP and NPE DR algorithms.

3.1. Locality Preserving Projections

LPP is an unsupervised graph-based embedding method [17]. LPP is designed with different objective principle from classical linear DR techniques. Due to locality preserving quality of LPP, it has been used in many information retrieval applications [26–28]. LPP can be simply functional everywhere unlike non-linear DR techniques. LPP could be carry out in the inventive space or in the reproducing kernel Hilbert space into which data points are mapped. The functioning of the LPP is defined as follows:

- *Constructing an adjacent graph:* using ϵ -neighborhood or k -neighborhood. In the ϵ -neighborhood: nodes i and j are connected by an edge if $\|p_i - p_j\|^2 < \epsilon$. In the k -neighborhood: nodes i and j are connected by an edge if i is in k nearest neighbors of j or j is in k nearest neighbors of i .
- *Choosing the weights:* Weights W_{ij} are given to edges between nodes i and j by a symmetric weight matrix W , computed either by uniform weight or Gaussian weight of Euclidean distance. W is often defined as:

$$W_{ij} = \begin{cases} \frac{\exp\|p_i - p_j\|^2}{t}, & p_j \in \text{KNN of } p_i \text{ or } p_i \in \text{KNN of } p_j, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

where parameter t is a positive constant.

- *Eigenmaps:* by computing eigenvectors and eigenvalues for the below eigenvector problem:

$$PLP^T \mathbf{a} = \lambda PDP^T \mathbf{a}. \quad (7)$$

where D is a diagonal matrix whose entities are column (or row) sum of W , $D_{ii} = \sum_j W_{ij}$, $L = D - W$ is the Laplacian matrix and \mathbf{a} is a transformation vector.

3.2. Neighborhood Preserving Embedding

The neighborhood preserving embedding (NPE) method is identical to LPP as both are targeted to discover the local structure of the original data manifold, yet their objective functions are totally different [19]. Linearity makes NPE a fast and suitable technique for practical applications [29–36]. NPE can be performed in either supervised or unsupervised mode. When the class information is available, it can be utilized to build a better weight matrix. It models the manifold structure directly by constructing the nearest-neighbor (NN) graph that reveals neighborhood relations of data points, and then retains the graph structure in the projection. The algorithmic procedure of NPE is defined as follows:

- *Constructing an adjacency graph:* using k -neighborhood or ϵ -neighborhood. In the k -neighborhood: a *directed* edge from nodes i and j if p_j is in k nearest neighbors of p_i . In the ϵ -neighborhood: put an edge between nodes i and j if $\|p_i - p_j\|^2 \leq \epsilon$.
- *Computing the weights:* Weights W_{ij} are given to edges between nodes i and j by a symmetric weight matrix W , and weight 0 if there is no such edge. The weight is calculated by minimizing following objective function:

$$\begin{aligned} \min \sum_i \|p_i - \sum_j W_{ij} p_j\|^2, \\ \text{s.t. } \sum_j W_{ij} = 1, j = 1, 2, \dots, N. \end{aligned} \quad (8)$$

- *Computing the projections:* final projections are obtained by minimizing the following eigenvector problem:

$$PMP^T \mathbf{a} = \lambda PDP^T \mathbf{a}. \quad (9)$$

where $M = (I - W)^T(I - W)$, I is an identity matrix and D is a diagonal matrix as in Equation (7). The optimal mapping transformation matrix A is embedded by the optimal mapping transformation vectors $(\mathbf{a}_0, \mathbf{a}_1, \dots, \mathbf{a}_{d-1})$, which are arranged in the order of their corresponding eigenvalues as $\lambda_0 \leq \dots \leq \lambda_{d-1}$.

4. The Proposed WNPEE Method

The aforementioned NPE is based on graph embedding and the neighbor relationship is measured by an artificially-constructed adjacent graph. Usually the KNN [2] criteria is used to construct an adjacent graph which makes the performance of the NPE very sensitive to the parameter of the neighborhood size. Thus, to overcome the neighborhood size sensitivity in NPE, we are motivated to propose a novel DR technique called weighted neighborhood preserving ensemble embedding (WNPEE). The proposed WNPEE is designed to capture the locality of the original data in the phase of manifold learning using an ensemble of adjacent graphs with varying the number of nearest neighbors. This graph ensemble building helps in obtaining the optimal embedded graph with low-dimensional projections pursued in a joint optimization way. Simultaneously, WNPEE makes the projections less sensitive to the parameter of the neighborhood size.

Let us assume a high-dimensional data set $X = \{x_i \in \mathbb{R}^D\}_{i=1}^N$, where x_i are data samples, D represents the dimensions, and N is the total number of samples. WNPEE aims to reduce the original D -dimensional data into reduced d -dimensional data ($d \ll D$) by constructing an optimal transformation matrix A . Let $Y = \{y_i \in \mathbb{R}^d\}_{i=1}^N$ represent the reduced d -dimensional data such that A maps x_i to y_i or we can say y_i represents x_i , where $y_i = A^T x_i$ in which locality of x_i is preserved by linear dimensionality reduction. In our proposed WNPEE method, the low-dimensional projections are carried out through the following steps:

Step 1: Firstly, it constructs an ensemble of adjacent graphs unlike NPE, i.e., constructing only one graph. Let G_1, G_2, \dots, G_L are adjacent graphs computed by varying the neighborhood size parameter ranges from 1 to L . These graphs can be formed either by K nearest neighbors (KNN) or ϵ -neighborhood. In proposed WNPEE, we assume the KNN criteria for constructing an ensemble of adjacent graphs. Each adjacent graph having N nodes, where the i -th node is corresponds to data point x_i . Using the KNN method, a directed edge is placed from node i to node j if x_j is among K nearest neighbors of x_i . Simultaneously weights are given to multiple adjacent graphs to ensemble them, by using parameter α_k where k is the neighborhood size parameter.

Step 2: Secondly, we compute the weights on the edges with fixed parameter α_k where the neighborhood size $k = 1$ to L . By using a fixed α_k we assume that each adjacent graph has an equal weight, i.e., if $L = 10$ then each adjacent graph have weight $1/10$ while computing the weights on the edges. Let W_k be an $N \times N$ weight matrix for every k adjacent graph, with $W_{k,ij}$ is the weight of the edge from node i to j , and 0 if there is no such edge. Minimization of the below objective function is used for weighting the edges:

$$\begin{aligned} \min \sum_i ||x_i - \sum_j \sum_k \alpha_k W_{k,ij} x_j||^2, \\ \text{s.t } \sum_j W_{k,ij} = 1, j = 1, 2, \dots, N. \end{aligned} \tag{10}$$

Step 3: Then the low-dimensional projections are obtained by solving the below generalized eigenvalue problem:

$$XMX^T \mathbf{a} = \lambda XX^T \mathbf{a}. \tag{11}$$

where X is high-dimensional data, $M = (I - \sum \alpha_k W_k)^T(I - \sum \alpha_k W_k)$, and $I = \text{diag}(1, \dots, 1)$ is a diagonal matrix. By solving Equation (11), the low-dimensional projection vectors $\mathbf{a}_0, \mathbf{a}_1, \dots, \mathbf{a}_{d-1}$

are obtained with their corresponding eigenvalues $\lambda_0 \leq \dots \leq \lambda_{d-1}$. Thus, WNPEE embedding is as follows:

$$x_i \rightarrow y_i = A^T x_i, A = (\mathbf{a}_0, \mathbf{a}_1, \dots, \mathbf{a}_{d-1}). \tag{12}$$

where y_i is a d -dimension vector and A is a $D \times d$ matrix.

Step 4: Finally, we compute the weight coefficients $\alpha_1, \alpha_2, \dots, \alpha_L$ according to Equation (26), on G_1, G_2, \dots, G_L adjacent graphs while taking the projection vector \mathbf{a} fixed obtained from Equation (11). With these updated weights on ensemble adjacent graphs, the weighted neighborhood preserving ensemble embedding projection vector \mathbf{a} as well as α_k are iteratively computed until the loss converges. The whole WNPEE procedure is summarized in Algorithm 1.

Algorithm 1. The weighted neighborhood preserving ensemble embedding algorithm.

Input: High-dimensional data X

Output: projection vector \mathbf{a}

Parameter: α_k

Initialize:

- Construct an ensemble of adjacent graphs using KNN with α_k weights

while loss not converged **do**

1. Computing the weights on edges with fix α_k acc. to (10)
2. Obtain projection vector \mathbf{a} acc. to (11)
3. Fix \mathbf{a} and update α_k acc.to (26)
4. Compute current loss

End while

Obtaining Parameter α_k in the Proposed WNPEE

In this paper we proposed an approach to make NPE less sensitive to neighborhood size k while preserving the local manifold structure of the input data during dimensionality reduction. To do so we introduce a new ensemble of adjacent graphs framework to NPE manifold learning technique. We therefore discuss how to obtain the value for the parameter α_k , which is the weight coefficient for the ensemble adjacent graphs which satisfies $\alpha_k \geq 0$ and $\sum_{k=1}^L \alpha_k = 1$.

In WNPEE each node of the adjacent graph can be represented by a linear combination of its neighbors with the weight $W_{k,ij}$ on the edge. Thus, the local spatial distribution of these patches can be characterized by linear coefficients that reconstruct each data point from its neighbors. With proper constraints and fixed $W_{k,ij}$ weights [18], the reconstruction errors are measured by the following cost function:

$$\Phi(y) = \sum_i \left(y_i - \sum_j \sum_k \alpha_k W_{k,ij} y_j \right)^2. \tag{13}$$

Although the projections are linear, i.e., $y^T = a^T X$, where the i -th column vector of X is x_i it can be written as:

$$z_i = y_i - \sum_j \sum_k \alpha_k W_{k,ij} y_j, \tag{14}$$

or in vector form:

$$z = y - \sum \alpha_k W_k y, \tag{15}$$

$$z = \left(I - \sum \alpha_k W_k \right) y. \tag{16}$$

where I is a identity matrix. Further the cost function can be reduced to:

$$\Phi(y) = \sum_i (z_i)^2, \tag{17}$$

$$\Phi(y) = z^T z, \tag{18}$$

$$\Phi(y) = y^T \left(I - \sum \alpha_k W_k \right)^T \left(I - \sum \alpha_k W_k \right) y, \tag{19}$$

$$\Phi(y) = \mathbf{a}^T X \left(I - \sum \alpha_k W_k \right)^T \left(I - \sum \alpha_k W_k \right) X^T \mathbf{a},$$

$$s.t. \sum_{k=1}^L \alpha_k = 1. \tag{20}$$

However from Equation (20), we can see that $\Phi(y)$ is linear with respect to α_k . It is also noted that the optimal solution of the linear programming will always be the extreme points, that is either $\alpha_k = 0$ or $\alpha_k = 1$, which means there will only be one adjacent graph selected contrary to our objective of exploring the rich complementation of multiple adjacent graphs. Thus, to overcome this problem, we make a relaxation by considering α_k to be α_k^r (r represents a control parameter for the weights of the multiple adjacent graphs). Thus, while $r > 1$ the outcome is a balance of multiple adjacent graphs [37,38]. Thus, by adding a Lagrangian multiplier, we obtain the formulation of WNPEE as:

$$\mathcal{L}(y, \lambda) = \mathbf{a}^T X \left(I - \sum \alpha_k^r W_k \right)^T \left(I - \sum \alpha_k^r W_k \right) X^T \mathbf{a} - \lambda \left(\sum \alpha_k^r - 1 \right). \tag{21}$$

Let $B = \left(I - \sum \alpha_k^r W_k \right)$, and taking partial derivative by using Leibniz’s notation as follows:

$$\frac{\partial \mathcal{L}(y, \lambda)}{\partial (\alpha_k)} = \frac{\partial (\Phi(y))}{\partial (B)} \frac{\partial (B)}{\partial (\alpha_k)}, \tag{22}$$

$$\frac{\partial \mathcal{L}(y, \lambda)}{\partial (B)} = 2 \operatorname{tr} \left[\left(I - \sum \alpha_k^r W_k \right) X^T \mathbf{a} \mathbf{a}^T X \right], \tag{23}$$

$$\frac{\partial (B)}{\partial (\alpha_k)} = -\operatorname{tr} (W_k) r \alpha_k^{r-1}, \tag{24}$$

Therefore:

$$\frac{\partial \mathcal{L}(y, \lambda)}{\partial (\alpha_k)} = r \alpha_k^{r-1} \operatorname{tr} \left\{ \left[\left(I - \sum \alpha_k^r W_k \right) X^T \mathbf{a} \mathbf{a}^T X \right]^T W_k \right\} - \lambda, \tag{25}$$

Thus, when projections are fixed, and considering the constraint $\sum_{k=1}^L \alpha_k = 1$, we can further derive that:

$$\alpha_k = \frac{r^{-1} \sqrt{\frac{1}{\operatorname{tr}(\sum_k)}}}{\sum_{j=1}^L r^{-1} \sqrt{\frac{1}{\operatorname{tr}(\sum_j)}}}. \tag{26}$$

where $\sum_k = \left[\left(I - \sum \alpha_k^r W_k \right) X^T \mathbf{a} \mathbf{a}^T X \right]^T W_k$. Therefore, by computing the value of α_k we can iteratively obtain the weights on ensemble of adjacent graphs. Furthermore using this graph ensemble building, our proposed WNPEE DR algorithm can obtain the low-dimensional projections with optimal embedded graph pursuing in a joint optimization way. As from [18] it reveals that NPE essentially tries to obtain the linear approximations to the eigenfunctions of the iterated Laplacian Beltrami operator. Therefore, WNPEE also provides a way to linearly approximate the eigenfunctions. In the next section, extensive experiments are carried out on a variety of face databases to verify performances in both recognition and lower sensitivity to the neighborhood size parameter with other state-of-the-art DR methods.

5. Experimental Results and Analysis

In this section, we have conducted experiments on four publicly available high-dimensional face databases ORL, Georgia Tech (GT), CMU PIE and Yale respectively. To demonstrate the effectiveness of our proposed WNPEE, results are compared with other competitive DR algorithms including Eigenfaces, Fisherfaces, LPP, and NPE. The performance is evaluated by considering the recognition rates corresponding to dimensions. Furthermore, to show the superiority of the proposed WNPEE method, its sensitivity to neighborhood parameter k is computed for different databases. The environment for the experiments is MATLAB (R2014a) (version 8.3.0.532, MathWorks, Natick, MA, USA) on an Intel® Core™ i5-6500 CPU @3.2 GHz, DELL machine with 4.0 GB RAM. In all experiments in this section, k nearest neighbor classifier is employed to construct the adjacent graphs. The value of the parameter of neighborhood size is pre-set to 15 with a step size of one, i.e., $k = 1, 3, \dots, 15$. The weight control parameter r is taken as 6 [37]. To overcome the small sample size problem, the PCA preprocessing is applied in the LPP method by keeping nearly 98% image energy. The eigenvector dimension taken in all experiments is fixed to 50 to compare the recognition results.

5.1. Experiments on the ORL Face Database

The ORL (Olivetti Research Laboratory) database [39] contains 400 images of 40 persons (10 images per person). To be subjected, the images were taken at different times, varying the light, facial expressions (open or closed eyes, smiling or not smiling), and facial details (glasses or no glasses). The size of each image is 112×92 pixels, with 256 gray levels per pixel. In our experiments, the images were normalized and cropped to a resolution of 32×32 pixels. Figure 1 shows some sample images of one person from the ORL face database.

The average recognition rates (%) of five different methods used with their correspondent standard deviation and dimension of the best subspace are shown in Table 1. We randomly select $l = 4, 5, 6, 7$ images for each individual for training and rest to be used as testing samples. For effective performance, experiments are repeated 10 times for each value of l . It can be easily seen from the table that the proposed WNPEE method achieves the best recognition results for value of $l = 4, 6, 7$ and NPE has the best recognition when $l = 5$. The best average recognition rates for Eigenfaces, Fisherfaces and LPP are, respectively, lower than NPE and WNPEE.



Figure 1. Sample face images of the ORL face database.

Table 1. The average recognition rates of each method on four face databases with the corresponding STDs and dimension of best subspace.

Database	No. of Trains	Eigenface	Fisherface	LPP	NPE	WNPEE
ORL	$l = 4$	$34.02 \pm 1.91(41)$	$66.69 \pm 1.21(41)$	$70.20 \pm 0.94(43)$	$82.05 \pm 1.42(37)$	$84.80 \pm 1.22(48)$
	$l = 5$	$21.61 \pm 0.91(43)$	$64.62 \pm 1.75(47)$	$74.28 \pm 1.46(41)$	$86.01 \pm 1.41(27)$	$82.56 \pm 1.74(37)$
	$l = 6$	$18.35 \pm 0.73(50)$	$69.16 \pm 1.83(34)$	$78.76 \pm 1.57(24)$	$86.68 \pm 1.42(43)$	$89.37 \pm 1.79(23)$
	$l = 7$	$16.18 \pm 0.63(50)$	$71.36 \pm 1.92(39)$	$80.05 \pm 1.41(17)$	$86.05 \pm 1.31(22)$	$86.99 \pm 1.78(14)$
GT	$l = 3$	$10.24 \pm 0.71(50)$	$30.40 \pm 0.72(33)$	$34.11 \pm 0.68(49)$	$46.28 \pm 0.92(24)$	$47.85 \pm 1.94(19)$
	$l = 5$	$7.82 \pm 0.46(50)$	$39.51 \pm 0.94(37)$	$49.37 \pm 1.53(50)$	$56.65 \pm 1.23(49)$	$63.84 \pm 1.42(22)$
	$l = 7$	$7.50 \pm 0.35(48)$	$42.01 \pm 0.94(45)$	$56 \pm 1.34(32)$	$59.59 \pm 1.22(39)$	$69.83 \pm 1.48(17)$
	$l = 9$	$9.22 \pm 0.83(50)$	$50.40 \pm 1.13(41)$	$59.33 \pm 1.33(23)$	$63.46 \pm 1.33(38)$	$73.26 \pm 1.51(20)$
CMU PIE	$l = 10$	$70.88 \pm 2.43(50)$	$57.40 \pm 2.91(50)$	$64.05 \pm 1.76(50)$	$62.48 \pm 1.84(50)$	$74.69 \pm 1.83(47)$
	$l = 15$	$73.24 \pm 2.72(48)$	$77.23 \pm 1.22(49)$	$76.26 \pm 2.31(49)$	$74.54 \pm 2.17(47)$	$84.35 \pm 1.91(47)$
	$l = 20$	$73.52 \pm 2.65(47)$	$74.50 \pm 2.15(48)$	$82.16 \pm 2.14(37)$	$79.77 \pm 2.28(49)$	$88.13 \pm 1.95(31)$
	$l = 25$	$76.25 \pm 2.61(33)$	$75.67 \pm 2.85(39)$	$80.17 \pm 2.13(33)$	$79.10 \pm 2.25(47)$	$88.92 \pm 1.93(29)$
YALE	$l = 6$	$29.57 \pm 1.42(47)$	$43.86 \pm 0.98(14)$	$48.90 \pm 0.89(15)$	$49.22 \pm 1.28(24)$	$55.65 \pm 0.90(27)$
	$l = 7$	$25.86 \pm 1.37(39)$	$39.86 \pm 0.73(24)$	$48.83 \pm 0.91(41)$	$51.43 \pm 0.9(42)$	$59.33 \pm 0.89(25)$
	$l = 8$	$23.51 \pm 1.27(49)$	$44.97 \pm 1.19(24)$	$49.95 \pm 0.81(35)$	$55.77 \pm 1.14(25)$	$69.68 \pm 1.13(9)$
	$l = 9$	$32.53 \pm 1.32(47)$	$47.33 \pm 0.90(42)$	$53.60 \pm 0.61(48)$	$58 \pm 1.13(13)$	$71.80 \pm 1.13(20)$

Figure 2 shows the recognition rate versus reduced dimensions of different DR methods used in experiments with our proposed WNPEE algorithm. Figure 2a–d for different values of l . From the figures we conclude that Eigenfaces have lowest and proposed WNPEE method have the highest recognition rate. Almost all DR algorithms starts from their lowest value of recognition and achieves the peak value when number of dimensions is 25 and after that almost stable.

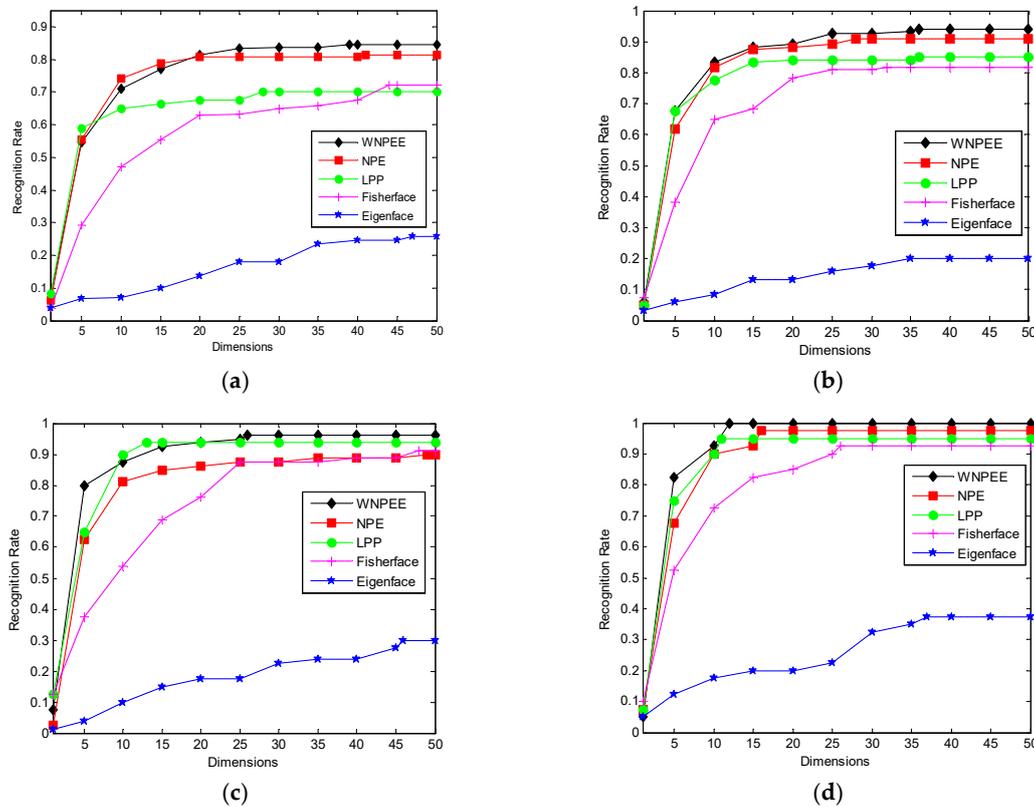


Figure 2. Average recognition rate versus reduce dimensions on ORL dataset. (a) $l = 4$, (b) $l = 5$, (c) $l = 6$, and (d) $l = 7$.

We list the computational time (in seconds) of each method for one run with different values of training samples l , shown in Table 2. As can be seen from the time results that the time taken by proposed method is slightly more than the other methods and the difference is not much. This is because of one more loop added in our algorithm. However, the performance of the WNPEE is superior to the other algorithms.

Table 2. Computation time on ORL face database under different trains for one run (time in seconds).

No. of Trains	Eigenface	Fisherface	LPP	NPE	WNPEE
$l = 3$	1.21	1.35	2.16	1.81	2.44
$l = 5$	2.08	2.87	2.98	2.08	3.16
$l = 7$	3.07	3.17	3.48	2.27	4.11
$l = 9$	2.80	2.98	3.74	2.41	4.27
Average	2.29	2.59	3.09	2.14	3.50

5.2. Experiments on the GT Face Database

Georgia Tech (GT) face database [40] contains 750 images of 50 people taken in two or three sessions between at the Center for Signal and Image Processing at Georgia Institute of Technology. All people in the database are represented by 15 color JPEG images with cluttered background. These images show frontal and/or tilted faces with different facial expressions, lighting conditions and scale.

In our experiments, all the images in the data set were manually cropped to size to 32×32 pixels and converted into gray scale images for training and testing. Figure 3 shows some sample images of GT database.



Figure 3. Sample face images of GT face database.

The experimental results in form of average recognition rates (%) of each DR methods with their corresponding standard deviation and best subspace dimension are shown in Table 1. The training sets are chosen randomly with $l = 3, 5, 7, 9$ images of each individual and rest to be used as testing samples. For different values of l the experiments are performed for 10 repetitions for satisfactory results. The best results are highlighted in bold face. Results clearly shows that the proposed WNPEE has the best recognition rates among all five comparative DR methods. Additionally, WNPEE achieves its best performance with the increase in value of l .

The recognition rate corresponds to reduce dimensions for different DR algorithms are shown in Figure 4a–d for different values of l . It is observed from graphs that mostly all comparable methods achieves their peak values of recognition rates in between dimensions 15 to 20 and then becomes stable for higher dimensions. We can also deduce that WNPEE shows the best results whereas NPE overcomes the other three graph-based DR methods.

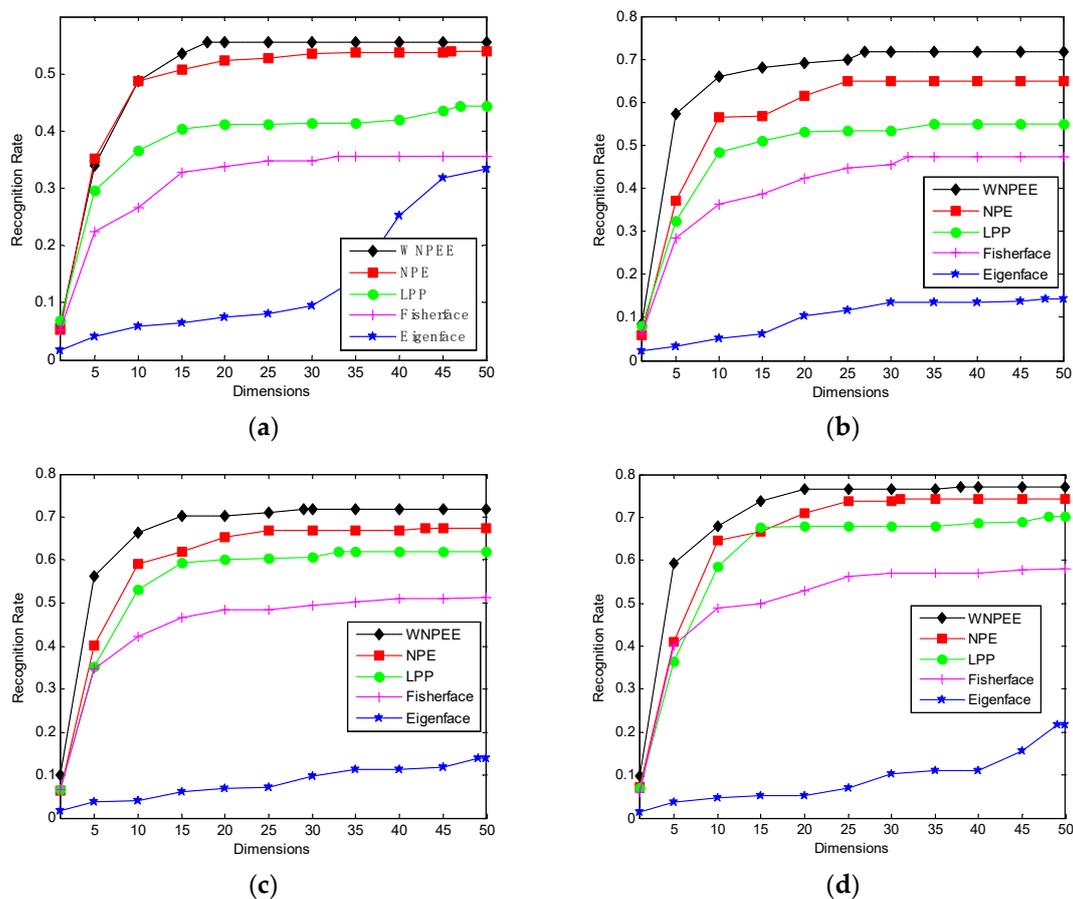


Figure 4. Average recognition rate versus reduce dimensions on GT dataset. (a) $l = 3$, (b) $l = 5$, (c) $l = 7$, and (d) $l = 9$.

According to the computational time results shown in Table 3, it is detected that the time consumed by WNPEE is different from other methods. On the other hand it shows that the time difference is not so large, it is in some seconds that make our proposed method suitable for many real time applications.

Table 3. Computation time on GT face database under different trains for one run (time in seconds).

No. of Trains	Eigenface	Fisherface	LPP	NPE	WNPEE
$l = 3$	4.31	4.28	4.31	4.25	4.63
$l = 5$	5.23	5.12	5.09	5.08	5.78
$l = 7$	5.72	5.53	5.58	5.54	6.23
$l = 9$	5.70	5.54	5.62	5.51	7.04
Average	5.24	5.12	5.15	5.09	5.92

5.3. Experiments on the CMU PIE Face Database

The CMU PIE face database [41] contains 41,368 images of 68 people, each person under 13 different poses, 43 different illumination conditions, and with four different expressions. Images are taken under 15 view points and 19 illumination conditions while displaying a range of facial expressions. All images are moved to gray scale and cropped to 32×32 pixels in our experiments. Some sample images are shown in Figure 5.



Figure 5. Sample face images of the CMU PIE face database.

Table 1 shows the average recognition rates (%) of five DR algorithms with their corresponding standard deviation and best subspace dimension, where the highest rates are in boldface. We choose the first $l = 10, 15, 20, 25$ images for the training and rest for the testing samples. Experiments are repeated ten times for best average recognition rates. From the results shown in Table 1 it is found that proposed WNPEE outperform other four comparative DR methods with different training sets. Furthermore, NPE achieves the second best results due to its superiority over other three methods.

Figure 6a–d illustrates the recognition rate against dimensions for each value of l . The graph results demonstrate that our proposed WNPEE outperforms all other comparative DR methods used in experiments. Although when the value of $l = 10$, NPE shows more superior recognition but, as the value of l increases, our proposed WNPEE outperforms all methods. It is also noted that with higher values of l overall performance of all DR methods increases.

To evaluate the computation complexity we record the time consumed for all DR techniques for one run which are shown in Table 4. The time is in seconds with different values of trains. Time results reveals that proposed method is almost equal in terms of computations. However, a slight difference in time makes our proposed WNPEE slow, but its performance makes it a competitive algorithm in the field of DR.

Table 4. Computation time on CMU PIE face database under different trains for one run (time in seconds).

No. of Trains	Eigenface	Fisherface	LPP	NPE	WNPEE
$l = 3$	1.04	2.14	2.57	2.08	2.61
$l = 5$	1.13	2.23	2.75	2.11	2.79
$l = 7$	1.12	2.16	2.78	2.13	2.75
$l = 9$	1.01	2.11	2.95	2.01	3.06
Average	1.07	2.16	2.76	2.08	2.80

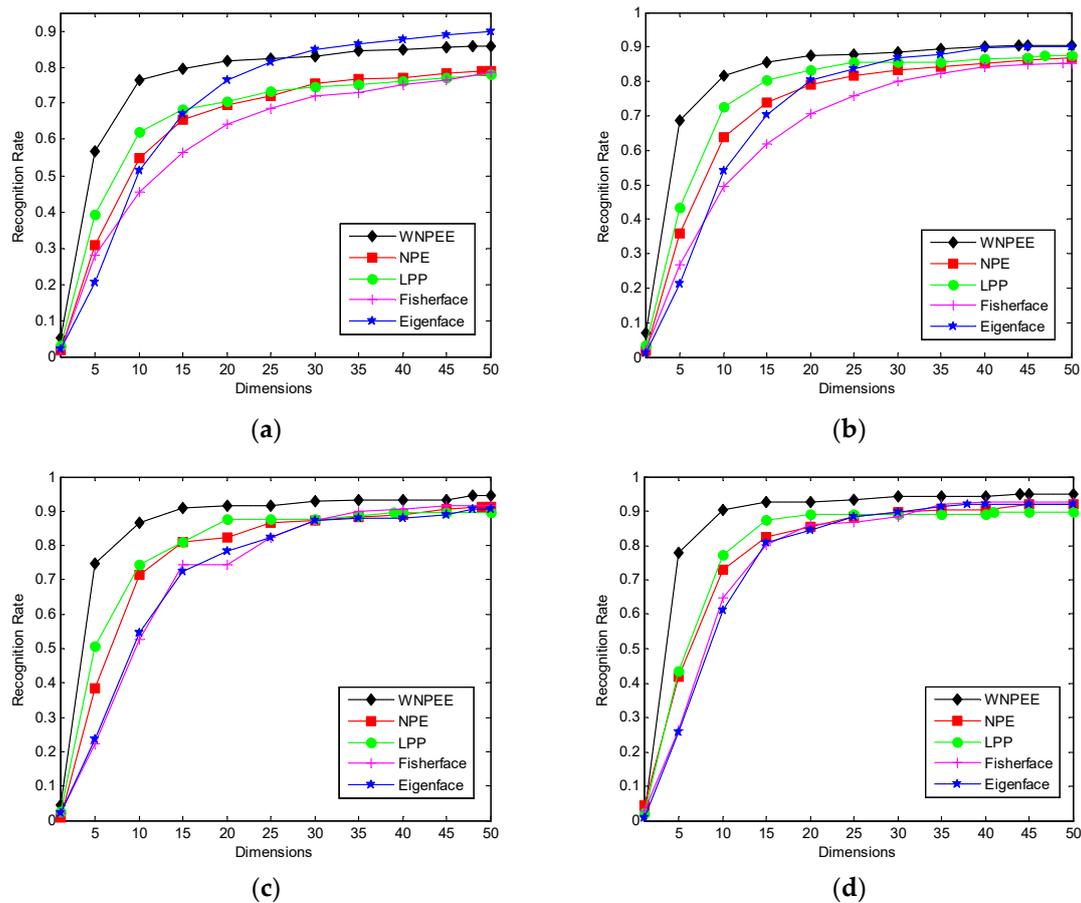


Figure 6. Average recognition rate versus reduce dimensions on CMU PIE dataset. (a) $l = 10$, (b) $l = 15$, (c) $l = 20$, and (d) $l = 25$.

5.4. Experiments on the YALE Face Database

The Yale database [42] was used to evaluate the system performance under conditions where the facial expressions and illumination are varied. It contains 165 gray scale images of 15 individuals under various facial expression (normal, happy, sad, sleepy, surprise, and wink), lighting conditions (left-light, center-light, right-light), and facial details (with glasses or without). The original size of the images is 243×320 pixels but in our experiments we have normalized each image to 32×32 pixels. Some sample images with different face expressions are shown in Figure 7.



Figure 7. Sample face images of the YALE face database.

In this experiment, we have evaluated the performance of the five methods: Eigenfaces, Fisherfaces, LPP, NPE, and WNPEE. The recognition rates with STDs and dimensions are recorded by taking their mean for 10 runs shown in Table 1. For better implementation of the algorithms different training sets for $l = 6, 7, 8, 9$ images of each individual are taken and rest are for testing sets. The result shows the superiority of the WNPEE and also it reveals that with the use of an optimal weighted graph, the performance can be increased for a DR algorithm.

The result graphs are in the form of recognition rate with respect to the dimensions as shown in Figure 8. For different values of l different results are recorded as shown in the figure. Graphs

demonstrate two interesting facts, i.e., with a greater amount of training, the greater the accuracy will be, and KNWPE outperforms the other four DR methods in terms of recognition rate. This makes our WNPEE more superior to use in many tasks, including face recognition and pattern analysis.

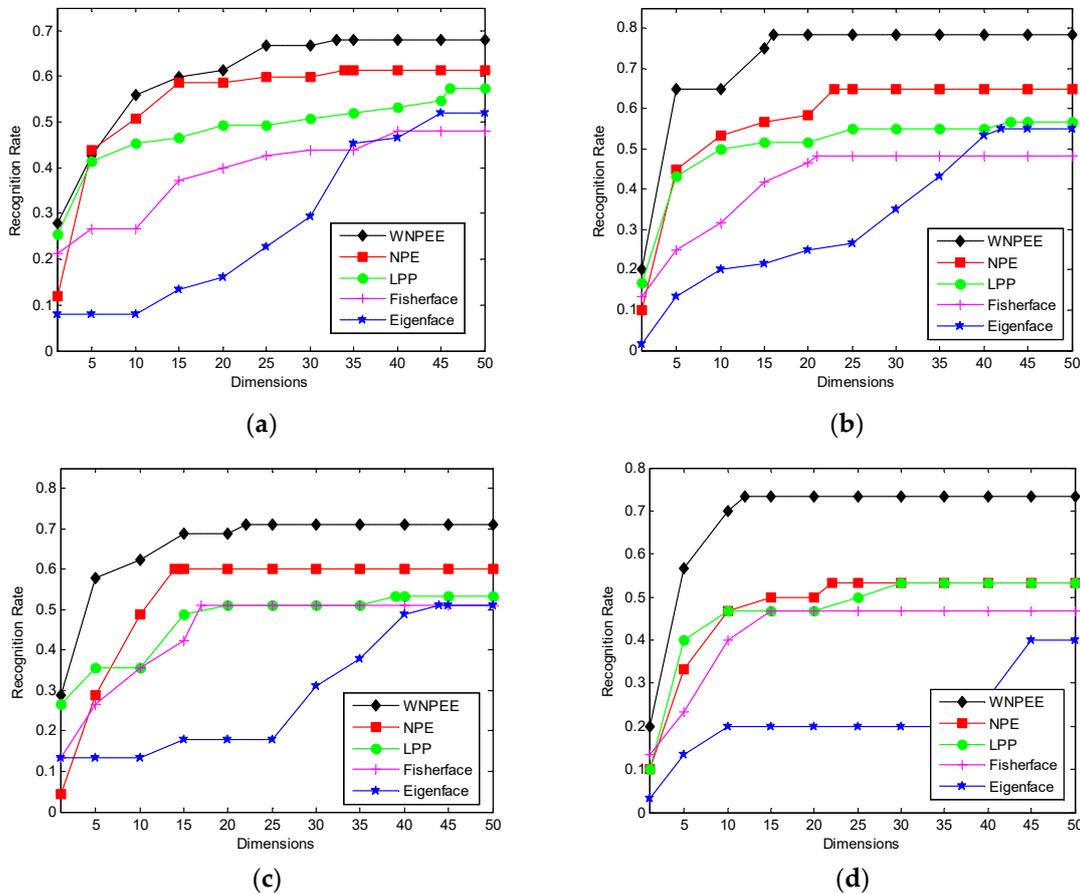


Figure 8. Average recognition rate versus reduce dimensions on YALE dataset. (a) $l = 6$, (b) $l = 7$, (c) $l = 8$, and (d) $l = 9$.

For any algorithm to be of practical use, it should have a lower computation time. As revealed from Table 5, the overall time for proposed WNPEE is slightly more than the other DR methods with a difference of 0.38 s. The slightly greater time is practically negligible if the performance is far better than the other methods. Thus, we can easily use this proposed WNPEE in practice.

Table 5. Computation time on the YALE face database under different training for one run (time in seconds).

No. of Trains	Eigenface	Fisherface	LPP	NPE	WNPEE
$l = 6$	0.91	1.9	2.19	2.93	3.03
$l = 7$	0.93	1.86	2.39	2.89	3.14
$l = 8$	1.14	1.96	2.94	3.07	3.59
$l = 9$	1.71	1.87	3.09	4.02	4.65
Average	1.17	1.89	2.65	3.22	3.60

5.5. Analyzing the Sensitivity to Neighborhood Size Parameter

As mentioned before the NPE is among the best graph-based DR methods, but still its performance is very sensitive to the neighborhood size parameter. Thus, to overcome this sensitivity problem in classical NPE we proposed a new WNPEE DR technique. To evaluate the WNPEE sensitivity to the

neighborhood size parameter we have conduct experiments on four face databases ORL, GT, CMU PIE, and Yale. The experimental results are observed in form of error rates with varying the parameter of neighborhood size. In these experiments, we vary the value of parameter k from 1 to 15 with a step size of one and plot the corresponding average error rates, as shown in Figure 9. Figure 8a–d are for ORL, GT, CMU PIE, and Yale databases, respectively.

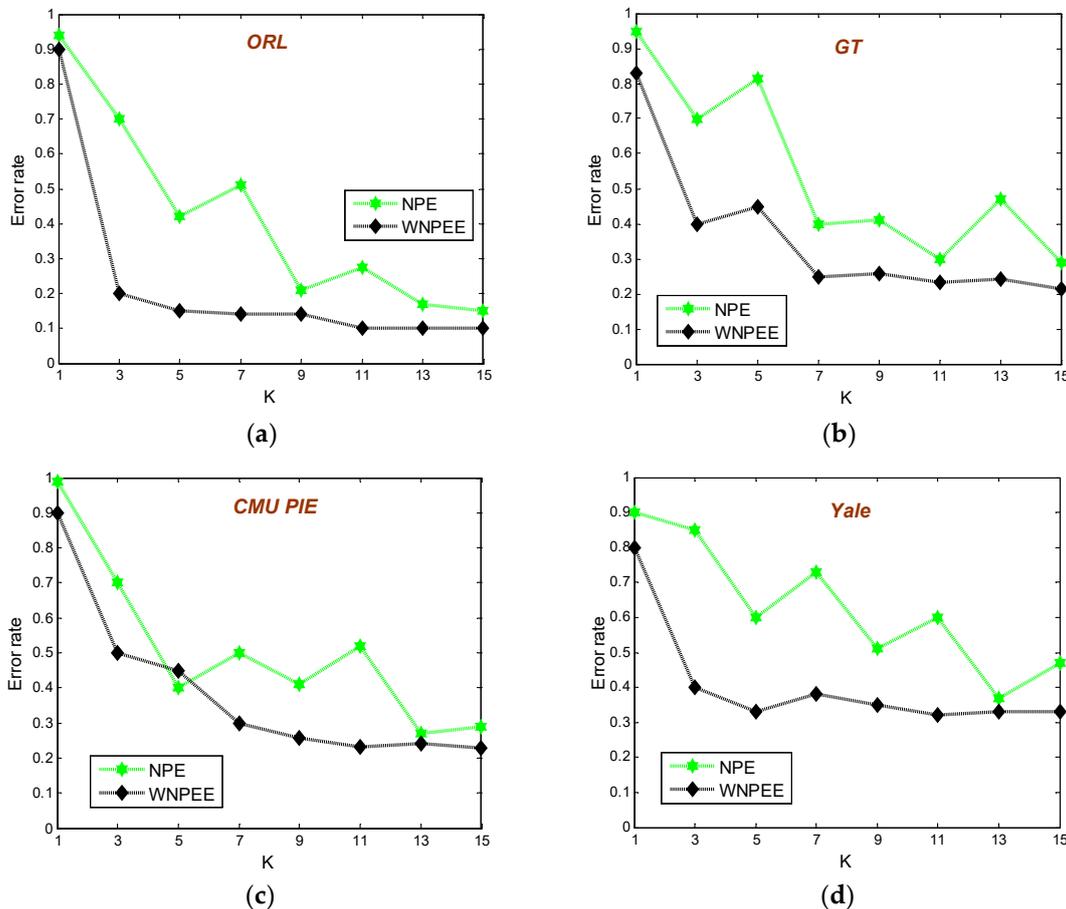


Figure 9. Error rates of NPE and proposed KWNPE corresponding to parameter of neighborhood size k on different databases (a) ORL with $l = 7$, (b) GT with $l = 9$, (c) CMU PIE with $l = 25$, and (d) Yale with $l = 9$, respectively.

For ORL database, each individual with $l = 7$ images are selected randomly for training and rest are used as testing. Figure 8a shows the average error rate versus parameter of neighborhood size on ORL database. From the figure we observe that the error rate for NPE is continuously changing with varying neighborhood size. Firstly it decreases fast when parameter changes from 1 to 3, then suddenly increases up to $k = 7$, and continue varying with parameter k . On the other hand, the average error rate of our proposed WNPEE shows a sharp down when parameter changes from 1 to 3 and then it is almost stable with contrast of neighborhood size. The above observation clearly shows that proposed WNPEE is very less sensitive to parameter of neighborhood size.

As compared to ORL, Georgia Tech face database is more complex because of increased number of expressions for various pose faces. In our experiment, we choose $l = 9$ for everyone to form training and balance were used as testing sets. The average error rates with varying neighborhood size parameter for 10 iterations are presented in Figure 8b. The results show that NPE error rates are different with different value of the neighborhood size parameter. Oppositely, the error rates of WNPEE changes sharply at the start, then up to parameter value 7 it changes little and, further, it

become stable. The reason behind this may be the GT database has a very complex bunch of data that needs more time to achieve a stable accuracy as compared to the ORL database.

In CMU PIE face database, a random subset of $l = 25$ training samples are selected and the rest were taken for the testing set. The average error rates of NPE and WNPEE versus the parameter of neighborhood size for 10 iterations is shown in Figure 8c. From the plot it is clear that both the error rates for NPE and WNPEE are changing with different values of parameter k , but WNPEE will become stable after $k = 7$. This shows that WNPEE is less sensitive to neighborhood size as compared to NPE.

The Yale database experiment results are shown in Figure 8d. In this experiment, each individual with $l = 9$ images are selected as the training set and the rest are taken for the testing set. In the same way, the error rates are used to measure the sensitivity to parameter of neighborhood size. As shown in plot 8d, the performance of the WNPEE is less sensitive to the neighborhood parameter than NPE.

Based on the above results on four face databases; we can realize two important facts. First is that NPE accuracy continuously varies with changes in the parameter of neighborhood size. Additionally, the accuracy of proposed WNPEE changes with respect to the neighborhood size parameter, but it becomes relatively stable after the parameter reaches 7 in mostly cases. Another fact is that the proposed WNPEE has attained greater accuracy than NPE in all four experiments. Therefore, an ensemble of adjacent graphs can be utilized to get improved face recognition performance of NPE which is pursued in a joint optimization way. From the above on-going discussions without any obstruction we can conclude that our proposed WNPEE shows a very significant improvement in terms of performance and is less sensitive to the neighborhood size parameter.

5.6. Computation Complexity

The most significant goal for dimension reduction is to help alleviate the problem of data complexities so as to improve the performance of algorithms. In this section, a detailed complexity analysis of the proposed WNPEE algorithm, and the computation cost compared with other approaches on ORL, Georgia Tech, CMU PIE, and Yale datasets are discussed.

The proposed WNPEE algorithm consists of two parts: one is for multiple graph construction and the other is for the iterative solution of the regularization framework. From Algorithm 1, computational convergence of WNPEE subjects to continual update of the weight coefficients α_k for multiple adjacent graphs. Suppose a training set X of dimensions $m \times n$, and k number of adjacent graphs for ensemble embedding, then it will take $O(kn^2)$ to compute α_k . Since our method is an eigenvalue problem, a complexity of $O(n^3)$ for obtaining the low-dimensional projection vector \mathbf{a} . Therefore, the total complexity of the proposed WNPEE method is $O(t(n^3 + kn^2))$, where t is the number of iterations for convergence.

The computation times for obtaining the low-dimensional projections on ORL, Georgia Tech, CMU PIE and Yale datasets are reported in Tables 2–5. It can be seen that the proposed method is slightly time consuming as compared to the rest of the methods. This is due to the utilization of multiple adjacent graphs in our method. However, the proposed WNPEE obtained superior recognition rates, as shown in Table 1. In conclusion, except for the slightly increased computational time consumption, the proposed WNPEE generally obtains better recognition rates and is less sensitive to neighborhood size k compared to the rest of the DR methods. Thus, there is no constraint to the practical application of the proposed method.

6. Conclusions

In this paper, we proposed a new graph-based DR algorithm, called weighted neighborhood preserving ensemble embedding (WNPEE). The proposed WNPEE method aims at improving the sensitivity of traditional NPE to parameter of neighborhood size, as well as to increase its face recognition performance as compared to other comparative methods. WNPEE significantly enhances the face recognition performance mainly from two key factors. The first improvement resides in the fact that in the WNPEE algorithm constructs an ensemble of adjacent graphs with varying

parameters of neighborhood size. With this graph ensemble building, WNPEE can obtain the low-dimensional projections with optimal embedded graph pursuing in a joint optimization way. This helps WNPEE to preserve more local information that best determines the essential manifold structure of high-dimensional data. The second improvement is that WNPEE provides an efficient weighting scheme for eigenspace which makes the projections much less sensitive to the neighborhood size parameter as compared to NPE. Four different DR approaches, Eigenfaces, Fisherfaces, LPP, and NPE, have been compared, to calculate the performance of the proposed WNPEE algorithm. Experimental results on four face databases ORL, GT, CMU PIE, and Yale demonstrate that the proposed WNPEE DR method achieves the highest recognition rates. Additionally, the proposed WNPEE is less sensitive to the neighborhood size parameter when compared to NPE.

Therefore, the proposed WNPEE method can be suitable for advanced driver assistance systems (ADAS) in various tasks, such as object recognition, data classification, obstacle detection, signal processing, text categorization, and various deep learning tasks. In future work we plan to extend ensemble of adjacent graphs framework to other graph embedding techniques.

Author Contributions: X.-J.S. and S.M. fabricated the algorithm; S.M. performed the experiments; B.-S.Z. and X.-J.S. analyzed the results and provide supervision; S.M. drafted the manuscript; and X.-J.S. and B.-S.Z. reviewed the paper.

Funding: This research was funded by National Natural Science Foundation of China, grant number 61572240.

Acknowledgments: The authors thank Abeo Timothy Apasiba, Arfan Ali Nagra, and Huang Chang Bin for their kind assistance, and also thanks to Nitish Mehta for motivation.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gou, J.; Yi, Z. Locality-Based Discriminant Neighborhood Embedding. *Comput. J.* **2013**, *56*, 1063–1082. [[CrossRef](#)]
2. Mehta, S.; Shen, X.; Gou, J.; Niu, D. A New Nearest Centroid Neighbor Classifier Based on K Local Means Using Harmonic Mean Distance. *Information* **2018**, *9*, 234. [[CrossRef](#)]
3. Huang, P.; Liab, T.; Gao, G.; Yao, Y.; Yang, G. Collaborative representation based local discriminant projection for feature extraction. *Digit. Signal Process.* **2018**, *76*, 84–93. [[CrossRef](#)]
4. Wang, Y.; Wu, Y. Complete neighborhood preserving embedding for face recognition. *Pattern Recognit.* **2010**, *43*, 1008–1015. [[CrossRef](#)]
5. Harikumar, R.; Kumar, P.S. Dimensionality Reduction Techniques for Processing Epileptic Encephalographic Signals. *Biomed. Pharmacol. J.* **2015**, *8*, 103–106. [[CrossRef](#)]
6. Dobry, G.; Hecht, R.M.; Avigal, M.; Zigel, Y. Supervector Dimension Reduction for Efficient Speaker Age Estimation Based on the Acoustic Speech Signal. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *19*, 7. [[CrossRef](#)]
7. Du, Z.; Zhang, Y.; Zheng, R.; Jiang, L. Chinese Text Feature Dimension Reduction Based on Semantics. In *Chinese Lexical Semantics*; CLSW; Liu, P., Su, Q., Eds.; Springer: Berlin, Germany, 2013; Volume 8229.
8. Kim, H.; Howland, P.; Park, H.; Christianini, N. Dimension Reduction in Text Classification with Support Vector Machines. *J. Mach. Learn. Res.* **2005**, *6*, 37–53.
9. Yan, S.; Xu, D.; Zhang, B.; Zhang, H.J.; Yang, Q.; Lin, S. Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 40–51. [[CrossRef](#)]
10. Jolliffe, I.T. Principal Component Analysis and Factor Analysis. *Princ. Compon. Anal.* **1986**, 115–128. [[CrossRef](#)]
11. Turk, M.A.; Pentland, A.P. Face Recognition Using Eigenfaces. In Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), Maui, HI, USA, 3–6 June 1991; pp. 586–591. [[CrossRef](#)]
12. Tenenbaum, J.B.; Desilva, V.; Langford, J.C. A global geometric framework for nonlinear dimensionality reduction. *Science* **2000**, *290*, 2319–2323. [[CrossRef](#)]
13. Roweis, S.T.; Saul, L.K. Nonlinear dimensionality reduction by locally linear embedding. *Science* **2000**, *290*, 2323–2326. [[CrossRef](#)] [[PubMed](#)]

14. Belkin, M.; Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **2003**, *15*, 1373–1396. [[CrossRef](#)]
15. Bengio, Y.; Paiement, J.; Vincent, P.; Dellalau, O.; Roux, L.; Quimet, M. Out-of-Sample Extensions for LLE, Isomap, MDS, Eigen-maps, and Spectral Clustering. *Adv. Neural Inf. Process. Syst.* **2004**, *8*, 177–184.
16. Belhumeur, P.N.; Hefanaha, J.P.; Kriegman, D.J. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 711–720. [[CrossRef](#)]
17. He, X.; Niyogi, P. Locality Preserving Projections. *Adv. Neural Inf. Process. Syst.* **2003**, 153–160.
18. He, X.; Cai, D.; Yan, S.; Zhang, H.J. Neighbourhood Preserving Embedding. In Proceedings of the IEEE International Conference on Computer Vision, Beijing, China, 20–25 June 2005; pp. 1208–1213.
19. Abeo, T.A.; Shen, X.J.; Gou, J.P.; Mao, Q.R.; Bao, B.K.; Li, S. Dictionary-induced least squares framework for multi-view dimensionality reduction with multi-manifold embeddings. *IET Comput. Vis.* **2019**, *13*, 97–108. [[CrossRef](#)]
20. Song, B.; Tan, S.; Shi, H.B. Process monitoring via enhanced neighborhood preserving embedding. *Control Eng. Pract.* **2016**, *50*, 48–56. [[CrossRef](#)]
21. Zhao, X.Q.; Wang, T. Tensor dynamic neighborhood preserving embedding algorithm for fault diagnosis of batch process. *Chem. Int. Lab. Syst.* **2017**, *162*, 94–103. [[CrossRef](#)]
22. Wang, Y.; Zhang, H.; Yang, F. A Weighted Sparse Neighbourhood-Preserving Projection for Face Recognition. *IETE J. Res.* **2017**, *63*, 358–367. [[CrossRef](#)]
23. Xu, Y.; Zhu, Q.; Fan, Z.; Qiu, M.; Chen, Y.; Liu, H. Coarse to fine K nearest neighbor classifier. *Pattern Recognit. Lett.* **2013**, *34*, 980–986. [[CrossRef](#)]
24. Gou, J.; Yi, Z.; Du, L.; Xiong, T. A local mean-based k-nearest centroid neighbor classifier. *Comput. J.* **2011**, *55*, 1058–1071. [[CrossRef](#)]
25. He, X.; Yan, S.; Hu, Y.; Niyogi, P.; Zhang, H. Face recognition using laplacian faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 328–340. [[CrossRef](#)] [[PubMed](#)]
26. Zhang, D.; Zhou, Z.H.; Chen, S. Semi-supervised dimensionality reduction. In Proceedings of the 2007 SIAM International Conference on Data Mining, Minneapolis, MN, USA, 30 April–2 May 2007; pp. 629–634. [[CrossRef](#)]
27. Cevikalp, H.; Verbeek, J.; Jurie, F.; Klaser, A. Semi-supervised Dimensionality Reduction Using Pairwise Equivalence Constraints Using Graph Embedding for Face Analysis. In Proceedings of the 3th International Conference Computer Vision Theory and Applications, Funchal, Portugal, 22–25 January 2008.
28. Yu, G.; Peng, H.; Wei, J.; Ma, Q. Enhanced locality preserving projections using robust path based similarity. *Neurocomputing* **2011**, *74*, 598–605. [[CrossRef](#)]
29. Liu, X.; Yin, J.; Feng, Z.; Dong, J.; Wang, L. Orthogonal Neighborhood Preserving Embedding for Face Recognition. In Proceedings of the IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007. [[CrossRef](#)]
30. Koringa, P.A.; Mitra, S.K. ONPPn: Orthogonal Neighborhood Preserving Projection with Normalization and its applications. *Image Vis. Comput.* **2018**, *76*, 64–75. [[CrossRef](#)]
31. Kishan, K.C.; Rui, L.; Feng, C.; Anne, R.H. GNE: A deep learning framework for gene network inference by aggregating biological information. *Bioinformatics* **2018**. [[CrossRef](#)]
32. Liang, J.; Chen, C.; Yi, Y.; Xu, X.; Ding, M. *Bilateral Two-Dimensional Neighborhood Preserving Discriminant Embedding for Face Recognition*; IEEE Access: New York, NY, USA, 2017; Volume 5, pp. 2169–3536. [[CrossRef](#)]
33. Han, P.Y.; Jin, A.T.B.; Abas, F.S. Neighbourhood preserving discriminant embedding in face recognition. *J. Vis. Commun. Image Represent.* **2009**, *20*, 532–542. [[CrossRef](#)]
34. Liu, Y.; Chan, K.; Hua, K. Hybrid manifold embedding. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *99*, 1–8. [[CrossRef](#)]
35. Du, S.; Ward, R.K. Adaptive region-based image enhancement method for robust face recognition under variable illumination conditions. *IEEE Trans. Circuits Syst. Video Technol.* **2010**, *20*, 1165–1175. [[CrossRef](#)]
36. Zhang, D.; Liu, H.; Lu, L.; Luo, B. Image recognition with extension of neighborhood preserving embedding on matrices. *J. Inf. Comput. Sci.* **2012**, *9*, 1511–1520.
37. Wang, M.; Hua, X.S.; Hong, R.; Tang, J.; Qi, G.J.; Song, Y. Unified Video Annotation via Multigraph Learning. *IEEE Trans. Circuits Syst. Video Technol.* **2009**, *19*, 733–746. [[CrossRef](#)]
38. Zhang, Q.; Zhang, Q.; Zhang, L.; Tao, D.; Huang, X.; Du, B. Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding. *Pattern Recognit.* **2015**, *48*, 3102–3112. [[CrossRef](#)]

39. The ORL Face Database. Available online: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>. (accessed on 28 July 2018).
40. Georgia Tech Face Database. Available online: http://www.anefian.com/research/face_reco.htm (accessed on 28 July 2018).
41. Sim, T.; Baker, S.; Bsat, M. The CMU pose, illumination, and expression database. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 1615–1618. [[CrossRef](#)]
42. Yale University Face Database. Available online: <http://cvc.yale.edu/projects/yalefaces/yalefaces.html> (accessed on 28 July 2018).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).