http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 11 (12): 1751-1757, 2012
ISSN 1812-5638 / DOI: 10.3923/itj.2012.1751.1757

Adaptive Quasiconformal Kernel Principle Component Analysis for MSTAR SAR Recognition

Chuang Lin, Xirui Sun, Binghui Wang and Xiangjie Kong
School of Software, Dalian University of Technology, No. 8 Road, Developing District, Dalian 116000, China

Abstract: In this study, an Adaptive Quasiconformal Kernel Principle Component Analysis (AQKPCA) algorithm is proposed. We apply a quasiconformal kernel to generate a new feature space (quasiconformal feature space) to maximize the distance of each class in terms of Maximum Margin Criterion (MMC) and then, the quasiconformal kernel based KPCA and corresponding classifiers are adopted for linear classification in the feature space. The extensive experiments are conducted on MSTAR SAR dataset. The experimental results indicate the superiority of our method compare with other algorithms.

Key words: Adaptive quasiconformal kernel, maximum margin criterion, SAR recognition, kernel principle component analysis

INTRODUCTION

Principal Component Analysis (PCA) is a well-known method for dimensionality reduction and feature extraction. In order to maintain as much as the input data structure, PCA algorithm calculates the eigenvectors (lower dimension) of the covariance matrix of input training samples substituted for input training vectors (higher dimension). However, in most case, when we face nonlinear classification problem, PCA cannot work as it is neither a nonlinear dimensionality reduction method nor a supervised algorithm. Thus, kernel trick is applied to overcome the problem of linear division in feature space. Kernel method is originally used for SVM. Later, it has been generalized into many algorithms calculating the term of inner products. Kernel PCA is one type of nonlinear PCA which is developed by generalizing the kernel into PCA (Scholkopf *et al*., 1998) KPCA firstly maps the original input training samples into feature space with high dimension. Then, PCA is adopted for dimensionality reduction in this feature space generalized by kernel. Since then, PCA and KPCA have become the popular algorithms for feature extraction.

However, when improper kernel is used for feature extraction, the geometric structure of data (sample) that maps into feature space could be even worse. Different kernel matrices determine different data structure in feature space. We can see that an appropriate kernel can greatly improve the performance of kernel learning. In here, optimized kernel can be an effective method for kernel learning. The main idea of optimized kernel is to generate a new kernel that rely on the original kernel and input data structure. The new kernel is the so-called optimized kernel. Adaptive quasiconformal kernel belongs to the field of optimized kernel. Here, we introduce a adaptive quasiconformal kernel which was studied in the previous work (Pan *et al*., 2008; Heisterkamp *et al*., 2001; Xiong *et al*., 2005; Daoudi and Idrissi, 2010), where the geometrical structure of data in the feature space is changeable with different basic kernels and different parameters of the quasiconformal kernel.

Recently, many kernel and optimal kernel methods for MSTAR SAR image target recognition have been proposed. MSTAR SAR image is provided by Defense Advanced Research Project Agency and Air Force Research Laboratory (DARPA/AFRL) which contains three classes of data (BMP2, BTR70 and T72), in the experiments. Ping *et al*. (2003) applied KPCA to SAR recognition. Yi *et al*. (2009) and Jing *et al*. (2008) proposed the optimized KPCA and local-feature based methods on the basis of Ping *et al*. (2003). Ying *et al*. (2010) proposed a new Target feature extraction methods based on Kernel Singular Value Decomposition (KSVD) and PCA. Lu *et al*. (2008) proposed KPCA based on multi-linear principal component analysis.

In this study, the method of adaptive quasiconformal kernel with principle component analysis is based on MMC (Li *et al*., 2006). To investigate the performance of

Corresponding Author: Chuang Lin, School of Software, Dalian University of Technology, No. 8 Road, Developing District, Dalian 116000, China

the extraction in feature space, with the methods proposed in this study, MSTAR SAR image set is applied as the experiment database. Experiments results demonstrate the superiority of the proposed algorithm compared with conventional KPCA algorithm in most cases.

AQKPCA ALGORITHMS

Kernel principle component analysis: As mentioned in the previous part, KPCA is one approach by generalizing PCA into non-linear case using the kernel trick. Given a set of input training vectors $x_i \in \mathbb{R}^n (i = 1, 2, ..., 1)$, each of which is of n dimension and the number of training sample is l. X_i is mapped into $\Phi(x_i)$ with m dimension.

KPCA firstly maps the original input vectors x_i into a higher dimensional feature space $\Phi(x_i)$. Then, with the same idea as PCA, eigenvectors in $\Phi(x_i)$ are calculated. By mapping x_i into $\Phi(x_i)$, KPCA solves the eigenvalue problem as follows:

$$\lambda_i w_i = C w_i \qquad i = 1, 2, ..., l \tag{1}$$

Where:

$$C = \frac{1}{l}\sum_{i=1}^{1} \Phi(\boldsymbol{x}_i)\Phi(\boldsymbol{x}_i)^T$$

is the sample covariance matrix of $\Phi(x_i)$. λ_i and w_i are one of the non-zero eigenvalues and eigenvector of the covariance matrix C.

By using kernel trick (1) can be transformed as follows:

$$\tilde{\lambda}_i \boldsymbol{\alpha}_i = K\boldsymbol{\alpha}_i \quad i = 1, 2,..., l \tag{2}$$

where, K is the l×l kernel matrix. The value of each element k_{ij} in K can be regarded as the distance between $\Phi(x_i)$ and $\Phi(x_j)$. It is the same as the inner product of two vectors $\Phi(x_i)$ and $\Phi(x_j)$. That is, $k(x_i, x_j) = <\Phi(x_i), \Phi(x_j)>$. $\tilde{\lambda}_i$ is one of the eigenvalues of K, satisfying $\tilde{\lambda}_i = l\lambda_j$ and α_i is the corresponding eigenvector of K, satisfying:

$$w_i = \sum_{j=1}^{1} \alpha_i(j)\Phi(x_j)$$

where, $\alpha(j)$ j = 1, 2,...., l are the components of α_i. Each eigenvector α_i should be normalized as:

$$\tilde{\alpha}_i = \frac{\alpha_i}{\sqrt{\tilde{\lambda}_i}}$$

Finally, for the training sample x_t, $(t \in [1, l])$, the corresponding principle component $z_t(i)$, (i = 1, 2,..., l) are calculated as follows:

$$z_t(i) = \boldsymbol{w}_i^T \Phi(\boldsymbol{x}_t) = \sum_{j=1}^{1} \tilde{a}_i(j) k(x_j, x_t), \ \ i = 1, 2,....., l \tag{3}$$

where, $z_t(i)$, (i = 1, 2,...., l) denote the ith kernel PCA components for x_t. However, in most cases, there is no need to calculate all the principle components of x_t, only the m largest of them is used for the sake of calculating convenience.

For the given training samples x_i and x_j, the Euclidean distance between two input samples in feature space is calculated as follows:

$$\text{Dis}^2(x_i, x_j) = \|\Phi(x_i) - \Phi(x_j)\|^2 = K_{ii} - 2K_{ij} + K_{jj} \tag{4}$$

The next part presents the adaptive quasiconformal kernel.

Adaptive quasiconformal kernel: Adaptive quasiconformal kernel is a kind of optimized kernel. Some algorithms have been proposed over the past decade. Amari and Wu have modified a support vector machine with a quasiconformal mapping (Amari and Wu, 1999). Heisterkamp and Peng adopted conformal and quasiconformal mapping in the field of image retrieval (Heisterkamp *et al*., 2001).

If f(x) is a positive real valued function of x, then a new kernel can be created by K_0. Where K_0 denotes the basic kernel matrix responding to kernel function $k_0(x, y)$ such as Gaussian kernel and Polynomial kernel:

$$\tilde{k}(x, y) = f(x)f(y)k_0(x, y) \tag{5}$$

We call it a quasiconformal kernel. Let us design $\tilde{K}$ is a quasiconformal kernel matrix and K_0 denotes the basic kernel matrix. Like Eq. 4, the new Euclidean distance metric of $\tilde{k}(x, y)$ is shown as:

$$\begin{aligned}\widetilde{\text{Dis}}^2(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \tilde{K}_{ii} - 2\tilde{K}_{ij} + \tilde{K}_{jj} \\ &= f^2(\boldsymbol{x}_i)K_{0(ii)} - 2f(\boldsymbol{x}_i)f(\boldsymbol{x}_j)K_{0(ij)} + f^2(\boldsymbol{x}_j)K_{0(jj)}\end{aligned} \tag{6}$$

where, $K_{0(ij)}$ is the ith row, jth column element of matrix K_0.

Here, Amari and Wu (1999) expanded the spatial resolution in the margin of a SVM by using:

$$f(\boldsymbol{x}) = \sum_{i \in SV} a_i e^{-s\|x-\tilde{x}_i\|^2} \quad (7)$$

where, SV is a set of support vector, $\tilde{x}_i$ is the ith support vector. α_i is a positive number representing the contribution of $\tilde{x}_i$, we call them "expansion coefficients" and δ is a free parameter. In this study, we generalize Amari and Wu's method as follows:

$$f(x) = b_0 + \sum_{n=1}^{N_{XV}} b_n e(x, \tilde{x}_n) \quad (8)$$

Where:

$$e(x, \tilde{x}_n) = e^{-\delta\|x-\tilde{x}_n\|^2}$$

in this study and δ is a free parameter. It can be seen from Eq. 8 that the quasiconformal kernel is determined by the "expansion coefficients" with the determinative "expansion vectors" and a free parameter δ. And here, $\tilde{x}_n (1 \le n \le N_{XV})$, are called the "expansion vectors (XVs)" in this paper. N_{XV} is the number of XVs and $b_i \in R$ is the "expansion coefficient" associated with $\tilde{x}_n$.

So, the first thing is to choose all these procedural parameters. In this study, XVs, which chosen to solve the expansion coefficients, are defined as follows:

$$e(x, \tilde{x}_n) = e(x, \bar{x}_I) = \exp(-\delta\|x-\bar{x}_I\|^2) \quad (9)$$

where, $\bar{x}_I$ is the mean of each class. In here, we desire to obtain the "expansion coefficients" b_i by VXs instead of f(x). According to Eq. 7, it can be reformed as follows:

$$\Lambda 1_l = E_\beta \quad (10)$$

where, $\Lambda = \text{diag}(f(x_1), f(x_2), ..., f(x_l))$, $\beta = [b_0, b_1, b_2, ..., b_{N_{xv}}]^T$ (i = 0, 1, 2,..., $b_{N_{xv}}$), 1_l is an l-dim vector whose entries equal to unity and E is defined as:

$$E = \begin{bmatrix} 1 & e(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_1) & \cdots & e(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_{N_{XVs}}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e(\boldsymbol{x}_M, \bar{\boldsymbol{x}}_1) & \cdots & e(\boldsymbol{x}_M, \bar{\boldsymbol{x}}_{N_{XVs}}) \end{bmatrix}$$

In this study, MMC (Li *et al*., 2006; Zheng *et al*., 2005) is adopted to solve the "expansion coefficient" vector β.

MAXIMUM MARGIN CRITERION

MMC is first proposed by Li *et al*. (2006) in order to extract the feature by maximizing the average margin between different classes of data in the feature space. Pan *et al*. (2008) use it to measure the average margin between different classes of data in the quasiconformal kernel mapping space. The average margin between two classes C_i and C_j in the new kernel mapping space is defined as:

$$\tilde{J}_{MMC} = \frac{1}{2l}\sum_{i=1}^{C}\sum_{j=1}^{C} l_i l_j d(C_i, C_j) \quad (11)$$

where, l_i, l_j denotes the number of samples in C_i and C_j and the margin d (C_i, C_j) is defined as:

$$d(C_i, C_j) = d(m_i^\Phi, m_j^\Phi) - (s(C_j)) \quad (12)$$

where d (m^Φ_i, $m\Phi_j$) means the distance between mean vectors of the class C_i and C_j , $s(C_i)$ and $s(C_j)$ denote the measure of the scatter of the class C_i and C_j, respectively. Pan *et al*. (2008) used the within-class scatter matrix of class i and j to represent $s(C_i)$ and $s(C_j)$, that is, $s(C_i) = tr(\tilde{S}_i^\Phi)$ and $s(C_j) = tr(\tilde{S}_j^\Phi)$, Let us design $\tilde{S}_b^\Phi$ and $\tilde{S}_w^\Phi$ be the between-class scatter matrix and the within-class scatter matrix in the quasiconformal kernel mapping space. We say the objective function $\tilde{J}_{MMC}$ as follows:

$$\tilde{J}_{MMC} = tr(\tilde{S}_b^\Phi) - tr(\tilde{S}_w^\Phi) \quad (13)$$

Eq. 13 can be reformed as:

$$\tilde{J}_{MMC} = \mathbf{1}_l^T(\tilde{B} - \tilde{W})\mathbf{1}_n \quad (14)$$

Where:

$$\tilde{B} = \text{diag}(\frac{1}{l_1}\tilde{K}_{11}, ..., \frac{1}{l_C}\tilde{K}_{CC}) - \frac{1}{l}\begin{bmatrix} \tilde{K}_{11} & \cdots & \tilde{K}_{1C} \\ \vdots & \ddots & \vdots \\ \tilde{K}_{C1} & \cdots & \tilde{K}_{CC} \end{bmatrix},$$
$$\tilde{W} = \text{diag}(\tilde{k}_{11}, ..., \tilde{k}_{ll}) - \text{diag}(\frac{1}{l_1}\tilde{K}_{11}, ..., \frac{1}{l_C}\tilde{K}_{CC}) \quad (15)$$

The proof of Eq. 14 is shown in appendix. Where $\tilde{K}_{ij}(i = 1, 2, ..., C; j = 1, 2, ..., C)$ represent the sub-matrices of quasiconformal kernel $\tilde{K}$ and the size of $\tilde{K}_{ij}$ is $l_i \times l_j$.

For the basic kernel K_0, the matrices B and W are similar with Eq. 15:

$$B = \text{diag}(\frac{1}{l_1}K_{0(11)}, ..., \frac{1}{l_C}K_{0(CC)}) - \frac{1}{l}\begin{bmatrix} K_{0(11)} & \cdots & K_{0(1C)} \\ \vdots & \ddots & \vdots \\ K_{0(C1)} & \cdots & K_{0(CC)} \end{bmatrix} \quad (16)$$

$$W = \text{diag}(k_{0(11)}, ..., k_{0(ll)}) - \text{diag}(\frac{1}{l_1}K_{0(11)}, ..., \frac{1}{l_C}K_{0(CC)})$$

where $K_{0(ij)}$ denotes sub-matrices of basic kernel K_0, Let K_0 and $\tilde{K}$ denote the basic kernel matrix and quasiconformal kernel matrix, the same definition as above. From formula (Eq. 5), we have:

$$\tilde{K} = [f(x_i)\, f(x_j)\, k_0(x_i, x_j)_{1\times 1}] = \Lambda K_0 \Lambda \quad (17)$$

From Eq. 17, we can obtain:

$$\tilde{B} = \Lambda B \Lambda, \tilde{W} = \Lambda W \Lambda \quad (18)$$

Thus, form 10-18 and the objective function (14) can be written as:

$$\tilde{J}_{MMC} = \beta^T E^T (B - W) E\beta \quad (19)$$

Given a basic kernel k_0 (x, y) and relative quasiconformal kernel coefficients, E^T (B-W)E is a constant matrix. So, $\tilde{J}_{MMC}$ is a function with its variable β, and it is reasonable to seek the optimal expansion coefficient vector β by maximizing $\tilde{J}_{MMC}$.

Now the problem can be regarded as follows:

$$\arg\max_\beta \beta^t E_t (B\text{-}W) E\beta \text{ S.t } \beta^t\beta\text{-}1 = 0 \quad (20)$$

By using Lagrangian method, we finally calculate:

$$E^t (B\text{-}W)\, E\beta = \lambda\beta \quad (21)$$

We can obtain the optimal expansion coefficient vector β, that is, the eigenvector of E^t (B-W)E corresponding to the largest eigenvalue. It is easy to see that the quasiconformal kernel with β is adaptive to the input data, so we call it adaptive quasiconformal kernel.

The next part introduces the procedures of AQKPCA algorithm.

AQKPCA algorithm: The main procedures of AQKPCA are mainly describes as follows:

- **Input:** A set of training face images $\{x_1, x_2, \ldots\ldots x_l\}$, each image is represented with a n-dimensional vector
- **Output:** A low m-dimensional representation z_t of x_t t = 1, 2,....., l

Step 1: Calculate B-M with selected basic kernel function k_0 (x, y), δ and $\{\bar{x}_i\}_{i=1,2,\ldots,NXVs}$

Step 2: Obtain the adaptive expansion coefficients vector β^* by solving equation (21)

Step 3: Calculate the adaptive quasiconformal kernel $\tilde{k}$ and its kernel matrix $\tilde{K}$ with the optimal expansion coefficient vector β^* and real value of f(x), f(y)

Step 4: Calculate the main component z_t by adopting Kernel Principle Component Analysis (KPCA)

Step 5: Adopt different classifiers for MSTAR classification

RESULTS AND DISCUSSION

The present, experiments are designed to evaluate the performance of the proposed algorithm. Here, we present all the experiments on MSTAR SAR image (Fig. 1) set provided by Defense Advanced Research Project Agency and Air Force Research Laboratory (DARPA/AFRL).

We select images of BMP2sn-c21, BTR70sn-c71 and T72sn-132 in 17 depression angle as the training samples (numbers of each class are 233, 233, 233). The testing samples are selected as BMP2 sn-9563, BMP2 sn-9566 and T72 sn-812, T72sn-s7 in 15 depression angle (numbers of each class are 195, 196, 195, 191). The testing targets have small configuration differences to the training targets. There is a need to explain that, in this paper, AQKPCA extracts features of all MSTAR images with different aspect angles directly and the process does not need to form different aspect windows and before recognition, we chipping images are chipped into 48*48 pixels. For each class of MSTAR SAR, 200 images are processed as training samples, other 150 for recognition test.

In order to compare different kernel PCA algorithms which maps data into different kernel space, we conduct the first experiments to compare the performance of a series of PCA methods (PCA, KPCA and KOPCA) with

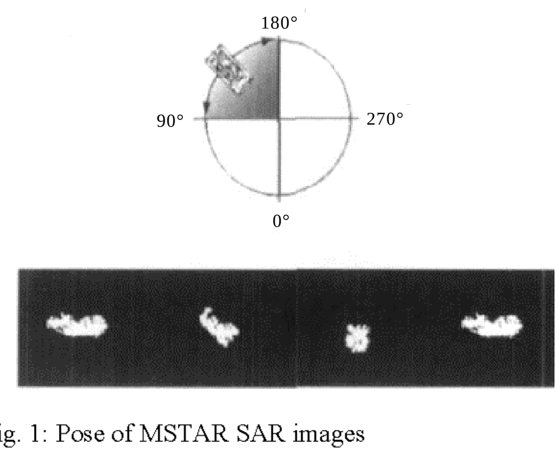

Fig. 1: Pose of MSTAR SAR images

the proposed method under NN classifier. In KOPCA, the Xiong's method (Daoudi and Idrissi, 2010) is adopted. Different from the algorithm proposed in this paper, Fisher Criterion is an iterative method. In here, we adopt MMC, which can avoid iterating procedures. The second experiment is carried out to investigate the performance of AQKPCA under different classifier. In this part, Nearest Neighbor Classifier (NNC), linear regression classifier (LRC) and sparse representation-based classifier are applied for AQKPCA algorithm.

PCA KPCA and KOPCA AQKPCA comparison: In this experiment, PCA, KPCA, KOPCA and AQKPCA are conducted to test the performance of all the PCA-like algorithms. In here, we adopt the Gaussian kernel function:

$$k(x,y) = \exp\left(\frac{-\|x-y\|^2}{s}\right)$$

With the parameter σ ranges from le 1 to le 10 and Polynomial kernel function:

$$k(x, y) = (<x.y>+1)^d,\ d \in N$$

With parameter d range from 1 to 10. We set parameter of the function e(.) in formula (9) as δ = le 6. Moreover, the feature dimensions are set 100 in all KPCA, KOPCA and AQKPCA methods. In order to reflect the performance of adaptive quasiconformal kernel in real application, the simplest Nearest Neighbor (NN) classifier is selected.

The recognition of PCA is 66.22% conducted on MSTAR SAR dataset with 200 training samples and 150 test samples for each class. Table 1 presents the kernel method of PCA algorithms. The polynomial kernel function is adopted.

From Table 1 and 2, we can see when adopting the adaptive quasiconformal kernel with appropriate coefficients of δ, the proposed methods outperforms other PCA methods in both Gaussian kernel and polynomial kernel. In other words, class separability in quasiconformal feature space seems more promising for linear classification.

Classifier comparison: Since, PCA is not a supervised method for classification, neither nor KPCA and AQKPCA methods. Thus, there is a need to carry out an experiment of AQKPCA with different classifiers. In this experiment, Linear Regression (LR) and Sparse Representation-based (SR) Classifiers are adopted for recognition. Here, we consider Gaussian kernel function and Polynomial kernel function. The as the previous experiments, parameters of σ in Gaussian function and d in Polynomial are ranges from le 1 to le 10 and 1 to 10, respectively. When dealing with coefficients of the function (8), we choose δ = le 6. Table 3 shows the recognition rate using LRC and SRC under different Polynomial coefficients. Table 4 gives the same experiment using Gaussian function.

From Table 3 and 4, we can see that Polynomial function seems more suitable for MSTAR SAR recognition than Gaussian function. The adaptive quasiconformal kernel with appropriate classifier can improve the performance of test result. Moreover, compare Table 1 and 3, AQKPCA with sparse representation-based classifier (SRC) outperforms the others (NNC and LRC). There is about 10-15% improvement of recognition rate. Seeing Table 3 and 4, we can also learn that AQKPCA: SRC are more suitable for Polynomial function, AQKPCA: NN are relatively stable for Gaussian one.

Table 1: Recognition rates of KPCA, KOPCA and AQKPCA with polynomial kernel using NN classifier

Polynomial kernel (d)	1	2	3	4	5	6	7	8	9	10
KPCA: NN	67.56	68.00	67.11	66.67	66.22	75.11	75.33	75.33	77.56	80.00
KOPCA: NN	77.78	77.56	77.56	77.11	77.11	76.67	76.00	75.56	75.33	74.67
AQKPCA: NN	85.11	85.33	87.78	83.56	83.33	84.22	82.89	80.89	81.33	81.33

Table 2: Recognition rates of KPCA, KOPCA and AQKPCA with Gaussian kernel using NN classifier

Gaussian kernel (σ)	le 1	le 2	le 3	le 4	le 5	le 6	le 7	le 8	le 9	le 10
KPCA: NN	62.00	62.00	62.00	66.11	68.00	67.11	62.00	61.33	34.22	31.11
KOPCA: NN	75.11	75.33	75.33	77.56	80.00	77.78	76.67	75.56	45.33	44.89
AQKPCA: NN	85.33	86.67	86.67	85.56	84.89	84.89	85.56	85.11	84.44	85.33

Table 3: Recognition rates of AQPCA with polynomial kernel using different classifier

Polynomial kernel (d)	1	2	3	4	5	6	7	8	9	10
AQKPCA: LRC	88.67	92.44	93.78	94.44	93.56	93.56	92.89	92.22	91.56	91.56
AQKPCA: SRC	98.44	98.44	98.22	98.00	97.56	98.22	98.00	97.56	97.78	97.78

Table 4: Recognition rates of AQKPCA with Gaussian kernel using different classifier

Gaussian kernel (σ)	le 1	le 2	le 3	le 4	le 5	le 6	le 7	le 8	le 9	le 10
AQKPCA: LRC	96.89	96.89	96.89	96.89	93.33	55.78	55.78	55.78	55.78	32.89
AQKPCA: SRC	94.89	94.89	94.89	95.11	95.56	95.33	32.67	31.78	31.78	31.78

CONCLUSION

In this study, AQKPCA is proposed for MSTAR SAR recognition. The main contributions of this study can be summarized as follows:

- An adaptive quasiconformal kernel and kernel principle component analysis have been combined that map kernel space into a new optimized kernel space
- We have adopted MMC to improve the classification in feature space after quasiconformal mapping
- We have applied proposed method to MSTAR SAR classification, and have compared different classifiers for recognition test

ACKNOWLEDGMENTS

We thank the reviewers for helpful comments. The work was supported in part by Natural Science Foundation of China and Fundamental Research Funds for the Central Universities, No. 61033012, No. 611003177, No.1600-852016, and No.1600-85201.

APPENDIX

Note the quasiconformal mapping Φ': X→Y, that is, $y_i = \Phi'(x_i)$. Let $y = \{y^T_i | i = 1, 2,..., l\}$, $Y_i = \{y^T_{ij} | j = 1, 2,...., l_i\}$, $i = 1, 2,..., C$. Then, we have:

$$m_i^{\Phi'} = \frac{1}{l_i}\sum_{j=1}^{l_i} y_j = \frac{1}{l_i} Y_i^T 1_{l_i}$$

and

$$m_0^{\Phi'} = \frac{1}{l}\sum_{k=1}^{l} y_k = \frac{1}{l} Y^T 1_l$$

As the quasiconformal kernel space preserves the dot product, that is:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_C \end{bmatrix} \begin{bmatrix} Y_1^T, & Y_2^T, \cdots, & Y_C^T \end{bmatrix} = \begin{bmatrix} Y_1Y_1^T & Y_1Y_2^T & \cdots & Y_1Y_C^T \\ Y_2Y_1^T & Y_2Y_2^T & \cdots & Y_2Y_C^T \\ \vdots & \vdots & \ddots & \vdots \\ Y_CY_1^T & Y_CY_2^T & \cdots & Y_CY_C^T \end{bmatrix} = YY^T$$

$$= K = \begin{bmatrix} K_{11} & K_{12} & \dots & K_{1C} \\ K_{21} & K_{22} & \dots & K_{2C} \\ \vdots & \vdots & \ddots & \vdots \\ K_{C1} & K_{C2} & \dots & K_{CC} \end{bmatrix}$$

Therefore, the proof of (14) is shown as follows:

$$\mathrm{tr}(\tilde{S}_b^{\Phi}) = \sum_{i=1}^{C} l_i (m_i^{\Phi} - m_0^{\Phi})^T (m_i^{\Phi} - m_0^{\Phi}) = \sum_{i=1}^{C} l_i (m_i^{\Phi})^T m_i^{\Phi} - (m_0^{\Phi})^T m_0^{\Phi}$$

$$= \sum_{i=1}^{C} l_i \frac{1}{N_i} \mathbf{1}_{l_i}^T Y_i \frac{1}{l_i} Y_i^T \mathbf{1}_{l_i} - \mathbf{1}_l^T Y \frac{1}{l} Y^T \mathbf{1}_l = \sum_{i=1}^{C} \frac{1}{l_i} \mathbf{1}_{l_i}^T Y_i Y_i^T \mathbf{1}_{l_i} - \frac{1}{l} \mathbf{1}_l^T Y Y^T \mathbf{1}_l$$

$$= \sum_{i=1}^{C} \frac{1}{l_i} \mathbf{1}_{l_i}^T \tilde{K}_{ii} \mathbf{1}_{l_i} - \frac{1}{l} \mathbf{1}_l^T \tilde{K} \mathbf{1}_l = \mathbf{1}_l^T \{\mathrm{diag}(\frac{1}{l_1} \tilde{K}_{11}, ..., \frac{1}{l_C} \tilde{K}_{CC})$$

$$- \frac{1}{l} \begin{bmatrix} \tilde{K}_{11} & \cdots & \tilde{K}_{1C} \\ \vdots & \ddots & \vdots \\ \tilde{K}_{C1} & \cdots & \tilde{K}_{CC} \end{bmatrix} \} \mathbf{1}_l = \mathbf{1}_l^T B \mathbf{1}_l$$

Similarly:

$$\mathrm{tr}(\tilde{S}_w^{\Phi})$$

can be reformed as follows:

$$\mathrm{tr}(\tilde{S}_w^F) = \sum_{i=1}^{C} \sum_{j=1}^{l_i} (y_i^j - m_i^F)^T (y_i^j - m_i^F) = \sum_{i=1}^{C} (\sum_{j=1}^{l_i} (y_i^j)^T y_i^j - l_i (m_i^F)^T m_i^F)$$

$$= (\sum_{k=1}^{N} y_k^T y_k - \sum_{i=1}^{C} l_i (m_i^F)^T m_i^F) = \sum_{k=1}^{N} y_k^T y_k - \sum_{i=1}^{C} \frac{1}{l_i} 1_{l_i}^T \tilde{K}_{ii} 1_{l_i}$$

$$= l_l^T \{\mathrm{diag}(\tilde{k}_{11}, ..., \tilde{k}_{ll}) - \mathrm{diag}(\frac{1}{l_1} \tilde{K}_{11}, ..., \frac{1}{l_C} \tilde{K}_{CC})\} 1_l = 1_l^T W 1_l$$

Therefore:

$$\tilde{J}_{MMC} = \mathrm{tr}(\tilde{S}_b^{\Phi}) - \mathrm{tr}(\tilde{S}_w^{\Phi}) = \mathbf{1}_l^T (\tilde{B} - \tilde{W}) \mathbf{1}_l$$

REFERENCES

Amari, S.I. and S. Wu, 1999. Improving support vector machine classifiers by modifying kernel functions. Neural Networks, 12: 783-789.

Daoudi, I. and K. Idrissi, 2010. A kernel-based active learning strategy for content-based image retrieval. Proceedings of the International Workshop on Content-Based Multimedia Indexing, June 23-25, 2010, Grenoble, pp: 1-6.

Heisterkamp, D., J. Peng and H.K. Dai, 2001. An adaptive quasiconformal kernel metric for image retrieval. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, December 8-14, 2001, IEEE Computer Security, USA., pp: 388-393.

Jing, Z., W. Guohong and Y. Zhi, 2008. SAR image recognition based on local feature kernel principal component analysis. J. Astronautics, 29: 995-1001.

Li, H., T. Jiang and K. Zhang, 2006. Efficient and robust feature extraction by maximum margin criterion. IEEE Trans. Neural Networks, 17: 157-165.

Lu, H., K.N. Plataniotis and A.N. Venetsanopoulos, 2008. MPCA-multilinear principal component analysis of tensor objects. IEEE Trans. Neural Networks, 19: 18-39.

Pan, J.S., J.B. Li and Z.M. Lu, 2008. Adaptive quasiconformal kernel discriminant analysis. Neuro Computing, 71: 2754-2760.

Ping, H., W. Renbiao and W. Zhaohua, 2003. SAR automatic target recognition based on KPCA criterion. J. Electron. Inf. Technol., 25: 1297-1301.

Scholkopf, B., A. Smola and K.R. Muller, 1998. Nonlinear component analysis as a Kernel eigenvalue problem. Neural Comput., 10: 1299-1319.

Xiong, H., M.N.S. Swamy and M.O. Ahmad, 2005. Optimizing the kernel in the empirical feature space. IEEE Trans. Neural Networks, 16: 460-474.

Yi, L., Z. Jing and L. Wu, 2009. SAR image recognition based on modified kernel principal component analysis. J. Naval Aeronaut. Eng. Inst., 24: 307-312.

Ying, L.I., H.L. Gong, J.X. Liang and Y.N. Zhang, 2010. SAR image target feature extraction based on KSVD and PCA. Geely Uni. J., 40: 1336-1339.

Zheng, W., C. Zou and L. Zhao, 2005. Weighted maximum margin discriminant analysis with kernels. Neurocomputing, 67: 357-362.