Research Article

Multiview Discriminative Geometry Preserving Projection for Image Classification

Ziqiang Wang, Xia Sun, Lijun Sun, and Yuchun Huang

School of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China

Correspondence should be addressed to Ziqiang Wang; zi-qiang-wang@hotmail.com

Received 19 December 2013; Accepted 22 January 2014; Published 9 March 2014

Academic Editors: X. Meng, Z. Zhou, and X. Zhu

Copyright © 2014 Ziqiang Wang 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 many image classification applications, it is common to extract multiple visual features from different views to describe an image. Since different visual features have their own specific statistical properties and discriminative powers for image classification, the conventional solution for multiple view data is to concatenate these feature vectors as a new feature vector. However, this simple concatenation strategy not only ignores the complementary nature of different views, but also ends up with "curse of dimensionality." To address this problem, we propose a novel multiview subspace learning algorithm in this paper, named multiview discriminative geometry preserving projection (MDGPP) for feature extraction and classification. MDGPP can not only preserve the intraclass geometry and interclass discrimination information under a single view, but also explore the complementary property of different views to obtain a low-dimensional optimal consensus embedding by using an alternating-optimization-based iterative algorithm. Experimental results on face recognition and facial expression recognition demonstrate the effectiveness of the proposed algorithm.

1. Introduction

Many computer vision and pattern recognition applications involve processing data in a high-dimensional space. Directly operating on such high-dimensional data is difficult due to the so-called “curse of dimensionality.” For computational time, storage, and classification performance considerations, dimensionality reduction (DR) techniques provide a means to solve this problem by generating a succinct and representative low-dimensional subspace of the original high-dimensional data space. Over the past two decades, many dimensionality reduction algorithms have been proposed and successfully applied to face recognition [1]. The most representative ones are principal component analysis (PCA) and linear discriminant analysis (LDA) [2].

PCA is an unsupervised dimensionality reduction method, which aims to project the high-dimensional data into a low-dimensional subspace spanned by the leading eigenvectors of a covariance matrix. LDA is supervised and its goal is to pursue a low-dimensional subspace by maximizing the ratio of between-class variance to within-class variance. Due to the utilization of label information, LDA usually outperforms PCA for classification tasks when sufficient labeled training data are available. While these two algorithms have attained reasonable good performance in pattern classification, they may fail to discover a highly nonlinear submanifold embedded in the high-dimensional ambient space as they seek only a compact Euclidean subspace for data representation and classification [3].

Recently, there has been considerable interest in manifold learning algorithms for dimensionality reduction and feature extraction. The basic consideration of these algorithms is that the high-dimensional data may lie on an intrinsic nonlinear low-dimensional manifold. In order to detect the underlying manifold structure, nonlinear dimensionality reduction algorithms such as ISOMAP [4], locally linear embedding (LLE) [5], and Laplacian eigenmap (LE) [6] have been proposed. All of these algorithms are defined only on the training data, and the issue of how to map new testing data remains difficult. Therefore, they cannot be applied to classification problem directly. To overcome the above so-called out-of-sample problem, He and Niyogi [7] developed the locality preserving projection (LPP), in which the linear projection function is adopted for mapping new data samples. As LPP is originally unsupervised, some recent attempts have exploited the discriminant information and derived many discriminant
manifold learning algorithms to enhance the classification performance. The representative algorithms include local discriminant embedding (LDE) [8], locality sensitive discriminant analysis (LSDA) [9], margin Fisher analysis (MFA) [10], local Fisher discriminant analysis (LFDA) [11], and discriminative geometry preserving projection (DGPP) [12]. Despite having different assumptions, all these algorithms can be unified into a general graph embedding framework (GEF) [10] with different constraints. While these algorithms have utilized both local geometry and the discriminative information for dimensionality reduction and achieved reasonably good performance in different pattern classification tasks, they assume that the data are represented in a single vector. They can be regarded as single-view-based methods and thus cannot handle data described by multiview features directly. In many practical pattern classification applications, different views (visual features) have their own specific statistical properties, and each view represents the data partially. To address this problem, the traditional solution for multiple view data is to simply concatenate vectors of different views into a new long vector and then apply dimensionality reduction algorithms directly on the concatenated vector. However, this concatenation ignores the diversity of multiple views and thus cannot explore the complementary nature and specific statistical properties of different views. Recent studies have provided convincing evidence of this fact [13–15]. Hence, it is more reasonable to assign different weights to different views (features) for feature extraction and classification. In computer vision and machine learning research, many works have shown that leveraging the complementary nature of the multiple views can better represent the data for feature extraction and classification [13–15]. Therefore, an efficient manifold learning algorithm that can cope with multiview data and place proper weights on different views is of great interest and significance.

Motivated by the above observations and reasons, we propose unifying different views under a discriminant manifold learning framework called multiview discriminative geometry preserving projection (MDGPP). Under each view, we can implement the discrimination and local geometry preserving projections as those used in discriminative geometry preserving projection (DGPP) [12]. Unifying different views in such a multiview discriminant manifold learning framework is meaningful, since data with different features can be appropriately integrated to further improve the classification performance. Specifically, we first implement the discrimination preservation by maximizing the average weighted pairwise distance between samples in different classes and simultaneously minimizing the average weighted pairwise distance between samples in the same class. Meanwhile, the local geometry preservation is implemented by minimizing the reconstruction error of samples in the same class. Then, we learn a low-dimensional feature subspace by utilizing both intraclass geometry and interclass discrimination information, such that the complementary nature of different views (features) can be fully exploited when classification is performed in the derived feature subspace. Experimental results on face recognition and facial expression recognition are presented to demonstrate the effectiveness of the proposed algorithm.

The remainder of the paper is organized as follows. Section 2 reviews the related works. Section 3 presents the details of the proposed MDGPP algorithm. Experimental results on face recognition are presented in Section 4, and the concluding remarks are provided in Section 5.

2. Related Works

Multiview learning is one important topic in the machine learning and pattern recognition communities. In such a setting, view weight information is introduced to measure the importance of different features in characterizing data, and different weights reflect different contribution to the learning process. The aim of multiview learning is to exploit more complementary information of different views rather than only a single view to further improve the learning performance. The traditional solution for multiview data is to concatenate all features into one vector and then conduct machine learning for such feature space. However, this solution is not optimal as these features usually have different physical properties. Simply concatenating them will ignore the complementary nature and specific statistical properties of different views, and thus causing performance degradation. In addition, this simple concatenation will end up with the curse of dimensionality problem for the subsequent learning task.

In order to perform multiview learning, much effort has been focused on multiview metric learning [14], multiview classification and retrieval [16], multiview clustering [15], and multiview semisupervised learning [17]. All these approaches demonstrated that the learning performance can be significantly enhanced if the complementary nature of different views is exploited and all views are appropriately integrated. It is very natural that multiview learning idea should also be considered in dimensionality reduction. However, most of the existing dimensionality reduction algorithms are designed only for single view data and cannot cope with multiview data directly. To address this problem, Long et al. [18] first proposed multiple view spectral embedding (MVSE) method. MVSE performs a dimensionality reduction process on each view independently, and then based on the obtained low-dimensionality representation, it constructs a common low-dimensional embedding that is close to each representation as much as possible. Although MVSE allow selecting different dimensionality reduction algorithms for each view, the original multiview data are invisible to the final learning process. Thus, MVSE cannot well explore the complementary information of different views. Xia et al. [19] proposed multiview spectral embedding (MSE) method to find low-dimensional and sufficiently smooth embedding based on the patch alignment framework [20]. However, MSE ignores the flexibility of allowing shared information between subset of different views owing to the global coordinate alignment process. To unify different views for dimensionality reduction under a probabilistics, Xie et al. [21] extended the stochastic
neighbor embedding (SNE) to its multiview version and pro-
presented multiview stochastic neighbor embedding (MSNE). 

Although MSNE operates on a probabilistic framework, it 
is an unsupervised method and its classification abilities 
may be limited since the class label information is not 
used in the learning process. More recently, inspired by 
the recent advances of sparse coding technique, Han et al. [22] 
proposed spectral sparse multiview embedding (SSMVE) 
method to deal with dimensionality reduction for multiview 
data. Although SSMVE can impose sparsity constraint on 
the loading matrix of multiview dimensionality reduction, it is 
unsupervised and does not explicitly consider the manifold 
structure on which the high dimensional data possibly reside. 

In the next section, focusing on the manifold learning 
and pattern classification, we propose a novel multiview 
discriminative geometry preserving projection (MDGPP) 
for multiview dimensionality reduction, which explicitly 
considers the local manifold structure and discriminative 
information as well as the complementary characteristics of 
different views in high-dimensional data.

3. Multiview Discriminative Geometry 
Preserving Projection (MDGPP)

In this section, we propose a new manifold learning algo-
rithm called multiview discriminative geometry preserving 
projection (MDGPP), which aims to find a unified low-
dimensional and sufficiently smooth embedding over all 
views simultaneously. To better explain the algorithm details 
of the proposed MDGPP, we introduce some important 
notations used in the remainder of this paper. Capital letters 
such as $X$ denote data matrices, and $X_{ij}$ represents the $(i, j)$ 
entry of $X$. Lower case letters such as $x$ denote data vectors, 
and $x_i$ represents the $i$th data element of $x$. Superscript 
($i$) such as $X^{(i)}$ and $x^{(i)}$ represents data from the $i$th view, 
respectively. Based on these notations, MDGPP can be 
described as follows according to the DGPP framework [12].

Given a multiview data set with $n$ data samples 
and each with $m$ feature representations, that is, $X = 
\{X^{(1)}, x^{(2)}, \ldots, x^{(m)}\} \in \mathbb{R}^{p \times n}$, wherein $X^{(i)}$ 
represents the feature matrix for the $i$th view, the aim of MDGPP 
is to find a projective matrix $W \in \mathbb{R}^{d \times d}$ to map $X^{(i)}$ into a 
low-dimensional representation $Y^{(i)}$ through $Y^{(i)} = W^T X^{(i)}$, 
where $d$ denotes the dimension of low-dimensional feature 
representation and satisfies $d < p_i$ ($1 \leq i \leq m$). The 
workflow of MDGPP can be simply described as follows. 
First, MDGPP builds a part optimization for a sample on a 
single view by preserving both the intraclass geometry and 
interclass discrimination. Afterward, all parts of optimization 
from different views are unified as a whole via view weight 
coefficients. Then an alternating-optimization-based iterative 
algorithm is derived to obtain the optimal low-dimensional 
embedding from multiple views.

Given the $i$th view $X^{(i)} = \{x^{(i)}_1, x^{(i)}_2, \ldots, x^{(i)}_n\} \in \mathbb{R}^{p \times n}$, 
MDGPP first makes an attempt to preserve discriminative 
information in the reduced low-dimensional space by 
maximizing the average weighted pairwise distance between 
samples in different classes and simultaneously minimizing 
the average weighted pairwise distance between samples in 
the same class on the $i$th view. Thus, we have

$$
\begin{align*}
\text{arg max}_{w^{(i)}} & \sum_{j=1}^{n} \| y^{(i)}_j - w_i x^{(i)}_j \|^2 \\
= \text{arg max}_{y^{(i)}} & \sum_{j=1}^{n} \| y^{(i)}_j - y^{(i)}_i \|^2 \\
= \text{arg max}_{y^{(i)}} & \text{Tr} \left( \sum_{j=1}^{n} h^{(i)}_{jj} (y^{(i)}_j - y^{(i)}_i) (y^{(i)}_j - y^{(i)}_i)^T \right) \\
= \text{arg max}_{y^{(i)}} & \text{Tr} \left( \sum_{j=1}^{n} h^{(i)}_{jj} (y^{(i)}_j)^T (y^{(i)}_j)^T - (y^{(i)}_i)^T (y^{(i)}_i)^T \right) \\
= \text{arg max} & 2 \text{Tr} \left( Y^{(i)} D^{(i)} (Y^{(i)})^T - Y^{(i)} H^{(i)} (Y^{(i)})^T \right) \\
= \text{arg max} & 2 \text{Tr} \left( Y^{(i)} L^{(i)} (Y^{(i)})^T \right) \\
= \text{arg max} & 2 \text{Tr} \left( U^T X^{(i)} L^{(i)} (X^{(i)})^T U \right),
\end{align*}
$$

(1)

where $\text{Tr}(\cdot)$ denotes the trace operation of matrix, $L^{(i)} = D^{(i)} - H^{(i)}$ is the graph Laplacian on the $i$th view, $D^{(i)}$ is a diagonal matrix with its element $D^{i}_{jj} = \sum_j H^{i}_{jj}$ on the $i$th view, and $H^{i} = [h^{i}_{jj}]_{n \times n}$ is the weighting matrix which encodes both the distance weighting information and the class label information on the $i$th view

$$
h^{i}_{jj} = \begin{cases} 
\frac{1}{n} \left( 1 - \frac{1}{n_i} \right), & \text{if } l_j = l_i = l \\
\frac{1}{n_i}, & \text{if } l_j \neq l_i,
\end{cases}
$$

(2)

where in $l_j$ is the class label of sample $x^{(i)}_j$, $n_i$ is the number of samples belonging to the $l_i$th class, and $c_j$ is set as $\exp(-\|x^{(i)}_j - x^{(i)}_i\|^2/\sigma^2)$ according to LPP [7] for locality preservation.

Second, we try to implement the local geometry preservation by assuming that each sample $x^{(i)}_j$ can be linearly 
reconstructed by the samples $x^{(i)}_k$ which share the same class 
label with $x^{(i)}_j$ on the $i$th view. Thus, we can obtain the reconstruction 
coefficient $w^{(i)}_j$ by minimizing the reconstruction error

$$
\begin{align*}
\text{arg min}_{w^{(i)}} & \sum_{j=1}^{n} \| e_j \|^2 \\
= \text{arg min}_{w^{(i)}} & \sum_{j=1}^{n} \| x^{(i)}_j - \sum_{l \neq l_i} w^{(i)}_j x^{(i)}_j \|^2
\end{align*}
$$

(3)
under the constraint
\[
\sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} = 1, \quad w_{jt}^{(i)} = 0 \quad \text{for} \ i \neq j,
\]  
(4)

Then, by solving (3) and (4), we have
\[
w_{jt}^{(i)} = \frac{\sum_p C_{j,p}^{(i)}}{\sum_{p,q} C_{p,q}^{(i)}}
\]
(5)

where \(C_{p,q} = (x_p^{(i)} - x_q^{(i)})^T(x_p^{(i)} - x_q^{(i)})\) denotes the local Gram matrix and \(l_p = l_q = l\).

Once obtaining the reconstruction coefficient \(w_{jt}^{(i)}\) on the \(i\)th view, then MDGPP aims to reconstruct \(y_j^{(i)} = U^T x_j^{(i)}\) from \(y_j^{(i)} = U^T x_j^{(i)}\) (where \(l_i = l_j\)) with \(w_{jt}^{(i)}\) in the projected low-dimensional space; thus we have
\[
\arg\min_{y_j^{(i)}} \sum_{j=1}^n \left\| y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} \right\|^2
\]
(6)
\[
= \arg\min_{y_j^{(i)}} \sum_{j=1}^n \left\| y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} \right\|^2
\]
\[
= \arg\min_{y_j^{(i)}} \text{Tr} \left( \sum_{j=1}^n \left( y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} \right)^T \right)
\]
\[
= \arg\min_{y_j^{(i)}} 2\text{Tr} \left( y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} \right)^T y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} - y_j^{(i)} - \lambda \sum_{j=1}^n \left\| y_j^{(i)} - \sum_{t \in \mathcal{L}_j} w_{jt}^{(i)} y_j^{(i)} \right\|^2
\]
(7)

where \(Q^{(i)} = X^{(i)} L^{(i)} X^{(i)^T} - \lambda X^{(i)} (I^{(i)} - W^{(i)}) (X^{(i)^T} U)^T\), and \(\lambda\) is a tradeoff coefficient which is empirically set as 1 in this experiment.

Based on the local manifold information encoded in \(L^{(i)}\) and \(W^{(i)}\), (7) aims at finding a sufficiently smooth low-dimensional embedding \(Y^{(i)} = U^T X^{(i)}\) by preserving the interclass discrimination and intraclass geometry on the \(i\)th view.

Because multiview could provide complementary information in characterizing data from different viewpoints, different views certainly have different contributions to the low-dimensional feature subspace. In order to well discover the complementary information of data from different views, a nonnegative weighted set \(\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_m]\) is imposed on each view independently. Generally speaking, the larger \(\sigma_i\) is, the more the contribution of the view \(X^{(i)}\) is made to obtain the low-dimensional feature subspace. Hence, by summing over all parts of optimization defined in (7), we can formulate MDGPP as the following optimization problem:
\[
\arg\max_{U,\sigma} \sum_{i=1}^m \sigma_i \text{Tr} \left( U^T Q^{(i)} U \right)
\]
(8)
subject to
\[
U^T U = I, \quad \sum_{i=1}^m \sigma_i = 1, \quad \sigma_i \geq 0.
\]
(9)

The solution to \(\sigma\) in (8) subject to (9) is \(\sigma_k = 1\) corresponding to the maximum \(\text{Tr}(U^T Q^{(i)} U)\) over different views, and \(\sigma_k = 0\) otherwise, which means that only the best view is finally selected by this method. Consequently, this solution cannot meet the demand for exploring the complementary characteristics of different views to get a better low-dimensional embedding than that based on a single view. In order to avoid this problem, we set \(\sigma_i \leftarrow \sigma_i^r\) with \(r > 1\) by following the trick utilized in [16–19]. In this condition, \(\sum_{i=1}^m \sigma_i^r = 1\) achieves its maximum when \(\sigma_i = 1/m\) according to \(\sum_{i=1}^m \sigma_i = 1\) and \(\sigma_i \geq 0\). Similarly \(\sigma_i\) for different views can be obtained by setting \(r > 1\); thus each view makes a specific contribution to obtaining the final low-dimensional embedding. Consequently, the new objective function of MDGPP can be defined as follows:
\[
\arg\max_{U,\sigma} \sum_{i=1}^m \sigma_i^r \text{Tr} \left( U^T Q^{(i)} U \right)
\]  
(10)
subject to
\[ U^T U = I, \quad \sum_{i=1}^{m} \sigma_i = 1, \quad \sigma_i \geq 0. \] (11)

The above optimization problem is a nonlinearly constrained nonconvex optimization problem, so there is no direct approach to find its global optimal solution. In this paper, we derive an alternating-optimization-based iterative algorithm to find a local optimal solution. The alternating optimization iteratively updates the projection matrix \( U \) and weight vector \( \sigma = [\sigma_1, \sigma_2, \ldots, \sigma_m] \).

First, we update \( \sigma \) by fixing \( U \). The optimal problem (10) subject to (11) becomes
\[
\arg \max_{U, \sigma} \sum_{i=1}^{m} \sigma_i^r \operatorname{Tr}(U^TQ^{(i)}U) \] (12)
subject to
\[ \sum_{i=1}^{m} \sigma_i = 1, \quad \sigma_i \geq 0. \] (13)

Following the standard Lagrange multiplier, we construct the following Lagrangian function by incorporating the constraint (13) into (12):
\[
L(\sigma, \lambda) = \sum_{i=1}^{m} \sigma_i^r \operatorname{Tr}(U^TQ^{(i)}U) - \lambda \left( \sum_{i=1}^{m} \sigma_i - 1 \right), \] (14)
where the Lagrange multiplier \( \lambda \) satisfies \( \lambda \geq 0 \).

Taking the partial derivation of the Lagrangian function \( L(\sigma, \lambda) \) with respect to \( \sigma_i \) and \( \lambda \) and setting them to zeros, we have
\[
\frac{\partial L(\sigma, \lambda)}{\partial \sigma_i} = r \sigma_i^{r-1} \operatorname{Tr}(U^TQ^{(i)}U) - \lambda = 0, \] (15)
\[
\frac{\partial L(\sigma, \lambda)}{\partial \lambda} = \sum_{i=1}^{m} \sigma_i - 1 = 0. \] (16)

Hence, according to (13) and (14), the weight coefficient \( \sigma_i \) can be calculated as
\[
\sigma_i = \left( \frac{1}{\operatorname{Tr}((U^TQ^{(i)}U))^{1/(r-1)}} \right) \sum_{i=1}^{m} \left( \frac{1}{\operatorname{Tr}((U^TQ^{(i)}U))^{1/(r-1)}} \right)^{1/r}. \] (17)

Then, we can make the following observations according to (15): if \( r \to \infty \), then the values of different \( \sigma_i \) will be close to each other. If \( r \to 1 \), then only \( \sigma_i = 1 \) corresponding to the maximum \( \operatorname{Tr}(U^TQ^{(i)}U) \) over different views, and \( \sigma_i = 0 \) otherwise. Thus, the choice of \( r \) should respect to the complementary property of different views. The effect of the parameter \( r \) will be discussed in the later experiments.

Second, we update \( U \) by fixing \( \sigma \). The optimal problem (10) subject to (11) can be equivalently transformed into the following form:
\[
\arg \max_{U} \operatorname{Tr}(U^TQU) \] (18)
subject to
\[ U^T U = I, \] (19)

where \( Q = \sum_{i=1}^{m} Q^{(i)} \). Since \( Q^{(i)} \) defined in (7) is a symmetric matrix, \( Q \) is also a symmetric matrix.

Obviously, the solution of (18) subject to (19) can be obtained by solving the following standard eigendecomposition problem
\[ QU = \lambda U. \] (20)

Let the eigenvectors \( U_1, U_2, \ldots, U_d \) be solutions of (20) ordered according to eigenvalues \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \). Then, the optimal projection matrix \( U \) is given by \( U = [U_1, U_2, \ldots, U_d] \). Now, we discuss how to determine the reduced feature dimension \( d \) by using the Ky Fan theorem [23].

Ky Fan Theorem. Let \( H \) be a symmetric matrix with eigenvalues \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \) and the corresponding eigenvectors \( U = [U_1, U_2, \ldots, U_n] \). Then
\[
\lambda_1 + \lambda_2 + \cdots + \lambda_k = \arg \max_{X^T=X} \text{Tr}(X^T H X). \] (21)

Moreover, the optimal \( X^* \) is given by \( X^* = U = [U_1, U_2, \ldots, U_k] R \), where \( R \) is an arbitrary orthogonal matrix.

From the above Ky Fan theorem, we can make the following observations. The optimal solution to (18) subject to (19) is composed of the largest \( d \) eigenvectors of the matrix \( Q \), and the optimal value of objective function (18) equals the sum of the largest \( d \) eigenvalues of the matrix \( Q \). Therefore, the optimal reduced feature dimension \( d \) is equivalent to the number of positive eigenvalues of the matrix \( Q \).

Alternately updating \( \sigma \) and \( U \) by solving (17) and (20) until convergence, we can obtain the final optimal projection matrix \( U \) for multiple views. A simple initialization for \( \sigma \) could be \( \sigma = \frac{1}{m}, \ldots, \frac{1}{m} \). According to the aforementioned statement, the proposed MDGPP algorithm is summarized as follows.

Algorithm 1 (MDGPP algorithm).

Input. A multiview data set \( X = \{X^{(i)} \in R^{p_i \times m} \}_{i=1}^{m} \), the dimension of the reduced low-dimensional subspace \( d < p_i, 1 \leq i \leq m \), tuning parameter \( r > 1 \), iteration number \( T_{\text{max}} \), and convergence error \( \varepsilon \).

Output. Projection matrix \( U \).

Algorithm.

Step 1. Simultaneously consider both intraclass geometry and interclass discrimination information to calculate \( Q^{(i)} \) for each view according to (7).

Step 2 (initialization).

(1) Set \( \sigma = \frac{1}{m}, \frac{1}{m}, \ldots, \frac{1}{m} \);
(2) obtain \( U^0 \) by solving the eigendecomposition problem (20).
Step 3 (local optimization). For $t = 1, 2, \ldots, T_{\text{max}}$

1. calculate $\sigma$ as shown in (17);
2. solve the eigenvalue equation in (20);
3. sort their eigenvectors $U_1, U_2, \ldots, U_d$ according to their corresponding eigenvalues: $\lambda_1 > \lambda_2 > \cdots > \lambda_d$,
   and obtain $U^t = [U_1, U_2, \ldots, U_d]$;
4. if $t > 2$ and $|U^t - U^{t-1}| < \epsilon$, then go to Step 4.

Step 4 (output projection matrix). Output the final optimal projection matrix $U = U^t$.

We now briefly analyze the computational complexity of the MDGPP algorithm, which is dominated by three parts. One is for constructing the matrix $Q^0$ for different views. As shown in (7), the computational complexity of this part is $O((\sum_{i=1}^{m} p_i) \times n^2)$. In addition, each iteration involves computing view weight $\sigma$ and solving a standard eigendecomposition problem; the computational complexity of running two parts in each iteration is $O((m + d) \times n^2)$ and $O(n^3)$, respectively. Therefore, the total computational complexity of MDGPP is $O((\sum_{i=1}^{m} p_i) \times n^2 + ((m + d) \times n^2 + n^3) \times T_{\text{max}})$, where $T_{\text{max}}$ denotes the iteration number and is always set to less than five in all experiments.

4. Experimental Results

In this section, we evaluate the effectiveness of our proposed MDGPP algorithm for two image classification tasks including face recognition and facial expression recognition. Two widely used face databases including AR [24] and CMU PIE [25] are employed for face recognition evaluation, and the well-known Japanese female facial expression (JAFFE) [26] database is used for facial expression recognition. We also compare the proposed MDGPP algorithm with some traditional single-view-based dimensionality reduction algorithms, such as PCA [2], LDA [2], LPP [3], MFA [10], DGPP [12], and the three latest multiview dimensionality reduction algorithms, including MVSE [18], MSNE [21], MSE [19], and SSMVE [22]. The nearest neighbor classifier with eigendecomposition problem; the computational complexity is $O((\sum_{i=1}^{m} p_i) \times n^2 + ((m + d) \times n^2 + n^3) \times T_{\text{max}})$.

4.1. Data Sets and Experimental Settings. We conducted face recognition experiments on the widely used AR and CMU PIE face databases and facial expression recognition experiments on the well-known Japanese female facial expression (JAFFE) database.

The AR database [24] contains over 4,000 color images corresponding to 126 people (70 men and 56 women), which include frontal views with different facial expressions, illumination conditions, and occlusions (sun glasses and scarf). Each person has 26 different images taken in two sessions (separated by two weeks). In our experiments, we used a subset of 800 face images from 100 persons (50 men and 50 women) with eight face images of different expressions and lighting conditions per person. Figure 1 shows eight sample images of one individual from the subset of the AR database.

The CMU PIE database [25] comprises more than 40,000 facial images of 68 people with different poses, illumination conditions, and facial expressions. In this experiment, we selected a subset of the CMU PIE database which consists of 3060 frontal face images with varying expression and illumination from 68 persons with 45 images from each person. Figure 2 shows some sample images of one individual from the subset of the CMU PIE database.

The Japanese female facial expression (JAFFE) database [26] contains 213 facial images of ten Japanese women. Each facial image shows one of seven expressions: neutral, happiness, sadness, surprise, anger, disgust, or fear. Figure 3 shows some facial images from the JAFFE database. In this experiment, following the general setting scheme of facial expression recognition, we discard all the neutral facial images and only utilize the remainder 183 facial images which include six basic facial expressions.

For all the face images in the above three face databases, the facial part of each image was manually aligned, cropped, and resized into $32 \times 32$ according to the eye's positions. For each facial image, we extract the commonly used four kinds of low-level visual features to represent four different views. These four features include color histogram (CH) [27], scale-invariant feature transform (SIFT) [28], Gabor [29], and local binary pattern (LBP) [30]. For the CH feature extraction, we used 64 bins to encode a histogram feature for each facial image according to [27]. For the SIFT feature extraction, we densely sampled and calculated the SIFT descriptors of $16 \times 16$ patches over a grid with spacing of 8 pixels according to [28]. For the Gabor feature extraction, following [29], we adopted 40 Gabor kernel functions from five scales and eight orientations. For the LBP feature extraction, we followed the parameter settings in [30] and utilized 256 bins to encode a histogram feature for each facial image. For more details on these four feature descriptors, please refer to [27–30]. Because these four features are complementary to each other in representing facial images, we empirically set the tuning parameter $r$ in MDGPP to be five.

In this experiment, each facial image set was partitioned into the nonoverlap training and testing sets. For each database, we randomly selected 50% data as the training set and use the remaining 50% data as the testing set. To reduce statistical variation for each random partition, we repeated these trials independently ten times and reported the average recognition results.
4.2. Compared Algorithms. We compared our proposed MDGPP algorithm with the following dimensionality reduction algorithms.

(1) PCA [2]: PCA is an unsupervised dimensionality reduction algorithm.

(2) LDA [2]: LDA is a supervised dimensionality reduction algorithm. We adopted a Tikhonov regularization term $\mu I$ rather than PCA preprocessing to avoid the well-known small sample size (singularity) problem in LDA.

(3) LPP [3]: LPP is an unsupervised manifold learning algorithm. There is a nearest neighbor number $k$ to be tuned in LPP and it was empirically set to be five in our experiments. In addition, the Tikhonov regularization was also adopted to avoid the small sample size (singularity) problem in LPP.

(4) MFA [10]: MFA is a supervised manifold learning algorithm. There are two parameters (i.e., $k_1$ nearest neighbor number and $k_2$ nearest neighbor number) to be tuned in MFA. We empirically set $k_1 = 5$ and $k_2 = 20$ in our experiments. Meanwhile, the Tikhonov regularization was also adopted to avoid the small sample size (singularity) problem in MFA.

(5) DGPP [12]: DGPP is a supervised manifold learning algorithm. There is a tradeoff parameter $\lambda$ to be tuned in DGPP and it was empirically set to be one in our experiments.

(6) MVSE [18]: MSVE is an initially proposed multiview algorithm for dimensionality reduction.

(7) MSNE [21]: MSNE is a probability-based unsupervised multiview algorithm. We followed the parameter setting in [21] and set the tradeoff coefficient $\lambda$ to be five in our experiments.

(8) MSE [19]: MSE is a supervised multiview algorithm. There are two parameters (i.e., the nearest neighbor number $k$ and the tuning coefficient $r$) to be tuned in MSE. We empirically set $k = 5$ and $r = 5$ in our experiments.

(9) SSMVE [22]: SSMVE is a sparse unsupervised multiview algorithm. We followed the parameter setting method in [22] and set the regularized parameter $\lambda$ to be one in our experiments.

It is worth noting that since PCA, LDA, LPP, MFA, and DGPP are all single-view-based algorithms, these five algorithms adopt the conventional feature concatenation-based strategy to cope with the multiview data.

4.3. Experimental Results. For each face image database, the recognition performance of different algorithms was evaluated on the testing data separately. The conventional nearest neighbor classifier with the Euclidean distance was applied to perform recognition in the subspace derived from different dimensionality reduction algorithms. Tables 1, 2, and 3 report the recognition accuracies and the corresponding optimal dimensions obtained on the AR, CMU PIE, and JAFFE databases, respectively. Figures 4, 5, and 6 illustrate the recognition accuracies versus the variation of reduced dimensions on the AR, CMU PIE, and JAFFE databases.
respective. According to the above experimental results, we can make the following observations.

(1) As can be seen from Tables 1, 2, and 3 and Figures 4, 5, and 6, our proposed MDGPP algorithm consistently outperforms the conventional single-view-based algorithms (i.e., PCA, LDA, LPP, MFA, and DGPP) and the latest multiview algorithms (i.e., MVSE, MSNE, MSE, and SSMVE) in all the experiments, which implies that extracting a discriminative feature subspace by using both intraclass geometry and interclass discrimination and explicitly considering the complementary information of different facial features can achieve the best recognition performance.

(2) The multiview learning algorithms (i.e., MVSE, MSNE, MSE, SSMVE, and MDGPP) perform much better than single-view-based algorithms (i.e., PCA, LDA, LPP, MFA, and DGPP), which demonstrates that simple concatenation strategy cannot duly combine features from multiple views, and the recognition performance can be successively improved by exploring the complementary characteristics of different views.
Figure 8: Recognition accuracy of MDGPP versus different numbers of iteration on the CMU PIE database.

Figure 9: Recognition accuracy of MDGPP versus different numbers of iteration on the JAFFE database.

Figure 10: Recognition accuracy of MDGPP versus varying parameter $\lambda$ on the AR database.

Figure 11: Recognition accuracy of MDGPP versus varying parameter $\lambda$ on the CMU PIE database.

Figure 12: Recognition accuracy of MDGPP versus varying parameter $\lambda$ on the JAFFE database.

Figure 13: Recognition accuracy of MDGPP versus varying parameter $r$ on the AR database.
(5) Although MVSE, MSNE, and SSMVE are all unsupervised multiview learning algorithms, SSMVE performs much better than MVSE and MSNE. The possible explanation is that the SSMVE algorithm adopts the sparse coding technique, which is naturally discriminative in determining the appropriate combination of different views.

(6) Among the compared multiview learning algorithms, MVSE performs the worst. The reason is that MVSE performs a dimensionality reduction process on each view independently. Hence it cannot fully integrate the complementary information of different views to produce a good low-dimensional embedding.

(7) MDGPP can improve the recognition performance of DGPP. The reason is that MDGPP can make use of multiple facial feature representations in a common learned subspace such that some complementary information can be explored for recognition task.

4.4. Convergence Analysis. Since our proposed MDGPP is an iteration algorithm, we also evaluate its recognition performance with different numbers of iteration. Figures 7, 8, and 9 show the recognition accuracy of MDGPP versus different numbers of iteration on the AR, CMU PIE, and JAFFE databases, respectively. As can be seen from these figures, we can observe that our proposed MDGPP algorithm can converge to a local optimal optimum value in less than five iterations.

4.5. Parameter Analysis. We investigate the parameter effects of our proposed MDGPP algorithm: tradeoff coefficient $\lambda$ and tuning parameter $r$. Since each parameter can affect the recognition performance, we fix one parameter as used in the previous experiments and test the effect of the remaining one. Figures 10, 11, and 12 show the influence of the parameter $\lambda$ in the MDGPP algorithm on the AR, CMU PIE, and JAFFE databases, respectively. Figures 13, 14, and 15 show the influence of the parameter $r$ in the MDGPP algorithm on the AR, CMU PIE, and JAFFE databases, respectively. From Figure 10 to Figure 15, we can observe that MDGPP demonstrates a stable recognition performance over a large range of both $\lambda$ and $r$. Therefore, we can conclude that the performance of MDGPP is not sensitive to the parameters $\lambda$ and $r$.

5. Conclusion

In this paper, we have proposed a new multiview learning algorithm, called multiview discriminative geometry preserving projection (MDGPP) for feature extraction and classification by exploring the complementary property of different views. MDGPP can encode different features from different views in a physically meaningful subspace and learn a low-dimensional and sufficiently smooth embedding over all views simultaneously with an alternating-optimization-based iterative algorithm. Experimental results on three face image databases show that the proposed MDGPP algorithm
outperforms other multiview and single view learning algorithms.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grants no. 70701013 and 6174056, the Natural Science Foundation of Henan Province under Grant no. 102300410020, the National Science Foundation for Postdoctoral Scientists of China under Grant no. 2011M500035, and the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant no. 20110023110002.

References

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: a literature survey,” ACM Computing Surveys, vol. 35, no. 4, pp. 399–458, 2003.
[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. fisherfaces: recognition using class specific linear projection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711–720, 1997.
[3] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, “Face recognition using Laplacianfaces,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 3, pp. 328–340, 2005.
[4] 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.
[5] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” Science, vol. 290, no. 5500, pp. 2323–2326, 2000.
[6] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” Neural Computation, vol. 15, no. 6, pp. 1373–1396, 2003.
[7] X. He and P. Niyogi, “Locality preserving projections,” Advances in Neural Information Processing Systems, vol. 16, pp. 585–591, 2003.
[8] H.-T. Chen, H.-W. Chang, and T.-L. Liu, “Local discriminant embedding and its variants,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR ’05), pp. 846–853, San Diego, Calif, USA, June 2005.
[9] D. Cai, X. He, K. Zhou, J. W. Han, and H. J. Bao, “Locality sensitive discriminant analysis,” in Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI ’07), pp. 708–713, Hyderabad, India, January 2007.
[10] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: a general framework for dimensionality reduction,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 1, pp. 40–51, 2007.
[11] M. Sugiyama, “Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis,” Journal of Machine Learning Research, vol. 8, pp. 1027–1061, 2007.