An Experimental Comparison of Semi-supervised Learning Algorithms for Multispectral Image Classification

Enmei Tu, Jie Yang, Jiangxiong Fang, Zhenghong Jia, and Nikola Kasabov

Abstract

Semi-Supervised Learning (SSL) method has recently caught much attention in the fields of machine learning and computer vision owing to its superiority in classifying abundant unlabelled samples using a few labeled samples. The goal of this paper is to provide an experimental efficiency comparison between graph based SSL algorithms and traditional supervised learning algorithms (e.g., support vector machines) for multispectral image classification. This research shows that SSL algorithms generally outperform supervised learning algorithms in both classification accuracy and anti-noise ability. In the experiments carried out on two data sets (hyperspectral image and Landsat image), the mean overall accuracies (OAs) of supervised learning algorithms are 15 percent and 86 percent, while the mean OAs of SSL algorithms are 26 percent and 99 percent. To overcome the polynomial complexity of SSL algorithms, we also developed a linear-complexity algorithm by employing multivariate Taylor Series Expansion (TSE) and Woodbury Formula.

Introduction

Multispectral image classification is a fundamental problem in scene change detection (Lu *et al.*, 2004; Radke *et al.*, 2005; Wilkinson *et al.*, 2008), land-use/land-cover investigation (Friedl *et al.*, 2002; Sobrino and Raissouni, 2000; Wu *et al.*, 2009), urban planning (Pauleit and Duhme, 2000; Yeh and Li, 2003), environment management (Zhou *et al.*, 2011), etc. Traditional image classification algorithms can be generally divided into two types:

1. Supervised learning method: First it trains a predefined model with manually labeled samples (samples could be raw image pixels or extracted features. In this paper we mean the former), and then classify a new unlabeled sample using the trained model, for example, Support Vector Machines (SVM) (Burges, 1998; Yang, 2011) and Artificial Neural Networks (ANN) (Patterson, 1998; Zhou and Yang, 2010). The distribution information associated with each class is extracted only from the labeled samples.
2. Unsupervised learning method: It clusters the unlabelled samples directly according to a certain similarity measurement and some presumed grouping rules. For example, spectral clustering (Luxburg, 2007) and *K*-means clustering (Kanungo *et al.*, 2002). The distribution information associated with each class is extracted only from the unlabeled samples.

Semi-Supervised Learning method (SSL) is a new machine learning method that differs from both supervised learning method and unsupervised learning method. It is capable of classifying abundant unlabeled samples with only a small number of labeled samples. By introducing unlabeled samples into training process, it can significantly improve classification results. In the last decade, semi-supervised learning method had drawn much attention in machine learning and computer vision fields (Chapelle *et al.*, 2006; Zhu, 2006; Zhu and Goldberg, 2009) due to its superiority over traditional supervised learning method, including Support Vector Machines (SVM) (Yang, 2011), Artificial Neural Networks (ANN) (Zhou and Yang, 2010), *K* Nearest Neighbors (*K*-NN) (Pal, 2011) etc. For a continuously, updated survey and book references, the reader is referred to Chapelle *et al.* (2006), Zhu (2006), and Zhu and Goldberg (2009).

The study of semi-supervised learning method in the field of remote sensing is particularly important due to the following reasons:

1. In most remote sensing applications, collecting sufficient high-quality labeled samples may cost much time and needs professional experience. On the other hand, huge amount of unlabeled data are relatively easy to collect.
2. In order to achieve low generalization error, traditional supervised learning method needs to be trained with sufficient well-chosen representative samples. However, for many remote sensing tasks, this is not easy because of the presence of transitional classes, restricted access to sites and uncertainties in classes, etc. (Bradley, 2009; Burnicki, 2011; Powell *et al.*, 2004).
3. Graph based semi-supervised learning method has been well researched and widely applied in the fields of machine learning and computer vision. Its merit has been demonstrated by many works. But a comparative and comprehensive study of graph-based, semi-supervised learning algorithms has not yet been found in the field of remote sensing. Therefore, we believe that a research of graph based semi-supervised learning algorithms in the field of remote sensing is not only necessary, but also imperative.

Enmei Tu, Jie Yang, and Jiangxiong Fang are with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University and the Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai, 200240, China (tuen@sjtu.edu.cn).

Zhenghong Jia is with Xinjiang University, School of Information Science and Engineering, Urumqi, 830046, China.

Nikola Kasabov is with the Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, New Zealand.

Photogrammetric Engineering & Remote Sensing
Vol. 79, No. 4, April 2013, pp. 347–357.

0099-1112/13/7904–347/$3.00/0

Although semi-supervised support vector machines (S^{3}VM) has been well studied for remote sensing image classification (Bruzzone *et al.*, 2006; Marconcini *et al.*, 2009), little attention has been paid to graph based semi-supervised learning method, despite the fact that it is very popular in the fields of computer vision and machine learning due to its solid mathematical background and excellent performance (Belkin *et al.*, 2006; Shi and Malik, 2000; Zhou *et al.*, 2004; Zhu *et al.*, 2003).

The goal of this paper is to provide an experimental efficiency comparison between graph based SSL algorithms and traditional supervised learning algorithms (e.g., support vector machines) for multispectral image classification. We design experiments to compare the most popular graph based semi-supervised learning algorithms with the supervised algorithm, in terms of their classification accuracy, time and space complexity, and anti-noise ability. To overcome the polynomial complexity of graph based semi-supervised learning method, we also propose an efficient algorithm which gives comparable classification accuracy, but requires considerably less space and computational time.

The remainder of this paper is organized as follows: The next section briefly reviews the selected graph-based, semi-supervised learning algorithms and their pros and cons, followed by our TSE based efficient semi-supervised learning algorithm. Finally, the simulations and comparisons are presented, followed by discussion and conclusions.

Graph-based, Semi-supervised Learning Method

General Notions of Graph

We first briefly introduce some notations. For readers who are interested in more details, we refer to these references: Bıyıkoglu *et al.* (2007), Chung (1997), and West (2001). Given a sample set $V = \{\mathbf{x}_1, \mathbf{x}_2,..., \mathbf{x}_l, \mathbf{x}_{l+1},..., \mathbf{x}_n | \mathbf{x}_i \in \mathbb{R}^d\}$ coming from c classes, and a class label set $\mathcal{L} = \{1,2,...,c\}$, we assume that the first l samples in V have their class labels $\{y_1, y_2,..., y_l | y_i \in \mathcal{L}\}$, and the class labels of the remaining n-l samples are unknown. Let us define a graph $G=(V, E)$ with vertex set V and edge set $E = \{e(\mathbf{x}_i, \mathbf{x}_j) | 1 \leqslant i, j \leqslant n\}$. Since there is a one-to-one correspondence between the graph vertices and the samples, we will use both terms "vertex $\mathbf{x}_i$" and "sample $\mathbf{x}_i$" indifferently to refer to an instance $\mathbf{x}_i$, according to the context. If there is an edge between vertices $\mathbf{x}_i$ and $\mathbf{x}_j$, then $e(\mathbf{x}_i, \mathbf{x}_j) = w_{ij}$; otherwise $e_{ij} = 0$. The weight w_{ij} is a nonnegative real number which measures the similarity between vertices $\mathbf{x}_i$ and $\mathbf{x}_j$. One popular choice of the weight function is the Gaussian kernel function $w_{ij} = \exp\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$, where parameter σ is the kernel width. Matrix $\mathbf{W} = \{w_{ij} | 1 \leqslant i, j \leqslant n\}$ is called the weight matrix or adjacency matrix. The volume of vertex $\mathbf{x}_i$ is defined as $vol(i) = \sum_j w_{ij}$. Matrix $\mathbf{S} = \mathbf{D}^{-1/2}\,\mathbf{W}\mathbf{D}^{-1/2}$ is the normalized weight matrix, where $\mathbf{D}$ is a diagonal matrix with diagonal elements $vol(1)$, $vol(2)$,...,$vol(n)$. Following the convention, a vector $f \in \mathbb{R}^n$ which assigns each vertex a real value will also be called a function defined on the graph.

A Brief Introduction of Graph-based, Semi-Supervised Learning Algorithms

We select four representative graph-based, semi-supervised learning algorithms for experimental comparison: the Harmonic Function (HF) (Zhu *et al.*, 2003), the Local and Global Consistency (LGC) (Zhou *et al.*, 2004), the Nystrom Approximation Method (NAM) (Camps-Valls *et al.*, 2007), and the Anchor Graph Regularization (AGR) (Liu *et al.*, 2010). We select these algorithms based on criteria of representativeness: HF and LGC are two of the most well-known and popular graph-based, semi-supervised learning algorithms in machine learning community; NAM is the first graph-based, semi-supervised learning algorithm that widely acknowledged in the remote sensing field, and AGR is a recently developed efficient algorithm. Here we review the selected algorithms briefly.

The Harmonic Function (HF)

The Harmonic Function algorithm (Zhu *et al.*, 2003; Zhu and Goldberg, 2009) attempts to find a function which optimizes the following constrained quadratic problem:

$$\begin{aligned} &\min_{f: f(\mathbf{x}) \in \mathbb{R}} \sum_{i,j=1}^{n} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2 \\ &s.t. \quad f(\mathbf{x}_j) = y_j,\ j = 1..l. \end{aligned} \tag{1}$$

Since w_{ij} measures the closeness of two vertices in Euclidean space, the optimal solution f^* is a piecewise constant function and it measures the in-class similarity, i.e., $\mathbf{x}_i$ and $\mathbf{x}_j$ are in the same class if and only if $f^*(\mathbf{x}_i)$ and $f^*(\mathbf{x}_j)$ are equal (Chung, 1997). The minimizer f^* is called harmonic function which satisfies the weighted average property on the unlabeled data:

$$\begin{aligned} &f^*(\mathbf{x}_j) = y_j,\ j = 1..l \\ &f^*(\mathbf{x}_j) = \frac{\sum_{k=1}^{n} w_{jk} f^*(\mathbf{x}_k)}{\sum_{k=1}^{n} w_{jk}},\ j = l+1..n\ . \end{aligned} \tag{2}$$

For a binary class classification problem, the harmonic function has a straightforward interpretation (Doyle and Snell, 2000): if the labeled and unlabeled samples together are treated as circuit nodes in an electrical network, and the edges are treated as wires with resistors equal to $1/w_{ij}$; then, by connecting positive voltage source to the labeled nodes of the first class and negative voltage source to the labeled nodes of the second class, the entries of the harmonic function are the voltage on each of the unlabeled nodes.

Local and Global Consistency (LGC)

Inspired by the work on spreading activation networks (Shrager *et al.*, 1987), the Local and Global Consistency algorithm (Zhou *et al.*, 2004) treats the classification problem as a process of certain activation spreading from labeled samples to unlabeled samples over the graph. Define an $n \times c$ initial labeling matrix $\mathbf{Y}$, and let $y_{ij} = 1$ if sample $\mathbf{x}_i$ belongs class j; $y_{ij} = 0$ otherwise. Let $\mathbf{F} = (f_1, f_2,..., f_c)$ be a $n \times c$ matrix. The LGC iterates $\mathbf{F}_{t+1} = \alpha\mathbf{S}\mathbf{F}_t + (1-\alpha)\mathbf{Y}$ until convergence, where $\alpha \in (0,1)$ is a learning rate parameter. During iterations the label information spreads from labeled samples to unlabeled samples through the graph edges. The authors have also shown that the iteration algorithm converges to a result same as the minimizer of the following regularization framework

$$\mathcal{Q}(\mathbf{F}) = \frac{1}{2}\left(\sum_{i,j=1}^{n} w_{ij} \left\| \frac{1}{\sqrt{vol(i)}}\mathbf{F}_i - \frac{1}{\sqrt{vol(j)}}\mathbf{F}_j \right\|^2 + \mu \sum_{i=1}^{n} \|\mathbf{F}_i - \mathbf{Y}_i\|^2\right) \tag{3}$$

where $\mathbf{F}_i$ and $\mathbf{Y}_i$ are the row vectors of the matrices $\mathbf{F}$ and $\mathbf{Y}$, respectively, and μ is a regularization parameter. The optimal label matrix is:

$$\mathbf{F}^* = (\mathbf{I} - \alpha\mathbf{S})^{-1}\mathbf{Y}. \tag{4}$$

Class labels of the unlabelled samples can be obtained by:

$$y_i = \arg\max_{j=\{1...c\}} \mathbf{F}^*_{ij}, \quad i = l+1..n. \tag{5}$$

Since all entries in $\mathbf{F}^*$ are nonnegative, if we perform a row-wise normalization on $\mathbf{F}^*$, denoted by $\tilde{\mathbf{F}}$, then the entry

$0 \leqslant \tilde{F}_{ij} \leqslant 1$ can also be treated as the likelihood of the sample $\mathbf{x}_i$ coming from the class j.

The Nystrom Approximation Method (NAM)
The Nystrom Approximation Method (Camps-Valls *et al.*, 2007; Fowlkes *et al.*, 2004) fully exploits the positive semi-definite and fast spectral-decaying properties of the graph adjacency matrix to further improve the LGC algorithm, making it scalable for large data set problem. According to Nystrom theory, the normalized weight matrix $\mathbf{S}$ can be well approximated by a subset of its rows (and hence columns) (Williams and Seeger, 2001) $\mathbf{S} \approx \mathbf{S}_{nm}\mathbf{S}^{+}_{mm}\mathbf{S}^{T}_{nm}$, where $\mathbf{S}_{nm}$ contains randomly chosen m rows of $\mathbf{S}$ and $\mathbf{S}_{mm}$ contains the intersection of the chosen rows and columns. By applying the Woodbury formula to Equation 4, the final label matrix is:

$$\mathbf{F} = (1-\alpha)(\mathbf{Y} - \mathbf{S}_{nm}(\mathbf{S}_{mm}\mathbf{S}^{T}_{nm}\mathbf{S}_{nm} - \alpha^{-1}\mathbf{I})^{-1}\mathbf{S}_{mm}\mathbf{S}^{T}_{nm}\mathbf{Y}). \quad (6)$$

The class label of an unlabeled sample can be determined by:

$$y_i = \arg \max_{j \in \{1 \ldots c\}} \mathbf{F}_{ij}. \quad (7)$$

Anchor Graph Regularization (AGR)
The Anchor Graph Regularization algorithm (Liu *et al.*, 2010) has a similar idea to the Local Linear Embedding (LLE) technique (Roweis and Saul, 2000). The core idea is to find an interpolation matrix and then project the learning problem from the original sample set to another smaller one: anchors of the original sample set V. The anchors are some representative points, usually the outputting prototypes of a certain clustering algorithm (such as K-means) on the original sample set V. AGR first optimizes a convex quadratic problems for each sample $\mathbf{x}_i$ to construct a matrix; let us call it interpolation matrix, $\mathbf{Z} \in \mathbb{R}^{n \times m}$, where n is the number of samples in V and m is the number of anchors.

$$\begin{aligned} &\min\ g(\mathbf{z}_i) = \|\mathbf{x}_i - \mathbf{U}\mathbf{z}_i\|^2 \\ &s.t. \quad \mathbf{z}_i^T\mathbf{1} = 1,\ \mathbf{z}_i \geqslant 0 \end{aligned} \quad (8)$$

where $\mathbf{U}$ contains $\mathbf{x}_i$'s k nearest neighbors in the anchor set, and $\mathbf{z}_i$ contains the nonzero entries of the i^{th} row of the interpolation matrix. $\mathbf{1}$ is a vector whose entries are all 1. After obtaining matrix $\mathbf{Z}$, the algorithm optimizes the following quadratic problem to obtain the label matrix $\mathbf{F}$ on anchors:

$$\min \|\mathbf{Z}_l\mathbf{F} - \mathbf{Y}\|_F^2 + \gamma \mathrm{tr}(\mathbf{F}^T\mathbf{Z}^T\mathbf{L}\mathbf{Z}\mathbf{F}) \quad (9)$$

where matrix $\mathbf{Z}_l$ contains those rows in $\mathbf{Z}$ corresponding to the labeled samples. Matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is called the Laplacian of graph, which is a discrete version of Laplace-Beltrami operator and is always a positive semi-definite matrix (Chung, 1997). Function tr($\bullet$) computes matrix trace. γ is a regularization parameter. Class label of an unlabeled sample can be obtained by:

$$y_i = \arg \max_{j \in \{1 \ldots c\}} \frac{\mathbf{Z}_{i\bullet}\mathbf{F}_{\bullet j}}{\lambda_j} \quad (10)$$

where $\mathbf{Z}_{i\bullet}$ is the i^{th} row of $\mathbf{Z}$, and $\mathbf{F}_{\bullet j}$ is the j^{th} column of $\mathbf{F}$. $\lambda_j = \mathbf{1}^T\mathbf{Z}\mathbf{F}_{\bullet j}$ is a normalization factor.

Discussion on Graph-based, Semi-supervised Learning Method

There are three basic assumptions in graph-based, semi-supervised learning method:

1. Smoothness assumption: Samples that are close to each other tend to be in the same class. The closeness is measured by some distance metric, usually Euclidean distance or geodesic distance.
2. Manifold assumption: Samples that are distributed on the same manifold (usually its dimension is much lower than the ambient space) tend to be in the same class,
3. In addition to the labeled samples, sufficient i.i.d. (independently and identically distributed) unlabeled samples are available.

To capture what the assumptions mean, it is helpful to consider a toy data set, a binary classification problem in Figure1a. The upper half-moon and the lower half-moon represent two class distributions. The big solid circle and square represent two labeled samples, one for each class. The small black dots are unlabelled samples. Four dots A, B, C and D in Figure 1a are also selected and illustrated by arrows for explanation purpose. The first assumption says that dot A is more likely to be in the same class with dot B than with dot C, because the distance between A and B is much smaller than the distance between A and C. The second assumption says that dot A is more likely to be in the same class with dot D than with dot C, even though the distance *between A and D* is larger than the distance between A and C, *because A and D* lie on the same manifold (i.e., the lower half-moon). The third assumption says that if no sufficient unlabelled samples are available, as shown in Figure 1b (the number of unlabeled samples reduced to one tenth of Figure 1a), the underlying intrinsic manifold structure cannot be fully represented by the unlabeled samples. Thus, the unlabeled samples provide less helpful information of the manifold distribution to the algorithm, so the binary classification problem becomes more difficult.

The main attractiveness of graph-based, semi-supervised learning method is that given a tiny number of labeled samples (usually very insufficient to train a supervised learning algorithm) and under the above assumptions, it can significantly improve the image classification results. Indeed, compared with supervised learning method that uses only labeled samples, one can hope the semi-supervised learning method to produce a more accurate prediction by taking the unlabeled samples into account. To understand how the unlabeled samples work and improve the results, let us consider again the binary classification example in Figure 1a. We first define a graph by connecting each sample to its k nearest neighbors and weighting the edges by the Gaussian kernel, where k is a positive integer. Imagine that the two labeled samples are two fluid sources (such as liquid or electricity), and also imagine that the fluid spreads on the defined graph using the edges between samples. Then, an unlabeled sample will be affected more by the labeled sample which lies on the same moon with it than by the one which lies on the different moon. Mathematically, in a Bayes learning setting, if the labeled and unlabeled samples $\{\mathbf{x}_i | i = 1..n\}$ come from a probability distribution with density function $p(\mathbf{x})$, and their class labels $\{y_i | i = 1..n\}$ come from a probability distribution with density function $p(y)$; then the partial density $p(\mathbf{x})$ can also provide useful information to estimate the class-conditional density $p(y|\mathbf{x})$, through the joint density $p(\mathbf{x}, y)$. Figure 1c shows the classification result of a graph-based, semi-supervised learning algorithm (LGC) and Figure 1d shows the classification result of a supervised learning algorithm (SVM). The dot curve and line indicate the final classification boundary.

Another advantage of graph-based, semi-supervised learning method is that it does not suffer from the so called "Curse of Dimensionality" effect, and thus it is capable of handling high-dimension data, such as hyperspectral data whose dimensionality is more than 200. We will expand a more detailed discussion on this point in the *Experimental Comparison* section.

The main difference between graph based semi-supervised learning method and traditional supervised learning method is that the former is, generally, a tranductive learning method while the latter is an inductive learning method (Chapelle *et al.*, 2006). This means that graph based semi-supervised

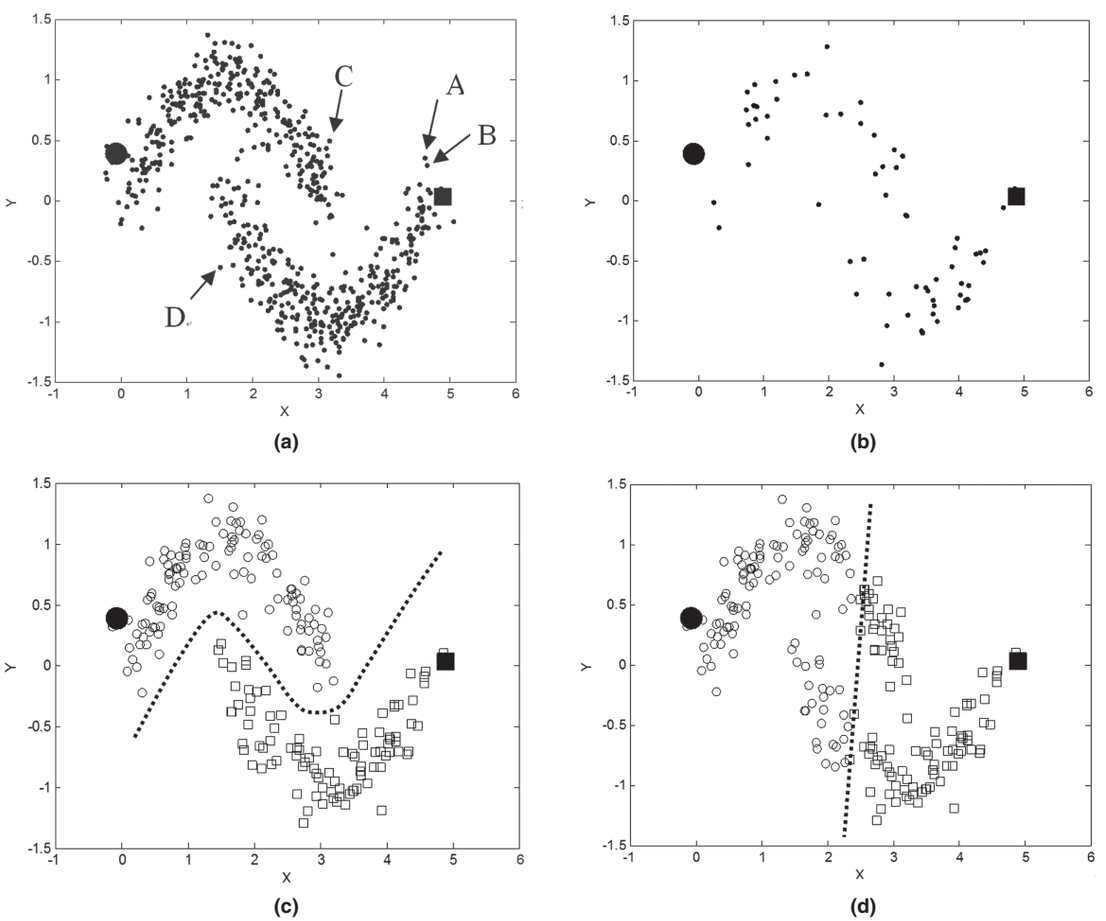

Figure 1. A binary classification problem: (a) A toy data set; the big solid circle and square represent two labeled samples, one for each class; the small dots are unlabelled samples, (b) Example of insufficient unlabelled samples, (c) Classification result of semi-supervised learning method, and (d) Classification result of supervised learning method.

learning method learns a label function only for the data set. Whenever a new unlabelled sample is added to the data set, the learning algorithm has to repeat the whole learning process again to determine the class label of the new comer. This can be very inefficient when samples are collected and classified one by one. On the other hand, supervised learning method can learns a function in the whole data space. Whenever a new sample comes, the learning algorithm can predict directly its class label using the previously learned model without re-learning. But in order to eliminate this difference, some inductive semi-supervised algorithms have also been proposed, such as the Manifold Regularization algorithm (Belkin *et al.*, 2006).

Efficient Graph-based Semi-supervised Learning Algorithm

The application of many graph-based, semi-supervised learning algorithms, including the HF and LGC, is impeded due to their polynomial complexity in both time and space. These algorithms need to construct explicitly a $n \times n$ matrix, where n is the number of samples. Some algorithms (i.e., HF) even have to invert the matrix. For large data set, n can be up to million and thus the matrix is huge and the problem becomes very intractable. Several fast algorithms have been reported in literature recently, including Nystrom Approximation Method (NAM) (Camps-Valls *et al.*, 2007), Anchor Graph Regularization (AGR) (Liu *et al.*, 2010) and the eigenfunction approach (Fergus *et al.*, 2009). From our experience, however, the Nystrom method tends to produce results which vary from time to time, due to the fact that the rows and columns used to approximate the kernel matrix are randomly chosen at each time. Besides, there is a trade-off between the performance and the number of rows and columns used for the approximation. The eigenfunction approach assumes that the samples are retrieved from a distribution which has a product form. This assumption is not always true for multispectral images, because remote sensors absorb not only the energy from electromagnetic waves of its main frequency, but also a fraction of the energy from a nearby frequency (Richards and Jia, 2006). The AGR algorithm can be efficient only when the anchor points and the interpolation matrix are computed, but the computation of these two ingredients is time consuming (finding anchors is itself a clustering problem and computing the interpolation matrix has to optimize a quadratic problem n times).

In this section, we propose an efficient algorithm based on the multivariate Taylor Series Expansion (TSE) theory. We do not construct the adjacency matrix explicitly, but first use Taylor expansion to approximate the Gaussian kernel function and then we convert a large matrix inverting problem to a much smaller one by adopting the well-known Woodbury formula. Simulated performance comparison of the algorithms will be presented in the *Experimental Comparison* Section.

Approximation of Adjacency Matrix Using Multivariate Taylor Expansion

We first construct a graph $G = (V, E)$ on $V = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_l, \mathbf{x}_{l+1}, ..., \mathbf{x}_n | \mathbf{x}_i \in \mathbb{R}^d\}$. Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n) \in \mathbb{R}^{d \times n}$ be a matrix with each column a d-dimension sample $\mathbf{x}_i \in \mathbb{R}^d$, $i = 1..n$. Let $\lambda_i = \exp\left(-\frac{\|\mathbf{x}_i\|^2}{2\sigma^2}\right)$ be a scalar. Then adjacency matrix can be rewritten as:

$$\mathbf{W} = \mathbf{\Lambda E \Lambda} \tag{11}$$

where $\mathbf{\Lambda}$ is a diagonal matrix with diagonal element $\Lambda_{ii} = \lambda_i$, $\mathbf{E}$ is a matrix with $e_{ii} = 0$ on diagonal and $e_{ij} = \exp\left(\frac{\mathbf{x}_i^T \mathbf{x}_j}{\sigma^2}\right)$ on off diagonal. Let us recall that the Taylor series expansion formula for a multivariate function $f(\mathbf{x})$ is:

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T \mathbf{H} (\mathbf{x} - \mathbf{x}_0) + O(\|\mathbf{x} - \mathbf{x}_0\|^2) \tag{12}$$

where $\nabla f(\mathbf{x}_0) = \left.\frac{\partial f}{\partial X}\right|_{\mathbf{x}=\mathbf{x}_0}$ is the gradient vector, and $\mathbf{H} = \left.\frac{\partial^2 f}{\partial \mathbf{x} \partial \mathbf{x}^T}\right|_{\mathbf{x}=\mathbf{x}_0}$ is the Hessian matrix. Let $f(\mathbf{x}, \mathbf{z}) = \exp\left(\frac{\mathbf{x}^T \mathbf{z}}{\sigma^2}\right)$. Expanding $f(\mathbf{x}, \mathbf{z})$ at origin results in:

$$f(\mathbf{x}, \mathbf{z}) = 1 + \frac{\mathbf{x}^T \mathbf{z}}{\sigma^2} + O(\|\mathbf{x}\|^2, \|\mathbf{z}\|^2). \tag{13}$$

Plugging Equation 13 into Equation 11 and ignoring the high orders, the adjacency matrix can be approximated by:

$$\tilde{\mathbf{W}} = \mathbf{\Lambda} \left(\mathbf{e}\mathbf{e}^T - \mathbf{I} + \frac{1}{\sigma^2} (\mathbf{X}^T \mathbf{X} - \tilde{\mathbf{\Lambda}}) \right) \mathbf{\Lambda}^T \tag{14}$$

where matrix $\tilde{\mathbf{\Lambda}}$ is a diagonal matrix with diagonal element $\tilde{\mathbf{\Lambda}}_{ii} = \|\mathbf{x}_i\|^2$, and vector $\mathbf{e} = (1,1,...,1)^T$. Furthermore, the volume of vertex $\mathbf{x}_i$ is:

$$vol(i) = \lambda_i C - \lambda_i^2 + \frac{\lambda_i}{\sigma^2} (\mathbf{x}_i^T \mathbf{y} - \lambda_i \tilde{\Lambda}_{ii}) \tag{15}$$

where $C = \sum_{j=1}^{n} \lambda_j$, $\mathbf{y} = \sum_{j=1}^{n} \lambda_j \mathbf{x}_j$. Thus, the normalized adjacency matrix can be approximated by:

$$\tilde{\mathbf{S}} = \boldsymbol{\alpha}\boldsymbol{\alpha}^T + \frac{1}{\sigma^2} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}} - \mathbf{T} \tag{16}$$

where $\mathbf{T} = \mathbf{D}^{-½} \mathbf{\Lambda \Lambda D}^{-½} + (\mathbf{D}^{-½} \mathbf{\Lambda} \tilde{\mathbf{\Lambda}} \mathbf{\Lambda D}^{-½})/\sigma^2$, $\boldsymbol{\alpha} = \mathbf{D}^{-½} \mathbf{\Lambda e}$ and $\tilde{\mathbf{X}} = \mathbf{X \Lambda D}^{-½}$. It is worth noting that there are only n elements for vector $\boldsymbol{\alpha}$ or diagonal matrix $\mathbf{T}$ that need to be stored, and $\tilde{\mathbf{X}}$ is same size as original data matrix. Thus, the total space required for storing the normalized adjacency matrix is reduced significantly.

Using Woodbury Formula for Large Matrix Inversion

We define a $n \times (d+1)$ matrix $\mathbf{M} = \left[\boldsymbol{\alpha} \frac{1}{\sigma} \tilde{\mathbf{X}}^T\right]$, then the normalized adjacency matrix is

$$\tilde{\mathbf{S}} = \mathbf{M}\mathbf{M}^T - \mathbf{T}. \tag{17}$$

Using the Woodbury matrix inversion formula (Horn and Johnson, 1990), $(\mathbf{AB} + \mathbf{C})^{-1} = \mathbf{C}^{-1} - \mathbf{C}^{-1} \mathbf{A}(\mathbf{I} + \mathbf{BC}^{-1} \mathbf{A})^{-1} \mathbf{BC}^{-1}$, we have:

$$(\mathbf{I} - \gamma \tilde{\mathbf{S}})^{-1} = \mathbf{K}^{-1} + \gamma \mathbf{K}^{-1} \mathbf{M} (\mathbf{I}_{d+1} - \gamma \mathbf{M}^T \mathbf{K}^{-1} \mathbf{M})^{-1} \mathbf{M}^T \mathbf{K}^{-1} \tag{18}$$

where $\mathbf{K} = \mathbf{I} + \gamma \mathbf{T}$ is a diagonal matrix, and $\mathbf{I}_k$ is the a $k \times k$ identity matrix. Thus, the final optimal solution in Equation 4 is:

$$\mathbf{F}^* = \mathbf{K}^{-1} \mathbf{Y} + \mathbf{G} (\gamma \mathbf{I} - \gamma^2 \mathbf{M}^T \mathbf{G})^{-1} (\mathbf{G}^T \mathbf{Y}) \tag{19}$$

where $\mathbf{G} = \mathbf{K}^{-1} \mathbf{M}$.

Complexity Analysis

Inverting the diagonal matrix $\mathbf{K}$ consists only in computing the reciprocal of its diagonal elements, and this can be done efficiently. The dimension of both matrices $\mathbf{M}$ and $\mathbf{G}$ is $n \times (d+1)$, so $\gamma \mathbf{I} - \gamma^2 \mathbf{M}^T \mathbf{G}$ is a $(d+1) \times (d+1)$ matrix. Matrix $\mathbf{G}^T \mathbf{Y}$ is a $d \times c$ matrix, where d is the sample dimensionality, and c is the class number. For multispectral image classification, d and c are much smaller than n. Therefore, inverting and multiplying of these matrices are very computationally efficient. As it will be seen in the next section, the speed of our TES algorithm is faster than SVM. Besides, the memory-saving characteristic of our TES algorithm is also apparent, since the maximum space required for all the computations is approximately two $\mathbf{X}$'s space and several vectors' space.

Experimental Comparison

The experiments were carried out on two multispectral image data sets: AVIRIS image data set and Landsat-7 Enhanced Thematic Mapper (ETM+) imagery data set.

Methodology

Data Sets Description

The AVIRIS Image Data Set

The first data set is a 220-band hyperspectral image: AVIRIS image Indian Pine Test Site. It covers a 2 mile × 2 mile portion of Northwest Tippecanoe County, Indiana. It contains 16 land-cover classes. Four main land-cover classes are soybeans, corn, woods, and grass. The classes of soybeans, corn and grass are further divided into several finer subclasses. These subclasses belonging to one major class are very similar to each other (as will be shown in the *Experimental Results* Section), and thus make it a challenging work for a classification algorithm to classify them correctly. Furthermore, the crops (mainly corn and soybeans) in the area are very young in their growth circle and thus have a very small canopy cover. The spatial resolution of the spectral image is only 20 m. These factors make it very difficult to discriminate among these crops in the image. Thus, it is a challenging data set and suitable to validate the efficiency of an image classification algorithm. The data set (with its ground truth reference file and calibration information) is available on website: *https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html*.

The Landsat Data Set

The second experimental data set is made up of a Landsat-7 Enhanced Thematic Mapper (ETM+) imagery acquired in 01 September 1999 with satellite orbit of path 143 and row 029. Located in northwest China, known as the ancient Silk Road, this region is the gateway from East Asia to West Asia and Europe. On the north of this region is the second largest desert in China, Gurbantunggut Desert, and on the south is the Changji autonomous prefecture. The selected area is on the transition from plain to dessert and contains land-cover types varying greatly, and thus the data set is suitable

for testing the generalization ability of semi-supervised classifier. In this experiment, a sub-region of 660 × 660 pixels has been considered, and it is shown in Figure 2.

Parameters Settings

For LGC and HF, the width of the Gaussian kernel function σ is determined by searching in parameter space and finding the optimal one which produces the lowest error rate for a specific data set. The regularization parameter α is heuristically set to 0.95 in both LGC and NAM. For the Nystrom Approximation Method (NAM), according to the experiments carried out in Camps-Valls *et al.* (2007), the number of samples used to approximate kernel matrix is set to be 80, and the number of eigenvalues is set to $p = 10$ to make a balance between accuracy and computation cost. For the Anchor Graph Regularization (AGR), the number of anchors is set to 500 to make a better trade-off between clustering time and performance. The number of nearest neighbors for each anchor point is set to be 5, as suggested by the author. In our TSE algorithm, γ is set to be 0.7, and σ is determined manually to produce the best results for each data set. For the SVM algorithm, we use the well-known library LIBSVM (freely available on *http://www.csie.ntu.edu.tw/~cjlin/*). We use radial basis function (RBF) kernel SVM as it produces better results (Chapelle *et al.*, 2002; Hsu and Lin, 2002). The scale of RBF is set to 1, and the width is chosen manually to produce the best performance.

Collection of Training Set and Validation Set

The AVIRIS Image Data Set

There are 16 land-cover classes in the ground truth map. We choose 14 of them and discard the other two because they do not have enough samples to allow training the classifiers and performing validation. The remaining 14 land-cover classes and their sample numbers are: Corn-notill (1,434), Corn-min (834), Corn (234), Grass/Pasture (497), Grass/Trees (747), Alfalfa (54), Hay-windrowed (489), Soybeans-notill (968), Soybeans-min (2468), Soybean-clean (614), Wheat (212), Woods (1,294), Bldg-Grass-Tree-Drives (380), and Stone-steel towers (95). We generate training samples by randomly selecting k (k = 1, 3…50) samples at each time from each class and mix these 14k samples together to form the training set. All the remaining samples are put together to form the validation set.

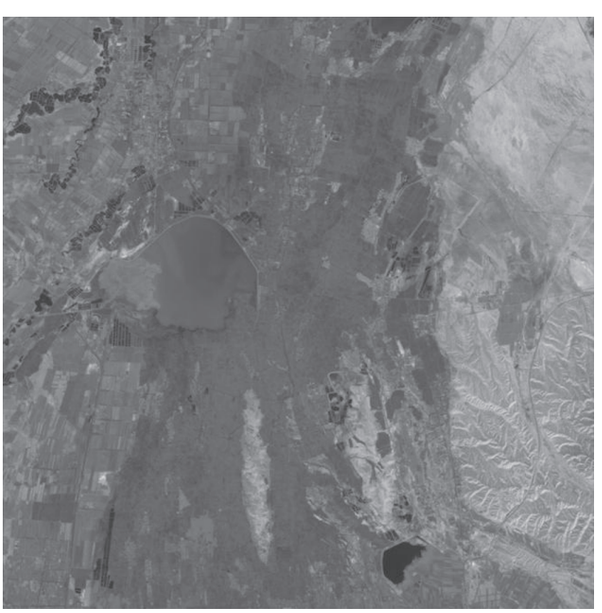

Figure 2. ETM+ Dataset: The Study Area.

The Landsat Data Set

In this data set, four typical land-cover classes (i.e., water, vegetation/plant, sand, and residential area) are defined and their corresponding labeled samples are manually determined from ground reference data. The capacities of these labeled sample sets are 1,734, 1,729, 1,087, and 1,011, respectively. Band 6 of the Landsat image is discarded because it has a lower spatial resolution than other bands and contributes less to the performance. In all experiments, the training samples for each land-cover type are selected randomly from their corresponding labeled sample sets. The rest of the labeled samples are mixed together to form the validation set. In order to compare the classification effectiveness of the graph-based, semi-supervised learning algorithms, the number of training samples of each class varies from 1 to 30. So, the final training sets size for four land-cover types is 4k, and the validation sets size is 5,561 - 4k, where k is the amount of training samples of each class, k = 1,2,...,30. Classification accuracy is evaluated on the validation sets in the assumption that all their labels are unavailable. For the semi-supervised learning algorithms, all samples are mixed together to define the graph.

Experimental Results and Discussion

In order to tune parameters under the same condition for all the algorithms, the AVIRIS data matrix is first normalized by its norm and then centralized by subtracting its mean. We perform the experiments using Matlab 2011a on a computer with 2.13GHZ CPU. The overall accuracy is evaluated on validation set by counting the total number of correctly classified samples. For each capacity of the training set, we run the algorithms five times independently and compute the final overall accuracy and time cost by averaging.

The AVIRIS Image Data Set

Figure 3 is the spectral curve of the subclasses of corn, grass, and soybeans, and Figure 4 is the mean overall accuracy. In Figure 3 we can see the spectral curves of the subclasses are very close to each other, (especially the subclasses of soybeans and grass). It is difficult to discriminate between them, and thus the classification task is very challenging. As shown in Figure 4, with a very low capacity of training set, these algorithms can hardly attain high accuracy. In addition, the high-dimensional nature of the hyperspectral image and the insufficiency of training samples can lead to the "Curse of Dimensionality" effect (which means that as the dimensionality of the samples increases, the complexity of the structure of the data increases exponentially, and as a result, the need of the training samples have to grow exponentially in order to obtain a reliable classifier) (Vapnik, 2000) for traditional statistical learning method, e.g., SVM algorithm. In contrast, graph-based, semi-supervised learning algorithms do not try to model the complex structure of the data set directly. Instead, they are only concerned with the underlying graph of the data set and learn a discriminant function from the graph topological structure indirectly by absorbing the information carried on both labeled and unlabeled data, as we have previously discussed. This property of graph-based, semi-supervised learning algorithms has a close relationship to the manifold learning method (Belkin and Niyogi, 2004; Chung, 1997) which has been demonstrated to be a powerful tool to handle a high-dimension data set (Belkin and Niyogi, 2003; Lin and Zha, 2008; Tenenbaum *et al.*, 2000; Zhang *et al.*, 2005).

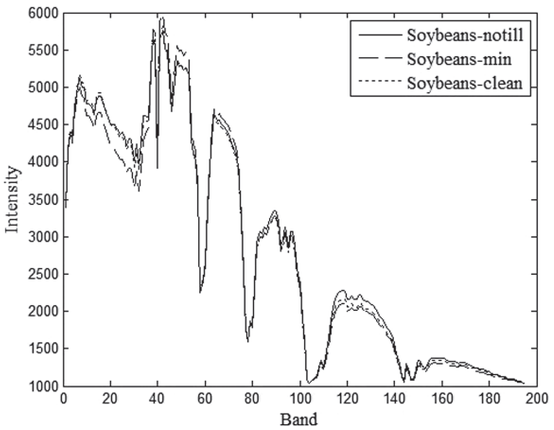

Figure 3. Spectral plot of subclasses of soybeans, corn and grass: (a) Spectral curve of subclasses of soybeans, (b) Spectral curve of subclasses of corn, and (c) Spectral curve of subclasses of grass.

Therefore, the graph-based, semi-supervised learning algorithms (i.e., the AGR, LGC, and HF in Figure 3) do not suffer from the curse of dimensionality effect, and thus achieve a considerable improvement in classification accuracy,

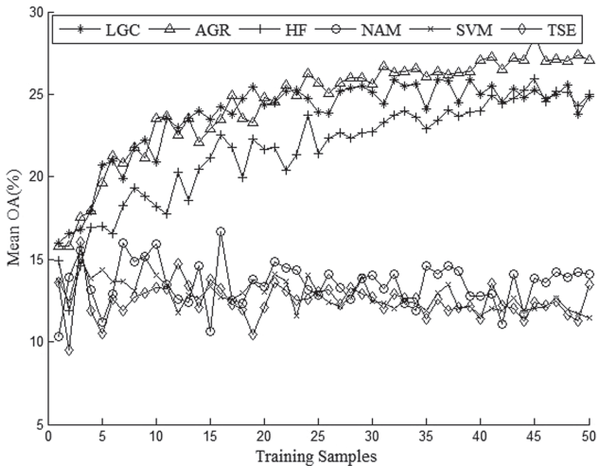

Figure 4. AVIRIS Data Set: Mean Overall Accuracy Comparison.

as shown in the upper part of the plot in Figure 4. The high similarity among the spectral curves and high-dimension characteristic of the hyperspectral image and insufficiency of training samples also cause the NAM and TSE algorithm failed to approximate the weight matrix with sufficient accuracy, and thus these algorithms produce a low overall accuracy.

Figure 5 shows the time cost of the algorithms. As expected, the time cost of SVM increases constantly as the number of training samples increases. Because the training samples are insufficient, all the labeled samples become support vector after training. As a result, an unlabeled sample needs more comparisons to determine its class label as the number of training samples increases. In contrast, the time cost of the semi-supervised learning algorithms remains nearly constant. This is due to the tranductivity property of the semi-supervised learning algorithms. Because tranductive learning algorithms use all the labeled and unlabeled samples to learn and when the total number of samples remains unchanged, the time cost will be constant.

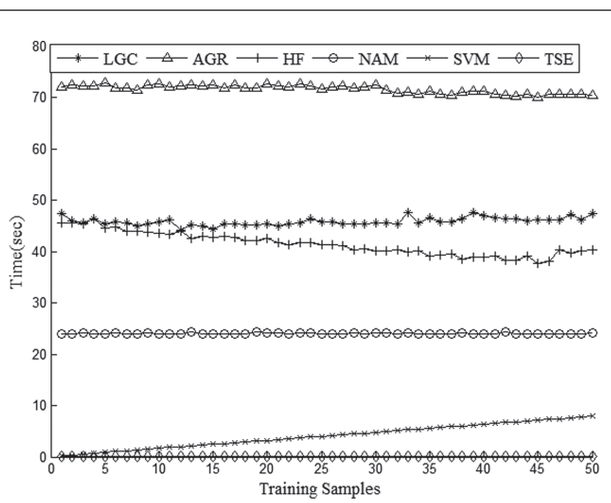

Figure 5. AVIRIS Data Set: Average Time Cost Comparison.

The Landsat Data Set

In this data set, we discard the sixth thermal infrared band because its spatial resolution is inconsistent with the other bands, and experiments also show that discarding the sixth band has no visible effect to the classification result in our classification task. Experimental results are shown in Figures 6 through 9. Figure 6 shows the spectral curves of water and plant. In this figure, we can see the spectrum of each class has a relative wide dynamic range. While SVM is trained by a very small number of labeled samples, it can hardly be able to model the spectral variation and results in a low and unstable performance. Figure 7 shows the mean overall accuracy versus training set size, and Figure 8 is plot of the Standard Deviation (STD) of overall accuracy versus training set size. From these figures we can infer that SVM can hardly produce a satisfactory result with a training set whose size is smaller than 60. In contrast, the semi-supervised learning algorithms are able to incorporate class distribution information contained in the large amount of unlabeled samples, and thus can produce content results even when only one or two training samples are used. At the same time, it is important to note that semi-supervised learning algorithms have a more stable performance while SVM tends to oscillate widely for small training set size. Because graph-based, semi-supervised learning algorithms learn a function on graph, and thus the connections between graph vertices (i.e., the vertex neighborhood relationships on the graph) play a fundamental role in the learning process. These connections are relatively robust. During learning process the class label information is propagated through these connections, and thus the performance is more stable. It is also worth noting that in most cases the proposed TSE algorithm achieves the best OA and at the same time has the lowest deviation.

Figure 9 shows a logarithmic plot of the time cost versus training set size. Again the time cost of SVM grows linearly with training set size, and the time cost of the other three semi-supervised learning algorithms almost remain constant. For semi-supervised learning algorithms, although the

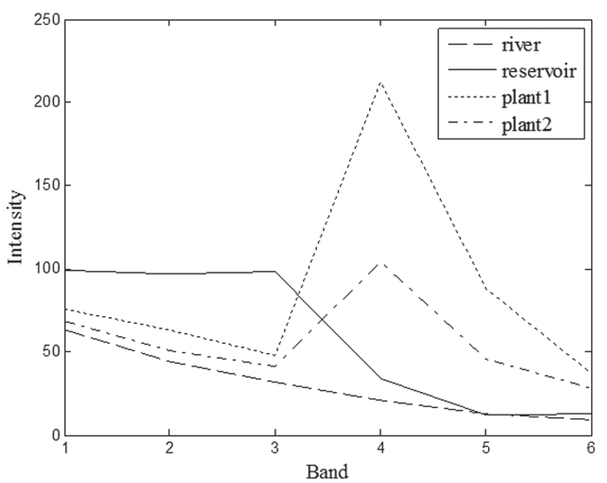

Figure 6. Spectral curves of water and plants.

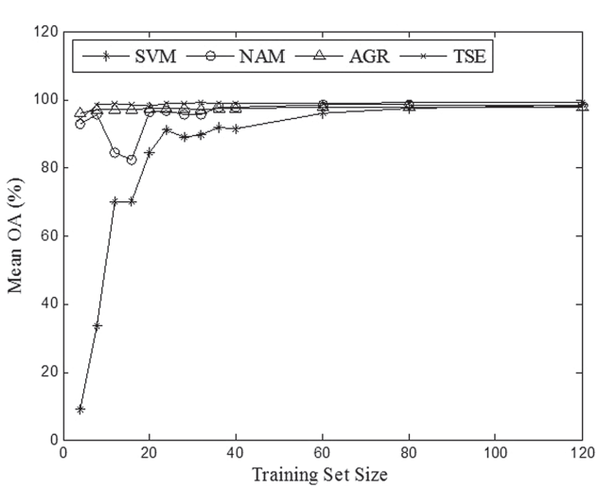

Figure 7. ETM+ Data Set: Mean Overall Accuracy Comparison.

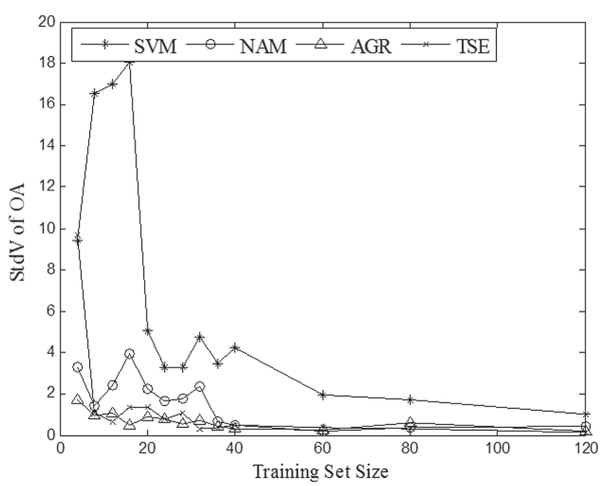

Figure 8. ETM+ Data Set: Standard Deviation (STD) of Overall Accuracy Comparison.

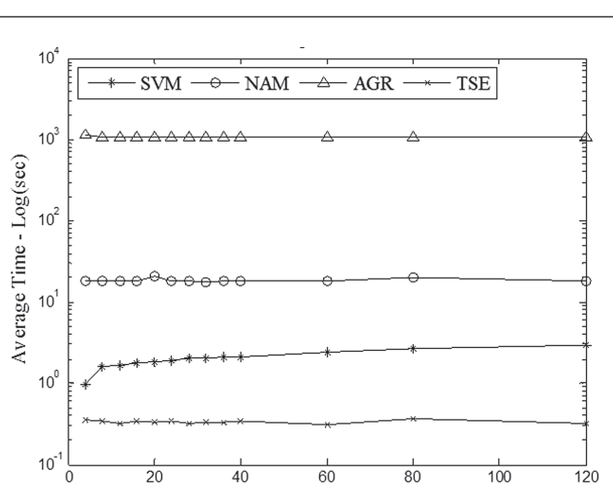

Figure 9. ETM+ Data Set: Average Time Cost Comparison.

training set size is increased, the total samples are unchanged, and thus the graph size remains unchanged. Therefore, the time cost is nearly constant. As shown in Figure 9, among all the three semi-supervised learning algorithms, the proposed TSE algorithm again has the lowest computation cost.

Noise Response of Classifiers

To evaluate the effect of noise, we consider the data set which is corrupted by different levels of noise. As shown in the left column of Figure 10, we add three levels of Gaussian noise with a standard deviation 5, 10, and 15 to the data set

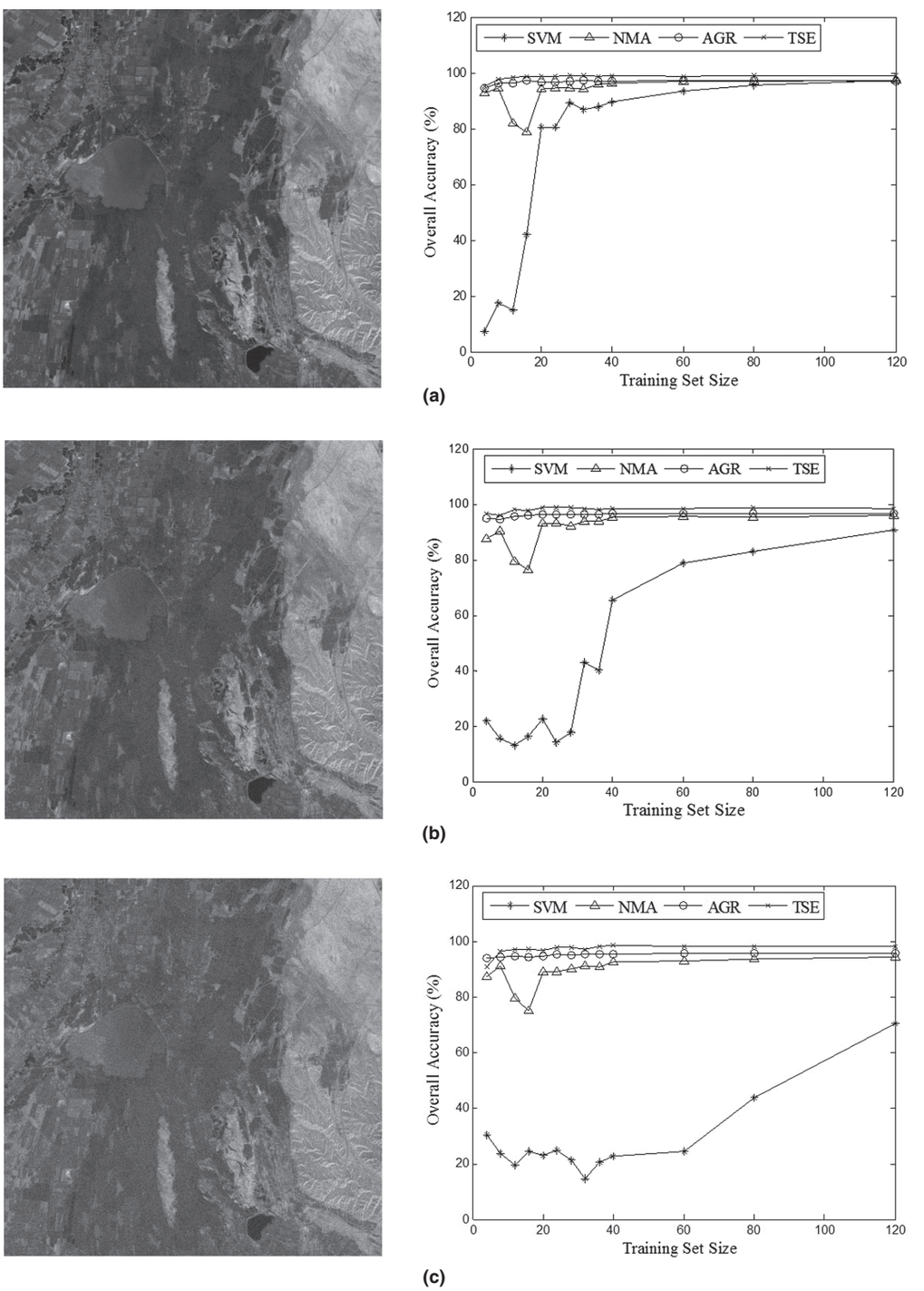

Figure 10. Gaussian noise with standard deviation 5, 10, and 15 corrupted remote sensing data (left column) and overall accuracy (right column) comparison of different noise levels: (a) Gaussian noise with standard deviation 5, (b) Gaussian noise with standard deviation 10, and (c) Gaussian noise with standard deviation 15.

by generating a noise matrix with the same size as the data matrix and then summing the noise matrix and data matrix together. The plots in the right column demonstrate the impact of different levels of noise on the four classifiers. As demonstrated in these figures, the proposed TSE algorithm and other semi-supervised learning algorithms are more robust to noise than supervised learning algorithm. This could be explained again by these two facts: (a) graph-based, semi-supervised learning algorithms use the information from both the labeled samples and many unlabeled samples, and (b) graph-based, semi-supervised learning algorithms learn a discriminant function on graph and propagate class label information through the robust vertices connections, i.e., the vertices' neighborhood relationships described in last section. These facts are less sensitive to noise and therefore guarantee the excellent performance even moderate noise added on the data set.

Conclusions

In this paper, we have evaluated several standard graph-based, semi-supervised learning algorithms as well as one supervised learning algorithm (i.e., support vector machines) in the context of the remote sensing image classification. Based on the experimental comparisons, we could draw several conclusions:

1. Given only a small number of training samples, supervised learning method (e.g., SVM) tend to suffer severely from the curse of dimensionality effect for high-dimension data set (e.g., hyperspectral image) while graph-based, semi-supervised learning method is generally not affected.
2. By incorporating the unlabelled samples into learning process, graph-based, semi-supervised learning method generally performs no worse (often much better) than supervised learning method in remote sensing image classification.
3. By exploiting the robust vertices, neighborhood relationships on graph, the stability and anti-noise ability of graph-based, semi-supervised learning method are much stronger than supervised learning method.
4. Graph-based, semi-supervised learning method is a tranductive learning method and learns a discriminant function only for the data set, while supervised learning method is a inductive learning method and can learn a discriminant function in the data space.

These properties of semi-supervised learning method show great potential in the remote sensing field. In order to overcome the polynomial complexity of graph-based, semi-supervised learning method, we have also proposed an efficient algorithm which has linear complexity in both time and space. Particularly, we first use the multivariate Taylor Series Expansion (TSE) to approximate the Gaussian kernel function, and then, by adopting the Woodbury formula, we reduce a large matrix inversion problem to a much smaller matrix inversion problem. In the future work, we will focus on two issues: an extension of our TSE algorithm to other kinds of graph construction method and a study of the theoretical error bound caused by Taylor approximation.

Acknowledgments

This work is supported by the International Cooperation Project of Ministry of Science and Technology of China with ID: 2009DFA12870 and National Science Foundation No. 61105001. This research is partly supported by National Science Foundation, China (No: 61273258) and Committee of Science and Technology, Shanghai (No: 11530700200).

References

Belkin, M., and P. Niyogi, 2003. Laplacian eigenmaps for dimensionality reduction and data representation, *Neural Computation*, 15(6):1373–1396.

Belkin, M., and P. Niyogi, 2004. Semi-supervised learning on Riemannian manifolds, *Machine Learning*, 56(1):209–239.

Belkin, M., P. Niyogi, and V. Sindhwani, 2006. Manifold regularization: A geometric framework for learning from labelled and unlabelled examples, *The Journal of Machine Learning Research*, 7(2006):2399–2434.

Bıyıkoglu, T., J. Leydold, and P. F. Stadler, 2007. *Laplacian Eigenvectors of Graphs, Volume 1915 of Lecture Notes in Mathematics*, Springer.

Bradley, B.A., 2009. Accuracy assessment of mixed land cover using a GIS-designed sampling scheme, *International Journal of Remote Sensing*, 30(13):3515–3529.

Bruzzone, L., M. Chi, and M. Marconcini, 2006. A novel transductive SVM for semi-supervised classification of remote-sensing images, *IEEE Transactions on Geoscience and Remote Sensing*, 44(11):3363–3373.

Burges, C.J.C., 1998. A tutorial on support vector machines for pattern recognition, *Data Mining and Knowledge Discovery*, 2(2): 121–167.

Burnicki, A.C., 2011. Modelling the probability of misclassification in a map of land cover change, *Photogrammetric Engineering & Remote Sensing*, 77(1):39–49.

Camps-Valls, G., T. Bandos Marsheva, and D. Zhou, 2007. Semi-supervised graph-based hyperspectral image classification, *IEEE Transactions on Geoscience and Remote Sensing*, 45(10): 3044–3054.

Camps-Valls, G., B. Marsheva, and D. Zhou, 2007. Semi-supervised graph-based hyperspectral image classification, *IEEE Transactions on Geoscience and Remote Sensing*, 45(10):3044–3054.

Chapelle, O., B. Schölkopf, and A. Zien, 2006. *Semi-supervised Learning*, MIT Press, Cambridge, Massachusetts.

Chapelle, O., V. Vapnik, O. Bousquet, and S. Mukherjee, 2002. Choosing multiple parameters for support vector machines, *Machine Learning*, 46(1):131–159.

Chung, F.R.K.,1997. *Spectral Graph Theory*, American Mathematical Society.

Doyle, P.G., and J.L. Snell, 2000. *Random Walks and Electric Networks*, Carus Mathematical Monographs.

Fergus, R., Y. Weiss, and A. Torralba, 2009. Semi-supervised learning in gigantic image collections, *Proceedings of the Twenty-Third Annual Conference on Neural Information Processing Systems.*

Fowlkes, C., S. Belongie, F. Chung, and J. Malik, 2004. Spectral grouping using the Nyström method, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225.

Friedl, M., D. McIver, J. Hodges, X. Zhang, D. Muchoney, A. Strahler, C. Woodcock, S. Gopal, A. Schneider, and A. Cooper, 2002. Global land cover mapping from MODIS: Algorithms and early results, *Remote Sensing of Environment*, 83(1-2):287–302.

Horn, R.A., and C.R. Johnson, 1990. *Matrix Analysis*, Cambridge University Press, Cambridge, UK.

Hsu, C.W., and C.J. Lin, 2002. A comparison of methods for multiclass support vector machines, *IEEE Transactions on Neural Networks*, 13(2):415–425.

Kanungo, T., D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, and A.Y. Wu, 2002. An efficient k-means clustering algorithm: Analysis and implementation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):881–892.

Lin, T., and H. Zha, 2008. Riemannian manifold learning, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 30(5):796–809.

Liu, W., J. He, and S.F. Chang, 2010. Large graph construction for scalable semi-supervised learning, *Proceedings of the International Conference on Machine Learning.*

Lu, D., P. Mausel, E. Brondizio, and E. Moran, 2004. Change detection techniques, *International Journal of Remote Sensing*, 25(12):2365–2401.

Luxburg, U., 2007. A tutorial on spectral clustering, *Statistics and Computing*, 17(4):395–416.

Marconcini, M., G. Camps-Valls, and L. Bruzzone, 2009. A composite semi-supervised SVM for classification of hyperspectral images, *IEEE Transactions on Geoscience and Remote Sensing Letters*, 6(2):234–238.

Pal, M., 2011. Modified nearest neighbour classifier for hyperspectral data classification, *International Journal of Remote Sensing*, 32(24):9207–9217.

Patterson, D.W., 1998. *Artificial Neural Networks: Theory and Applications*, Prentice Hall New Jersey.

Pauleit, S., and F. Duhme, 2000. Assessing the environmental performance of land cover types for urban planning, *Landscape and Urban Planning*, 52(1):1–20.

Powell, R., N. Matzke, and C. De Souza, 2004. Sources of error in accuracy assessment of thematic land-cover maps in the Brazilian Amazon, *Remote Sensing of Environment*, 90(2):221–234.

Radke, R.J., S. Andra, O. Al-Kofahi, and B. Roysam, 2005. Image change detection algorithms: A systematic survey, *IEEE Transactions on Image Processing*, 14(3):294–307.

Richards, J.A., and X. Jia, 2006. *Remote Sensing Digital Image Analysis: An Introduction*, Springer-Verlag, Berlin, Germany.

Roweis, S.T., and L.K. Saul, 2000. Nonlinear dimensionality reduction by locally linear embedding, *Science*, 290(5500):2323–2326.

Shi, J., and J. Malik, 2000. Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Shrager, J., T. Hogg, and B.A. Huberman, 1987. Observation of phase transitions in spreading activation networks, *Science*, 236(4805):1092–1094.

Sobrino, J., and N. Raissouni, 2000. Toward remote sensing methods for land cover dynamic monitoring: application to Morocco, *International Journal of Remote Sensing*, 21(2):353–366.

Tenenbaum, J.B., V. De Silva, and J.C. Langford, 2000. A global geometric framework for nonlinear dimensionality reduction, *Science*, 290(5500):2319–2323.

Vapnik, V.N., 2000. *The Nature of Statistical Learning Theory*, Springer-Verlag, Inc., New York.

West, D.B., 2001. *Introduction to Graph Theory*, Prentice Hall Upper Saddle River, New Jersey.

Wilkinson, D., R. Parker, and D. Evans, 2008. Change detection techniques for use in a state wide forest inventory program, *Photogrammetric Engineering & Remote Sensing*, 74(7):893–901.

Williams, C., and M. Seeger, 2001. Using the Nyström method to speed up kernel machines, *Proceedings of the Fifteenth Annual Conference on Neural Information Processing Systems*.

Wu, B., B. Huang, and T. Fung, 2009. Projection of land use change patterns using kernel logistic regression, *Photogrammetric Engineering & Remote Sensing*, 75(8):971–979.

Yang, X., 2011. Parameterizing support vector machines for land cover classification, *Photogrammetric Engineering & Remote Sensing*, 77(1):27–37.

Yeh, A.G.O., and X. Li, 2003. Simulation of development alternatives using neural networks, cellular automata, and GIS for urban planning, *Photogrammetric Engineering & Remote Sensing*, 69(9):1043–1052.

Zhang, J., S. Li, and J. Wang, 2005. Manifold learning and applications in recognition, *Intelligent Multimedia Processing with Soft Computing*, pp. 281–300.

Zhou, D., O. Bousquet, T.N. Lal, J. Weston, and B. Schölkopf, 2004. Learning with local and global consistency, *Proceedings of the Eighteenth Annual Conference on Neural Information Processing Systems*.

Zhou, G., X. Xu, H. Du, H. Ge, Y. Shi and, Y. Zhou, 2011. Estimating aboveground carbon of Moso bamboo forests using the k nearest neighbors technique and satellite imagery, *Photogrammetric Engineering & Remote Sensing*, 77(11):1123–1131.

Zhou, L., and X. Yang, 2010. Training algorithm performance for image classification by neural networks, *Photogrammetric Engineering & Remote Sensing*, 76(8):945–951.

Zhu, X., 2006. *Semi-supervised Learning Literature Survey*, University of Wisconsin-Madison.

Zhu, X., Z. Ghahramani, and J. Lafferty, 2003. Semi-supervised learning using Gaussian fields and harmonic functions, *Proceedings of the International Conference on Machine Learning*.

Zhu, X., and A.B. Goldberg, 2009. Introduction to semi-supervised learning, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130.

(Received 14 May 2012; accepted 09 August 2012; final version 31 October 2012)