IEEE International Conference on Computer Vision in Remote Sensing, Xiamen, China, 16-18 December 2012

Hyperspectral Image Classificaiton with SVM-based Domain Adaption Classifiers

Zhuo Sun[1], Cheng Wang[1,2*], Peng Li[3], Hanyun Wang[3], Jonathan Li[2,4]

[1]RSSI Lab, Department of Computer Science
Xiamen University
Xiamen, China
*E-mail: cwang@xum.edu.cn
[2]School of Information Science and Engineering
Xiamen University
Xiamen, China

[3]School of Electronics Science and Engineering
National University of Defense Technology
Changsha, China
[4]GeoSTARS Lab, Faculty of Environment
University of Waterloo
Waterloo, Canada
E-mail: junli@uwaterloo.ca

Abstract—**A common assumption in hyperspectral image classification is that the distribution of the classes is stable for all the areas of hyperspectral image. However, this assumption is often incorrect due to the inner-class variety over even short distance on the ground. In this paper, we present a semi-supervised support vector machine (SVM) framework to learn the cross-domain kernels from both the source and target domain in hyperspectral data. The proposed method simultaneously learns the cross-domain kernel mapping and a robust SVM classifier, which is done by minimizing both the Maximum Mean Discrepancy and structural risk functional of SVM. Experiments are carried out on two real data sets and results show that the proposed model can achieve high classification accuracy and provide robust solutions.**

Keywords: Domain adapation, remote sensing, hyperspectral image classification, support vector machines, maximum mean discrepancy

I. Introduction

High spectral resolution images obtained from hyperspectral sensors offer the possibility to increase the classification performance of land-cover classes. A subset of training samples is usually labeled as ground truth to generate the classifiers. Supervised classifiers such as support vector machines (SVMs) [1] provide a robust and effective solution for utilizing the labeled information with the maximum margin and an appropriate loss function.

However, in practice, the collection of labeled training data requires expensive and time-consuming human labor. Moreover, the labeled class signatures are suffering a sample selection bias, which means that the statistical distribution of the training data is not suitably applied to other testing areas. Given though large amounts of the labeled training data in the source domain, the classifiers that trained from them may perform poorly on the test data of the target domain. The shift in the distribution from source labeled domain to target domain in hyperspectral data can be attributed to the differences in illumination conditions, in the phonological state of the vegetative cover and in the shadowing effects caused by satellite view angles or solar elevations, etc. How to exploit the knowledge transfer from the labeled source domain to the unlabeled target domain becomes a new challenge in remote sensing classification.

The domain adaption (DA) technology (also known as transfer learning) is to mine the available knowledge on a given source domain to construct a classifier built on the target domain for which *a priori* information is not available [2]. In remote sensing field, the DA technology can be applied in two situations: 1) disjoint areas that have some similar characteristics or 2) the same geographical area at different times. In this paper, we focus on the investigation of learning the kernels from different area of hyperspectral images. Recently, active learning [3] [4] [5] has been applied to learn the data set shift and perform sample query over the target domain for training. Suju et. al. [6] develop a novel framework to exploit the contextual information in existing classifier and help to classify the data obtained from spatially and temporally different area. Lonrenzo et. al. [7] extends support vector machines technique to the domain-adaption problem and permits the addressing of land-cover map updating where no ground truth is available. They also propose a circular accuracy assessment strategy for the validation of the classifiers. Kanchan et. al. [8] presented a novel DA method to identify whether the new classes have appeared or some of the existing classes disappeared in the new considered image using change-detection technology.

A new criterion Maximum Mean Discrepancy (MMD) [9] is proposed to measure the distribution distance across different domains in a Reproducing Kernel Hilbert Space (RKHS). Pan et.al [10] learn a kernel matrix of samples using the MMD criterion and then apply it for SVMs training. Their method, which is called transfer component analysis (TCA), is applied in feature extraction for remote sensing data by Giona et. al. [11] and reveal improvements for domain adaption classification. In this paper, we introduce the domain-adaption framework (Domain-Transfer SVM)

This work was supported in part by the Natural Science Foundation of China under Grants 40971245.

[12] to solve hyperspectral classification problem. It incorporates MMD criterion and SVM structural risk simultaneously. In order to make use of the SVM software, the kernel function is assumed as a linear combination of multiple base kernels. The methodology combines the kernel learning and SVM parameters learning in one-step, and follows an iterative optimization procedure to get the optimal SVM parameters. Comparison with traditional SVMs and simple Multiple Kernel Learning (MKL) [13] is carried out on two hyperspectral data sets. The indicators include overall accuracy (OA), kappa coefficient, training time and test time.

The rest of the paper is outlined as follows. Section II reviews the definition of MMD and the framework of domain-transfer SVM (DTSVM). Section III presents the multiple kernel learning for DTSVM and the corresponding learning algorithm. Section IV shows the experiment results. Finally, Section V concludes for this paper.

II. Domain-Transfer Support Vector Machine Framework

In the proposed framework, it is expected to minimize the divergence of the distribution between the source and target domains after transformation and the structural risk of SVM classifiers.

Let $D^S = \{X^S, Y^S\} = \{(\mathbf{x}_i^S, y_i^S)\}_{i=1}^{n_s}$ be the n_s labeled source training data $\mathbf{X}^S = \{\mathbf{x}_j^S\}_{j=1}^{n_S}$ and the n_T unlabeled target test data $X^T = \{\mathbf{x}_j^T\}_{j=1}^{n_T}$, with samples $\mathbf{x}_i^S, \mathbf{x}_j^T \in \mathbb{R}^d\ \forall i, j$ and labels $y_i^S \in \Omega = \{\omega_c\}_{c=1}^{C}$. The goal is to predict labels by exploiting the knowledge from the labeled and unlabeled data. We look for the latent representation for both X^S and X^T that preserve the data high order information after transformation. Let the mapping be for both domains: $X^S \rightarrow \psi(X^S) = X^{S*}, X^T \rightarrow \psi(X^T) = X^{T*}$.

A. Maximum Mean Discrepancy (MMD)

In domain adaptation learning, the most significant thing is to reduce the difference between the source and target domains. Many distance measurements have been proposed to evaluate the difference of two distributions: Kullback-Leibler (KL) divergence[14], Jensen-Shannon divergence, Bhattacharyya distance [15]. However, many of these criteria requires the intermediate parametric estimation, which process is quite trivial and difficult to scale to high dimensionality data. Recently, Borgward et al. [9] proposed the Maximum Mean Discrepancy as a new indicator for comparing distributions computed in the Reproducing Kernel Hilbert Space (RKHS), namely,

$$MMD(X^S, X^T) = \left\| \frac{1}{n_S} \sum_{i=1}^{n_S} \psi(\mathbf{x}_i^S) - \frac{1}{n_T} \sum_{i=1}^{n_T} \psi(\mathbf{x}_i^T) \right\| \qquad (1)$$

Therefore, the distance between distributions of two classes can be well-estimated by the means of the two samples mapped into a RKHS. By virtue of the well-known kernel trick, the distance in (1) can be rewritten as:

$$MMD(X^S, X^T) = tr(\mathbf{KL}), \qquad (2)$$

Where

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}^{S,S} & \mathbf{K}^{S,T} \\ \mathbf{K}^{T,S} & \mathbf{K}^{T,T} \end{pmatrix} \in \mathbb{R}^{(n_S + n_T) \times (n_S + n_T)}, \qquad (3)$$

with $\mathbf{K}^{S,S}$, $\mathbf{K}^{T,T}$, $\mathbf{K}^{S,T}$, $\mathbf{K}^{T,S}$ being the kernel matrices (of elements) obtained from the data of the source domain, target domain and cross domains, respectively; and $\mathbf{L} = [L_{ij}] \succeq 0$ with $L_{ij} = 1/n_S^2$ if $x_i, x_j \in \mathbf{X}^S$; $L_{ij} = 1/n_T^2$, if $x_i, x_j \in \mathbf{X}^T$; otherwise, $-1/n_S n_T$.

B. Formulation

DTSVM minimizes distribution distance between the two domains, as well as the structural risk functional of SVM. The optimization problem can be formulated as:

$$[K, f] = \arg\min\, [\ \Phi(Dist_K(X^S, X^T)) + \mu SVM_{K,f}(X^S)\ ] \qquad (4)$$

where $\Phi(.)$ is any monotonic increasing function, and $\mu > 0$ is a weight parameter to balance the difference of data distribution from two domains and the structural risk functional SVM for labeled samples. The kernel function K and the SVM decision function f can be learned in one optimization problem.

The first term in equation (4) is used to measure the mismatch of data distribution between the two domains. In order to reduce the differences between two distributions, the MMD criterion is applied as the distance function. Taking advantage of the kernel trick, the raw feature vector can be non-linearly mapped into a higher-dimensional feature space to shorten the distance of data distribution between the source and target domains.

The second term in equation (4) is to minimize the structural risk functional of SVMs for better classification generalization in the target domain. Let $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_n]'$ be a vector of the dual variables α_i for each labeled sample, $\mathbf{y} = [y_1, ..., y_n]'$ as the label vector, and $K^{S,S} = \psi(\mathbf{x}_i, \mathbf{x}_j)$. SVM is usually solved by its dual problem:

$$\max_{\boldsymbol{\alpha} \in \mathrm{A}} [\boldsymbol{\alpha}'\mathbf{1} - \frac{1}{2}(\boldsymbol{\alpha} \circ \mathbf{y})'\mathbf{K}^{S,S}(\boldsymbol{\alpha} \circ \mathbf{y})], \qquad (5)$$

which is in form of the Quadratic programming problem,. Here, $\mathrm{A} = \{\boldsymbol{\alpha} \in R^n \mid C\mathbf{1} \geq \boldsymbol{\alpha} \geq 0, \boldsymbol{\alpha}'\mathbf{y} = 0\}$ is the feasible set of $\boldsymbol{\alpha}$, with the regularization parameter C for SVM. The element-wise product between two matrices A and B is represented as $\mathbf{A}\circ\mathbf{B}$.

Considering summing up the two terms, we obtained the following saddle-point minimax problem:

$$\min_{\mathbf{K}\succeq 0}\left(\Phi(tr(\mathbf{KL}))+\mu\left(\max_{\boldsymbol{\alpha}\in A}\boldsymbol{\alpha}'\mathbf{1}-\frac{1}{2}(\boldsymbol{\alpha}\circ\mathbf{y})'\mathbf{K}^{S,S}(\boldsymbol{\alpha}\circ\mathbf{y})\right)\right) \quad (6)$$

By utilizing both criteria in equation(4), the unlabeled samples in target domain can be used to promote the performance of SVM classifiers in the target domain. Moreover, an effective kernel function can be learned for a better representation of data in different domains.

III. LERANING DTSVM CLASSIFIERS

A. Multiple Kernel Learning for DTSVM

In order to simplify the kernel learning and facilitate the usage of the existing SVM solver (e.g., LIBSVM), we assume that the kernel function k is a linear combination of a set of base kernel functions, i.e., $k=\sum_{m=1}^{M}d_m k_m$, where $d_m\geq 0,\sum_{m=1}^{M}d_m=1$. We further assume that

$$\Phi(tr(\mathbf{KL}))=\frac{1}{2}(tr(\mathbf{KL}))^2. \quad (7)$$

This quadratic term is strictly convex, and can help to accelerate the convergence speed in kernel learning.

Let us define two kernel matrices $\mathbf{K}=\sum_{m=1}^{M}d_m\mathbf{K}_m$, $\mathbf{K}^{S,S}=\sum_{m=1}^{M}d_m\mathbf{K}_m^{S,S}$, where $\mathbf{K}_m\in\mathbb{R}^{(n_S+n_T)\times(n_S+n_T)}$ and $\mathbf{K}^{S,S}\in\mathbb{R}^{n_S\times n_S}$ are the *m*-th base kernel matrices defined for both domains and for the labeled patterns, respectively. Note that the base kernel matrices $\mathbf{K}_m\succeq 0$ (resp. $K_m^{S,S}\succeq 0$) ,so $\mathbf{K}\succeq 0$ (resp. $\mathbf{K}^{S,S}\succeq 0$) . We then achieve the multi-kernel version DTSVM as:

$$\min_{d\in M}\frac{1}{2}(tr(\sum_{m=1}^{M}d_m\mathbf{K}_m\mathbf{L}))^2+\mu R(\mathbf{d}), \quad (8)$$

where

$$R(\mathbf{d})=\frac{1}{2}\sum_{m=1}^{M}\frac{\|w_m\|^2}{d_m}+C\sum_{i=1}^{n}Loss(y_i(\sum_{m=1}^{M}\mathbf{w}'\psi(\mathbf{x}_i)+b)) \quad (9)$$

with $\mathrm{Loss}(\mathrm{f})=\max(0,1-\mathrm{f})$ is the hinge loss, and ψ_m is the feature mapping function induced form the base kernel $\mathbf{K}_m$. Applying the kernel trick and minimizing the SVM term (9) is equivalent to the Lagrange dual problem, i.e., maximizing

$$J(\mathbf{d})=\boldsymbol{\alpha}'\mathbf{1}-\frac{1}{2}(\boldsymbol{\alpha}\circ\mathbf{y})'\left(\sum_{m=1}^{M}d_m\mathbf{K}_m^{S,S}\right)(\boldsymbol{\alpha}\circ\mathbf{y}) \quad (10)$$

where the kernel weight vector $\mathbf{d}=[d_1,...,d_m]'$. Finally, we simply the optimization problem as:

$$\min_{d\in M}\left(\frac{1}{2}(tr(\sum_{m=1}^{M}d_m\mathbf{K}_m\mathbf{L}))^2+\mu\max_{\alpha\in A}J(\mathbf{d})\right). \quad (11)$$

B. Optimization Algorithm

Solving (11) is a saddle-point minimax problem, and standard iterative update procedures may not converge. Let us define $p_m=tr(\mathbf{K}_m\mathbf{L})$ and $\mathbf{p}=[p_1,...,p_M]'$. Then (11) can be transformed as:

$$\min_{\mathbf{d}\in A}h(\mathbf{d})=\min_{\mathbf{d}\in A}\frac{1}{2}\mathbf{d}'\mathbf{pp}'\mathbf{d}+\mu\max_{\alpha\in A}J(\mathbf{d}). \quad (12)$$

We update the different variables ($\boldsymbol{\alpha}$ and $\mathbf{d}$) iteratively to obtain the globally optimization solution.

With a fixed $\mathbf{d}$, a standard SVM software (such as LIBSVM) can be used to solve the dual problem to maximize $J(\mathbf{d})$. The kernel matrix $\mathbf{K}^{S,S}=\sum_{m=1}^{M}d_m\mathbf{K}_m^{S,S}$ is pre-computed and set for SVM solver to get the optimal $\boldsymbol{\alpha}$.

When we hold the SVM parameter $\boldsymbol{\alpha}$, *(11)* can be updated using second-order Newton's method to get the kernel weights $\mathbf{d}$. Note that $\mathbf{pp}'$ is not full rank, to avoid numerical instability, we replace $\mathbf{pp}'$ as $\mathbf{pp}'+\varepsilon\mathbf{I}$ where ε is set to 0.0001. Then we have

$$\nabla h=(\mathbf{pp}'+\varepsilon\mathbf{I})\mathbf{d}+\mu\nabla J \quad (13)$$

where ∇J is the constant gradient of J in (10) when $\boldsymbol{\alpha}$ is fixed. Since the hessian matrix $\nabla^2 h=\mathbf{pp}'+\varepsilon\mathbf{I}\succ 0$, we use the second-order Newton's method to converge faster. The update direction

$$\mathbf{g}=-(\nabla^2 h)^{-1}\nabla h=-\mathbf{d}-\mu(\mathbf{pp}'+\varepsilon\mathbf{I})^{-1}\nabla J. \quad (14)$$

To maintain $\mathbf{d}\in\mathrm{M}$, the update direction $\mathbf{g}$ should be reduced, so the updated weight is:

$$\mathbf{d}_{t+1}=\mathbf{d}_t+\eta_t\mathbf{g}_t\in M \quad (15)$$

where $\mathbf{d}_t$ and $\mathbf{g}_t$ are the weight vector $\mathbf{d}$ and the reduced update direction $\mathbf{g}$ at the *t*-th iteration respectively, and η_t is the corresponding learning rate.

IV. Experiments and Results

A. Data Description

Two hyperspectral data sets are used in our experiments. The first hyperspectral data set, shown in Figure 1(a), is a part of the scene taken over the Washington DC mall (1280 x 307 pixels) by the HYperspectral Digital Imagery Collection Experiment (HYDICE) sensor in 1995 (noted as WDC for short). The bands are collected in the 400-2500nm region of the visible and infrared spectra. Some atmospheric water absorption bands are discarded in the preprocessing, and 191 spectral channels are remained. Five classes are considered, including roof (C1), street (C2), path (C3), grass (C4), trees (C5).

The second data set, shown in Figure 1(b), is a 1.3m spatial resolution hyperspectral image acquired by the ROSIS-03 optical sensor over the city center of Pavia (noted as PC for short). According to specifications, the number of bands of the ROSIS-3 sensor is 115 with a spectral coverage ranging from 430-860 nm. The image contains 1096x715 pixels. The remaining 102 spectral dimensions are processed. Five classes of interest are considered, *i.e.*, water (C1), trees (C2), asphalt (C3), bitumen (C4) and bared soil (C5).

B. Experimental Setup

We set up two sampling strategy to validate the robustness of the proposed method. The first strategy is that sampling the training and testing data randomly all over the image. The second strategy is that we split the image as source domain part (pixels within green rectangles in Figure 1) and target domain part (pixels outside the green rectangles) and sample from them as training and testing data, respectively. To make the results comparable, the sample amounts for both two strategies are set to equal size. The overall training and testing number is illustrated in Table I. To reliably evaluate the performance of the different methods, all the results are averaged over ten different randomly selected training and validation data sets for every experiment. The training and testing ratios for Washington DC data set is 0.1 and 0.1, while for Pavia Center data set is 0.5 and 0.1.

For each pair of two classes, a binary classifier is trained using DTSVM method. Based on the trained binary classifiers, one-against-one strategy [16] is proposed to make the final prediction for the multiclass problem of hyperspectral data. The results are compared with SVMs and MKL approaches. For SVM, we use Gaussian kernel (*i.e.* $k(x_i,x_j)=\exp(-\lambda\|x_i-x_j\|^2)$) as the default kernel, and the λ is set 0.125 while the regularization parameter C is set 4. For DTSVM and MKL, we add three other types of kernels : Laplacian kernel(*i.e.* $k(x_i,x_j)=\exp(-\sqrt{\lambda}\|x_i-x_j\|)$), inverse square distance kernel (*i.e.* $k(x_i,x_j)=1/(\lambda\|x_i-x_j\|^2+1)$) and inverse distance kernel (*i.e.* $k(x_i,x_j)=1/(\sqrt{\lambda}\|x_i-x_j\|+1)$).

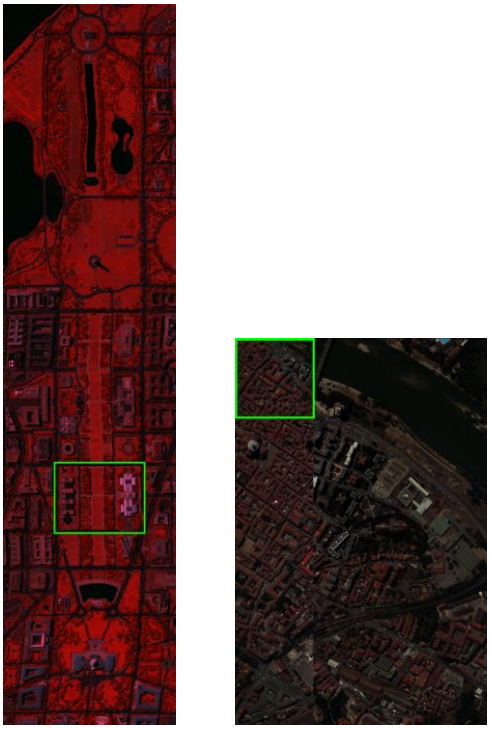

Figure 1. Pseudo-color image for hyperspectral data. (a) Washington DC data set. (b) Pavia Center data set. The two green rectangles stand for the source domain area for training.

TABLE I
NUMBERS OF ALL THE TRAINING AND TESTING SAMPLES OF AND WASHINGTON DC AND PAVIA CENTER DATA SETS

Class Index	Washington DC		Pavia Center	
	Train	**Test**	**Train**	**Test**
C1	1531	19564	693	65278
C2	1634	8191	133	7465
C3	312	793	185	2905
C4	1924	19645	35	6549
C5	444	7612	98	2765
Total	5845	55805	1144	84962

For each kind of kernel, the λ has five possible values $\{2,2^{-1},2^{-2},2^{-3},2^{-4}\}$. So there are 20 base kernels for training. The tradeoff parameter μ in equation (11) is fixed as 1 for DTSVM. All the experiments are done on Core 2 CPU 2.53 GHz with 8-GB RAM.

C. Classification Performance Evaluation

The classification accuracies and kappa statistics of the proposed method and other methods (traditional SVMs and simple MKL) are summarized in Table II with the maximum values in every row in bold. Under the random sampling strategy, we can see that all the three approaches have comparable performance of overall accuracy for the two data sets and MKL is slightly outperform the other two methods. However, when the sampling for training data is constrained within the source domain area, there is obvious drop of the

TABLE II
OVERALL ACCURACY AND KAPPA RESULTS WITH DIFFERENT METHODS AND SAMPLING STRATEGIES ON WDC AND PC DATA SETS

		SVM		MKL		DTSVM	
		OA	Kappa	OA	Kappa	OA	Kappa
Random Sampling	WDC	0.9383	0.9142	**0.9477**	**0.9271**	0.9439	0.9220
	PC	0.9855	0.9635	**0.9862**	**0.9651**	0.9854	0.9631
Domain Sampling	WDC	0.8345	0.7789	0.8637	0.8140	**0.8712**	**0.8243**
	PC	0.9222	0.8116	0.9376	0.8461	**0.9476**	**0.8696**

TABLE III
COMPUTATIONAL TIMES (SECONDS) WITH DIFFERENT METHODS AND SAMPLING STRATEGIES ON WDC AND PC DATA SETS

		SVM		MKL		DTSVM	
		Train	Test	Train	Test	Train	Test
Random Sampling	WDC	4.8	16.3	87	16.4	216	16.1
	PC	2.4	22.9	115	21.6	267	19.4
Domain Sampling	WDC	6.1	28.5	114	17.4	258	13.4
	PC	2.7	25.8	107	23.3	408	19.5

classification accuracies for the conventional SVM and MKL classifiers, which is caused by the data distribution of the source domain failed to apply to the target domain. The classifier that is trained biased to the training area is lack of generalization ability to adapt to the testing domain area. DTSVM performs the best among the three methods in term of OA and kappa coefficient for the domain sampling strategy. For the Washington DC data set, DTSVM has a gain of +3.7%, +0.8% over SVM and MKL methods, respectively. For the Pavia Center data set, DTSVM has a gain of +2.5%, +1.0% over SVM and MKL methods, respectively. This can be explained that the proposed DTSVM can overcome the domain adaption problem by mapping both domain data into the similar distribution in the RHKS.

D. Computation Performance Comparison

As seen in Table III, the training time for DTSVM is more than the other two traditional methods. The training for DTSVM classifiers is more than 10 and 2 times slower than SVM and MKL methods, respectively. By practical profiling analysis, most time is spent on calculating the kernel matrices from labeled and unlabeled data. The learning process to transfer knowledge across domains indeed increases the computation complexity. In term of testing time, the time cost for all the three methods are at the same level. In most applications, testing times are more critical. Therefore, DTSVM's testing computation performance can provide the promising potential for learning in cross-domain problems in practice.

V. CONCLUSION

In this paper, a unified domain adaption framework DTSVM is utilized for the cross-domain classification of hyperspectral data. The proposed method minimizes the structural risk of SVMs and Maximum Mean Discrepancy at one step and provides an efficient algorithm to solve the convex optimization problem. Experimental results show that DTSVM outperforms the traditional learning methods with acceptable computation time in the two real data sets.

ACKNOWLEDGMENT

The authors would like to thank David A. Landgrebe from Purdue University and Paolo Gamba from University of Pavia for providing the hyperspectral data.

REFERENCES

[1] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," Geoscience and Remote Sensing, IEEE Transactions on, vol. 42, pp. 1778-1790, 2004.

[2] H. Daumé III and D. Marcu, "Domain adaptation for statistical classifiers," Journal of Artificial Intelligence Research, vol. 26, pp. 101-126, 2006.

[3] P. Mitra, B. Uma Shankar, and S. K. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," Pattern Recognition Letters, vol. 25, pp. 1067-1074, 2004.

[4] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," Geoscience and Remote Sensing, IEEE Transactions on, vol. 47, pp. 2218-2232, 2009.

[5] D. Tuia, E. Pasolli, and W. J. Emery, "Dataset shift adaptation with active queries," 2011, pp. 121-124.

[6] S. Rajan, J. Ghosh, and M. M. Crawford, "Exploiting class hierarchies for knowledge transfer in hyperspectral data," Geoscience and Remote Sensing, IEEE Transactions on, vol. 44, pp. 3408-3417, 2006.

[7] L. Bruzzone and M. Marconcini, "Toward the automatic updating of land-cover maps by a domain-adaptation SVM classifier and a circular validation strategy," Geoscience and Remote Sensing, IEEE Transactions on, vol. 47, pp. 1108-1122, 2009.

[8] K. Bahirat, F. Bovolo, L. Bruzzone, and S. Chaudhuri, "A Novel Domain Adaptation Bayesian Classifier for Updating Land-Cover Maps With Class Differences in Source and Target Domains," Geoscience and Remote Sensing, IEEE Transactions on, pp. 1-17, 2012.

[9] K. M. Borgwardt, A. Gretton, M. J. Rasch, H. P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," Bioinformatics, vol. 22, pp. e49-e57, 2006.

[10] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," Neural Networks, IEEE Transactions on, pp. 1-12, 2009.

[11] G. Matasci, M. Volpi, D. Tuia, and M. Kanevski, "Transfer component analysis for domain adaptation in image classification," 2011, p. 12.

[12] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer svm for video concept detection," 2009, pp. 1375-1381.

[13] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," Journal of Machine Learning Research, vol. 9, pp. 2491-2521, 2008.

[14] S. Kullback and R. A. Leibler, "On information and sufficiency," The Annals of Mathematical Statistics, vol. 22, pp. 79-86, 1951.

[15] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," Bull. Calcutta Math. Soc, vol. 35, p. 4, 1943.

[16] N. Garcia-Pedrajas and D. Ortiz-Boyer, "Improving multiclass pattern recognition by the combination of two strategies," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 28, pp. 1001-1006, 2006.