KERNEL-BASED MANIFOLD-ORIENTED STOCHASTIC NEIGHBOR PROJECTION METHOD

Jianwei Zheng, Hong Qiu, Qiongfang Huang, Wanliang Wang and Xinli Xu
School of Computer Science and Technology
Zhejiang University of Technology
310023, Hangzhou, China
E-mail: zjw@zjut.edu.cn

KEYWORDS
Kernel method, Dimensionality reduction, Nonlinear feature extraction, Manifold learning.

ABSTRACT

A new method for performing a nonlinear form of manifold-oriented stochastic neighbor projection method is proposed. By the use of kernel functions, one can operate in the feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. The proposed method is termed as kernel-based manifold-oriented stochastic neighbor projection(KMSNP). By two different strategies, KMSNP is divided into two methods: KMSNP1 and KMSNP2. Experimental results on several databases show that, compared with the relevant methods, the proposed methods obtain higher classification performance and recognition rate.

INTRODUCTION

Kernel-based methods(kernel methods for short) have become a new hot topic in machine learning fields in recent years, their theoretical basis is statistical learning theory. Kernel methods are a class of algorithms for pattern analysis, whose best known element is the support vector machine(SVM) (Dardas and Georganas 2011).The methods skillfully introduce kernel function which not only reduces the curse of dimensionality (Cherchi and Guevara 2012, Xue et al. 2012), but also effectively solves the local minimum and incomplete statistical analysis in traditional pattern recognition methods on the premise of no additional computational capacity. As an availability way to resolve the problem of nonlinear pattern recognition, kernel methods approach the problem by mapping the data into a high-dimensional feature space, where each coordinate corresponds to one feature of the data items, transforming the data into a set of points in a Euclidean space (Chen and Li 2011, Zhang et al. 2008).

The theory of kernel methods can be traced back to 1909, Mercer proposed Mercer's theorem (Mercer 1909) which indicates that any ‘reasonable’ kernel function corresponds to some feature space. 1964, the use of Mercer's theorem for interpreting kernels as inner products in a feature space was introduced into machine learning by Aizerman et al. (AizermanI et al. 1964), but no sufficient importance has been attached to it. Until 1992, Vapnik et al. (Boser et al. 1992) successfully extended the SVM to the non-linear SVM by using kernel functions, it began to show its potential and advantages. Subsequently, more and more kernel-based methods were presented, such as: kernel principal component analysis(KPCA) (Xiao et al. 2012), kernel fisher discriminator(KFD) (Yang et al. 2005), kernel independent component analysis (KICA) (Zhang et al. 2013), kernel partial least squares(KPLS) (Helander et al. 2012) and so on.

In this paper, we propose to use the kernel idea and present a method called kernel-based manifold-oriented stochastic neighbor projection(KMSNP) method through improving the manifold-oriented stochastic neighbor projection(MSNP) (Wu et al. 2011) technique. MSNP is based on stochastic neighbor embedding(SNE) (Hinton and Roweis 2002) and t-SNE (Maaten and Hinton 2008). The basic principle of SNE is to convert pairwise Euclidean distances into probabilities of selecting neighbors to model pairwise similarities while t-SNE uses student t-distribution to model pairwise dissimilarities in low-dimensional space. Different from SNE and t-SNE, MSNP converts pairwise dissimilarities of inputs to probability distribution related to geodesic distance in high-dimensional space and uses Cauchy distribution to model stochastic distribution of features. Furthermore, it recovers the manifold structure through a linear projection by requiring the two distributions to be similar. Experiments demonstrate MSNP has unique advantages in terms of visualization and recognition task, but there are still two drawbacks in it: firstly, MSNP is an unsupervised method and lack of the idea of class label, so it is not suitable for pattern identification; secondly, since MSNP is a linear feature dimensionality reduction algorithm, it cannot effectively settle the nonlinear feature extraction problem. To overcome the disadvantages of MSNP, we have done some preliminary work. On the first, we introduced the idea of class label and presented a method called discriminative stochastic neighbor embedding analysis(DSNE) (Zheng et al. 2012, Chen

Proceedings 27th European Conference on Modelling and Simulation ©ECMS Webjørn Rekdalsbakken, Robin T. Bye, Houxiang Zhang (Editors)
ISBN: 978-0-9564944-6-7 / ISBN: 978-0-9564944-7-4 (CD)

and Wang 2012). On the second, we think KMSNP can overcome the disadvantage mentioned above well.

The rest of this paper is organized as follows: in Section 2, we provide a brief review of MSNP. Section 3 describes the detailed algorithm derivation of KMSNP. Furthermore, experiments on various databases are presented in Section 4. Finally, we provide some concluding remarks and describe several issues for future works in Section 5.

MSNP

Considering the problem of representing d-dimensional data vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$, by r-dimensional ($r << d$) vectors $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N$ such that $\boldsymbol{y}_i$ represents $\boldsymbol{x}_i$. The basic principle of MSNP is to convert pairwise dissimilarity of inputs to probability distribution related to geodesic distance in high-dimensional space, and then using Cauchy distribution to model stochastic distribution of features, finally, MSNP recovers the manifold structure through a linear projection by requiring the two distributions to be similar. Mathematically, the similarity of datapoint $\boldsymbol{x}_i$ to datapoint $\boldsymbol{x}_j$ is depicted as the following joint probability p_{ij} which means $\boldsymbol{x}_i$ how possible to pick $\boldsymbol{x}_j$ as its neighbor:

$$p_{ij} = \frac{\exp(-D_{ij}^{geo}/2)}{\sum_{k \neq i} \exp(-D_{ik}^{geo}/2)} \tag{1}$$

where D_{ij}^{geo} is the geodesic distance for $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. In practice, MSNP calculates geodesic distance by using a two-phase method (Wu et al. 2011). Firstly, an adjacency graph G is constructed by K-nearest neighbor strategy. Secondly, the desired geodesic distance is approximated by the shortest path of graph G. This procedure is proposed in Isomap to estimate geodesic distance and the detail calculation steps can be found in (Tenenbaum et al. 2000).

For low-dimensional representations, MSNP employs Cauchy distribution with γ degree of freedom to construct joint probability q_{ij}. The probability q_{ij} indicates how possible point i and point j can be stochastic neighbors is defined as:

$$q_{ij} = \frac{\left[\pi\gamma\left(1 + \left\|\frac{\boldsymbol{y}_i - \boldsymbol{y}_j}{\gamma}\right\|\right)\right]^{-1}}{\sum_{k \neq l}\left[\pi\gamma\left(1 + \left\|\frac{\boldsymbol{y}_k - \boldsymbol{y}_l}{\gamma}\right\|\right)\right]^{-1}} \tag{2}$$

where γ is the freedom degree parameter of Cauchy distribution and through a linear projection: $\boldsymbol{y}_i = \mathbf{A}\boldsymbol{x}_i$ ($\mathbf{A} \in \mathbf{R}^{r \times d}$), formula (2) can be formulated as:

$$q_{ij} = \frac{\left[\gamma^2 + d_{ij}{}^2(\mathbf{A})\right]^{-1}}{\sum_{k \neq l}\left[\gamma^2 + d_{ij}{}^2(\mathbf{A})\right]^{-1}} \tag{3}$$

In formula (3), $d_{ij}(\mathbf{A}) = \|\boldsymbol{y}_i - \boldsymbol{y}_j\| = \|\mathbf{A}\boldsymbol{x}_i - \mathbf{A}\boldsymbol{x}_j\| = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{A}(\boldsymbol{x}_i - \boldsymbol{x}_j)}$ is the Euclidian distance between two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

MSNP finds the optimal low-dimensional representations for matching p_{ij} and q_{ij} to the greatest extent. This is achieved by minimizing the following penalty function, which is the sum of Kullback–Leibler divergences measuring the difference between two probability distributions:

$$C_i(\mathbf{A}) = KL(P_i \| Q_i) = \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{4}$$

MSNP calculates the optimal low-dimensional representation $\mathbf{A}$ by minimizing $C(\mathbf{A})$ overall datapoints with a gradient descent search, i.e.:

$$\min C(\mathbf{A}) = \sum_{i,j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{5}$$

ALGORITHM DERIVATION OF KMSNP

Obviously, MSNP is a linear feature dimensionality reduction algorithm and has unique advantages in terms of visualization and recognition task, but it cannot effectively settle the nonlinear feature extraction problem. So the remainder of this section is devoted to extend MSNP to a nonlinear scenario using techniques of kernel methods.

Let kernel function κ defines the inner product in an embedding high dimensional space F with the associated nonlinear map $\varphi(\boldsymbol{x})$: $R^d \rightarrow F$:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle\varphi(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_j)\rangle \tag{6}$$

which allows us to compute the value of the inner product in F without having to carry out the map.

It should be noted that we use φ_i to denote $\varphi(\boldsymbol{x}_i)$ for brevity in the following. Next, we express the transformation $\mathbf{A}$ with:

$$\mathbf{A} = \left[\sum_{i=1}^{N} b_i^{(1)}\varphi_i, \ldots, \sum_{i=1}^{N} b_i^{(r)}\varphi_i\right]^{\mathrm{T}} \tag{7}$$

We define $\mathbf{B} = \left[b^{(1)}, \ldots, b^{(r)}\right]^{\mathrm{T}}$ and $\mathbf{\Phi} = \left[\varphi_1, \ldots, \varphi_N\right]^{\mathrm{T}}$, then $\mathbf{A} = \mathbf{B\Phi}$. Based on above definition, the Euclidian distance between two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the embedding space F can be formulated as:

$$\begin{aligned} &d_{ij}^F(\mathbf{A}) \\ &= \left\|\mathbf{A}(\varphi_i-\varphi_j)\right\| = \left\|\mathbf{B\Phi}(\varphi_i-\varphi_j)\right\| \\ &= \left\|\mathbf{B}(K_i-K_j)\right\| = \sqrt{(K_i-K_j)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_i-K_j)} \end{aligned} \tag{8}$$

where $K_i = \left[\kappa(\boldsymbol{x}_1,\boldsymbol{x}_i),...,\kappa(\boldsymbol{x}_N,\boldsymbol{x}_i)\right]^{\mathrm{T}}$ is a column vector. It is clear that the distance in the kernel embedding space is related to a kernel function and the matrix **B**.

In this section, we propose two methods to construct the objective function. The first strategy is to parameterize the objective function by the matrix **B**. Firstly, we replace $D_{ij}^{geo\,F}$ with D_{ij}^{geo} in formula (1) and $d_{ij}(\mathbf{A})$ with $d_{ij}^F(\mathbf{A})$ in formula (3) so that p_{ij}^1, q_{ij}^1 which are defined to be applied in the embedding high dimensional space F can be written as:

$$p_{ij}^1 = \frac{\exp(-D_{ij}^{geo^F}/2)}{\sum_{k\neq i}\exp(-D_{ik}^{geo^F}/2)} \tag{9}$$

$$q_{ij}^1 = \frac{\left[\gamma^2+(d_{ij}^F(\mathbf{A}))^2\right]^{-1}}{\sum_{k\neq l}\left[\gamma^2+(d_{kl}^F(\mathbf{A}))^2\right]^{-1}} \tag{10}$$

By simple algebra formulation, formula (10) can be formulated as:

$$q_{ij}^1 = \frac{\left[\gamma^2+(K_i-K_j)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_i-K_j)\right]^{-1}}{\sum_{k\neq l}\left[\gamma^2+(K_t-K_l)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_t-K_l)\right]^{-1}} \tag{11}$$

Then, we denote $C(\mathbf{B})$ by modifying $C(\mathbf{A})$ via substituting **A** with **B** into the regularization term of formula (5). In this work, we use the conjugate gradient method to minimize $C(\mathbf{B})$ as in (Wu et al. 2011). In order to make the derivation less cluttered, we first define two auxiliary variables w_{ij} and u_{ij} as:

$$w_{ij} = [\gamma^2+(K_i-K_j)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_i-K_j)]^{-1} \tag{12}$$

$$u_{ij} = (p_{ij}^1-q_{ij}^1)w_{ij} \tag{13}$$

Afterwards differentiating $C(\mathbf{B})$ with respect to the matrix **B** gives the following gradient, which we adopt for learning:

$$\begin{aligned} &\frac{dC(\mathbf{B})}{d(\mathbf{B})} = \sum_{i,j}\frac{p_{ij}^1}{q_{ij}^1}(q_{ij}^1)' \\ &= 2\mathbf{B}[\sum_{i,j}p_{ij}\frac{(K_i-K_j)(K_i-K_j)^{\mathrm{T}}}{\gamma^2+(K_i-K_j)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_i-K_j)}] \\ &-2\mathbf{B}[\sum_{i,j}p_{ij}^1\frac{\sum_{k\neq l}(\gamma^2+(K_k-K_l)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-2}}{\sum_{k\neq l}(\gamma^2+(K_k-K_l)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}} \\ &\qquad\frac{(K_k-K_l)(K_k-K_l)^{\mathrm{T}}}{\sum_{k\neq l}(\gamma^2+(K_k-K_l)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}}] \\ &= 2\mathbf{B}[\sum_{i,j}p_{ij}^1 w_{ij}(\boldsymbol{x}_i-\boldsymbol{x}_j)(\boldsymbol{x}_i-\boldsymbol{x}_j)^{\mathrm{T}} \\ &\qquad -\sum_{k\neq l}q_{kl}^1 w_{kl}(\boldsymbol{x}_k-\boldsymbol{x}_l)(\boldsymbol{x}_k-\boldsymbol{x}_l)^{\mathrm{T}}] \\ &= 2\mathbf{B}\sum_{i,j}u_{ij}(K_i-K_j)(K_i-K_j)^{\mathrm{T}} \end{aligned} \tag{14}$$

Let **U** be the N order matrix with element u_{ij}. Note that **U** is a symmetric matrix; therefore, **D** can be defined as a diagonal matrix that each entry is column(or row) sum of **U**, i.e., $\mathbf{D}_{ii} = \sum_j \mathbf{U}_{ij}$. With this definition, the gradient expression (14) can be reduced to:

$$\begin{aligned} \frac{dC(\mathbf{B})}{d(\mathbf{B})} &= 2\mathbf{B}\sum_{i,j}u_{ij}(K_i-K_j)(K_i-K_j)^{\mathrm{T}} \\ &= 2\mathbf{B}(\sum_{i,j}u_{ij}K_iK_i^{\mathrm{T}}+\sum_{i,j}u_{ij}K_jK_j^{\mathrm{T}} \\ &\qquad -\sum_{i,j}u_{ij}K_iK_j^{\mathrm{T}}-\sum_{i,j}u_{ij}K_jK_i^{\mathrm{T}}) \\ &= 4\mathbf{B}(\mathbf{KDK}^{\mathrm{T}}-\mathbf{KUK}^{\mathrm{T}}) \end{aligned} \tag{15}$$

Once the gradient is calculated, our optimal problem can be solved by an iterative procedure based on the conjugate gradient method. For convenience, we name this kernel method as KMSNP1. The description of KMSNP1 algorithm can be given by:

Step1. Make sure the sample matrix **X** and the kernel function, set K-nearest neighborhood parameter k(for geodesic distance), Cauchy distribution parameter γ and the maximum iteration times Mt.

Step2. Compute geodesic distance $D_{ij}^{geo\,F}$ between two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ on **X**, compute the joint probability p_{ij}^1 by utilizing formula (9).

Step3. Set t=1 : Mt, we search for the solution in loop: firstly, compute the joint probability q_{ij}^1 by utilizing formula (11); then, compute gradient $dC(\mathbf{B})/d(\mathbf{B})$ by utilizing formula (15); finally, update $\mathbf{B}^t$ based on $\mathbf{B}^{t-1}$ by conjugate gradient operation.

Step4. Judge whether C^t-C^{t-1}<ε (in this paper, we take ε=1e-7) converges to a stable solution or t reaches the maximum value Mt; If these prerequisites are met, Step5 are performed, otherwise we repeat Step3.

Step5. output $\mathbf{B} = \mathbf{B}^t$.

Another strategy is that we let $C^F(\mathbf{A})$ be the objective function in the embedding space F. Its gradient can be written as:

$$\begin{aligned} &\frac{dC^F(\mathbf{A})}{d(\mathbf{A})} = \sum_{i,j}\frac{p_{ij}^1}{q_{ij}^1}(q_{ij}^1)' \\ &= -2[\sum_{i,j}p_{ij}^1\frac{\mathbf{B}(K_i-K_j)(\boldsymbol{\varphi}_i-\boldsymbol{\varphi}_j)^{\mathrm{T}}}{\gamma^2+(\mathbf{B}(K_i-K_j))^{\mathrm{T}}\mathbf{B}(K_i-K_j)}] \end{aligned}$$

$$
\begin{aligned}
&+2[\sum_{i,j} p_{ij}{}^{1} \frac{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-2}}{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}} \\
&\quad \frac{\mathbf{B}(K_k-K_l)(\boldsymbol{\varphi}_k-\boldsymbol{\varphi}_l)^{\mathrm{T}}}{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}}] \\
&=-2[\sum_{i,j} p_{ij}{}^{1} \frac{\mathbf{B}Q_{ij}^{(K_i-K_j)}\mathbf{\Phi}}{\gamma^2+(\mathbf{B}(K_i-K_j))^{\mathrm{T}}\mathbf{B}(K_i-K_j)}] \\
&+2[\sum_{i,j} p_{ij}{}^{1} \frac{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-2}}{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}} \\
&\quad \frac{\mathbf{B}Q_{kl}^{(K_k-K_l)}\mathbf{\Phi}}{\sum_{k\neq l}(\gamma^2+(\mathbf{B}(K_k-K_l))^{\mathrm{T}}\mathbf{B}(K_k-K_l))^{-1}}] \\
&=-2\left[\sum_{i,j} p_{ij}{}^{1} w_{ij}\mathbf{B}Q_{ij}^{(K_i-K_j)} - \sum_{i,j} q_{kl}{}^{1} w_{kl}\mathbf{B}Q_{kl}^{(K_k-K_l)}\right]\mathbf{\Phi}
\end{aligned}
\tag{16}
$$

in this form, $Q_{ij}^{(K_i\text{-}K_j)}$ can be regard as the $N\times N$ matrix with vector K_i-K_j in the ith column, vector K_j-K_i in the jth column and the other columns are all zeros.

This method is called KMSNP2. Note that $\mathbf{\Phi}$ is a constant matrix, furthermore, the observations of formula (16) make us know that updating the matrix **A** in the optimization only means updating the matrix **B**. More important, $\mathbf{\Phi}$ does not need to compute since $d_{ij}^{F}(\mathbf{A})$ only depend upon **B**, K_i and K_j. Therefore, we don't need to explicitly perform the nonlinear map $\varphi(\boldsymbol{x})$ to minimize the objective function $C^F(\mathbf{A})$. The description of KMSNP2 algorithm is the same as that of KMSNP1, except for Step3. The Step3 in KMSNP2 should be described as: set t=1 : Mt, we search for the solution in loop: firstly, compute the joint probability $q_{ij}{}^{1}$ by utilizing formula (11); then, compute gradient $d_{ij}^{F}(\mathbf{A})/d(\mathbf{A})$ by utilizing formula (16); finally, update $\mathbf{B}^t$ based on $\mathbf{B}^{t-1}$ by conjugate gradient operation. Additionally, the computational complexity of KMSNP1 and KMSNP2 is respectively O($2rN^2$ $+rNK$) and O($2rKN$ + rN^2) in each iteration. Hence, it is obviously that KMSNP2 is faster than KMSNP1 during each iteration.

EXPERIMENTS

In this section, we evaluate the effectiveness of our KMSNP1 and KMSNP2 methods for feature extraction. Several experiments are carried out on U.S. Postal Service (USPS) and ORL face databases to demonstrate their good behavior on classification performance and recognition task. Moreover, the Gaussian RBF kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\| \boldsymbol{x} - \boldsymbol{x}' \|^2/2\sigma^2)$ is chosen as the kernel function of KMSNP1 and KMSNP2, where σ is set as the variance of the training sample set of **X**.

USPS and ORL databases

The databases used in our experiments are summarized as follows:

USPS includes 10 digit characters and 1100 samples in total. The original data format is of 16 x 16 pixels. Figure 1 shows samples of the cropped images from USPS handwritten digits database.

Figure 1: Samples of the cropped images from USPS handwritten digits database.

ORL consists of gray images of faces from 40 distinct subjects, with 10 pictures for each subject. For every subject, the images were taken with varied lighting condition and different facial expressions. The original size of each image is 112 x 92 pixels, with 256 gray levels per pixel. Figure 2 illustrates a sample subject of ORL database.

Figure 2: Sample face images from ORL database.

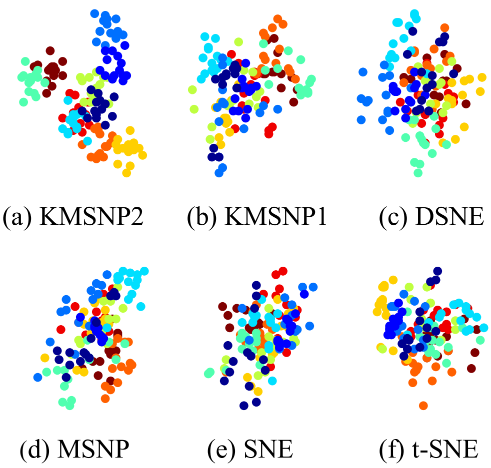
Visualization using KMSNP1 and KMSNP2

In this subsection, we focus on the classification performance of the proposed methods which are compared with that of the relevant algorithms, including MSNP (Wu et al. 2011), SNE (Hinton and Roweis 2002), t-SNE (Maaten and Hinton 2008) and DSNE (Zheng et al. 2012). The experiments are carried out respectively on USPS and ORL databases. For the sake of computational efficiency as well as noise filtering, we first adjust the size of each image to 32 x 32 pixels on ORL, and then we select fourteen samples from each class on USPS and five samples from each class on ORL.

The experimental procedure is to extract a 20-dimensional feature for each image by SNE, t-SNE, DSNE, MSNP, KMSNP1 and KMSNP2, respectively; then to evaluate the quality of features through visual presentation of the first two-dimensional feature. The parameters are set as: the local neighbor parameter of all the methods is k=15; for MSNP, KMSNP1 and KMSNP2, the degree freedom of Cauchy distribution is γ=4 and the iteration number is Mt =1000; for DSNE, the perplexity parameter is λ=0.1 and the iteration number is 1000 as well; for SNE and t-SNE, the perplexity parameter is $perp$=20 and the iteration number is the same as DSNE.

Figure 3 and Figure 4 show the visual presentation results of the methods mentioned above on USPS and ORL databases. The visual presentation is represented as a scatterplot in which different color determines different class information. The figures reveal that

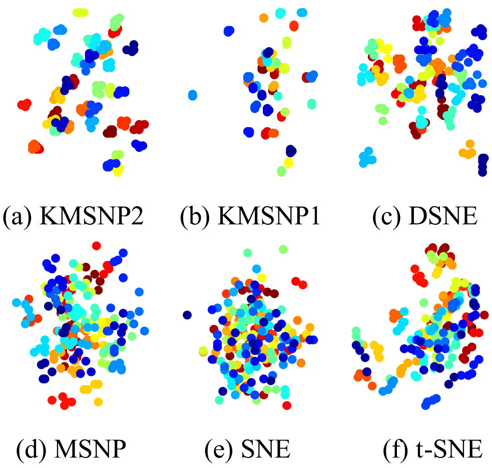
KMSNP1 and KMSNP2 give considerably better classification result than SNE, t-SNE, DSNE, and MSNP on all databases, for the separation between classes are quite obvious. In particular, it is clearly observed that though DSNE provides much better insight into the manifold structure, the clustering qualities and separation degree of KMSNP1 and KMSNP2 scatterplots are obviously better than the one of DSNE; for MSNP, it has smaller intra-class scatter, but exists overlapping phenomenon among classes; SNE and t-SNE not only get less separation for the inter-class data but also produce larger intra-class scatter, so, it is difficulty to find obvious manifold structure from their results. With regard to KMSNP1 and KMSNP2, we can find from the figures that KMSNP1 shows the best classification performance among all the algorithms on ORL face database, while not on the other database, thereinto, the classification performance of KMSNP1 is inferior to KMSNP2 on USPS.



(a) KMSNP2 (b) KMSNP1 (c) DSNE

(d) MSNP (e) SNE (f) t-SNE

Figure 3: Visualization of 140 images from USPS handwritten digits database.



(a) KMSNP2 (b) KMSNP1 (c) DSNE

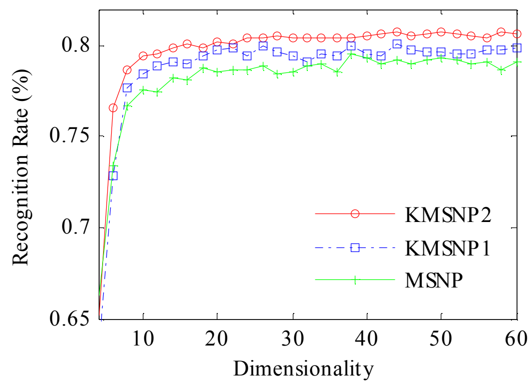
(d) MSNP (e) SNE (f) t-SNE

Figure 4: Visualization of 200 face images from ORL faces database.

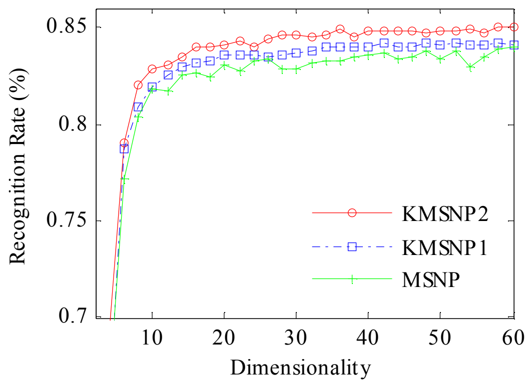
Recognition using KMSNP1 and KMSNP2

In this subsection, we apply KMSNP1 and KMSNP2 to recognition task to verify their feature extraction capability. We compare the performance with MSNP, since it was proved to be better than existing feature extraction algorithms such as SNE, t-SNE and so on in Ref. (Wu et al. 2011). The procedure of recognition is designed as follows: firstly, divide dataset into training sample set $\mathbf{X}_{train}$ and testing sample set $\mathbf{X}_{test}$ randomly; secondly, the training process for the optimal matrix $\mathbf{A}$ or $\mathbf{B}$ is taken for MSNP, KMSNP1 and KMSNP2; thirdly, feature extraction is accomplished for all samples using $\mathbf{A}$ or $\mathbf{B}$; finally, a testing image is identified by a nearest neighbor classifier. The freedom degree γ of Cauchy distribution in MSNP is determined by cross validation and the maximum number of iterations is set to be 1000. Let l denotes the number of training samples in each class, we set the K-nearest neighborhood parameter k in MSNP, KMSNP1 and KMSNP2 as $k = l$-1.

Figure 5 is the results of the experiment in USPS ((a):l=14, (b):l=25), and Figure 6 shows the recognition rate versus subspace dimension on ORL ((a):l=3, (b):l=5). The maximal recognition rate of each method and the corresponding dimension are given in Table 1, where the number in bold stands for the highest recognition rate. From Table 1, we can find that KMSNP1 and KMSNP2 outperform MSNP on USPS and ORL. Obviously, KMSNP2 gives better recognition task than KMSNP1 on USPS while the result is just on the contrary on ORL. That means the performances of KMSNP1 and KMSNP2 vary with databases. Besides, KMSNP1 and KMSNP2 achieve considerable recognition accuracy when feature dimension is 10 on USPS and 20 on ORL. It indicates that KMSNP1 and KMSNP2 grasp the key character of face images relative to identification with a few features. From the nature of dimensional reduction, this result demonstrates that MSNP is inferior to KMSNP1 and KMSNP2.

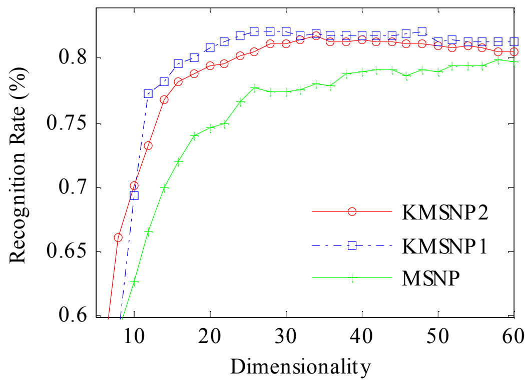


(a) l=14

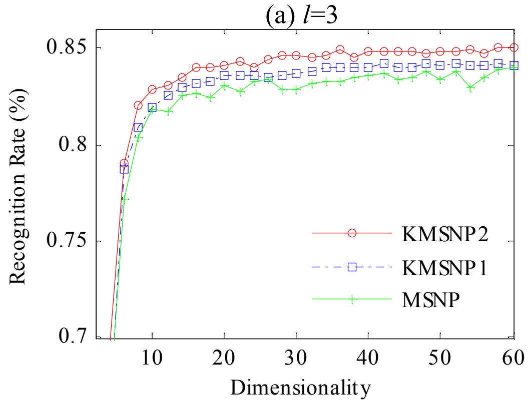


(b) l=25

Figure 5: Recognition rate (%) versus subspace dimension on USPS. (a) l=14. (b) l=25



(a) l=3



(b) l=5

Figure 6: Recognition rate (%) versus subspace dimension on ORL. (a) l=3. (b) l=5

Table 1: The maximal recognition rates(%) of algorithms versus the dimensions on USPS and ORL

	USPS l=14	USPS l=25	ORL l=3	ORL l=5
MSNP	0.7958 (38)	0.8395 (58)	0.7989 (59)	0.8690 (58)
KMSNP1	0.8007 (44)	0.8422 (42)	**0.8206 (28)**	**0.8825 (34)**
KMSNP2	**0.8077 (58)**	**0.8503 (60)**	0.8170 (34)	0.8780 (40)

CONCLUSION AND FUTURE WORK

On the basis of MSNP, we present a method called kernel-based manifold-oriented stochastic neighbor projection(KMSNP), we extend MSNP to a nonlinear scenario using techniques of kernel methods and present two kernel-based methods by two different strategies: KMSNP1 and KMSNP2. In this paper, it has been shown that the proposed methods may result in performance comparable with the previously used classification methods. Experimental results on USPS and ORL databases demonstrate the superior performance of the proposed methods. Our future work might consider class label information in the feature extraction. We believe that the labeled samples are of great value for classification purpose and recognition task. Therefore, supervised extension of KMSNP may result in more effective features for pattern recognition task. We are currently exploring these problems in theory and practice.

REFERENCES

AizermanI, M.A.; E.M. Braverman; and L.I. Rozonoer. 1964. "Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning." *Automation and Remote Control*, No.6 (Apr), 821-837.

Boser, B.; I. Guyon; and V. Vapnik. 1992. "A Training Algorithm for Optimal Margin Classifiers." *In Proc. Annual Conf. Computational Learning Theory* (D. Haussler, ed.). ACM Press, Pittsburgh, 144-152.

Chen, S.Y. and Y.F. Li, "Determination of Stripe Edge Blurring for Depth Sensing", IEEE Sensors Journal, Vol. 11, No. 2, Feb. 2011, pp. 389-390.

Chen, S.Y. and Z.J. Wang, "Acceleration Strategies in Generalized Belief Propagation", IEEE Transactions on Industrial Informatics, Vol. 8, No. 1, 2012, pp. 41-48.

Cherchi, E. and C.A. Guevara. 2012. "A Monte Carlo experiment to analyze the curse of dimensionality in estimating random coefficients models with a full variance–covariance matrix." *Transportation Research Part B: Methodological*, No.2 (Feb), 321-332.

Dardas, N.H. and N.D. Georganas. 2011. "Real-Time Hand Gesture Detection and Recognition Using Bag-of-Features and Support Vector Machine Techniques." *Instrumentation and Measurement, IEEE Transactions on*, No.11 (Aug), 3592-3607.

Hinton, G.; and S. Roweis. 2002. "Stochastic neighbor embedding." *Advances in neural information processing systems* No.15 (Dec), 833-840.

Helander, E.; T. Virtanen; and M. Gabbouj. 2012. "Voice Conversion Using Dynamic Kernel Partial Least Squares Regression." *Audio, Speech, and Language Processing, IEEE Transactions on*, No.3 (Mar), 806-817.

Mercer, J. 1909. "Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations." *Philosophical Transactions of the Royal Society of London. Series A. Containing Papers of a Mathematical or Physical Character*, No.209 (Jan), 415-446.

Maaten, L.; and G. Hinton. 2008. "Visualizing data using t-SNE." *Journal of Machine Learning Research*, No.9 (Aug), 2579-2605.

Tenenbaum, J.B.; V. de Silva; and J.C. Langford. 2000. "A global geometric framework for nonlinear dimensionality reduction." *Science*, No.290 (Aug), 2319-2323.

Wu, S.; Sun, M.; and Yang, J. 2011. "Stochastic neighbor projection on manifold for feature extraction." *Neurocomputing*, No.17 (Oct), 2780-2789.

Xiao, Y.; H. Wang; W. Xu; and J. Zhou. 2012. "L1 norm based KPCA for novelty detection." *Pattern Recognition*, NO.1 (Oct), 389-396.

Xue, J., et al., "Bound Maxima as a Traffic Feature under DDOS Flood Attacks", Mathematical Problems in Engineering, Vol. 2012, Article ID 419319, 2012, 20 pages.

Yang, J. et al. 2005. "KPCA plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition." *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, No.2 (Mar), 230-244.

Zhang, J., et al., "Normalized weighted shape context and its application in feature-based matching", Optical Engineering, Vol. 47, No. 9, Sep. 2008, 097201.

Zheng, J.; H. Qiu; Y. Jiang; and W. Wang. 2012. "Discriminative Stochastic Neighbor Embedding Analysis Method." *Computer-Aided Design & Computer Graphics,* No.11 (Nov), 1477-1484.

Zhang, Y.; J. An; and H. Zhang. 2013. "Monitoring of time-varying processes using kernel independent component analysis." *Chemical Engineering Science*, No.0 (Jan), 23-32.

AUTHOR BIOGRAPHIES

JianWei Zheng was born in Shengzhou, China and went to the Zhejiang University of Technology, where he studied computer science and technology and obtained his doctor degree in 2010. Now he is a lecturer of the Zhejiang University of Technology and his research interest covers machine learning and feature extraction. As a researcher, he has spirit of innovation and yielded some achievements in scientific research. His e-mail address is : `zjw@zjut.edu.cn`

Hong Qiu was born in Ningbo, China and went to the Zhejiang University of Technology, where she studied computer science and technology and obtained her undergraduate degree in 2012. Now she is a master student at Zhejiang University of Technology. Her research interest covers image processing and machine learning. She has a practical and realistic attitude towards science. Her e-mail address is : `qh@stu.zjut.edu.cn`

Qiongfang Huang was born in Zunyi, China and went to the Zhejiang University of Technology, where she studied computer science and technology and obtained her undergraduate degree in 2013. Now she is a master student at Zhejiang University of Technology. Her research interest covers image processing and machine learning. She has a great passion towards her major. Her e-mail address is : gdhqf@sina.cn

Wanliang Wang was born in Jiangshu, China and went to the TongJi University, where he studied control theory and control engineering and obtained his doctor degree in 2001. He has devoted nearly 20 years to educational work and now he is the dean of School of Computer Science and Technology at the Zhejiang University of Technology. As a researcher, he is leading a large research group in the field of simulation for small hydropower projects and many valuable achievements have made. His research interest covers intelligent algorithms and network control. His e-mail address is : `wwl@zjut.edu.cn`

Xinli Xu was born in Yuyao, China and went to the Zhejiang University of Technology, where she studied control theory and control engineering and obtained her doctor degree in 2003. Now she is an associate professor of the Zhejiang University of Technology and her research interest covers scheduling optimization, network science. Her e-mail address is : `xxl@zjut.edu.cn`