Hyperspectral feature recognition based on kernel PCA and relational perspective map

Hongjun Su (苏红军) and Yehua Sheng (盛业华)*

Key Laboratory of Virtual Geographic Environment (Ministry of Education), Nanjing Normal University, Nanjing 210046, China

**E-mail: shengyehua@njnu.edu.cn*

Received February 26, 2010

A novel joint kernel principal component analysis (PCA) and relational perspective map (RPM) method called KPmapper is proposed for hyperspectral dimensionality reduction and spectral feature recognition. Kernel PCA is used to analyze hyperspectral data so that the major information corresponding to features can be better extracted. RPM is used to visualize hyperspectral data through two-dimensional (2D) maps, and it is an efficient approach to discover regularities and extract information by partitioning the data into pieces and mapping them onto a 2D space. The experimental results prove that the KPmapper algorithm can effectively obtain the intrinsic features in nonlinear high dimensional data. It is useful and impressing for dimensionality reduction and spectral feature recognition.

OCIS codes: 100.5010, 280.4788, 300.6170.

doi: 10.3788/COL20100808.0811.

Hyperspectral data can provide fine and detailed spectral information by contiguous spectral range and narrow spectrum interval. As a result, some important and typical spectral features that cannot be expressed in broadband remote sensed data will be revealed and extracted obviously, which is significant to target identification, endmember extraction, anomaly diagnosis, fine classification, and sophisticated applications of remote sensing[1−3]. But for the given spectra acquired from field measurement with spectrometer or pixels on hyperspectral remote sensing image by aerial or spaceborne sensors, how to extract those significant features that can characterize the objects is still the most important topic for hyperspectral applications. Two critical issues arise from the above situations. One is "Hughes phenomenon"[4] caused by high dimensionality; the other is spectral feature recognition.

Dimensionality reduction (DR) is the approach to eliminate the impact of "Hughes phenomenon". Generally speaking, there are two kinds of DR approaches: one is band selection that selects some interesting bands or those bands with more information and weak inter-correlations; the other is feature extraction, which compresses all bands by certain mathematic transformation[5]. In the past few years, many DR methods have been presented in remote sensing with linear techniques such as principal component analysis (PCA)[6], linear discriminant analysis (LDA)[7], independent component analysis (ICA)[8], and so on. However, the nonlinear features which can be the major properties in spectral space for hyperspectral data are ignored.

On the other hand, different methods to extract characteristic spectral features have been researched in the recent years[9,10]. Manifold learning, as a new approach, has been applied to high dimensional data[11,12], and it can model the nonlinear features (manifold) of high dimensional data, while the nonlinear properties are well preserved. Such manifold learning algorithms as kernel PCA (KPCA)[13], multi-dimensional scaling (MDS)[14], isometric mapping (ISOMAP)[11], diffusion maps[15], local linear embedding[12,16], Laplacian eigenmap[17], and local tangent space alignment[18] have devoted to pattern recognition and machine learning while ignoring their applications in hyperspectral remote sensing. In fact, those manifold learning algorithms are able to efficiently reveal geometrical structures and regularities which indwell in high dimensional space from hyperspectral data[19]. In this letter, a novel joint KPCA and relational perspective map (RPM) method called KPmapper for DR and spectral feature recognition is proposed. Experiments are designed to validate the performance of the proposed algorithm.

PCA is one of the classic linear algorithms in pattern recognition[6]. Its nonlinear version, KPCA, is used to deal with hyperspectral data in this letter. The details of KPCA algorithm can be found in Ref. [13]. Suppose that there is a dataset of centered random vector $\boldsymbol{X} \in \mathbb{R}^n$ with N observations $x_i, i \in [1, \cdots, N]$. Firstly, we mapped the data onto another dot product space $\mathbb{Q}^d$ by $\phi : \mathbb{R}^n \to \mathbb{Q}^d, \mathbf{X} \to \phi(\mathbf{X})$. The covariance matrix $\phi(\mathbf{X})$ can be defined as $C_{\phi(\boldsymbol{X})} = \frac{1}{N}\sum_{i=1}^{N} \phi(x_i)\phi(x_i)^{\mathrm{T}}$. Let $\mathbf{v} \in \mathbb{Q}^d (v \neq 0)$ be an eigenvector of $C_{\phi(\mathbf{X})}$ that corresponds to a positive eigenvalue λ of $C_{\phi(\mathbf{X})}$. Similar to PCA, we can get

$$\lambda v = C_{\phi(\boldsymbol{X})}\mathbf{v}, \tag{1}$$

where $\mathbf{v} = \sum_{i=1}^{N} \boldsymbol{\alpha}_i \phi(x_i)$ is lying in the span of $\{\phi(x_1), \phi(x_2), \cdots, \phi(x_N)\}$. In addition, by multiplying Eq. (1) with $\phi(x_k)$ from the left and substituting the value of $\mathbf{v}$ into it, we can get $\lambda\phi(x)\cdot\mathbf{v}^k = \phi(x)\cdot C_{\phi(x)}\cdot\mathbf{v}^k$, where $\mathbf{v}^k = \sum_{i=1}^{N} \boldsymbol{\alpha}_i^k \phi(x_i), k \in [1, N]$. In order to construct the kernel function, we defined an $N \times N$ matrix $\mathbf{K}$ as $\mathbf{K}_{i,j} = \phi(x_i)\cdot\phi(x_j) = k(x_i, x_j)$. Consider an eigenvalue

decomposition for the expansion of $\boldsymbol{\alpha}^k$ by using the kernel matrix $\mathbf{K}$ as

$$\lambda \boldsymbol{\alpha}^k = \mathbf{K}\boldsymbol{\alpha}^k, \tag{2}$$

where $\boldsymbol{\alpha}^k = (\alpha_1^k, \alpha_2^k, \cdots, \alpha_N^k)^{\mathrm{T}}$. The solution of Eq. (2) $(\lambda_k, \boldsymbol{\alpha}^k)$ has to be normalized by $\lambda_k(\boldsymbol{\alpha}^k \cdot \boldsymbol{\alpha}^k) = 1$ and be centered by substituting centered $\mathbf{K}_{\mathrm{c}}$ for the matrix $\mathbf{K}$. $\mathbf{K}_{\mathrm{c}}$ is given by

$$\mathbf{K}_{\mathrm{c}} = \mathbf{K} - \mathbf{1}_N \cdot \mathbf{K} + \mathbf{1}_N \cdot \mathbf{K} \cdot \mathbf{1}_N, \tag{3}$$

where $\mathbf{1}_N$ is an $N \times N$ matrix whose elements are all equal to $1/N$. To get the k first principal components, $\phi(x)$ is mapped on the vector $\mathbf{v}^k$, namely,

$$\mathbf{v}^k \cdot \phi(x) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i^k (\phi(x_i) \cdot \phi(x)) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i^k k(x_i, x). \tag{4}$$

In this letter, Gaussian kernel is used for hyperspectral data, that is,

$$K_{\mathrm{Gauss}}(x_i, x_j) = \exp(-\left\|x_i - x_j\right\|^2). \tag{5}$$

Then, the hyperspectral data can be processed by KPCA algorithm with Gaussian kernel.

As a new unsupervised learning algorithm, RPM[20] provides a new approach for hyperspectral data processing. Its general ideas are as follows: the points in the high dimensional space can be represented by some independent variables, and a manifold in a high dimensional observed space is formed by those finite interactive variables; then the dimensionality can be reduced by unfolding the curled manifold in the observed space or finding the main internal variables. Its main objective is to extract the intrinsic geometrical structures and regularities from the dataset which indwells in the space when the space is represented as a manifold. Figure 1 depicts the model of the RPM method. It first maps data points onto the surface of a torus, then onto the flat rectangle by a vertical and a horizontal cut. RPM focuses on short distance information, while relaxing the constraints posed by long distance information.

Let the distance matrix of $\mathbf{X}$ be δ_{ij}, where $i, j = 1, \cdots, N$. The RPM algorithm maps data points x_i into image points t_i in a two-dimensional (2D) space (torus surface) with visual distance d_{ij} between the image points. So, the image points $t_i, i \in [1, \cdots, N]$ provide a 2D visualization of distance transformation of the initial data. Furthermore, δ_{ij} and d_{ij} are the relational distance and image distance matrix, respectively. Mathematically, RPM uses the gradient descent algorithm to minimize the following energy function:

$$E_p = \sum_{i<j} \frac{\delta_{ij}}{p d_{ij}^p} \tag{6}$$

with $E_0 = -\sum_{i<j} \delta_{ij} \ln(d_{ij})$, where p is the rigidity, a parameter that controls how strong the algorithm biases local distance information, and p can be a value between –1 and $+\infty$. The forces between the points are defined by

$$f_{ij} = \frac{\partial E_p}{\partial d_{ij}} = -\frac{\delta_{ij}}{d_{ij}^{p+1}}, \quad i < j. \tag{7}$$

Assume that $\mathbf{T} = [0, w] \times [0, h] \in \mathbb{R}^2$ denotes the rectangle plane of width w and height h in the 2D Cartesian coordinate system, a torus mapping can be represented as a visualization of

$$\varphi : \mathbf{X} \to \mathbf{T}, \ x_i \to t_i = (x_i, y_i). \tag{8}$$

Then the distance between t_i and t_j from $\mathbf{T}$ can be defined as

$$\begin{aligned} d(t_i, t_j) = &\min(\left|x_i - x_j\right|, w - \left|x_i - x_j\right|) \\ &+ \min(\left|y_i - y_j\right|, h - \left|y_i - y_j\right|). \end{aligned} \tag{9}$$

With the above distance function, the opposite edges of $\mathbf{T}$ are actually stuck together, so that it becomes topologically equivalent to a torus in Fig. 1. The goal of RPM algorithm is thus to find a torus mapping ϕ in Eq. (8) that minimizes Eq. (6).

The RPM algorithm adopts the Newton-Raphson (NR) method to minimize Eq. (6). So, we can get the first and second order partial derivatives of E with respect to all variables x_i and y_i:

$$\begin{aligned} \frac{\partial E}{\partial x_i} &= \sum_{k \neq i} -\frac{\delta_{ik}}{d_{ik}^{p+1}} \frac{\partial d_{ik}}{\partial x_k} = \sum_{k \neq i} h_{ik} f_{ik}, \\ \frac{\partial^2 E}{\partial x_i^2} &= (p+1) \sum_{k \neq i} \frac{f_{ik}}{h_{ik}}, \end{aligned} \tag{10}$$

where $h_{ik} = \partial d_{ik} / \partial x_k$. Then we can get the iterative formula to find the minimum energy by

$$\begin{aligned} x_i^{(m+1)} &= x_i^{(m)} - \frac{\partial E / \partial x}{\partial^2 E / \partial x^2} \\ &= x_i^{(m)} - \frac{1}{p+1} \frac{\sum\limits_{k \neq i} h_{ik} f_{ik}}{\sum\limits_{k \neq i} f_{ik} / h_{ik}}. \end{aligned} \tag{11}$$

Replacing the constant $\frac{1}{p+1}$ in Eq. (11) by a parameter $c^m = r a^m$, which is the learning speed at the step m, we can get the modified version as

$$x_i^{(m+1)} = x_i^{(m)} - c^{(m)} \frac{\sum\limits_{k \neq i} h_{ik} f_{ik}}{\sum\limits_{k \neq i} f_{ik} / h_{ik}}. \tag{12}$$

In addition, $c^{(m)}$ should approach zero as m increases, and r is the initial learning speed, $a \in (0; 1)$. Both r and a are determined empirically.

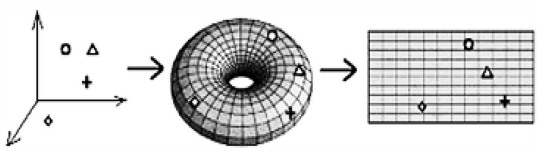

Fig. 1. Model of the RPM method[20,21].

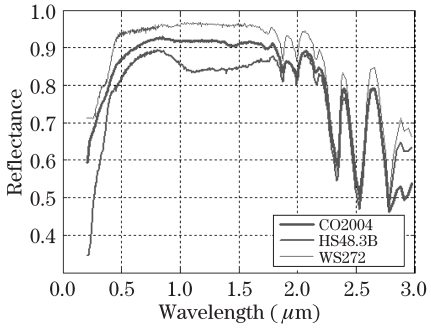

Fig. 2. Original curves of Calcite data from USGS spectral library.

From the above analysis, we can see that KPCA algorithm has the ability of nonlinear DR for high dimensional data by mapping the original data into a featured space. RPM is a robust tool for extracting intrinsic geometrical structure and regularities by mapping the data onto a 2D space without overlapping. The above two points encourage us to use them together for DR and spectral feature recognition. So, a novel DR and spectral feature recognition method called KPmapper, which combines KPCA and RPM, is put forward as the following steps.

1) Calculate the centered vector $\mathbf{X} \Leftarrow \mathbf{X} - E\{X\}$ of original data $\mathbf{X} \in \mathbb{R}^n$ with N observations $x_i, i \in [1, \cdots, N]$.

2) Get the Gaussian kernel matrix and the entered $\mathbf{K}_c$ with Eqs. (5) and (3), respectively.

3) Extract the first k principal components by Eq. (4).

4) Select a mapping ϕ that maps k first principal components to randomly selected points $\{(x_i^{(0)}, y_i^{(0)}) \in T | i = 1, \cdots, N\}$, set $m = 0$.

5) Calculate $x_i^{(m+1)}$ and $y_i^{(m+1)}$according to Eq. (12).

6) If the total change of $\sum_{i=1}^{N} \left|x_i^{(m+1)} - x_i^{(m)}\right| + \left|y_i^{(m+1)} - y_i^{(m)}\right|$ is smaller than a threshold (e.g., 1×10^{-5}), go to step 7.

7) Set $m = m + 1$; go to step 5.

8) If the total change of $\sum_{i=1}^{N} \left|x_i^{(m+1)} - x_i^{(m)}\right| + \left|y_i^{(m+1)} - y_i^{(m)}\right|$ is greater than a threshold (e.g., 1×10^{-5}), stop.

It should be noted that the image coordinates are constrained to a limited space (that is topologically equivalent to the surface of a torus) so that the image distances d_{ij} cannot grow unlimited.

To validate the proposed method, we designed some experiments. The spectral data used in this letter were obtained from US Geological Survey (USGS) spectral library (http://speclab.cr.usgs.gov/spectral-lib.html). There are 499 spectral data of 444 objects, of which the Calcite CO2004 collected in Alligator Ridge Mine, Nevada was selected as the test example data to validate our proposed method. The spectral data ranging from 0.2 to 3.0 μm are corrected to absolute reflectance using a US National Institute of Standards and Technology (NIST) Halon standard and can be used as the reference in objects recognition. Figure 2 depicts the original curve map for the Calcite data, and each record in the dataset is taken as an internal body object randomly positioned in the map.

Table 1. Principal Component Results with Gaussion Kernel

Component	P0	P1	P2	P3	P4	P5	P6
Gaussian	0.2057	0.0539	0.0077	0.0033	0.0018	0.0006	0.0003
Contribution (%)	75.27	19.72	2.82	1.21	0.66	0.22	0.11

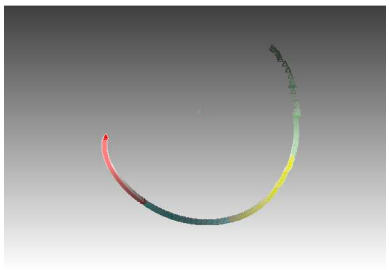

Fig. 3. Gaussion kernel results of KPCA (colorful online).

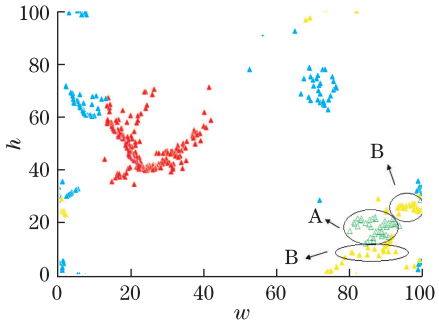

Fig. 4. 2D RPM mapping for Calcite CO2004.

In the experiment, the spectral data of Calcite CO2004 was processed by KPCA algorithm with Gaussian kernel and 7 principal components were obtained. It should be noted that the Gaussian kernel parameter γ is set as 1. From the results shown in Table 1, we can see that the first three principal components contribute to 97.8% of all components. KPCA results map is shown in Fig. 3 and the records which have the same characteristics are concentrated in the same regions.

The data derived from KPCA is then partitioned into pieces and mapped into a 2D space without overlapping using RPM algorithm. In RPM, the learning speed parameters r and a are set to 0 at the beginning, and they will change in the next steps. Figure 4 shows the results using RPM algorithm, and all the data points are mapped into the 2D map which also called torus surface. From Refs. [20] and [21], we can see that the relations between the values of the parameters w (the width of the rectangle plane), h (the height of the rectangle plane), and the visualization of hyperspectral data are hard to determine. In order to get the best visualization results, 10 experiments have been done for every couple of parameters w and h (for example, $w = 100$, $h = 100$ or $w = 100$, $h = 1000$), and it is found that when $w = h = 100$, the results are the best (Fig. 4).

From Fig. 4, we can see that all the spectral points are partitioned into four parts; like the clustering algorithm, the points which have the nearer distance are mapped in the same group such as the red, cyan, yellow (B), and

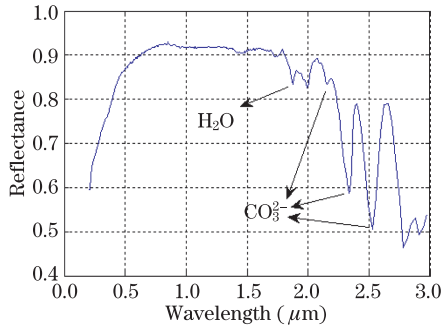

Fig. 5. Original curve and derived spectral absorption features using KPmapper method for Calcite CO2004.

green (A) groups on the 2D map. In the results, some most important spectral features are extracted and the points with the same characteristics are clustering into one group. Combining Fig. 4 with Fig. 5, we can get the following results. The red group presented the curve that between 0.2131 and 0.859 μm in the whole origin spectral curve of Calcite CO2004, the cyan group between 0.871 and 1.4785 μm, the yellow (B) group between 1.4835 and 2.115 μm, and the green (A) group between 2.125 and 2.976 μm.

Meaningful results can be found from Figs. 4 and 5 if we take the spectral absorption features of Calcite into consideration. We know that Calcite has three main absorption features of anion CO_3^{2-} located in 2.3 – 2.4, 2.1 – 2.2, and 2.5 μm, respectively and one main absorption feature of H_2O located at 1.9 μm[22]. By comparing the extracted feature results of KPmapper with the original Calcite CO2004 data, we can see that the A and B groups are in fact the absorption features of CO_3^{2-} and H_2O, respectively. It illustrated that our proposed method is an effective tool for spectral feature recognition. In addition, we can use this method for mineral identification and geological mapping in the future.

The experimental results also proved that KPCA algorithm has powerful ability for DR using a nonlinear method. RPM is more preferable to reveal details in an easy way by partitioning complex dataset and presenting them in a non-overlapping manner. From geometric point of view, our proposed KPmapper algorithm has combined advantages of these two algorithms, it can visualize data through 2D maps and reveal the important features directly to the human eyes; it has extremely high capability for DR and pattern recognition.

It should be noted that although only Calcite CO2004 experiment is considered, we have tested all the Calcite and some other mineral objects spectrum data in the USGS spectral library, and the experimental results are similar. These results are not included here.

In conclusion, we propose a novel method for DR and spectral feature recognition, which is called KPmapper, based on KPCA and RPM algorithms. Instead of the original data for RPM, the eigenvalue data derived from KPCA are analyzed and used as the input of RPM. By the experiments on the spectral data from USGS spectral library, it is proved that the proposed approach can be used in hyperspectral DR, and is suitable to discover spectral features, effectively identify and discriminate typical minerals based on their spectra.

This work was partially supported by the National Natural Science Foundation of China (No. 40901200), the China Scholarship Council (No. 2009686004), and the Outstanding Postgraduate Dissertation Cultivating Program of Nanjing Normal University (No. 1243211601040).

References

1. Y. Du, C.-I. Chang, H. Ren, C.-C. Chang, J. O. Jensen, and F. M. D'Amico, Opt. Eng. **43,** 1777 (2004).
2. C.-I. Chang (ed.), *Hyperspectral Data Exploitation*: *Theory and Applications* (Wiley, Hoboken, 2007).
3. X. Liu, H. Zhao, and N. Li, Acta Opt. Sin. (in Chinese) **3,** 844 (2009).
4. G. F. Hughes, IEEE Trans. Inf. Theory **14,** 55 (1968).
5. H. Su, Y. Sheng, and P. Du, in *Proceedings of the ISPRS* **7,** 279 (2008).
6. Q. Miao and B. Wang, Chin. Opt. Lett. **6,** 104 (2008).
7. T. V. Bandos, L. Bruzzone, and G. Camps-Valls, IEEE Trans. Geosci. Remote Sens. **47,** 862 (2009).
8. V. Zarzoso and P. Comon, IEEE Trans. Neural Networks **21,** 248 (2010).
9. P. Du, H. Su, and W. Zhang, Proc. SPIE **6752,** 675204 (2007).
10. G. He and L. Peng, Chinese J. Lasers (in Chinese) **36,** 2983 (2009).
11. J. B. Tenenbaum, V. de Silva, and J. C. Langford, Science **290,** 2319 (2000).
12. S. T. Roweis and L. K. Saul, Science **290,** 2323 (2000).
13. B. Schölkopf, A. Smola, and K.-R. Müller, Neural Computation **10,** 1299 (1998).
14. A. Amar, Y. Wang, and G. Leus, IEEE Signal Processing Lett. **17,** 473 (2010).
15. S. Lafon and A. B. Lee, IEEE Trans. Pattern Anal. and Machine Intell. **28,** 1393 (2006).
16. G. Chen and S.-E. Qian, J. Appl. Remote Sens. **1,** 013509 (2007).
17. M. Belkin and P. Niyogi, Neural Computation **15,** 1373 (2003).
18. L. Ma, M. M. Crawford, and J. W. Tian, Electron. Lett. **46,** 497 (2010).
19. C. M. Bachmann, T. L. Ainsworth, and A. R. Fusina, IEEE Trans. Geosci. Remote Sens. **43,** 441 (2005).
20. J. X. Li, Information Visualization **3,** 49 (2004).
21. R. Karbauskaitė, V. Marcinkevičus, and G. Dzemyda, Technological and Economic Development of Economy **12,** 289 (2006).
22. S. J. Gaffey, American Mineralogist **71,** 151 (1986).