EVALUATION OF KERNELS FOR MULTICLASS CLASSIFICATION OF HYPERSPECTRAL REMOTE SENSING DATA

Mathieu Fauvel $^{*\diamond}$, *Jocelyn Chanussot* *

Jon Atli Benediktsson $^{\diamond}$

*Laboratoire des Images et des Signaux
LIS-INPG
BP 46 - 38402 St Martin d'Heres - FRANCE

$^{\diamond}$University of Iceland
Dept. of Electrical and Computer Eng.
Hjardarhagi 2-6, 107 Reykjavik-ICELAND

ABSTRACT

Classification of hyperspectral remote sensing data with support vector machines (SVMs) is investigated. SVMs have been introduced recently in the field of remote sensing image processing. Using the kernel method, SVMs map the data into higher dimensional space to increase the separability and then fit an optimal hyperplane to separate the data. In this paper, two kernels have been considered. The generalization capability of SVMs as well as the ability of SVMs to deal with high dimensional feature spaces have been tested in the situation of very limited training set. SVMs have been tested on real hyperspectral data. The experimental results show that SVMs used with the two kernels are appropriate for remote sensing classification problems.

1. INTRODUCTION

With the development of remote sensing sensors, hyperspectral remote sensing images are now widely available. They are characterized by hundreds of spectral bands. For a classification task, the increased dimensionality of the data increases the capability to detect various classes with a better accuracy. But at the same time, classical classification techniques are facing the problem of statistical estimation in high dimensional spaces. Due to the high number of features and small number of training samples, reliable estimation of statistical parameters is difficult [1]. Furthermore, it is proved that, with a limited training set, beyond a certain limit, the classification accuracy decreases as the number of features increases (Hughes phenomenon [2]). Recently, *support vector machines* (SVMs) have shown to be well suited for high dimensional classification problems [3, 4]. With SVMs, classes are not characterized by statistical criteria but by a geometrical criterion. SVMs seek a separating hyperplane maximizing the distance to the closest training samples for two classes. This approach gives SVMs very high generalization capabilities and, as a consequence, they only require a small number of training samples. In addition, for non linearly separable data, SVMs use the kernel method to map the data onto a higher dimensional space where they are linearly separable [5].

Early work in classification of remotely sensed images by SVMs showed promising results [6, 7]. In [8], several SVM-based classifiers were compared to other classical classifiers such as a K-nearest neighbors classifier and a neural network classifier. The SVMs using the kernel method outperformed the other classifiers in terms of accuracy. Multiclass SVMs performances were also positively compared with a discriminant analysis classifier, a decision tree classifier and a feedforward neural network classifier with a limited training set [9]. Though these experiments highlight the good generalization capability of SVMs, the data used were pre-processed, i.e., 3 selected bands were used for the classification and thus, performances in high dimensional space were not investigated. In both articles [8, 9], the *Gaussian radial basis* kernels were shown to produce the best results. In [10], several spectral-based kernels were tested on hyperspectral data. These kernels were designed to handle spectral meaning, and, in particular, various non-Euclidean metrics were considered to characterize the similarity between vectors.

In this paper, multiclass SVMs are investigated for the classification of hyperspectral data without any feature reduction. Two kernels are compared : The first one is based on Euclidean distance. It is the Gaussian radial basis with L2-norm distance. The second one is based on spectral angle mapper. It basically computes the angle between two vectors in the vector space. The spectral angle is known to be scale invariant and thus a good measure of spectral shape [10, 11].

The generalization capability is also studied in the case of a limited training set. Global and average accuracies as well as Kappa coefficient of agreement are used for evaluation and comparison.

The paper is organized as follows. SVMs are briefly presented in Section 2. Data and experimental scheme are outlined in Section 3. Experimental results are discussed in Section 4. Finally, conclusions are drawn.

The authors would like to thank the IAPR - TC7 for provinding the data and Prof. Paolo Gamba and Prof. Fabio Dell'Acqua of the University of Pavia, Italy, for providing reference data. This research was supported in part by the Research Fund of the University of Iceland and the Jules Verne Program of the French and Icelandic governments (PAI EGIDE).

2. SUPPORT VECTOR MACHINES

In this section we briefly recall the general mathematical formulation of SVMs. Starting from the linearly separable case, optimal hyperplanes are introduced. Then, the classification problem is modified to handle non-linearly separable data and a brief description of multiclass strategies is given. Kernel methods are discussed and the kernels used in the experiments are presented.

2.1. Linear SVMs

For a two-class problem in a n-dimensional space $\mathbb{R}^n$, we assume that N training samples, $\mathbf{x}_i \in \mathbb{R}^n$, are available with their corresponding labels $y_i = \pm 1$: $\{(\mathbf{x}_i, y_i) \mid i \in [1, N]\}$. The SVM method consists of finding the hyperplane that maximizes the margin (see Fig. 1), i.e, the distance to the closest training data points in both classes. Noting $\mathbf{w} \in \mathbb{R}^n$ as the vector normal to the hyperplane and $b \in \mathbb{R}$ as the bias, the hyperplane H_p is defined as

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \ \forall \mathbf{x} \in H_p \tag{1}$$

where $\mathbf{w} \cdot \mathbf{x}$ is the dot product between $\mathbf{w}$ and $\mathbf{x}$. If $\mathbf{x} \notin H_p$ then $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ is the distance of $\mathbf{x}$ to H_p. According to the previous statement, such a hyperplane has to satisfy:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) > 1, \ \forall i \in [1, N]. \tag{2}$$

Finally, the optimal hyperplane has to maximize the margin: $2/\left\|\mathbf{w}\right\|$. This is equivalent to minimizing $\left\|\mathbf{w}\right\|/2$ and leads to the following quadratic optimization problem:

$$\min \left[\frac{\left\|\mathbf{w}\right\|^2}{2} \right], \text{ subject to (2).} \tag{3}$$

For non-linearly separable data, *slack* variables ξ are introduced to deal with misclassified samples (see Fig. 1). Eq. (2) becomes

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) > 1 - \xi_i, \ \xi_i \geq 0 \quad \forall i \in [1, N]. \tag{4}$$

The final optimization problem becomes:

$$\min \left[\frac{\left\|\mathbf{w}\right\|^2}{2} + C \sum_{i=1}^{N} \xi_i \right], \text{ subject to (4)} \tag{5}$$

where the constant C controls the amount of penalty. These optimization problems are usually solved by quadratic programming [3].

As a conclusion, the SVM training process consists of seeking the optimal hyperplane from one training set. The classification is done by $y_u = sgn(\mathbf{w} \cdot \mathbf{x}_u + b)$ where $(\mathbf{w}, b)$ are the hyperplane parameters found during the training process and $\mathbf{x}_u$ is an unseen sample.

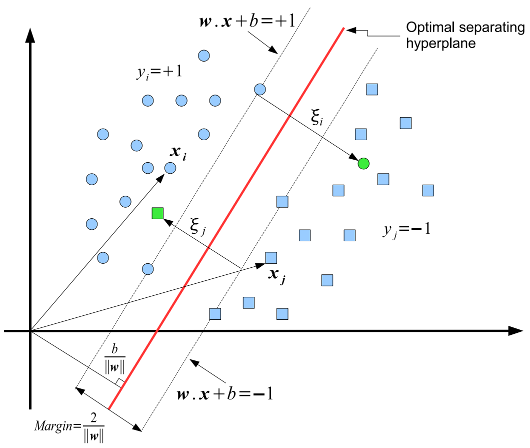
$\boldsymbol{w}.\boldsymbol{x}+b=+1$
Optimal separating hyperplane
$y_i=+1$
ξ_i
$\boldsymbol{x}_i$
ξ_j
$y_j=-1$
$\boldsymbol{x}_j$
$\frac{b}{\|\boldsymbol{w}\|}$
$\boldsymbol{w}.\boldsymbol{x}+b=-1$
$Margin=\frac{2}{\|\boldsymbol{w}\|}$

Fig. 1. Classification of a non-linearly separable case by SVMs. There is one non separable feature vector in each class.

2.2. Multiclass SVMs

SVMs are designed to solve binary problems where the class labels can only take two values: ±1. For a remote sensing application, several classes are usually of interest. Various approaches have been proposed to address this problem. They usually combine a set of binary classifiers. Two main approaches were originally proposed for a m-classes problem [5].

- **One Versus the Rest:** m binary classifiers are applied on each class against the others. Each sample is assigned to the class with the *maximum* output.

- **Pairwise Classification**: $\frac{m(m-1)}{2}$ binary classifiers are applied on each pair of classes. Each sample is assigned to the class getting the highest number of votes. A vote for a given class is defined as a classifier assigning the pattern to that class.

The *pairwise classification* has shown to be more suitable for large problems [12]. Even though the number of the used classifiers is larger than for the *one versus the rest* approach, the whole classification problem is decomposed into much simpler ones. Therefore, this second approach was used in our experiments.

2.3. Nonlinear SVMs

Kernel methods are a generalization of SVMs providing nonlinear hyperplanes and thus improving classification abilities. Input data are mapped onto a higher dimensional space $\mathbb{H}$ using a nonlinear function Φ:

$$\begin{array}{rcl} \mathbb{R}^n & \rightarrow & \mathbb{H} \\ \mathbf{x} & \rightarrow & \Phi(\mathbf{x}) \\ \mathbf{x}_i \cdot \mathbf{x}_j & \rightarrow & \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j). \end{array} \tag{6}$$

The expensive computation of $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$ in $\mathbb{H}$ is reduced using the *kernel trick* [5]:

$$\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j) \tag{7}$$

The kernel K should fulfill Mercer's condition [4]. Using kernels, we never explicitly work in $\mathbb{H}$, and all the computations are done in the original space $\mathbb{R}^n$.
For classification of remote sensing images, two kernels are widely used: the inhomogeneous polynomial function and the Gaussian radial basis function (RBF).

$$K_{POLY}(\mathbf{x}_i, \mathbf{x}_j) = [(\mathbf{x}_i \cdot \mathbf{x}_j) + 1]^p. \tag{8}$$

$$K_{GAUSS}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left[-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right]. \tag{9}$$

RBF can be written as follows [5]: $K(\mathbf{x}_i, \mathbf{x}_j) = f(d(\mathbf{x}_i, \mathbf{x}_j))$ where d is a metric on $\mathbb{R}^n$ and f is a function on $\mathbb{R}_0^+$. For the Gaussian RBF, $f(t) = \exp(-\gamma t^2)$, $t \in \mathbb{R}_0^+$, and $d(\mathbf{x}_i, \mathbf{x}_j)) = \|\mathbf{x}_i - \mathbf{x}_j\|$, i.e., the Euclidean distance. As mentioned in [11], Euclidean distance is not scale invariant, however due to atmospheric attenuation or variation in illumination, spectral energy can be different for two samples even if they belong to the same class. To handle such a problematic case, scale invariant metrics can be considered. *Spectral Angle Mapper* (SAM) is a well known scale invariant metric, it has been widely used in many remote sensing problems and it has been shown to be robust to variations in spectral energy [11]. This metric α focuses on the angle between two vectors:

$$\alpha(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \,.\, \|\mathbf{x}_j\|}\right). \tag{10}$$

In this paper, we compare RBF kernels with the Euclidean distance (9) and the spectral angle mapper (11).

$$K_{SAM}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left[-\gamma \alpha(\mathbf{x}_i, \mathbf{x}_j)^2\right]. \tag{11}$$

Both kernels fulfill Mercer's conditions and optimal hyperplanes can therefore be found.

3. DATA AND CLASSIFICATION SCHEME

The data used in the experiments are ROSIS (Reflective Optical System Imaging Spectrometer) provided by DLR. These data are very high-resolution hyperspectral data. The used imagery is of Pavia, Italy. It is 492 by 1096 pixels and contains 102 spectral bands. A three-color composite image is shown in Fig. 2.(a). Training and test sets are listed in Table 1. Small training sets were randomly extracted from the training set and were composed of 10, 20, 40, 60, 80 and 100 pixels by class, respectively. The SVMs were trained with each of these training subsets and then evaluated with the whole test set. These experiments were repeated five times (with five independent training subsets) and the mean accuracy values were reported. During each training process, the kernel parameter γ and the penalty term C were adjusted to maximize the estimated overall accuracy, which was computed using a fivefold *cross validation* [5]. The SVMs were computed using the LIBSVM library [13] and the program was modified to include SAM kernel.

Table 1. Information classes and samples.

Class		Samples	
No	Name	Train	Test
1	Water	745	65278
2	Trees	785	6508
3	Meadow	797	2900
4	Brick	485	2140
5	Soil	820	6549
6	Asphalt	816	7555
7	Bitumen	808	6479
8	Tile	223	3122
9	Shadow	195	2165
Total		5536	103504

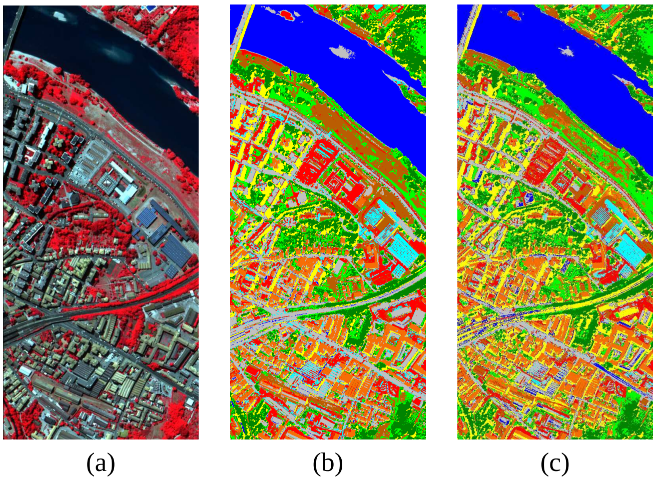

Fig. 2. (a) Original hyperspectral image, three-channel color composite. (b) Thematic map produced with the Gaussian RBF kernel SVMs with 10 training pixels by class. (c) Thematic map produced with the SAM RBF kernel SVMs with 10 training pixels by class.

4. EXPERIMENTS

Table 2 summarizes the results obtained using the Gaussian and the SAM RBF kernels. These values were extracted from the *confusion matrix* [14]. The overall accuracy (OA) is the percentage of correctly classified pixels whereas the average accuracy (AA) represents the average of class classification accuracies. Kappa coefficient is another criterion classically used in remote senig classification to measure the degree of agreement [14] and takes into account the correct classification that may have been obtained ”by chance” by weighting the measured accuracies.
SVMs generalizes very well: with only 10 training pixels per

Table 2. Classification Accuracies for the Gaussian and the SAM RBF kernel.

Training Set Size	**OA%**		**AA%**		**Kappa Coef.**	
	Gaussian	SAM	Gaussian	SAM	Gaussian	SAM
10	93,85	93,32	88,76	86,36	0,90	0,89
20	94,51	93,87	91,00	88,64	0,91	0,90
40	94,51	93,79	92,66	91,26	0,92	0,90
60	94,71	94,23	92,04	91,67	0,91	0,90
80	95,36	94,40	93,24	91,89	0,92	0,90
100	95,29	94,54	93,39	92,61	0,92	0,91
All	96,45	95,56	95,08	94,26	0,94	0,93

class more than 90% accuracy is reached by both kernels. It is also clear that the classification accuracy is correlated to the training set size. But the difference in terms of accuracy is fairly low: for instance, with the Gaussian RBF kernel, the OA obtained with only 10 training pixels per class is only $2,7\%$ lower than the OA obtained with the complete training set. However, the computation time (including optimal parameters selection, training and classification) requires only about 10 minutes with 10 training pixels compared to more than 12 hours with the full training set.

The use of the SAM kernel gives slightly degraded classification results for the OO, OA and the Kappa coefficient. However, with most of the accuracies over 90%, this kernel seems also promising for the classification of hyperspectral remote sensing images. Thematic maps provided by SVMs classification with the Gaussian and SAM kernel are shown in Fig. 2.(b) and (c), respectively.

5. CONCLUSIONS

In this paper, the classification of hyperspectral remote sensing data using support vector machines was investigated. SVMs proved to provide very accurate classification, even in the case of a very limited number of training sample and high dimensional data. Two kernels have been compared, the well known Gaussian RBF kernel and a kernel based on the spectral angle mapper. From our experiments, both gave excellent results in terms of classification accuracy, the Gaussian kernel slightly outperforming the SAM kernel. An explanation lies in the fact that urban scenes are less sensitive to spectral variations than agricultural areas, with weeds at various development steps and different layers casting shadows. A perspective of this work surely lies in the combination of these kernels to further improve the classification results using decision fusion.

6. REFERENCES

[1] C. Lee and D. A. Landgrebe, “Analysing high dimensional multispectral data,” *IEEE Trans. on Geoscience and Remote Sensing*, vol. 31, pp. 792–800, July 1993.

[2] G. F. Hughes, “On the mean accuracy of statistical pattern recognizers,” *IEEE Trans. on Information Theory*, vol. IT-14, pp. 55–63, January 1968.

[3] V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.

[4] C.J.C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data mining and knowledge discovery*, vol. 2, pp. 121–167, 1998.

[5] B. Scholkopf and A. J. Smola, *Learning with Kernels*, MIT Press, 2002.

[6] J. A. Gualtieri and S. Chettri, “Support vector machines for classification of hyperspectral data,” in *Geoscience and Remote Sensing Symposium*. IGARSS ’00. Proceedings, July 2000, vol. 2.

[7] G. H. Halldorsson, J. A. Benediktsson, and J. R. Sveinsson, “Support vector machines in multisource classification,” in *Geoscience and Remote Sensing Symposium*. IGARSS ’03. Proceedings, July 2003, vol. 3.

[8] F. Melgani and L. Bruzzone, “Classification of hyperspectral remote sensing images with support vector machines,” *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, pp. 1778–1790, August 2004.

[9] G. F. Foody and Ajay Mathur, “A relative evaluation of multiclass image classification by support vector machines,” *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, pp. 1335–1343, June 2004.

[10] M. Lennon G. Mercier, “Support vector machines for hyperspectral image classification with spectral-based kernels,” in *Geoscience and Remote Sensing Symposium*. IGARSS ’03. Proceedings, July 2003, vol. 1.

[11] N. Keshava, “Distance metrics and band selection in hyperspectral processing with application to material identification and spectral librairies,” *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, pp. 1552–1565, July 2004.

[12] C. W. Hsu and C. J. Lin, “A comparison of methods for multiclass support vector machines,” *IEEE Trans. on Neural Networks*, vol. 13, pp. 415–425, March 2002.

[13] C. Chang and C. Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at *http://www.csie.ntu.edu.tw/ cjlin/libsvm.*

[14] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*, Springer, 1999.