Exponential Locality Preserving Projections for Small Sample Size Problem

Su-Jing Wang[a], Hui-Ling Chen[a], Xu-Jun Peng[b], Chun-Guang Zhou[a,*]

[a] *College of Computer Science and Technology, Jilin University, Changchun 130012, China*

[b] *Raytheon BBN technologies, Boston, MA, 02138, USA*

Abstract

Locality Preserving Projections (LPP) is a widely used manifold reduced dimensionality technique. However, it suffers from two problems: (1) Small Sample Size problem; (2)the performance is sensitive to the neighborhood size k. In order to address these problems, we propose an Exponential Locality Preserving Projections (ELPP) by introducing the matrix exponential in this paper. ELPP avoids the singular of the matrices and obtains more valuable information for LPP. The experiments are conducted on three public face databases, ORL, Yale and Georgia Tech. And the results show that the performances of ELPP is better than those of LPP and the state-of-the-art LPP Improved1.

Keywords: locality preserving projections, small sample size problem, matrix exponential, facial recognition

1. Introduction

Automatic facial recognition is a longstanding challenge in the field of computer vision and pattern recognition for several decades. A real face image usually has a high dimensional data. In order to deal with the high dimensional image data adequately and avoid the curse of dimensionality, its dimensionality needs to be reduced. Dimensionality reduction is the transformation of high-dimensional data into a meaningful representation of dimensionality reduction. PCA[1] and LDA[2] are two widely used techniques

*Corresponding author

Email address: cgzhou@jlu.edu.cn (Chun-Guang Zhou)

for reduced dimensionality. Recently, a number of research efforts show that the high dimensional image information in the real world lies on or is close to a smooth nonlinear low dimensional manifold[3]. However, both PCA and LDA fail to discover the underlying manifold structure, due to the fact that they aim only to preserve the global structures of the image samples. In order to uncover the essential manifold structure of the facial images, *laplacianfaces*[4] are obtained by using LPP[5] to preserve the local structure of image samples, i.e., the neighbor relationship between samples.

The neighbor relationship is measured by the artificially constructed adjacent graph. Usually, the most popular adjacent graph construction manner is based on the k nearest neighbor or $\epsilon-$neighborhood criteria. Once an adjacent graph is constructed, the edge weights are assigned by various strategies such as 0-1 weights and heat kernel function. Unfortunately, such adjacent graph is artificially constructed in advance, thus it does not necessarily uncover the intrinsic local geometric structure of the samples. To make things worse, the performance of LPP is seriously sensitive to the neighborhood size k. To address the problem, some researches focus on how to construct the adjacent graph. Instead of predefining a same neighborhood size k for all samples, Sample-dependent Graph [6] is constructed based on samples in question to determine the neighbors of each sample and similarities between sample pairs. Locally Discriminating Projection (LDP) [7] uses label information to construct the adjacent graph. Sparsity Preserving Projections (SPP) [8] aims to preserve the sparse reconstructive relationship of samples, which is achieved by constructing the adjacent graph by minimizing a L1 regularization-related objective function.

Another problem of LPP is the fact that, like LDA, it also suffers from the Small Sample Size (SSS) problem, when the dimension of the sample is larger than the number of the samples, which causes a matrix to be singular. For LDA, *Fisherfaces* employed PCA to reduce dimension before executing the LDA. Exponential discriminant analysis (EDA)[9] introduced the matrix exponential to overcome the SSS problem of LDA. For LPP, *Laplacianfaces*[4] uses PCA to reduce dimension, and then applying the LPP. However, a potential problem is that the PCA criterion may not be compatible with the LPP criterion, thus the PCA step may discard the valuable information for LPP in the null space of $\mathbf{XLX}^T$. In order to address this issue, the Direct LPP [10] optimizes locality preserving criterion on high-dimensional images via simultaneously diagonalizing $\mathbf{XLX}^T$ and $\mathbf{XDX}^T$. Xu *et al.*[11] transforms $\mathbf{XLX}^T$ and $\mathbf{XDX}^T$ into the main space of $\mathbf{XDX}^T$, then find

the optimal solution in the main space of $\mathbf{XDX}^T$. The above methods can extract the feature vectors of $N-1$ dimensions at most.

To alleviate the above two problems of LPP: (1) the SSS problem; (2)the performance is sensitive to the neighborhood size k, we propose an Exponential Locality Preserving Projections (ELPP) in this paper. ELPP uses matrix exponential to avoid the singular of $\mathbf{XDX}^T$. The main advantages of the proposed method are two-fold. On the one hand, ELPP shows advantageous performance over LPP on complex face databases. Especially, on larger size of training set, ELPP significantly outperforms LPP in terms of the recognition rate. On the other hand, compared with LPP, ELPP is much less sensitive to the parameter k.

The rest of this paper is organized as follows: in Section 2, we briefly review the LPP algorithm; in Section 3, we give the background of matrix exponential, and introduce the ELPP algorithm; in Section 4, the experiments are conducted on three public face databases: ORL, Yale and Georgia Tech face database, and the results are analyzed which show that the performance of ELPP is better than those of LPP and the state-of-the-art LPP Improved1, especially for face database with different poses and cluttered background like the Georgia Tech face database; finally in Section 5, conclusions are drawn.

2. Locality Preserving Projections

Given a set of N samples $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{R}^D$, we attempt to find a transformation matrix $\mathbf{W}$ of size $D \times d$ to map: $\mathbf{y}_i = \mathbf{W}^T\mathbf{x}_i, \mathbf{y}_i \in \mathbb{R}^d$, such that $\mathbf{y}_i$ is easier to be distinguished in the projective subspace.

LPP[5] attempts to preserve the local structure of samples in the low-dimensional projective subspace as much as possible. The local structure of samples is measured by constructing a adjacency graph G. There are two ways to construct G: $\varepsilon-$ neighborhoods and k nearest neighbors. In this paper, we use the k nearest neighbors to construct adjacency graph G.

The similarity matrix $\mathbf{S}$ is usually defined by using the heat kernel function as following:

$$S_{ij} = \begin{cases} exp(-\|x_i - x_j\|^2/2t^2) & \text{nodes } i \text{ and } j \text{ are connected in } G \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

where t is a parameter that is determined empirically. In order to avoid to contain discriminant information, we do not use any label information

to construct the similarity matrix $\mathbf{S}$. We hope that the criterion function incurs a heavy penalty if neighboring points $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Therefore, minimizing to the criterion is an attempt to ensure that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are close, then $\mathbf{y}_i$ and $\mathbf{y}_j$ are close, as well. That means to minimize:

$$\sum_{i,j} (\mathbf{y}_i - \mathbf{y}_j)^2 S_{ij} \tag{2}$$

A reasonable criterion function of LPP is as follows:

$$\min_{\mathbf{W}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{W}=\mathbf{I}} \mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W} \tag{3}$$

where $\mathbf{D}$ is a diagonal matrix; its entries $D_{ii} = \sum_j S_{ij}$ measure the local density around $\mathbf{x}_i$. $\mathbf{L} = \mathbf{D} - \mathbf{S}$ is the Laplacian matrix. Finally, the transformation matrix consists of the eigenvectors associated with the smallest eigenvalues of the following generalized eigenvalue problem:

$$\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{w} = \lambda\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{w} \tag{4}$$

3. Exponential Locality Preserving Projections (ELPP)

3.1. Matrix Exponential

The matrix exponential is widely used in applications such as nuclear magnetic resonance spectroscopy [12][13], control theory[14], and Markov chain analysis[15]. In this section, the definition and properties of matrix exponential are introduced. Given an $n \times n$ square matrix $\mathbf{A}$, its exponential is defined as follows:

$$\exp(\mathbf{A}) = \mathbf{I} + \mathbf{A} + \frac{\mathbf{A}^2}{2!} + \cdots + \frac{\mathbf{A}^m}{m!} + \cdots \tag{5}$$

where $\mathbf{I}$ is a identity matrix with the size of $n \times n$. The properties of matrix exponential are listed as follows:

1. $\exp(\mathbf{A})$ is a finite matrix.
2. $\exp(\mathbf{A})$ is a full rank matrix
3. If matrix $\mathbf{A}$ commutes with $\mathbf{B}$, i.e.,$\mathbf{AB} = \mathbf{BA}$, then $\exp(\mathbf{A} + \mathbf{B}) = \exp(\mathbf{A})\exp(\mathbf{B})$.
4. If $\mathbf{B}$ is a nonsingular matrix, then $\exp(\mathbf{B}^{-1}\mathbf{AB}) = \mathbf{B}^{-1}\exp(\mathbf{A})\mathbf{B}$.

5. If $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ are eigenvectors of $\mathbf{A}$ that correspond to the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$, then $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ are also eigenvectors of $\exp(\mathbf{A})$ that correspond to the eigenvalues $e^{\lambda_1}, e^{\lambda_2}, \ldots, e^{\lambda_n}$. It is also well known that the matrix is non-singular.

A wide variety of methods for computing $\exp(\mathbf{A})$ were analyzed in the classic paper of Moler and Van Loan [16], which was reprinted with an update in [17]. The scaling and squaring method is one of the best methods for computing the matrix exponential. To facilitate the subsequent discussion, we first define the $[p/q]$ Padé approximant to $\exp(\mathbf{A})$ as following:

$$R_{pq}(\mathbf{A}) = [D_{pq}(\mathbf{A})]^{-1} N_{pq}(\mathbf{A}) \tag{6}$$

where

$$N_{pq}(\mathbf{A}) = \sum_{j=0}^{p} \frac{(p+q-j)!q!}{(p+q)!j!(q-j!)} (\mathbf{A})^j \tag{7}$$

and

$$D_{pq}(\mathbf{A}) = \sum_{j=0}^{q} \frac{(p+q-j)!q!}{(p+q)!j!(q-j!)} (\mathbf{A})^j \tag{8}$$

The scaling and squaring method is summarized in Algorithm 1. The cost of Algorithm 1 is $\pi_m + \lceil \log_2(\|\mathbf{A}\|_1/\theta_m) \rceil$ matrix multiplications, where m is the degree of Padé approximant used, and π_m is tabulated in Table 1, plus the solution of one matrix equation. For details, please refer to [18].

Table 1: Number of matrix multiplications, π_m, required to evaluate R_m

m	1	2	3	4	5	6	7	8	9	10	
π_m	0	1	2	3	3	4	4	5	5	6	
m	11	12	13	14	15	16	17	18	19	20	21
π_m	6	6	6	7	7	7	7	8	8	8	8

3.2. the ELPP Algorithm

We denote $\mathbf{S}_L = \mathbf{X}\mathbf{L}\mathbf{X}^T$ and $\mathbf{S}_D = \mathbf{X}\mathbf{D}\mathbf{X}^T$, then the eigen solution formulation of LPP (3) can be rewritten as follows:

$$\min_{\mathbf{W}^T\mathbf{S}_D\mathbf{W}=\mathbf{I}} \mathbf{W}^T\mathbf{S}_L\mathbf{W} \tag{9}$$

Algorithm 1 the scaling and squaring method

b(0: 13) = [64764752532480000, 32382376266240000, 7771770303897600, 1187353796428800, 129060195264000, 10559470521600, 670442572800, 33522128640, 1323241920, 40840800, 960960, 16380, 182, 1];
$\theta_3 = 0.01495585217958292$, $\theta_5 = 0.253939833006323$,$\theta_7 = 0.9504178996162932$, $\theta_9 = 2.097847961257068$,$\theta_{13} = 5.371920351148152$,
$\mu = trace(\mathbf{A})/n$
$\mathbf{A} \leftarrow \mathbf{A} - \mu\mathbf{I}$
$\mathbf{A} \leftarrow \mathbf{D}^{-1}\mathbf{AD}$, where $\mathbf{D}$ is a balancing transformation (or set $\mathbf{D} = \mathbf{I}$ if balancing does not reduce the 1-norm of $\mathbf{A}$).
for $m = [3, 5, 7, 9, 13]$ **do**
  if $\|\mathbf{A}\|_1 \leq \theta_m$ **then**
    $\mathbf{X} = R_m(\mathbf{A})$, where R_m is the $[m/m]$ Padé approximant to $\exp(\mathbf{A})$
    $\mathbf{X} = e^{\mu}\mathbf{DXD^{-1}}$
  end if
end for
$\mathbf{A} \leftarrow \mathbf{A}/2^s$, where s is a minimal integer such that $\|\mathbf{A}/2^s\|_1 \leq \theta_{13}$
$\mathbf{A}_2 = \mathbf{A}^2, \mathbf{A}_4 = \mathbf{A}_2^2, \mathbf{A}_6 = \mathbf{A}_2\mathbf{A}_4$
$\mathbf{U} = \mathbf{A}[\mathbf{A}_6(b_{13}\mathbf{A}_6 + b_{11}\mathbf{A}_+ b_9\mathbf{A}_2) + b_7\mathbf{A}_6 + b_5\mathbf{A}_4 + b_3\mathbf{A}_2 + b_1\mathbf{I}]$
$\mathbf{V} = \mathbf{A}_6(b_{12}\mathbf{A}_6 + b_{10}\mathbf{A}_+ b_8\mathbf{A}_2) + b_6\mathbf{A}_6 + b_4\mathbf{A}_4 + b_2\mathbf{A}_2 + b_0\mathbf{I}$
Solve $(-\mathbf{U} + \mathbf{V})R_{13} = \mathbf{U} + \mathbf{V}$ for R_{13}
$\mathbf{X} = R_{13}^{2^s}$ by repeated squaring.
$\mathbf{X} = e^{\mu}\mathbf{DXD^{-1}}$

Theorem 1. *Let D and N be the dimension of the sample and the number of the samples,respectively .If $D > N$, then the rank of $\mathbf{S}_L$ is at most $N-1$ and the rank of $\mathbf{S}_D$ is at most N.*

PROOF. According to the properties of the Laplacian matrix, it is easy to know that the determinant of $\mathbf{L}$ is 0. So, the rank of $\mathbf{L}$ is at most $N-1$. It is known that the maximum possible rank of the product of two matrices is smaller than or equal to the smaller of the ranks of the two matrices. Hence, $rank(\mathbf{S}_L) = rank(\mathbf{X}\mathbf{L}\mathbf{X}^T) \leq N-1$. Similarly, we have $rank(\mathbf{S}_D) \leq N$.

From Theorem 1, LPP also suffers from the SSS problem, due to the fact that the matrix $\mathbf{S}_L$ is singular when the SSS problem occurs. We denote the eigenvectors of $\mathbf{S}_L$ as $\mathbf{V}_L = [\mathbf{v}_{L1}, \mathbf{v}_{L2}, \ldots, \mathbf{v}_{Ln}]$ that correspond to the eigenvalues $\mathbf{\Lambda}_L = diag(\lambda_{L1}, \lambda_{L2}, \ldots, \lambda_{Ln})$. Similarly, the eigenvectors of $\mathbf{S}_D$ are denoted as $\mathbf{V}_D = [\mathbf{v}_{D1}, \mathbf{v}_{D2}, \ldots, \mathbf{v}_{Dn}]$ that correspond to the eigenvalues $\mathbf{\Lambda}_D = diag(\lambda_{D1}, \lambda_{D2}, \ldots, \lambda_{Dn})$. The Eq. (9) can be rewritten as follows:

$$\min_{\mathbf{W}^T(\mathbf{V}_D^T\mathbf{\Lambda}_D\mathbf{V}_D)\mathbf{W}=\mathbf{I}} \mathbf{W}^T(\mathbf{V}_L^T\mathbf{\Lambda}_L\mathbf{V}_L)\mathbf{W} \tag{10}$$

The matrix $\mathbf{S}_L$ is non-singular, when the SSS problem occurs. In order to address the problem, the PCA is adopted to reduce the dimension of the feature space to $N-1$, before applying the standard LPP defined by Eq. (10). Unfortunately, the valuable information for LPP in the null space of $\mathbf{S}_L$ may also be discarded in the PCA step.To extract this kind of valuable information for LPP, we replace λ_{Li}, i.e., the eigenvalues of $\mathbf{S}_L$, by $\exp(\lambda_{Li})$ and λ_{Di}, i.e., the eigenvalues of $\mathbf{S}_D$, by $\exp(\lambda_{Di})$. Then, Eq. (10) is transformed into

$$\begin{aligned} \min_{\mathbf{W}^T(\mathbf{V}_D^T\exp(\mathbf{\Lambda}_D)\mathbf{V}_D)\mathbf{W}=\mathbf{I}} & \mathbf{W}^T(\mathbf{V}_L^T\exp(\mathbf{\Lambda}_L)\mathbf{V}_L)\mathbf{W} \\ &= \min_{\mathbf{W}^T\exp(\mathbf{S}_D)\mathbf{W}=\mathbf{I}} \mathbf{W}^T\exp(\mathbf{S}_L)\mathbf{W} \end{aligned} \tag{11}$$

The above equation is the criterion function of ELPP. According to the properties of the matrix exponential, the $\exp(\mathbf{S}_L)$ is nonsingular. The valuable information for LPP in the null space of $\mathbf{S}_L$ can be extracted by Eq. (11). According to Rayleigh quotient, Eq. (11) is minimized if and only if the matrix $\mathbf{W}$ consists of d generalized eigenvectors, which are corresponding to the

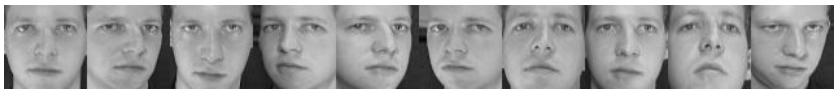


Figure 1: Sample images of one individual from the ORL database.

Figure 2: Sample images of one individual in the YALE database.

smallest d generalized eigenvalues of the matrix pencil $(\exp(\mathbf{S}_L), \exp(\mathbf{S}_D))$, which satisfy:

$$\exp(\mathbf{S}_L)\mathbf{w} = \lambda \exp(\mathbf{S}_D)\mathbf{w} \tag{12}$$

The transformation matrix $\mathbf{W}$ consists of the d eigenvectors corresponding to the d smallest eigenvalues.

4. Experiments

4.1. Databases

We conducted the experiments on three well-known face databases ORL[1], Yale[2] and Georgia Tech face databases [3].

The ORL database collects images from 40 individuals, where 10 different images are captured for each individual. For each individual, the images with different facial expressions and details are obtained at different times. The face in the images may be rotated, scaled and be tilting in some degree. Each image is manually cropped and resized to 32×32 pixels. The sample images of one individual from the ORL database are shown in Fig. 1.

There are total of 165 gray scale images for 15 individuals where each individual has 11 images in Yale face database. The images demonstrate variations in lighting condition, facial expression (normal, happy, sad, sleepy, surprised, and wink). Each image is manually cropped and resized to 32×32 pixels. The sample images of one individual from the Yale database are showed in Fig. 2.

[1] `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`
[2] `http://cvc.yale.edu/projects/yalefaces/yalefaces.html`
[3] `http://www.anefian.com/research/face_reco.htm`

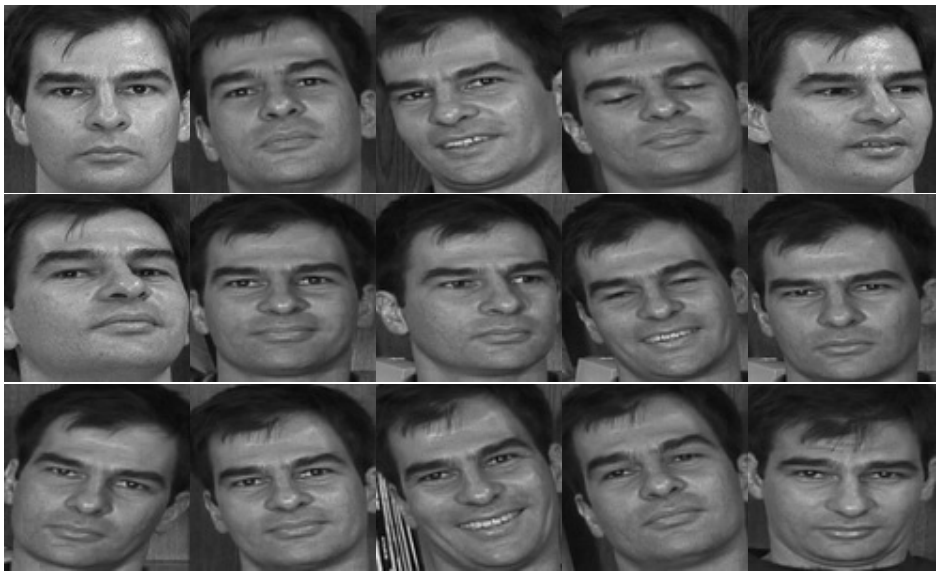


Figure 3: Sample images of one individual from the Georgia Tech database.

Georgia Tech face database contains images of 50 individuals taken in two or three sessions at different times. Each individual in the database is represented by 15 color JPEG images with cluttered background taken at resolution 640×480 pixels. The average size of the faces in these images is 150×150 pixels. The pictures show frontal and/or tilted faces with different facial expressions, lighting conditions and scale. In the experiments, 15 individuals were selected from 50 individuals. Each image was manually grayed, cropped and resized to 32×32 pixels. The sample images for one individual of the Georgia Tech database are showed in Fig. 3.

4.2. Experiments and results on the ORL database

The experiments are conducted on the ORL database. In our experiments, the similarity matrix $\mathbf{S}$ is defined by the heat kernel function. Empirically, the parameter t is set as the mean norm of the training set. The neighbors parameter k is searched from $\{2, 3, \ldots, N-1\}$. We randomly split the image samples so that p ($p = 2, 3, 4, 5, 6, 7, 8$) images for each individual are used as the training set and the remainings are used as the testing set. This process is repeated 50 times. Fig. 4 plots the relationship between the performances of two algorithms and the neighborhood size k, when $p = 2, 3, 4$. The warmer color represents the better performance in the figure. Compar-

ing the corresponding columns of Fig. 4(a) and Fig. 4(b), there's very little color difference in each column of Fig. 4(a). This means that the performance of ELPP is much less sensitive to the parameter k than that of LPP.

Analytically, we define the criterion to measure the sensitivity to the parameter k. The recognition accuracies are normalized to $[0, 1]$. Within the 50 random splits in our experiments, each split includes $N-1$ recognition accuracies corresponding $N-1$ values of k. For each split, the maximum difference of recognition accuracy is obtained by subtracting the minimum accuracy from the maximum accuracy. The criterion Mean Maximum Difference (MMD) is the mean value of all maximum differences of recognition accuracy. To a certain degree, the smaller value of MMD means more insensitive to k. The MMDs of ELPP and LPP are listed in Table 2. From the table, the MMDs of ELPP are less than that of LPP. This also shows that ELPP is less sensitive to k than LPP.

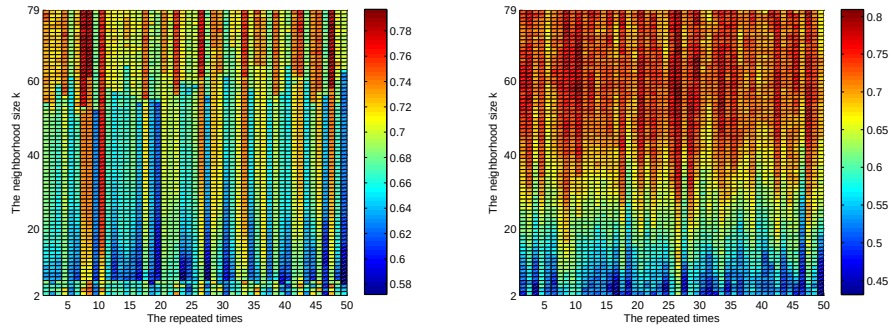
Table 2: MMD of ELPP and LPP on ORL database

p	2	3	4	5	6	7	8
ELPP	0.4258	0.3971	0.3890	0.4467	0.4793	0.3727	0.3357
LPP	0.6407	0.7074	0.7516	0.7009	0.6800	0.6191	0.6562

In order to investigate the performance of ELPP, we compare ELPP with PCA, LPP and LPP Improved1[19]. These methods are the dimensionality reduction algorithms without discriminant information. The results are illustrated in Fig. 5. The solid lines denote that the neighborhood size k traverses $\{2, 3, \ldots, N-1\}$. The dot-dash lines denote that the k is equal to 2. As shown in Fig. 5, the performances of ELPP are much better than those of LPP and LPP Improved1 in the case of $k = 2$. When k traverses $\{2, 3, \ldots, N-1\}$, the performances of ELPP, LPP and LPP Improved1 are not much different. With the increasing sample size, the performance of ELPP becomes better and better. In Fig. 5, we also find that the space between two curves of ELPP is much narrower than those of LPP and LPP Improved1. This also proves that the performance of ELPP is much less sensitive to the parameter k than those of LPP and LPP Improved1.

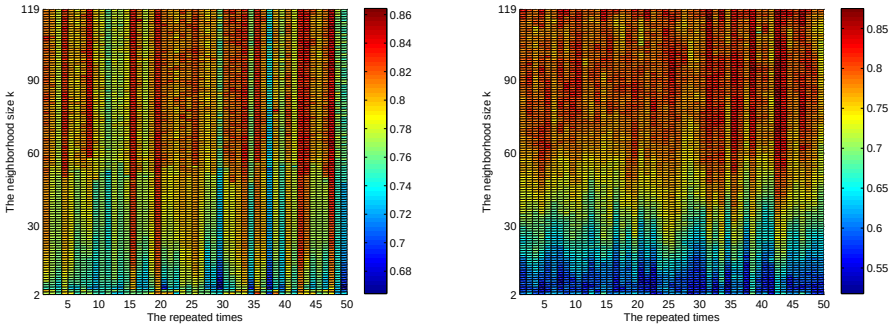
4.3. Experiments and results on the Yale database

Unlike ORL database, Yale database contains the facial images which are influenced by various expressions and lighting condition. The exponential



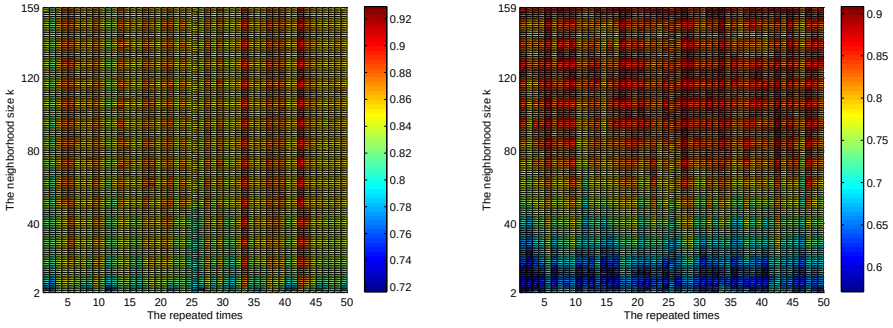
(a) ELPP (p=2)

(b) LPP (p=2)



(c) ELPP (p=3)

(d) LPP (p=3)



(e) ELPP (p=4)

(f) LPP (p=4)

Figure 4: The performances of two algorithms vs. the neighborhood size k on the ORL database.

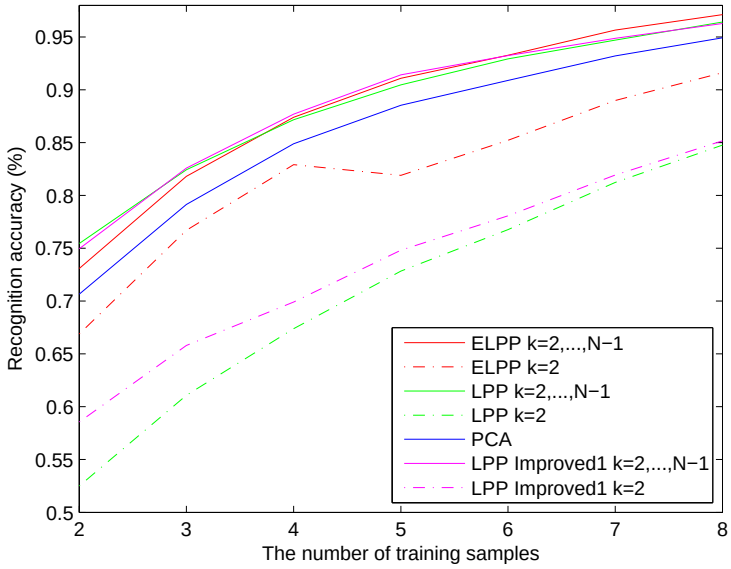


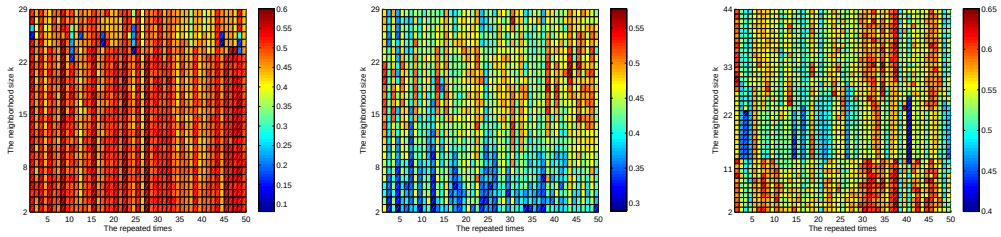
Figure 5: The performances of two algorithms on the ORL database.

setting is described in the above section. Fig. 6 plots the relationship between the performances of two algorithms and the neighborhood size k. Comparing the corresponding columns of Fig. 6(a) and Fig. 6(b), there's very little color difference in each column of Fig. 6(a). This means that the performance of ELPP is much less sensitive to the parameter k than that of LPP. The same conclusion can be drawn from the other sub-figures. We also list the MMDs of of ELPP and LPP in Table 3. From the table, the MMDs of ELPP are less than that of LPP.

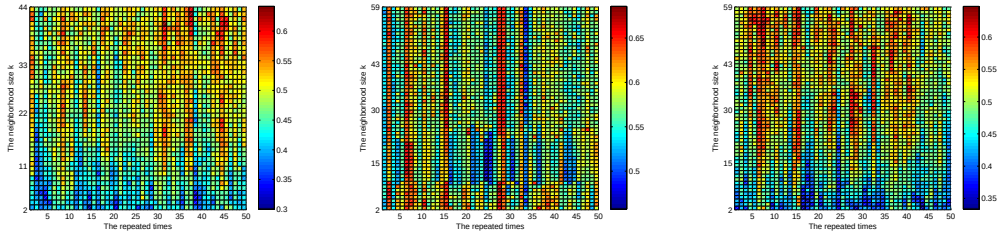
Table 3: MMD of ELPP and LPP on Yale database

p	2	3	4	5	6	7	8
ELPP	0.2617	0.3627	0.3625	0.3619	0.3170	0.3289	0.3333
LPP	0.4590	0.4800	0.5636	0.4984	0.5772	0.5817	0.5752

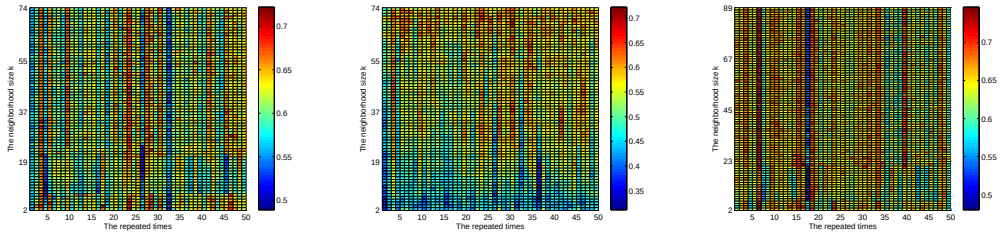
In the same way, we implement ELPP, LPP and LPP Improved1 on Yale database. The results are illustrated in Fig. 7. The legend is the same as the description in the above section. As shown in Fig. 7, the performances of



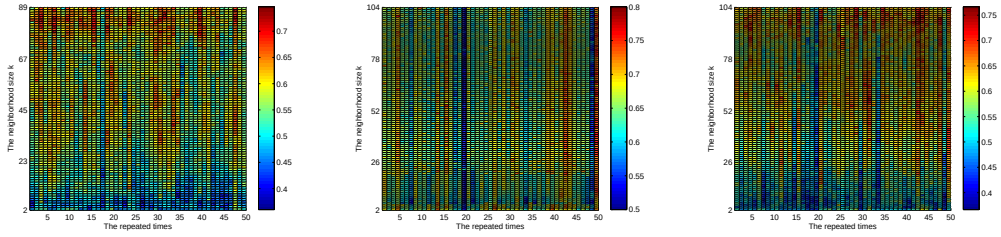
(a) ELPP (p=2) (b) LPP (p=2) (c) ELPP (p=3)



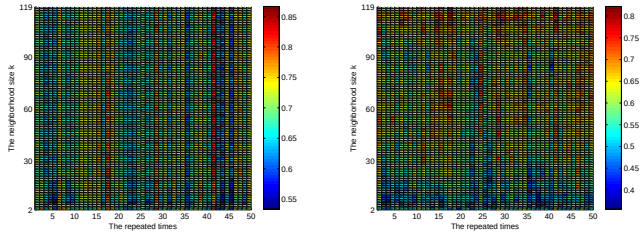
(d) LPP (p=3) (e) ELPP (p=4) (f) LPP (p=4)



(g) ELPP (p=5) (h) LPP (p=5) (i) ELPP (p=6)



(j) LPP (p=6) (k) ELPP (p=7) (l) LPP (p=7)



(m) ELPP (p=8) (n) LPP (p=8)

Figure 6: The performances of two algorithms vs. the neighborhood size k on the Yale database.

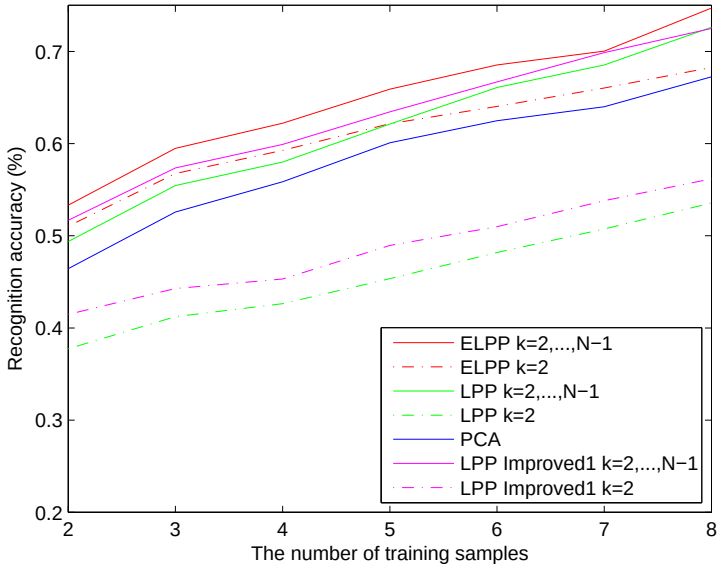


Figure 7: The performances of two algorithms on the Yale database.

ELPP are better than those of LPP and LPP Improved1 in two ranges of k. It is interesting that when the training sample size is small, the performance of ELPP with $k = 2$ is better than that of LPP with $k = 2, 3, \ldots, N-1$. It also illustrates that ELPP is more effective than LPP for SSS problem. In Fig. 7, we also find that the space between two curves of ELPP is much narrower than those of LPP and LPP Improved1. This also proves that the performance of ELPP is much less sensitive to the parameter k than those of LPP and LPP Improved1.

4.4. Experiments and results on the Georgia Tech face database

Georgia Tech face database is more complex than Yale database, because it contains various pose faces with different expressions on cluttered background. In this experiment, We randomly split the image samples so that p ($p = 2, 4, 6, 8, 10, 12$) images for each individual are used as the training set and the rest are used as the testing set. This process is repeated 30 times. Other setting is the same as Yale database. We plot the relationship between the performances and k in Fig. 8. In this figure, we not only see the similar phenomenon in Fig. 6 but also see the fact of that the better performance of ELPP occurs when k is small. The MMDs of ELPP and LPP are shown

in Table 4. As we can see in Table 4, in each split, the MMD of ELPP is less than that of LPP, except for $p = 2$. The similar conclusion can be drawn from the table. The experimental results about the performances are also illustrated in Fig. 9. As shown in the figure, the similar phenomenon as Fig. 7 can be seen. Moreover, we also find that the performance of ELPP with $k = 2$ is better than that of LPP with $k \in \{2, 3, \ldots, N-1\}$.

Table 4: MMD of ELPP and LPP on the Georgia Tech database

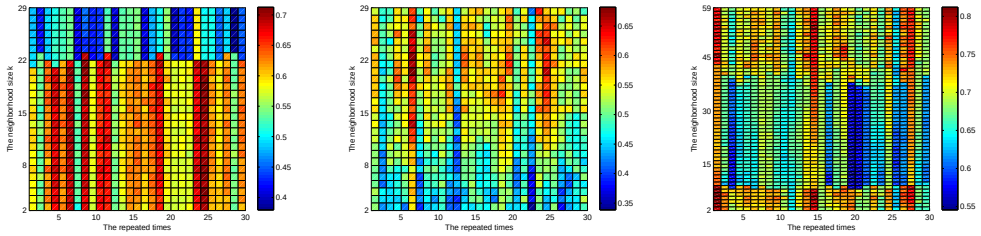
p	2	4	6	8	10	12
ELPP	0.4733	0.3856	0.3875	0.4648	0.3917	0.3048
LPP	0.4060	0.5896	0.6350	0.5981	0.6493	0.5825

4.5. Discussion

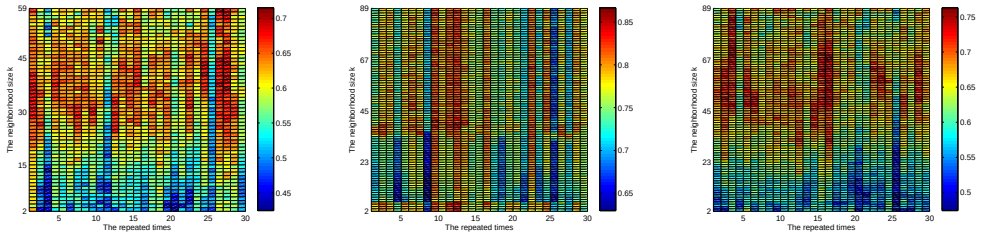
The experiments have been systematically performed on three public face databases: ORL, Yale and Georgia Tech. Among them, Yale database is the simplest one which only includes the facial images from various views. The Georgia Tech database is the most complex one which includes the non-aligned head image with cluttered background. On ORL database, the performance of PCA is better than the performances of the other three methods with $k = 2$ (see Fig. 5). This is due to the tuning parameters for the other three methods. On more complex Yale database, the performance of ELPP with $k = 2$ is superior to that of PCA (see Fig. 7). On the most complex Georgia Tech database, ELPP with $k = 2$ surpasses over LPP with $k \in \{2, 3, \ldots, N-1\}$ (see Fig. 9). Based on above analysis, ELPP shows outstanding performance on complex face databases.

5. Conclusion

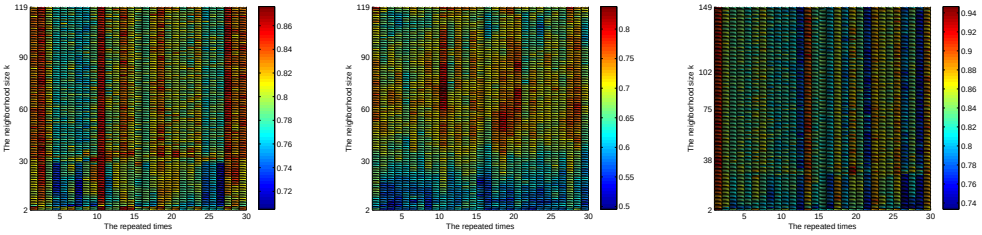
We have presented a new reduced dimensionality technique, which is named as Exponential Locality Preserving Projections. It addressed the two problems of LPP: (1) Small Sample Size problem; (2)the performance is sensitive to the neighborhood size k. ELPP avoids the singular of the matrices and obtains more valuable information for LPP. The experimental results prove the performances of ELPP was better than these of LPP and LPP Improved1 on three public face database: ORL, Yale and Georgia Tech. The results reveal a number of interesting remarks:



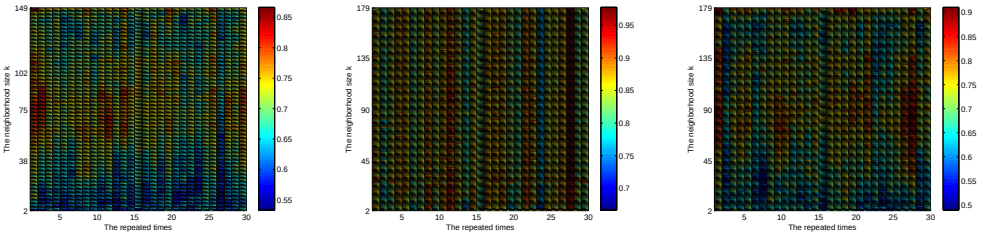
(a) ELPP (p=2) (b) LPP (p=2) (c) ELPP (p=4)



(d) LPP (p=4) (e) ELPP (p=6) (f) LPP (p=6)



(g) ELPP (p=8) (h) LPP (p=8) (i) ELPP (p=10)



(j) LPP (p=10) (k) ELPP (p=12) (l) LPP (p=12)

Figure 8: The performances of two algorithms vs. the neighborhood size k on the the Georgia Tech database.

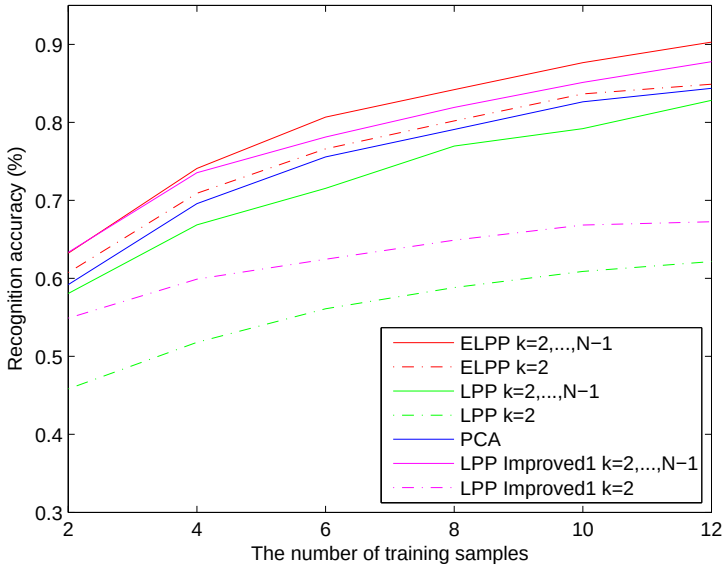


Figure 9: The performances of two algorithms on the Georgia Tech database.

1. ELPP shows outstanding performance on complex face databases.
2. ELPP shows better performance on larger size of training set.
3. ELPP is much less sensitive to the parameter k than LPP.

Our future work focus on applying matrix exponential to improve the varieties of LPP.

Acknowledgements

This paper is supported by (1)the National Natural Science Foundation of China under Grant No. 60873146, 60973092, 60903097, (2)project of science and technology innovation platform of computing and software science (985 engineering), (3)the Key Laboratory for Symbol Computation and Knowledge Engineering of the National Education Ministry of China.

[1] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1) (1991) 71–86.

[2] P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman, Eigenfaces vs. fisherfaces: recognition using class specific linear projection, IEEE Transac-

tions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 711–720.

[3] H. S. Seung, D. D. Lee, The manifold ways of perception, Science 290 (5500) (2000) 2268–2269.

[4] X. F. He, S. C. Yan, Y. X. Hu, P. Niyogi, H. J. Zhang, Face recognition using laplacianfaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 328–340.

[5] X. F. He, P. Niyogi, Locality preserving projections, in: Advances in neural information processing systems, Vol. 16, The MIT Press, 2004, pp. 153–160.

[6] B. Yang, S. Chen, Sample-dependent graph construction with application to dimensionality reduction, Neurocomputing.

[7] H. Zhao, S. Sun, Z. Jing, J. Yang, Local structure based supervised feature extraction, Pattern recognition 39 (8) (2006) 1546–1550.

[8] L. Qiao, S. Chen, X. Tan, Sparsity preserving projections with applications to face recognition, Pattern Recognition 43 (1) (2010) 331–341.

[9] T. Zhang, B. Fang, Y. Y. Tang, Z. Shang, B. Xu, Generalized discriminant analysis: A matrix exponential approach, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 40 (1) (2010) 186–197.

[10] G. Feng, D. Hu, Z. Zhou, A Direct Locality Preserving Projections (DLPP) Algorithm for Image Recognition, Neural Processing Letters 27 (3) (2008) 247–255.

[11] Y. Xu, A. Zhong, J. Yang, D. Zhang, LPP solution schemes for use with face recognition, Pattern Recognition.

[12] T. Havel, I. Najfeld, J. Yang, Matrix decompositions of two-dimensional nuclear magnetic resonance spectra, Proceedings of the National Academy of Sciences of the United States of America 91 (17) (1994) 7962.

[13] I. Najfeld, T. Havel, Derivatives of the matrix exponential and their computation, Advances in Applied Mathematics 16 (3) (1995) 321–375.

[14] G. Franklin, D. Powell, M. Workman, Digital control of dynamic systems.

[15] R. Sidje, W. Stewart, A numerical study of large sparse matrix exponentials arising in markov chains, Computational statistics & data analysis 29 (3) (1999) 345–368.

[16] C. Moler, C. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, SIAM review 20 (4) (1978) 801–836.

[17] C. Moler, C. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM review 45 (1) (2003) 3–49.

[18] N. Higham, The scaling and squaring method for the matrix exponential revisited, SIAM Journal on Matrix Analysis and Applications 26 (4) (2005) 1179–1196.

[19] Y. Xu, A. Zhong, J. Yang, D. Zhang, LPP solution schemes for use with face recognition, Pattern Recognition 43 (2010) 4165–4176.