Metric Learning with Two-Dimensional Smoothness for Visual Analysis

Xinlei Chen Zifei Tong Haifeng Liu† Deng Cai*
The State Key Lab of CAD&CG, †College of Computer Science, Zhejiang University
388 Yu Hang Tang Rd., Hangzhou, Zhejiang, China 310058
{endernewton, soariez}@gmail.com haifengliu@zju.edu.cn dengcai@cad.zju.edu.cn

Abstract

In recent years, metric learning methods based on pairwise side information have attracted considerable interests, and lots of efforts have been devoted to utilize these methods for visual analysis like content based image retrieval and face identification. When applied to image analysis, these methods merely look on an $n_1 \times n_2$ image as a vector in $\mathbb{R}^{n_1 \times n_2}$ space and the pixels of the image are considered as independent. They fail to consider the fact that an image represented in the plane is intrinsically a matrix, and pixels spatially close to each other may probably be correlated. Even though we have $n_1 \times n_2$ pixels per image, this spatial correlation suggests the real number of freedom is far less. In this paper, we introduce a regularized metric learning framework, Two-Dimensional Smooth Metric Learning (2DSML), which uses a discretized Laplacian penalty to restrict the coefficients to be two-dimensional smooth. Many existing metric learning algorithms can fit into this framework and learn a spatially smooth metric which is better for image applications than their original version. Recognition, clustering and retrieval can be then performed based on the learned metric. Experimental results on benchmark image datasets demonstrate the effectiveness of our method.

1. Introduction

In the area of computer vision and object recognition, images are usually represented as vectored data, which are easy to handle since they can be mathematically abstracted as points residing in an Euclidean space. One of the most successful example is the appearance-based method [16], where an image of size $n_1 \times n_2$ pixels are linearized to an $n_1 \times n_2$-dimensional vector. An appropriate distance metric in this space spanned by image data plays a key role in many learning algorithms, *e.g.*, k-means clustering, nearest-neighbor classification. However, commonly used distance metrics such as Euclidean distance and Mahalanobis distance are usually not very well adapted to most empirical problems, in that they cannot highlight the distinctive features for a certain task. Therefore, to apply a proper distance into these practical uses, many distance metric learning algorithms have been proposed to reveal the semantic meaning between different samples [24, 22, 7, 12, 18, 15].

Supervised distance metric learning aims to learn a distance metric from training data associated with either explicit class labels [10, 9, 22] or *side information* [23], which indicates whether two data points are similar or dissimilar in the form of pairwise constraints. Xing *et al*. [23] first formulated the problem into a constrained convex programming by minimizing the distance between similar data points under the constraint that dissimilar points are well separated. Following their work, Relevance Component Analysis (RCA) [2] seeks a distance metric minimizing the covariance matrix imposed by only the similar constraints. Information-Theoretic Metric Learning (ITML) [7] employs an information-theoretic regularization term and learns the distance metric by minimizing the Bregman divergence. Recently, Qi *et al*. [18] proposed Sparse Distance Metric Learning (SDML), by imposing a sparse prior on the off-diagonal elements of Mahalanobis matrix, which complies to the fact that the concentration matrix in high-dimensional space is often nearly sparse.

In [12], Hoi *et al*. presented the *semi-supervised* distance metric learning framework, which utilizes unlabeled data that are not originally involved in the pairwise constraints, resulting in a more generalized procedure: Laplacian Regularized Metric Learning (LRML). Baghshah and Shouraki [1] introduced a kernelized version for non-linear transformation. The idea was further extended in Semi-Supervised Sparse Metric Learning (S^3ML) [15], in which the affinity propagation technique and the sparse constraints are incorporated into the framework. With its adaptability to scarce or noisy pairwise constraints, semi-supervised distance metric learning has a broad spectrum of applications such as content-based image retrieval (CBIR) [12].

In recent years, though considerable efforts have been

*Corresponding author

978-1-4673-1228-8/12/$31.00 ©2012 IEEE

devoted to utilize the power of (semi-)supervised distance metric learning in visual analysis [12, 11], all of them consider an image as a high-dimensional vector. As a result, the pixels are considered as independent pieces of information. However, an $n_1 \times n_2$ image is intrinsically a matrix and pixels are likely to have spatial correlation to their neighborhood. Even if we have $n_1 \times n_2$ pixels per image, the spatial correlation suggests that the real number of freedom is far less [4].

On the other hand, there has been a growing interest in applying higher order smoothing approaches to image de-noising [3], image reconstruction [5], and face recognition [4]. Inspired by their work, we introduce a *Two-Dimensional Smooth Metric Learning* (2DSML) framework with a discretized Laplacian penalty to explicitly constrain the two-dimensional smoothness. Rather than considering the mapping function as an $n_1 \times n_2$-dimensional vector, we view it as a matrix, or a discrete function defined on an $n_1 \times n_2$ lattice. Thus, the discretized Laplacian [4] can be applied to the mapping functions to measure their smoothness along horizontal and vertical directions. This penalty allows us to incorporate the prior information that neighboring pixels are spatially correlated. Once the distance matrix is learned, many learning algorithms such as nearest-neighbor classification can be effectively applied.

2. Metric Learning with Pairwise Constraints

Generally, assume we have m $n_1 \times n_2$ images. Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^m \subset \mathbb{R}^n$ ($n = n_1 \times n_2$) denote their vector representations and $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m] \in \mathbb{R}^{n \times m}$. Two sets of pairwise constraints among these data points [23]:

$$\begin{aligned} \mathcal{S} &= \{(i,j) | \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are similar}\}, \\ \mathcal{D} &= \{(i,j) | \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are dissimilar}\}, \end{aligned} \tag{1}$$

are also provided, where $\mathcal{S}$ is the set of similar pairwise constraints, and $\mathcal{D}$ is the set of dissimilar pairwise constraints. Each pairwise constraint (i, j) indicates if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar or dissimilar judged by users under certain applications [23].

For any pair of points $\mathbf{x}_i$ and $\mathbf{x}_j$, the Mahalanobis distance $d_M(\mathbf{x}_i, \mathbf{x}_j)$ is:

$$\begin{aligned} d_M(\mathbf{x}_i, \mathbf{x}_j) &= \|\mathbf{x}_i - \mathbf{x}_j\|_M = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T M (\mathbf{x}_i - \mathbf{x}_j)} \\ &= \sqrt{\mathrm{Tr}\left(M(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T\right)}, \end{aligned} \tag{2}$$

where $\mathrm{Tr}(\cdot)$ stands for the trace operator, and $M \in \mathbb{R}^{n \times n}$ is a symmetric metric matrix. In general, the symmetric matrix M is a valid metric if and only if it satisfies the non-negativity and the triangle inequality properties. In other words, M must be positive semi-definite, *i.e.*, $M \succeq 0$. As a common case, when M is the identity matrix $I \in \mathbb{R}^{n \times n}$, the distance in Eq.(2) becomes the common Euclidean distance.

In practice, to learn a distance metric, one can assume that there is some linear mapping $U : \mathbb{R}^n \to \mathbb{R}^r$, where $U = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_r] \in \mathbb{R}^{n \times r}$ such that $M = UU^T$ [6]. Thus, the Mahalanobis distance between two examples can be computed as:

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \|U^T(\mathbf{x}_i - \mathbf{x}_j)\| \tag{3}$$

The goal of distance metric learning is to learn an optimal distance metric $M \succeq 0$ from a collection of data points $\mathcal{X}$, together with a set of pairwise constraints $\mathcal{S}$ and $\mathcal{D}$, based on certain loss function $f(M, \mathcal{X}, \mathcal{S}, \mathcal{D})$ [12]. Thus, formulating a proper objective function $f(\cdot)$ is crucial in metric learning.

In the following, we would like to briefly summarize the state-of-the-art objective functions $f(\cdot)$ devised for (semi-)supervised metric learning algorithms, which can be formulated in a regularization framework with different terms representing different priors.

2.1. Log-Determinant Divergence

A natural assumption of distance metric learning is that M is close to a known prior $M_0 \in \mathbb{R}^{n \times n}$. Here M_0 can be either the sample concentration matrix Σ^{-1} (Σ is the sample covariance matrix) which gives knowledge about the sample distribution, or the identity matrix I_n, which gives the most unbiased prior to the metric [18]. Based on this assumption, a distribution prior that parameterizes the Mahalanobis distance or the Euclidean distance is imposed. This term seeks to regularize M to be as close as possible to a given Mahalanobis metric M_0 by minimizing the log-determinant (or Kullback-Leibler) divergence function [7]:

$$f_{\mathcal{D}}(M) = D_g(M \| M_0) = \mathrm{Tr}(M_0^{-1} M) - \log\det(M) - n, \tag{4}$$

where the constant n can be ignored in the optimization.

2.2. Incorporating the Side Information

A natural extension of Eq.(4) is to incorporating information from both the pairwise constraints and the unlabeled data. In the literature [12, 15], this goal is accomplished by a term $f_{\mathcal{L}}(\cdot)$, which builds on a certain graph affinity matrix $W \in \mathbb{R}^{m \times m}$ that each entry w_{ij} measures the strength of similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. Specifically, $f_{\mathcal{L}}(\cdot)$ is defined as follows [12]:

$$\begin{aligned} &f_{\mathcal{L}}(M, \mathcal{X}, \mathcal{S}, \mathcal{D}) \\ =&\frac{1}{2} \sum_{i,j=1}^{n} \|\mathbf{x}_i - \mathbf{x}_j\|_M^2 w_{ij} = \mathrm{Tr}\left(XLX^T M\right), \end{aligned} \tag{5}$$

where D is a diagonal matrix with $d_{ii} = \sum_{j=1}^n w_{ij}$. $L = D - W$ is known as the *graph Laplacian*. Please see [12] for the detailed derivation.

In Laplacian Regularized Metric Learning (LRML) [12], two graph affinity matrices $W^{\mathcal{S}} \in \mathbb{R}^{m\times m}$ and $W^{\mathcal{D}} \in \mathbb{R}^{m\times m}$ are constructed to encode the side information:

$$w_{ij}^{\mathcal{S}} = \begin{cases} 1, \text{if } (i,j) \in \mathcal{S} \\ 0, \text{otherwise} \end{cases} \tag{6}$$

$$w_{ij}^{\mathcal{D}} = \begin{cases} -1, \text{if } (i,j) \in \mathcal{D} \\ 0, \text{otherwise} \end{cases}. \tag{7}$$

Moreover, to take the full advantage of unlabeled data and remedy over-fitting, they designed a nearest-neighbor graph $W^{\mathcal{N}} \in \mathbb{R}^{m\times m}$ aimed to absorb all data information:

$$w_{ij}^{\mathcal{N}} = \begin{cases} 1, \text{if } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \\ 0, \text{otherwise} \end{cases}, \tag{8}$$

where $\mathcal{N}_k(\mathbf{x}_i)$ stands for the k nearest neighbor list of $\mathbf{x}_i$ [12].

Then the graph W that LRML defines is

$$W = W^{\mathcal{N}} + \gamma_s W^{\mathcal{S}} + \gamma_d W^{\mathcal{D}}, \tag{9}$$

where γ_s and γ_d are two regularization parameters to balance the tradeoff between similar and dissimilar constraints as well as the nearest neighbor information.

Recently, Liu *et al*. [15] argued that such a linear combination may fail to absorb the pairwise constraints when $\mathcal{S}$ and $\mathcal{D}$ contain very few data instances. In order to take full advantage of the limited semi-supervision, in their method, Semi-Supervised Sparse Metric Learning (S^{3}ML), the affinity matrix W is constructed in an affinity propagation strategy:

$$W^* = (1-\alpha)(I_n - \alpha \bar{W}^{\mathcal{N}})^{-1}(W^{\mathcal{S}} + W^{\mathcal{D}}), \tag{10}$$

where $\bar{W}^{\mathcal{N}}$ is the normalized nearest neighbor graph with row entries summed up to 1. Then the final graph W is obtained by zeroing out all the entries of W^* whose absolute values are smaller than a certain threshold [15].

So far, the optimization problem for metric learning with side information is:

$$\begin{aligned} \min_{M\succeq 0} f_1(M,\mathcal{X},\mathcal{S},\mathcal{D}) &= f_{\mathcal{D}}(M) + \gamma f_{\mathcal{L}}(M,\mathcal{X},\mathcal{S},\mathcal{D}) \\ &= \mathrm{Tr}\left((M_0^{-1} + \gamma X L X^T)M\right) - \log\det(M). \end{aligned} \tag{11}$$

This problem has an analytical optimal solution M^* if $\Sigma = M_0^{-1} + \gamma X L X^T \succ 0$ holds [12]. And in practice, a regularization term for positive definiteness is needed to give the final M^*:

$$M^* = \left(M_0^{-1} + \gamma X L X^T + \sigma I_n\right)^{-1}, \tag{12}$$

where $\sigma > 0$ and $I_n \in \mathbb{R}^{n\times n}$ is the identity matrix [12].

2.3. Exploring the Sparsity

Regarding $\Sigma = M_0^{-1} + \gamma X L X^T$ as a sample covariance matrix, the problem of metric learning can be converted to the problem of learning the structure of a undirected graphical model in statistics [8]. Assuming the data observed have a multivariate Gaussian distribution, then wherever $m_{ij} = 0$, features i and j are conditionally independent given other features [8]. Empirically, the Mahalanobis matrix Σ^{-1} is nearly sparse in high-dimensional space, this motivates researchers to learn the zero-pattern of $M = \Sigma^{-1}$ by introducing the ℓ_1-norm of M, *i.e*., $f_{\mathcal{S}}(M) = \|M\|_1$ [8, 20]. Besides, the ℓ_1 regularization can also help speedup the training time because of a more sparse M. Therefore, Many researchers applied this idea to constitute a sparse metric [19, 13, 25], *e.g*., Qi *et al*. [18] formulated the Sparse Distance Metric Learning (SDML) as:

$$\begin{aligned} \min_{M\succeq 0} f_2(M,\mathcal{X},\mathcal{S},\mathcal{D}) &= f_1(M,\mathcal{X},\mathcal{S},\mathcal{D}) + \lambda f_{\mathcal{S}}(M) \\ &= \mathrm{Tr}\left((M_0^{-1} + \gamma X L X^T)M\right) \\ &\quad - \log\det(M) + \lambda\|M\|_1, \end{aligned} \tag{13}$$

where λ is the parameter trading off between sparsity prior and the others. S^{3}ML [15] followed this idea.

3. Visual Analysis with Spatial Smoothness

In this section, we describe how to apply Laplacian penalized functional to measure the smoothness of the mapping functions U in metric learning, which plays the key role in our Two-Dimensional Smooth Metric Learning (2DSML) approach. We begin with a general description of Laplacian smoothing.

3.1. Laplacian Smoothing

Let $u(\cdot)$ be a function defined on a region of interest, $\Omega \subset \mathbb{R}^d$. The function operator $\mathcal{L}$ is defined as follows [14]:

$$\mathcal{L}u(\mathbf{t}) = \sum_{j=1}^{d} \frac{\partial^2 u}{\partial (t^{(j)})^2}. \tag{14}$$

The Laplacian penalty functional, denoted by $\mathcal{G}$, is defined by:

$$\mathcal{G}(u) = \int_{\Omega} [\mathcal{L}u]^2 d\mathbf{t}. \tag{15}$$

Intuitively, $\mathcal{G}(u)$ measures the smoothness of the function u over the region Ω [4]. In this paper, we primarily focus on images, which are essentially two-dimensional signals. Therefore, we take d to be 2.

3.2. Discretized Laplacian Smoothing

As described previously, $n_1 \times n_2$ images can be represented as vectors in $\mathbb{R}^n$ ($n = n_1 \times n_2$). Let

$D^{(j)}$ be an $n_j \times n_j$ matrix that yields a discrete approximation to $\partial^2/\partial(t^{(j)})^2$ ($j = 1, 2$). Then if $\mathbf{v} = (v(t_1), v(t_2), \ldots, v(t_{n_j}))$ is an n_j-dimensional vector which is discretized version of a continuous function $v(t^{(j)})$ on j-th dimension, then $D^{(j)}$ should have the property that [4]:

$$[D^{(j)}\mathbf{v}]_i \approx \frac{\partial^2 v(t_i)}{\partial(t^{(j)})^2}, \quad i = 1, 2, \ldots, n_j. \tag{16}$$

There are many options for $D^{(j)}$ [21]. In this paper, we use $D^{(j)}$ by modifying the Neumann discretization [17]:

$$D^{(j)} = n_j \begin{bmatrix} -1 & 1 & & & & & 0 \\ 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \\ 0 & & & & & 1 & -1 \end{bmatrix}. \tag{17}$$

To give a more concrete idea of the nature of the discretized penalty, let us examine $\|D^{(j)}\mathbf{v}\|^2$ in the one-dimensional case:

$$\begin{aligned} \|D^{(j)}\mathbf{v}\|^2 &= \frac{[v(t_2) - v(t_1)]^2}{h_j^2} + \frac{[v(t_{n_j}) - v(t_{n_j-1})]^2}{h_j^2} \\ &\quad + \frac{1}{h_j}\sum_{i=2}^{n_j-1} h_j \left[\frac{v(t_{i-1}) + v(t_{i+1}) - 2v(t_i)}{h_j^2}\right]^2 \\ &\approx [v'(t_1)]^2 + [v'(t_{n_j})]^2 \\ &\quad + \frac{1}{h_j}\int \left[v''(t^{(j)})\right]^2 dt^{(j)}, \quad j = 1 \text{ or } 2, \end{aligned} \tag{18}$$

where $h_j = 1/n_j$ is the step length of integration. Eq.(16) shows that $\|D^{(j)}\mathbf{v}\|^2$ is proportional to the discrete approximation of Laplacian penalty of function v defined over $t^{(j)}$.

For an image of size $n_1 \times n_2$, the region of interest Ω is essentially a two-dimensional rectangle. And there are totally n_2 n_1-dimensional rows and n_1 n_2-dimensional columns in it. Let $\mathbf{u} \in \mathbb{R}^n$ be the mapping functions obtained by a certain distance metric learning scheme, and $\mathbf{u}^{(i,j)}$,$i = 1, 2, \ldots, n_j$ be a sub-vector of $\mathbf{u}$ corresponding to either the i-th row or the i-th column of the rectangle. Then according to Eq.(16), the two-dimensional smoothness on this lattice could be discretely approximated by the (weighted) sum of Laplacian penalty over all such subvectors:

$$\mathcal{G}(\mathbf{u}) = \sum_{j=1,2}\frac{1}{n_j}\sum_{i=1}^{n_j}\|D^{(j)}\mathbf{u}^{(i,j)}\|^2. \tag{19}$$

This objective function can be rewritten in a more compact form of $\|\Delta\mathbf{u}\|^2$, with Δ defined as follows [17]:

$$\Delta = \frac{1}{\sqrt{n_1}}D^{(1)} \otimes I^{(2)} + I^{(1)} \otimes \frac{1}{\sqrt{n_2}}D^{(2)}, \tag{20}$$

where $I^{(j)} \in \mathbb{R}^{n_j \times n_j}$ is the identity matrix for $j = 1, 2$, and $\otimes$ is the kronecker product.

3.3. The Algorithms

From the above description, we hereby present a two-dimensional smoothness term for metric learning in visual analysis:

$$\begin{aligned} f_{\mathcal{G}}(M) &= \|\Delta \cdot U\|_F^2 = \mathrm{Tr}(U^T\Delta^T\Delta U) \\ &= \mathrm{Tr}(\Delta^T\Delta U U^T) = \mathrm{Tr}(\Delta^T\Delta M), \end{aligned} \tag{21}$$

where $\|\cdot\|_F$ stands for the Frobenius norm. This term can be easily incorporated into the regularization framework of metric learning. Given objective functions defined in Eq.(11) and Eq.(13), our 2DSML approach can be formally defined as the following:

$$\begin{aligned} &\min_{M\succeq 0} f_1(M, \mathcal{X}, \mathcal{S}, \mathcal{D}) + \zeta f_{\mathcal{G}}(M) \\ &\quad = \mathrm{Tr}\left((M_0^{-1} + \gamma XLX^T + \zeta\Delta^T\Delta)M\right) \\ &\quad - \log\det(M), \end{aligned} \tag{22}$$

and

$$\begin{aligned} &\min_{M\succeq 0} f_2(M, \mathcal{X}, \mathcal{S}, \mathcal{D}) + \zeta f_{\mathcal{G}}(M) \\ &\quad = \mathrm{Tr}\left((M_0^{-1} + \gamma XLX^T + \zeta\Delta^T\Delta)M\right) \\ &\quad - \log\det(M) + \lambda\|M\|_1. \end{aligned} \tag{23}$$

With the choice of different W and with/without sparse regularization, our approach can give two-dimensional smooth versions of existing metric learning methods, and since the smoothness term $f_{\mathcal{G}}(M)$ is naturally incorporated into the trace norm, all the existing algorithms can be easily adapted.

4. Experimental Results

In this section, several experiments were conducted to demonstrate the effectiveness of the proposed Two-Dimensional Smooth Metric Learning (2DSML) framework on benchmark image datasets.

4.1. Datasets and Compared Algorithms

Two image datasets are used in our experiments. The first one is CMU PIE face database[1], which contains 32×32 cropped face images of 68 persons. Each person has 170 facial images under different illumination conditions and expressions. The second is YaleB face database[2]. It has 38 individuals and around 64 near frontal images under different illuminations. For numerical considerations, the features (gray-scale values) are manually scaled to $[0, 1]$ (divided by 256). Then the whole dataset is randomly partitioned into

[1] http://www.ri.cmu.edu/projects/project_418.html
[2] http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html

Table 1. Classification error rate ER with varying number of side information on PIE & YaleB.(%)

	$\lvert\mathcal{S}\rvert$	0	500	1000	1500	2000	2500	3000	Avg.
	$\lvert\mathcal{D}\rvert$	0	2500	5000	7500	10000	12500	15000	
PIE	**EU**	24.03±0.50							24.03
	MAH	14.54±0.60							14.54
	LRML	12.75 ± 0.44	12.72 ± 0.46	12.62 ± 0.44	12.58 ± 0.44	12.49 ± 0.46	12.38 ± 0.46	12.13 ± 0.45	12.53
	SDML	11.21 ± 0.44	11.15 ± 0.45	11.11 ± 0.45	11.05 ± 0.44	10.97 ± 0.47	10.91 ± 0.48	10.80 ± 0.45	11.02
	S^{3}ML	13.15 ± 0.36	11.05 ± 0.53	9.64 ± 0.41	9.34 ± 0.47	8.72 ± 0.41	8.47 ± 0.34	8.21 ± 0.39	9.80
	2DSML	**8.33 ± 0.37**	**8.13 ± 0.36**	**7.94 ± 0.34**	**7.77 ± 0.35**	**7.62 ± 0.34**	**7.49 ± 0.34**	**7.35 ± 0.37**	**7.81**
YaleB	**EU**	25.62±1.32							25.62
	MAH	32.39±1.62							32.39
	LRML	18.44 ± 1.33	18.11 ± 1.73	15.19 ± 1.85	13.11 ± 1.55	11.09 ± 1.48	10.22 ± 1.36	10.08 ± 1.23	13.75
	SDML	14.98 ± 1.80	11.62 ± 1.50	10.95 ± 1.32	10.01 ± 1.36	9.84 ± 1.23	9.20 ± 1.23	9.09 ± 1.44	10.82
	S^{3}ML	25.62 ± 1.34	19.30 ± 1.71	10.14 ± 1.60	9.82 ± 1.40	9.67 ± 1.52	9.22 ± 1.39	9.11 ± 1.23	13.27
	2DSML	**9.59 ± 1.23**	**9.39 ± 1.13**	**9.26 ± 1.08**	**8.91 ± 1.21**	**8.62 ± 1.34**	**7.92 ± 1.37**	**7.88 ± 1.18**	**8.80**

training and testing sets. Specifically, 50 images per person are selected and the remaining images are used for testing.

For each dataset, we apply seven metrics listed below:

Euclidean. The Euclidean distance as a baseline algorithm, denoted as "EU".

Mahalanobis. A standard Mahalanobis metric parameterized by the sample concentration, denoted as "MAH".

LRML [12]. Laplacian Regularized Metric Learning whose affinity graph is constructed as Eq.(9).

SDML [18]. Sparse Distance Metric Learning which works under pairwise constraints and produces sparse metrics.

S^{3}ML [15]. Semi-Supervised Sparse Metric Learning which produces sparse metrics and constructs affinity graph in an affinity propagation strategy.

2DSML. Two-Dimensional Smooth Metric Learning proposed in this paper. Due to space limitation, we only implemented Eq.(22) of 2DSML as a demonstration, which is a direct extension of LRML.

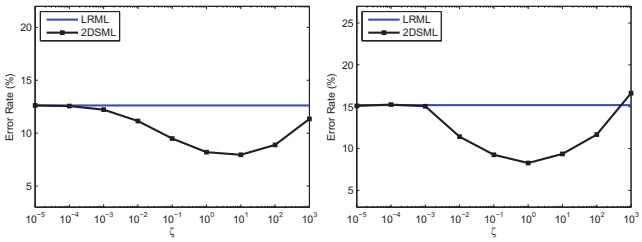
Note that we only compare the state-of-the-art metric learning algorithms supporting pairwise constraints. Popular methods [22, 9, 10] are not considered here for they need an explicit class label for each sample.

4.2. Nearest Neighbor Classification

We consider the 1-nearest neighbor classifier to evaluate the discriminating power of different metrics, since a good distance metric should yield high classification accuracy. For each testing sample $\mathbf{x}_i$, we find its nearest neighbor $\mathbf{x}'_i$ in the training set using different metrics. Let $c(\mathbf{x})$ be the ground truth class label for $\mathbf{x}$, the nearest neighbor classification error rate (ER) is defined as:

$$ER = 1 - \frac{1}{N}\sum_{n=1}^{N}\delta(c(\mathbf{x}_i), c(\mathbf{x}'_i)), \tag{24}$$

where N is the number of data points and $\delta(a,b) = 1$ if $a = b$ and 0 otherwise. In the training data, we randomly construct similar and dissimilar pairwise constraint sets of different sizes. In the extreme case, we provide no side information ($\mathcal{S}$ and $\mathcal{D}$ are empty) to see how the (semi-)supervised metric learning algorithms perform under totally unsupervised scheme. The evaluation process is repeated for 10 times. The sample concentration Σ^{-1} is chosen as the matrix to be approximated. Parameters $k = 6$, $\gamma_s = 1$ and $\gamma_d = 0.1$ are fixed for all methods. The propagation parameter α for W in S^{3}ML is 0.5. For 2DSML, we use two-fold cross validation on the training dataset to tune the smoothing parameter ζ, while fixing all the other parameters identical to LRML, its corresponding un-smoothed version.



(a) PIE (b) YaleB

Figure 1. Model selection for 2DSML.

With regard to the algorithm, LRML and 2DSML only need a matrix inverse, and alternating linearization method [20] are applied to solve the optimization in SDML and S^{3}ML.

Table 1 summarizes the results for all the compared methods. It can be seen that 2DSML significantly outperforms other metric learning algorithms in both datasets. The reason lies 2DSML explicitly considers the spatial relationship between the pixels in an image. The use of spatial information significantly reduces the number of degrees of freedom. Therefore, 2DSML can have good performance even when there is no side information available.

4.3. Model Selection

The $\zeta \geq 0$ is an essential parameter in our 2DSML model, which controls the smoothness of the estimator. When $\zeta = 0$, the 2DSML model will reduce to the ordinary metric learning approach. When $\zeta \to \infty$, the 2DSML model will choose a two-dimensional smooth mapping function and ignore the information from the data and the constraints. 2DSML with an appropriate ζ is a trade-off between these two extreme cases. Thus how to choose the parameter ζ, or how to select the model. In this subsection, we fix $|\mathcal{S}| = 1000$ and $|\mathcal{D}| = 5000$ and study the impact of parameter ζ on ER.

Figure 1 shows how the average performance of 2DSML varies with the parameter ζ. The result of LRML is also depicted as a baseline method. As we can see, 2DSML is very robust with respect to ζ. It achieves consistently significant better performance with ζ varying from 10^{-1} to 10^{1}.

5. Conclusions

This paper proposes a new family of metric learning methods termed Two-Dimensional Smooth Metric Learning (2DSML) for visual analysis. In contrast to other metric learning techniques, the proposed 2DSML explicitly considers the spatial relationship between the pixels in images. By introducing a discretized Laplacian penalized functional, the number of degrees of freedom is significantly reduced, resulting in a Mahalabonis metric smoother than those obtained by the existing metric learning algorithms. The promising results on PIE and YaleB datasets show that 2DSML is superior to the state-of-the-art metric learning approaches for visual analysis.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grants 91120302 and 60905001, National Basic Research Program of China (973 Program) under Grant 2011CB302206 and Fundamental Research Funds for the Central Universities.

References

[1] M. Baghshah and S. Shouraki. Semi-supervised metric learning using pairwise constraints. In *IJCAI*, 2009. 1

[2] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *JMLR*, 6(1):937, 2006. 1

[3] M. Berman. Automated smoothing of image and other regularly spaced data. *IEEE TPAMI*, 16(5):460–468, 1994. 2

[4] D. Cai, X. He, Y. Hu, J. Han, and T. Huang. Learning a spatially smooth subspace for face recognition. In *CVPR*, 2007. 2, 3, 4

[5] T. Chan, A. Marquina, and P. Mulet. High-order total variation-based image restoration. *SIAM Journal on Scientific Computing*, 22:503, 2000. 2

[6] J. Davis and I. Dhillon. Structured metric learning for high dimensional problems. In *SIGKDD*, 2008. 2

[7] J. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon. Information theoretic metric learning. In *ICML*, 2007. 1, 2

[8] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432, 2008. 3

[9] A. Globerson and S. Roweis. Metric learning by collapsing classes. In *NIPS 18*, 2005. 1, 5

[10] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *NIPS 17*, 2004. 1, 5

[11] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, 2009. 2

[12] S. Hoi, W. Liu, and S. Chang. Semi-supervised distance metric learning for collaborative image retrieval and clustering. *ACM TOMCCAP*, 6(3):1–26, 2010. 1, 2, 3, 5

[13] K. Huang, Y. Ying, and C. Campbell. Gsml: A unified framework for sparse metric learning. In *ICDM*, 2009. 3

[14] J. Jost. *Riemannian Geometry and Geometric Analysis*. Springer-Verlag, 2002. 3

[15] W. Liu, S. Ma, D. Tao, J. Liu, and P. Liu. Semi-supervised sparse metric learning using alternating linearization optimization. In *ACM SIGKDD*, 2010. 1, 2, 3, 5

[16] H. Murase and S. Nayar. Visual learning and recognition of 3-d objects from appearance. *IJCV*, 14(1):5–24, 1995. 1

[17] F. O'Sullivan. Discretized laplacian smoothing by fourier methods. *Journal of the American Statistical Association*, pages 634–642, 1991. 4

[18] G.-J. Qi, J. Tang, Z.-J. Zha, T.-S. Chua, and H.-J. Zhang. An efficient sparse metric learning in high-dimensional space via l_1-penalized log-determinant regularization. In *ICML*, 2009. 1, 2, 3, 5

[19] R. Rosales and G. Fung. Learning sparse metrics via linear programming. In *SIGKDD*, 2006. 3

[20] K. Scheinberg, S. Ma, and D. Goldfarb. Sparse inverse covariance selection via alternating linearization methods. In *NIPS 23*, 2010. 3, 5

[21] L. Trefethen and D. Bau. *Numerical linear algebra*. Society for Industrial Mathematics, 1997. 4

[22] K. Weinberger and L. Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10:207–244, 2009. 1, 5

[23] E. Xing, A. Ng, M. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *NIPS 16*, 2003. 1, 2

[24] L. Yang and R. Jin. Distance metric learning: a comprehensive survey. Technical report, Michigan State University, 2006. 1

[25] Y. Ying, K. Huang, and C. Campbell. Sparse metric learning via smooth optimization. In *NIPS 22*, 2009. 3