Supplementary Material for Manifold Partition Discriminant Analysis

Yang Zhou and Shiliang Sun

1 Detailed Derivation of S

By representing S as a block matrix, the within-class objective function becomes:

$$\min_{\boldsymbol{t},\boldsymbol{v}} \begin{pmatrix} \boldsymbol{t} \\ \boldsymbol{v} \end{pmatrix}^\top \begin{pmatrix} S_1 & S_2 \\ S_2^\top & S_3 \end{pmatrix} \begin{pmatrix} \boldsymbol{t} \\ \boldsymbol{v} \end{pmatrix} = \boldsymbol{f}^\top S \boldsymbol{f}. \quad (1)$$

In order to fix S, we decompose (1) into four additive terms as follows:

$$\begin{aligned} \boldsymbol{f}^\top S \boldsymbol{f} = & \underbrace{\sum_{i,j=1}^{n} W_{ij}((\boldsymbol{x}_i - \boldsymbol{x}_j)^\top \boldsymbol{t})^2}_{\text{term one}} + \\ & \underbrace{\sum_{i,j=1}^{n} W_{ij}\big(\boldsymbol{v}_{\pi_j}^\top T_{\pi_j}^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\big)^2}_{\text{term two}} + \\ & \underbrace{\sum_{i,j=1}^{n} W_{ij}\big[-2((\boldsymbol{x}_i - \boldsymbol{x}_j)^\top \boldsymbol{t})\boldsymbol{v}_{\pi_j}^\top T_{\pi_j}^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\big]}_{\text{term three}} + \\ & \underbrace{\gamma \sum_{i,j=1}^{n} W_{ij} \|\boldsymbol{v}_{\pi_i} - T_{\pi_i} T_{\pi_j}^\top \boldsymbol{v}_{\pi_j}\|_2^2}_{\text{term four}}, \end{aligned}$$

and examine their separate contributions to the whole S_p.

Term One

$$\begin{aligned} & \sum_{i,j=1}^{n} W_{ij}((\boldsymbol{x}_i - \boldsymbol{x}_j)^\top \boldsymbol{t})^2 \\ =& 2\boldsymbol{t}^\top X(D - W)X^\top \boldsymbol{t} = 2\boldsymbol{t}^\top X L X^\top \boldsymbol{t}, \end{aligned}$$

where D is a diagonal weight matrix with $D_{ii} = \sum_{j=1}^{n} W_{ij}$, and $L = D - W$ is the Laplacian matrix. Thus term one contributes to S_1 in (1).

- *Yang Zhou and Shiliang Sun (corresponding author) are with Shanghai Key Laboratory of Multidimensional Information Processing, Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, P. R. China (email: shiliangsun@gmail.com, slsun@cs.ecnu.edu.cn)*

Term Two

Define $B_{\pi_j i} = T_{\pi_j}^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)$. Then

$$\begin{aligned} & \sum_{i,j=1}^{n} W_{ij}\big(\boldsymbol{v}_{\pi_j}^\top T_{\pi_j}^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\big)^2 \\ = & \sum_{i,j=1}^{n} W_{ij}(\boldsymbol{v}_{\pi_j}^\top B_{\pi_j i})^2 = \sum_{j=1}^{n} \boldsymbol{v}_{\pi_j}^\top \big(\sum_{i=1}^{n} W_{ij} B_{\pi_j i} B_{\pi_j i}^\top\big)\boldsymbol{v}_{\pi_j}. \end{aligned}$$

Let $\Pi_p = \{i | \pi_i = p, i \in \{1, \ldots, n\}\}$ be a set that consists of the indices of the data belonging to the p-th linear subspace. Then we can group the terms with respect to $\boldsymbol{v}_{\pi_j}$ $(j = 1, \ldots, n)$ into P terms as follows:

$$\begin{aligned} & \sum_{j=1}^{n} \boldsymbol{v}_{\pi_j}^\top \big(\sum_{i=1}^{n} W_{ij} B_{\pi_j i} B_{\pi_j i}^\top\big)\boldsymbol{v}_{\pi_j} \quad (2) \\ = & \sum_{p=1}^{P} \boldsymbol{v}_p^\top \big(\sum_{j \in \Pi_p} H_j\big)\boldsymbol{v}_p, \end{aligned}$$

where we have defined matrices $\{H_j\}_{j=1}^n$ with $H_j = \sum_{i=1}^{n} W_{ij} B_{\pi_j i} B_{\pi_j i}^\top$.

Now we can define a block diagonal matrix S_3^H sized $mP \times mP$, where the block size is $m \times m$. Set the (i,i)-th block $(i = 1, \ldots, P)$ of S_3^H to be $\sum_{j \in \Pi_p} H_j$. Then the resultant S_3^H is the contribution of term two for S_3 in (1).

Term Three

Define vectors $\{F_p\}_{p=1}^P$ with $F_p = \sum_{i=1}^{n} \sum_{j \in \Pi_p} W_{ij} B_{\pi_j i} \boldsymbol{x}_i^\top$. Then term three can be

rewritten as:

$$\begin{aligned}
&\sum_{i,j=1}^{n} W_{ij}\big[-2((\boldsymbol{x}_i-\boldsymbol{x}_j)^\top \boldsymbol{t})\boldsymbol{v}_{\pi_j}^\top T_{\pi_j}^\top(\boldsymbol{x}_i-\boldsymbol{x}_j)\big]\\
=&\sum_{i,j=1}^{n} 2W_{ij}\big[((\boldsymbol{x}_j-\boldsymbol{x}_i)^\top \boldsymbol{t})\boldsymbol{v}_{\pi_j}^\top B_{\pi_j i}\big]\\
=&\sum_{p=1}^{P}\boldsymbol{t}^\top(\sum_{i=1}^{n}\sum_{j\in\Pi_p} -W_{ij}\boldsymbol{x}_i B_{\pi_j i}^\top)\boldsymbol{v}_p+\\
&\sum_{p=1}^{P}\boldsymbol{v}_p^\top(\sum_{i=1}^{n}\sum_{j\in\Pi_p} -W_{ij}B_{\pi_j i}\boldsymbol{x}_i^\top)\boldsymbol{t}+\\
&\sum_{p=1}^{P}\boldsymbol{t}^\top F_p^\top \boldsymbol{v}_p+\sum_{p=1}^{P}\boldsymbol{v}_p^\top F_p\boldsymbol{t}.
\end{aligned}$$

From this expression, we can give the formulation of S_2. Then the block matrix $S_2^\top$ in (1), which is its transpose, is ready to get.

Suppose we define two block matrices S_2^1 and S_2^2 sized $d\times mP$ each where the block size is $d\times m$, and S_2^2 is a block diagonal matrix. Set the p-th block $(p=1,\ldots,P)$ of S_2^1 to be $\sum_{i=1}^n\sum_{j\in\Pi_p} -W_{ij}\boldsymbol{x}_i B_{\pi_j i}^\top$, and the (p,p)-th block $(p=1,\ldots,P)$ of S_2^2 to be $F_p^\top$. Then, term three can be rewritten as: $\boldsymbol{t}^\top(S_2^1+S_2^2)\boldsymbol{v}+\boldsymbol{v}^\top(S_2^1+S_2^2)^\top\boldsymbol{t}$. It is clear that $S_2=S_2^1+S_2^2$.

Term Four

Denote matrix $T_{\pi_i}T_{\pi_j}^\top$ by $A_{\pi_i\pi_j}$. Then, with γ omitted temporarily,

$$\begin{aligned}
&\sum_{i,j=1}^{n} W_{ij}\|\boldsymbol{v}_{\pi_i}-T_{\pi_i}T_{\pi_j}^\top\boldsymbol{v}_{\pi_j}\|_2^2\\
=&\sum_{i,j=1}^{n} W_{ij}\boldsymbol{v}_{\pi_j}^\top A_{\pi_i\pi_j}^\top A_{\pi_i\pi_j}\boldsymbol{v}_{\pi_j}+\\
&\sum_{i,j=1}^{n} W_{ij}\boldsymbol{v}_{\pi_i}^\top\boldsymbol{v}_{\pi_i}-\sum_{i,j=1}^{n}2W_{ij}\boldsymbol{v}_{\pi_i}^\top A_{\pi_i\pi_j}\boldsymbol{v}_{\pi_j}.
\end{aligned}$$

Similarly, we can sum up the terms with respect to $\boldsymbol{v}_{\pi_j}$, which further leads to:

$$\begin{aligned}
&\sum_{i,j=1}^{n} W_{ij}\boldsymbol{v}_{\pi_j}^\top A_{\pi_i\pi_j}^\top A_{\pi_i\pi_j}\boldsymbol{v}_{\pi_j}+\\
&\sum_{i=1}^{n} D_{ii}\boldsymbol{v}_{\pi_i}^\top I\boldsymbol{v}_{\pi_i}-\sum_{i,j=1}^{n}2W_{ij}\boldsymbol{v}_{\pi_i}^\top A_{\pi_i\pi_j}\boldsymbol{v}_{\pi_j}\\
=&\sum_{p=1}^{P}\boldsymbol{v}_p^\top(\sum_{j\in\Pi_p}(D_{jj}I+C_j))\boldsymbol{v}_p-\\
&\sum_{p=1}^{P}\sum_{q=1}^{P}\boldsymbol{v}_p^\top(\sum_{i\in\Pi_p}\sum_{j\in\Pi_q}2W_{ij}A_{\pi_i\pi_j})\boldsymbol{v}_q.
\end{aligned}$$

where in the third line we have defined matrices $\{C_j\}_{j=1}^n$ with $C_j=\sum_{i=1}^n W_{ij}A_{\pi_i\pi_j}^\top A_{\pi_i\pi_j}$.

Now suppose we define two block matrices S_3^1 and S_3^2 sized $mP\times mP$ each where the block size is $m\times m$, and S_3^1 is a block diagonal matrix. Set the (p,p)-th block $(p=1,\ldots,P)$ of S_3^1 to be $\sum_{j\in\Pi_p}(D_{jj}I+C_j)$, and the (p,q)-th block $(p,q=1,\ldots,P)$ of S_3^2 to be $\sum_{i\in\Pi_p}\sum_{j\in\Pi_q}2W_{ij}A_{\pi_i\pi_j}$. Then the contribution of term four for S_3 would be $\gamma(S_3^1-S_3^2)$. Further considering the contribution of term two for S_3, we finally have $S_3=S_3^H+\gamma(S_3^1-S_3^2)$.

TABLE 1
Average error rates (dimensionality) on the USPS data sets.

Methods	USPS-eo	USPS-sl
Baseline	2.72%(256)	3.25%(256)
PCA	2.47%(37.9)	2.93%(38.75)
LDA	10.93%(9)	21.25%(9)
MFA	2.79%(26.3)	3.57%(29.45)
LSDA	3.39%(26.8)	4.21%(27.1)
LFDA	7.89%(39.25)	9.84%(20.9)
LLTSA	8.16%(28.4)	8.47%(29.85)
LSDR	-	-
LPFE	4.33%(184.4)	3.61%(180.25)
PMPDA	**1.76%(28.65)**	**2.20%(31.6)**
MPDA	**1.67%(35.25)**	**2.28%(30.2)**

2 USPS Data Set

In this section, we further evaluate the effectiveness of MPDA by conducting experiments on the data sets that have been tested by the authors of its counterparts. In this case, results from the existing algorithms can be cited from the corresponding original paper for fairer comparisons. According to Table 2 in our paper, LFDA seems to be the best algorithm except for MPDR and PMPDR. Because of this, we focus on comparing MPDA with LFDA. Specifically, our experiments are conducted on two binary classification data sets created from the USPS handwritten digit data set. The first task (USPS-eo) is to separate even numbers from odd numbers, and the second task (USPS-sl) is to separate small numbers ("0" to "4") from large numbers ("5" to "9"). We randomly chose 100 data points from each digit to form both the training and test set, so that there are 1000 data points for training and testing, respectively. The strategy of model selection is the same as that in our paper. We report the best classification results obtained by each method and the corresponding dimensionality at which the results are achieved. Every experimental result is obtained from the average over 20 times, where the best method and the comparable one based on Student's t-test with a p-value of 0.05 are highlighted in bold font. The above experimental configuration is identical to the one used in LFDAs original paper [1] except for two differences. The first is that the neighborhood size k is treated as a parameter for graph construction. We determine k via 4-fold cross validation, while k is set to be 7 in LFDA's original paper [1]. The second is that we search all the possible dimensionalities of embedding subspaces to report the best classification results, whereas the results

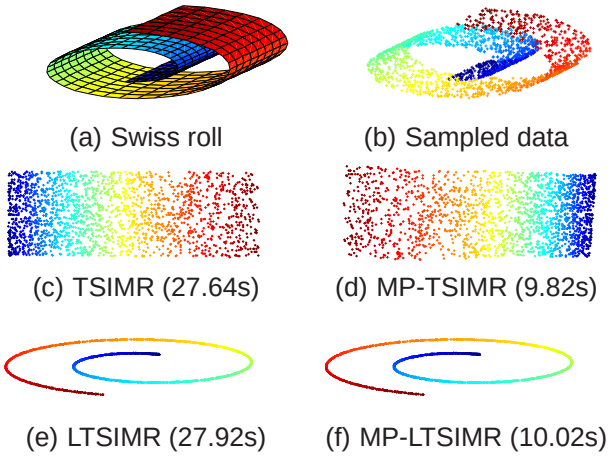


Fig. 1. The "Swiss roll" manifold and the corresponding embedding results (time) obtained by the dimensionality reduction methods with and without partitioning the manifold.

in [1] are obtained by selecting the dimensionality of embedding subspaces via 20-fold cross validation. This two differences imply that we are expected to get relatively better results than those reported in [1]. Table 1 shows that MPDA and PMPAD still outperform its counterparts with statistical significance. LSDR is not tested since the execution time is too long. It is worth noting that the reported classification results of LFDA in [1] are 9.0% for UPSP-eo and 12.9% for UPSP-sl, respectively. In our experiments, these results are improved as expected. However, they are still much worse than the results of MPDA and PMPDA.

3 Effectiveness of Partitioning the Manifold

As a crucial part of MPDA, the manifold partition strategy plays a key role in preserving the manifold structure with computational and storage efficiency. In fact, it can also be adopted by other tangent space base methods [2], [3] to make them more efficient. To verify its effectiveness, we apply the manifold partition strategy to Tangent Space Intrinsic Manifold Regularization (TSIMR) [3] and its linear variation (we call it LTSIMR). This further leads to two algorithms called MP-TSIMR and MP-LTSIMR, which can be viewed as the approximated versions of TSIMR and LTSIMR. To show the effectiveness of the manifold partition strategy, we compare the embedding results of these methods on the "Swiss roll" manifold.

The "Swiss roll" is a two-dimensional manifold in a three-dimensional ambient space as shown in Figure 1(a). The data set consists of 2000 points sampled from the manifold, which are depicted in Figure 1(b). The construction of the adjacency graph uses 10 nearest neighbors with the heat kernel. The parameter t of the heat kernel is fixed as the average of the squared distances between all points and their most nearest neighbors. The parameter γ is set to be $\gamma = 0$ for TSIMR and MP-TSIMR, and $\gamma = 10^9$ for LTSIMR and MP-LTSIMR. We fix the parameters $k' = 31$ and $M = 48$ for the manifold partitioning algorithm.

Figure 1 shows the two-dimensional embedding results obtained by different algorithms. As can be seen in Figure 1(c), TSIMR precisely reflects the intrinsic manifold structure. In addtion, Figure 1(d) shows that MP-TSIMR also gets a good embedding result besides a little distortion. In the case of linear dimensionality reduction, LTSIMR and MP-LTSIMR almost get identical results as shown in Figures 1(e) and 1(f). These embedding results demonstrate that MP-TSIMR and MP-LTSIMR have similar or the same embedding performance compared with TSIMR and LTSIMR. When it comes to the computational time, MP-TSIMR and MP-LTSIMR are three times faster than their counterparts. This is because both TSIMR and LTSIMR estimate 2000 tangent vectors and tangent spaces to discover the intrinsic manifold structure, whereas MP-TSIMR and MP-LTSIMR only need to estimate 65 tangent vectors and tangent spaces. Therefore, it is clear that the manifold partition strategy not only is useful for MPDA, but can accelerate other tangent space based methods without sacrificing their performances much.

4 Storage Overheads

Apart from high computational costs, many tangent based methods such as TSIMR [3] and PFE [2] are also storage consuming, which greatly hampers their applications. In contrast, the proposed MPDA algorithm is free from this limitation. In this section, we evaluate the storage overheads of MPDA compared with its tangent space based counterparts including PMPDA and LPFE on the COIL20, COIL100, Face Detection, MNIST, Opt-Digits, Semeion Handwritten and Vehicle data sets. The number of the training data for each data set is the same with the setting used in our paper. In order to obtain consistent results, the neighborhood size for constructing the within-class graph G is set to be $k = 7$, because the storage costs of PMPDA and LPFE directly depend on k. Figure 2 shows the peak memory costs of PMPDA, LPFE and MPDA on different data sets. As can be seen, MPDA has the least storage overheads in all cases (about 150 times and 60 times less than PMPDA and LPFE, respectively). In Figures 2(b) and 2(e), the peak memory costs of PMPDA and LPFE are extremely high as the number of data becomes relatively large, whereas MPDA still has very low memory costs. It should be noted that the above results are obtained under the condition of $k = 7$. In practice, k may be larger, say $k = 10$ or $k = 15$. In this case, the storage overheads of PMPDA and LPFE grow quickly as k becomes large. Consequently, LPFE can barely work normally, and PMPDA is no longer applicable because of out of memory. In contrast, MPDA still costs the same amount of memory because its storage overheads are independent of k. Therefore,

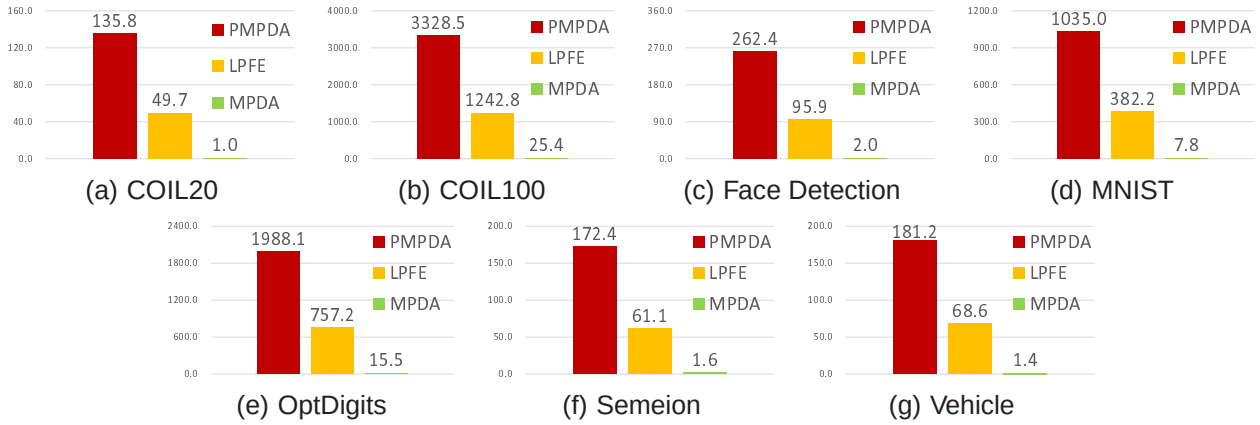


Fig. 2. Peak memory costs (Mb) of PMPDA, LPFE and MPDA on different data sets.

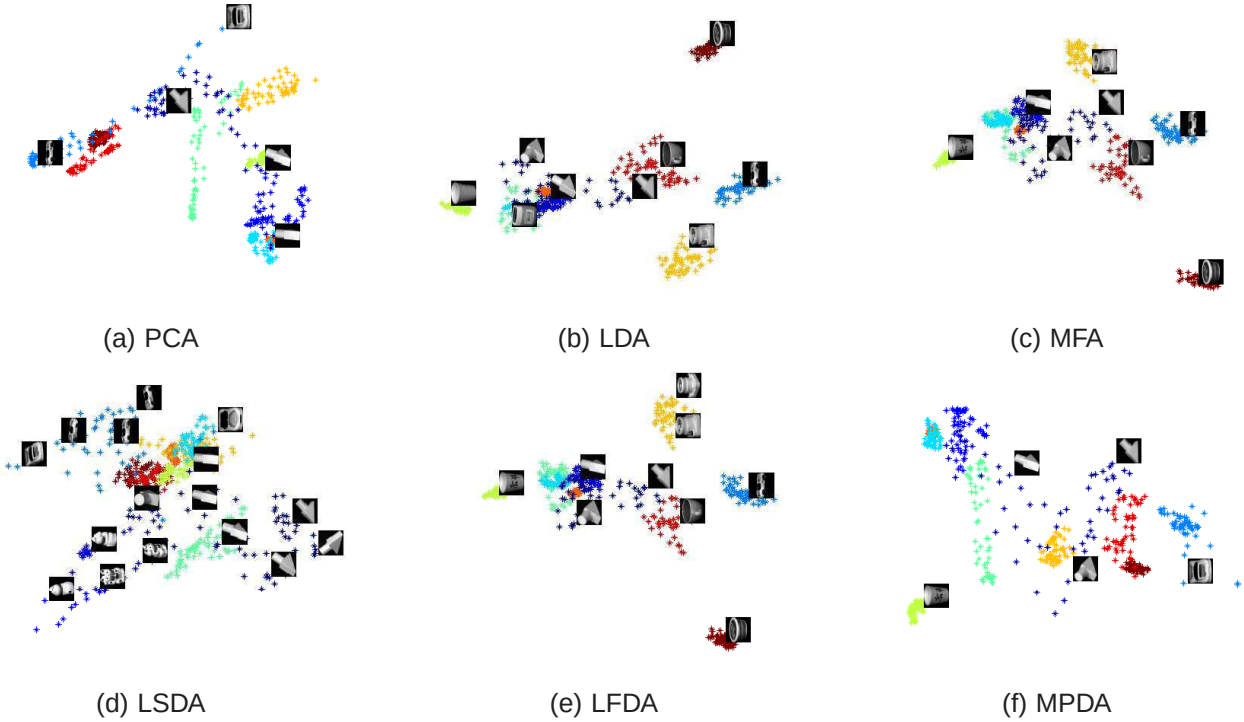


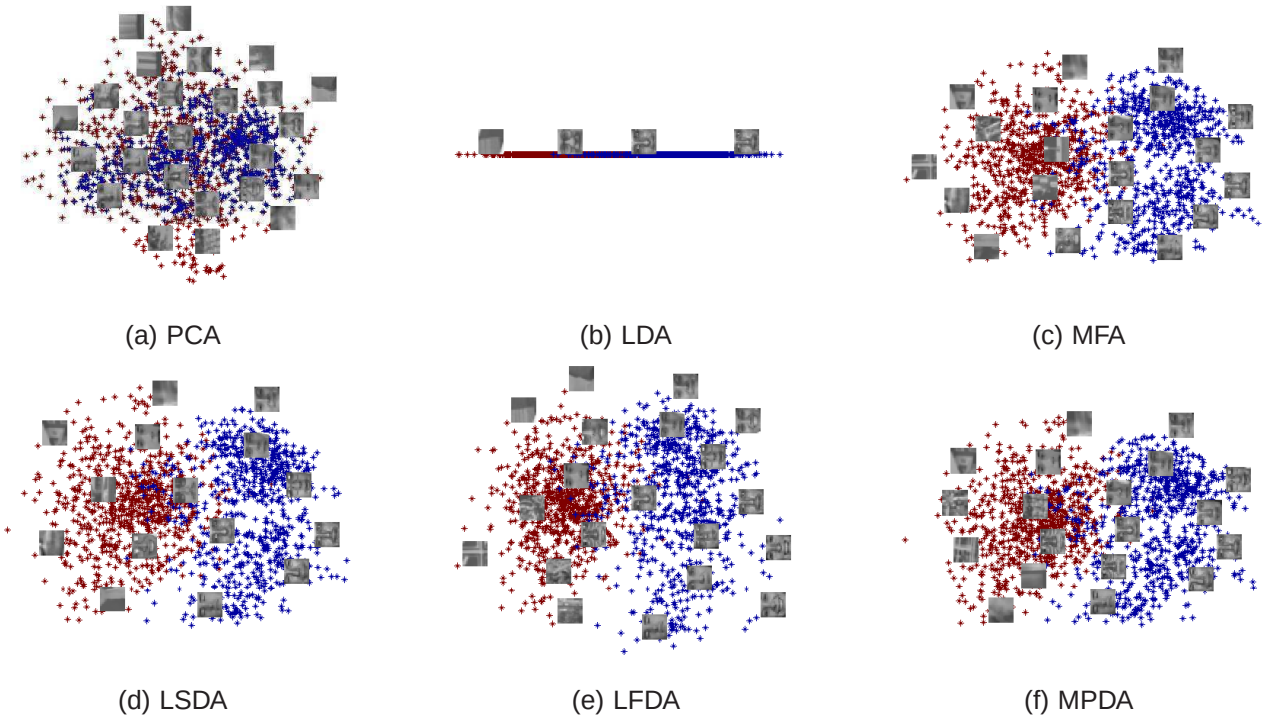
Fig. 3. Two-dimensional embedding results of the COIL20 dataset (better viewed in color).

the proposed MPDA algorithm is more scalable and practical than its counterparts.

5 2D Embedding Results

MPDA aims to find an embedding space where the manifold structure with respect to each class is preserved as much as possible, while nearby data from different classes are well separated. To verify whether this goal is achieved, we test the two-dimensional embedding results obtained by MPDA on the COIL20 and Face Detection image data sets. The data embeddings belonging to different classes are represented by different colors, and thumbnails of some images are also shown. In order to represent the data embeddings clearly, we only show the data from 10 out of 20 classes in the COIL20 data set. The embedding results obtained by other methods including PCA, LDA, MFA, LSDA and LFDA are also provided.

In Figure 3, all the methods except LSDA obtain reasonable results for the COIL20 data set. LDA, MFA and LFDA get similar embedding results because they essentially fall into the same framework [4]. Compared with other methods, the embeddings obtained by MPDA are quite different but still reasonable in gathering the data in the same class as well as separating those from different classes. As can be seen in Figure 4, the embeddings of each class obtained by PCA seriously overlap each other, because it has no ability to utilize the discriminant information from class labels. In addition, LDA also gets bad embedding results, because the Face Detection is a binary classification data set so that LDA can only map the data into a one-dimensional space. On



(a) PCA (b) LDA (c) MFA

(d) LSDA (e) LFDA (f) MPDA

Fig. 4. Two-dimensional embedding results of the Face Detection dataset (better viewed in color).

the other side, MFA, LSDA, LFDA and MPDA obtain reasonable results because they share the same spirit of gathering the data in the same class and separating those in different classes. It is worth noting that since Figures 3 and 4 only reflect the embedding performance of MPDA, we cannot draw any conclusion that whether or not the classification performance of MPDA is superior to other methods just from these figures..

REFERENCES

[1] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis," *Journal of Machine Learning Research*, vol. 8, pp. 1027–1061, 2007.

[2] B. Lin, X. He, C. Zhang, and M. Ji, "Parallel vector field embedding," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2945–2977, 2013.

[3] S. Sun, "Tangent space intrinsic manifold regularization for data representation," in *Proceedings of the IEEE China Summit and International Conference on Signal and Information Processing*, 2013, pp. 179–183.

[4] S. Yan, D. Xu, B. Zhang, H. 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.