Application of Feature Selection and Classifier Ensembles for the Classification of Hyperspectral Data

Y.Maghsoudi, A.Alimohammadi, M.J.Valadan Zoej and B. Mojaradi

Faculty of Geomatics Eng., K. N. Toosi University of Technology, Tehran, Iran

ymaghsoudi@yahoo.com, alimoh_abb@yahoo.com, valadanzouj@kntu.ac.ir, Mojaradi@alborz.kntu.ac.ir

Abstract:

The improved spectral resolution of modern hyperspectral sensors provides capabilities for discrimination of subtly different classes and objects. However, in order to obtain statistically reliable classification results, the number of required training samples increases exponentially as the number of spectral bands increases. However, in many situations, acquisition of the large number of training samples for the high-dimensional datasets may not be feasible. Multiple classifiers have been regarded as a promising solution for this problem. In this paper, creation of ensemble of classifiers based on feature selection has been evaluated and an effective strategy for generation of feature subsets has been proposed. The proposed method is based on generating multiple feature subsets by running feature selection algorithm several times, with the aim of discrimination of one class from the others each time. Each of the final subsets of features is selected so as to have the capability for discrimination of one of the classes. Each of these subsets is then passed to the maximum likelihood classifier. Finally a combination scheme is used to combine the outputs of individual classifiers. Practical examinations on the AVIRIS data for discrimination of different land cover classes demonstrate the effectiveness of the proposed strategy.

Keywords: hyperspectral, classification, ensembles, feature selection, class-based

1. Introduction

Recent developments in sensor technology have made it possible to collect hyperspectral data from 200 to 400 spectral bands. These data, can provide more effective information for monitoring of the earth surface and a better discrimination among ground cover classes than the traditional multispectral scanners. However, the data analysis approach that has been successfully applied to multispectral data in the past is not so effective for hyperspectral data. Because, the existing stochastic approaches often fail to achieve satisfactory results for hyperspectral data.

Classification of hyperspectral images is challenging. The classification performance in these images suffers from two important problems:

1. Curse of dimensionality*;* the accuracy of parameter estimation depends substantially on the ratio of the number of training samples to the dimensionality of the feature space. As the dimensionality increases, the number of training samples as needed for characterization of classes increases considerably. If the size of the training samples fails to satisfy the requirements, which is the case for the hyperspectral images, the estimated statistics, becomes very unreliable. Although increasing the number of spectral bands potentially provides more capabilities for discrimination of classes, this positive effect can be diluted by poor statistics estimation. As a result, the classification accuracy first grows and then declines with the number of spectral bands when the number of the training samples is low, finite and remains constant. This is often referred to as the Hughes Phenomenon or the curse of dimensionality [1]. As it is often difficult to provide adequate training samples for supervised classification, an ensemble of classifiers can be used to solve this problem.

2. Large hypothesis space; In general there are three spaces associated with any classification problem :(i) Input space , which is the space of all the features that are used in the classification process ,(ii) Output space which is the set of all observed classes. This space is the most powerful one from the standpoint of information extraction [2] and (iii) Hypothesis space which is the space of the models in which the desired classifier is sought. In other words it is a link between the input and output spaces. With increase in the input dimensionality, for a fixed number of classes and choice of a classifier family, the hypothesis space also grows exponentially. This problem makes the classification performance very unreliable. By using an ensemble of classifiers this problem can also be avoided.

An ensemble of classifiers requires two conditions to be met in order to reduce the generalization error of its constituent members [14]. Firstly the classifiers must be diverse. Obviously ensembling identical classifiers will not lead to any improvements. To be precise about what diversity means, classifiers should be independent i.e. making uncorrelated errors. Secondly the classifiers should be accurate. An accurate classifier is one that has an error rate of better than random guessing on a new data point. If the classifiers are an average accurate and diverse then we would expect that most of the classifiers will not make the same mistake on the same example. A simple majority voting schema would ensure that the correct classification is made.

Design of classifier ensembles consists of two parts. The first part is constructing multiple classifiers for creation of a set of diverse and accurate classifiers and the second part is the design of a combination scheme for implementation of fusion mechanism that can optimally combine the classifications.

In this paper a new method for constructing an ensemble of classifiers has been proposed. The method is based on finding the best features that can discriminate a class from the rest. A feature selection process is run several times each time for the discrimination of one of the classes. Each of the final selected subsets of features has the potential for optimal discrimination of one of the classes. Using an ensemble of these classifiers can lead to a better classification result.

The paper is organized into 3 sections. Section 2 presents a literature survey on feature selection algorithms. In section 3 different methods for constructing an ensemble of classifiers have been reviewed. Section 4 describes the proposed method. The experimental results on AVIRIS data have been presented in section 5.

2. Feature Selection Algorithms

A key stage in constructing classifiers is the selection of the best discriminative and informative features. The performance of most classifiers is improved when correlated or irrelevant features are removed. A large number of algorithms have been proposed for feature selection and some comparative studies have been carried out [3-6].

The feature selection problem can be stated as follows: Given a set of N features find the best subset of m features to be used for classification. Feature selection algorithms generally involve both a search strategy and an evaluation function [7-8]. The aim of the search algorithm is to generate subsets of features from the original feature space and the evaluation function compares these feature subsets in terms of discrimination. The output of the feature selection algorithm is the best feature subset found for this purpose.

Optimal search algorithms determine the best feature subset in terms of an evaluation function, whereas suboptimal search algorithms determine a good feature subset. Even for a medium sized feature set, optimal search algorithms are exhaustive and prohibitive.

Branch and bound method proposed by Narenda and Fukunaga [9] gives the optimal solution. It starts searching from the original feature space and proceeds by removing features from the set. A bound is placed on the value of evaluation function to create a rapid search. As the evaluation function obeys the monotonicity principle, any feature subset for which the value is less than the bound is removed from the search space.

Sequential methods include a well-established family for feature selection. They progressively add and discard features according to a certain strategy, ranging from sequential forward and backward selection methods (SFS, SBS) to the more complex generalized plus-l take-away-r algorithm [3]. SFS starts from an empty set and in each iteration it generates new feature sets by adding a feature selected by some evaluation function. On the other hand SBS starts from a complete set and in each iteration generates new subsets by removing a feature selected by some evaluation function. The main problems of these two algorithms are that the selected features in each iteration can not be removed in SFS and the discarded features can’t be reselected in SBS.

To overcome these problems Pudil et al. [5] proposed the floating versions of SFS and SBS. Sequential forward floating search algorithms (SFFS) can backtrack unlimitedly as long as the backtrack finds a better feature subset. SBFS is the backward version. For very high dimensional data these two methods are very effective.

Genetic feature selectors are a series of feature selection methods which use genetic algorithm to guide the selection process. The genetic algorithm for feature selection was proposed by Siedlecki and Sklansky [10]. In genetic feature selection each feature subset is represented by a chromosome which is binary string including 0's and 1's, which corresponds to a discarded or selected features respectively. New chromosomes are generated using crossover, mutation and reproduction operators. Ferri et al. [4] showed that the performance of genetic feature selectors deteriorates when the size of the complete feature set increases.

Serpico et all. [11] proposed steepest ascent (SA) search algorithm for feature selection in hyperspectral data. It is based on the representation of the problem solution by a discrete binary space and on the search for constrained local maximas of a criterion function in such space. A feature subset is a local maximum of the criterion function if the value of that feature subset criterion function is greater than or equal to the value the criterion function takes on any other point of the neighborhood of that subspace. They also proposed fast constrained (FC) search algorithm which is the computationaly reduced version of SA. The number of iteration in this algorithm is deterministic. Although this algorithm is expected to be less effective than SA, but it is always faster than or as fast as SA. A comparative study on AVIRIS data has shown that SA and FC allowed greater improvements than SFFS and differences between SA and FC are negligible.

Langley [12] put the feature selection methods into two groups: filters and wrappers. Filter methods are independent of the classifier to be used and the evaluation function is usually one of the inter-class distance measures e.g. Bhattacharyya or Jeffries-Matusita (JM) distance whereas wrappers utilize the classifier as the evaluation function. Since the classifier is ignored in filter methods there is no interaction between the biases of the feature selector and the classifier and the quality of the best selected feature subset is not as effective as the subset selected using a wrapper model [13].

3. Methods for Creating Multiple Classifiers

There are many methods for creating an ensemble with the mentioned properties in section 1. Three main families of ensemble generation methods are: Manipulating training data, Manipulating input features and Manipulating output classes. The first method of generating an ensemble of classifiers is to train classifiers on different sets of training data. Bagging [15] which uses sampling with replacement is one of the best known methods for generating a set of classifiers. In bagging we create n different training sets by sampling with replacement from the original training set. We then train a classifier on each set and combine their outputs using a simple voting. A popular alternative to Bagging is Boosting [16]. In Boosting the classifiers in the ensemble are trained serially, with the weights on the training instances set according to the performance of the previous classifiers. The main idea is that the classification algorithm should concentrate on the difficult instances. These methods are only effective for unstable learning algorithms. By "unstable" we mean the learning algorithms whose output predictions show considerable changes in response to a small change in the training samples (e.g. neural networks and decision trees). Another method for generating multiple classifiers is to manipulate the set of input features. In this method different feature subspaces are passed to different classifiers. In order to increase the diversity among different subspaces, the sampling of features from the original set of features can be randomly selected [17]. There are different sampling functions: sampling with replacement and sampling without replacement. In sampling with replacement a feature can be selected more than once. Obviously, this method works well when the input features are highly redundant. On the other hand this approach does not suffer from curse of dimensionality. The third method for generating a good ensemble of classifiers is through manipulating the output classes. Error Correcting Output Coding (ECOC) proposed by Dietterich and Bakiri [18] is one of these methods. In this method a multiclass problem is decomposed into multiple two-class problems. Suppose that the number of classes, k, is large. New learning problems can be constructed by randomly partitioning the k classes into two subsets A_m and B_m. The input data can then be relabeled so that any of the original classes in set A_m are given the label 0 and the original classes in set B_m are given the label 1. N classifiers are trained on each of these two-class problems made up of A_m and B_m. Finally an ensemble of N classifiers is obtained. For classification of a new object, the output of N classifiers are processed and the new object is assigned to the class received the highest number of votes. Dietterich and Bakiri [18] showed that ECOC could improve decision trees and neural networks and it does not work with classifiers that use local information such as the KNN approach.

4. The Proposed Method

In this paper a new method for creating multiple classifiers has been proposed. As mentioned in section 2 accuracy and diversity are two conditions for creating an effective set of classifiers. In most of the proposed methods one condition is usually sacrificed for another. For example in Random Subspace Method [17] although the classifiers are quite diverse but they are not expected to have a high accuracy, so their ensemble may not lead to a good result.

In order to overcome this problem and the problems mentioned in section 1 a class-based feature selection for creating an ensemble of classifiers has been proposed. The main idea of the method is that from the huge number of spectral bands in hyperspectral data there are some bands which can discriminate each class better than the others. Assume that there are k classes in the classification problem. In order to find the best features for each of the classes we applied a feature selection process. In this paper the Fast Constrained search algorithm is used as the search strategy. As mentioned in section 3 the Fast Constrained search algorithm is very fast and the quality of the selected features is comparable to many other search algorithms. The classification accuracy of each class was used as a criterion for the evaluation of feature subsets. Classification accuracy of each class is defined as the percentage of independent test samples of each class correctly classified by the classifier. Bayesian classifier was used as the classification algorithm. For each class, the feature subset with the highest evaluation function is subsequently passed to the Bayesian classifier. This process is repeated for all the classes. The method is schematically represented in Fig. 1. Finally a combination scheme is used to combine the outputs of individual classifiers. The first classifier shares its first class values in the final classified image, the second classifier plays this role by sharing its second class values and so on. For those pixels in the final classified image in which there is an overlap between the decisions of classifiers the pixel label with the highest value of probability is selected as the final decision.

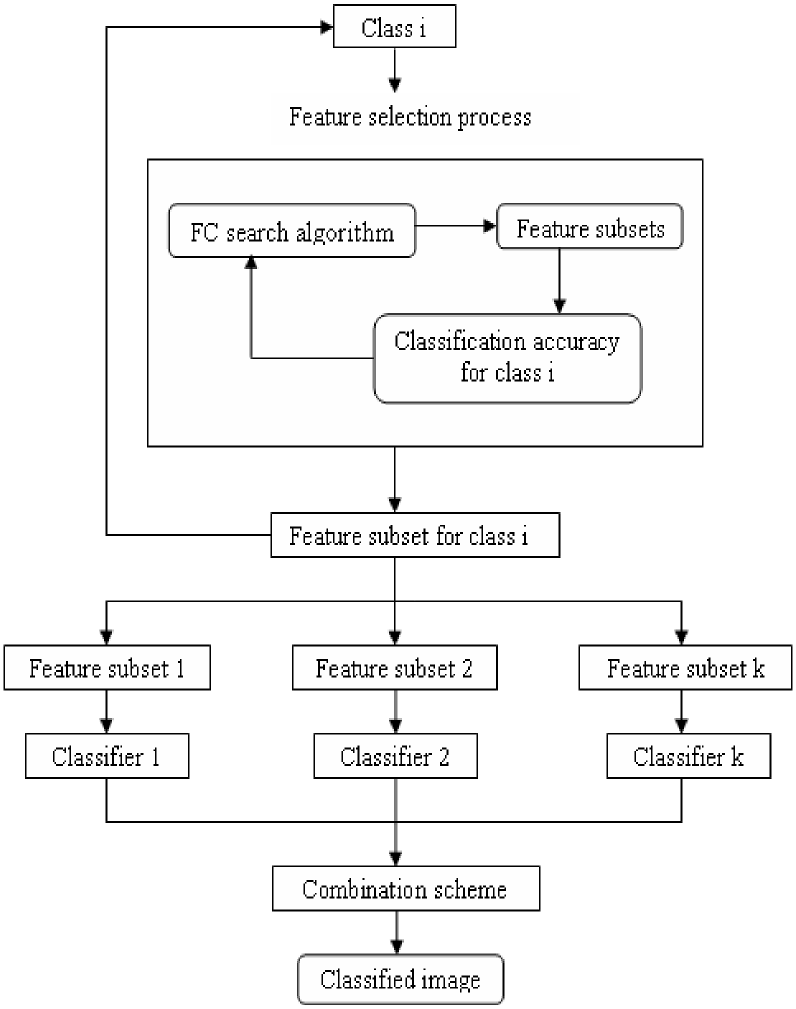

Fig. 1: A schematic illustration of the proposed method

Table 1. List of classes and training and test sample size

Land cover classes	Number of Training.	Number of Test
1.Alfalfa	34	44
2.corn_notill	375	201
3.Corn_min	213	195
4.Grass_pasture	200	136
5.Grass_Trees	307	221
6.Grass/pasture mowed	579	914
7.Hay_windrowed	257	126
8.Oatas	110	45
9.Soy_notil	355	277
10.Soy_clean	190	170
11.Woods	304	554
12.Corn	143	219
Total	3067	3102

Fig. 2 : Band 12 of the hyperspectral image utilized in the experiments.

5. Experiments

5.1. Dataset Description

The dataset used in this study is an AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) dataset downloaded from [19] along with a ground truth image, containing 12 classes. The considered dataset referred to the agricultural area of Indian pie in the Northern part of Indiana. Images were acquired by an AVIRIS in June 1992. The dataset was composed of 220 spectral channels (spaced at about 10 nm) acquired in the 0.4-2.5 um region. Hyperspectral image bands are often highly correlated and among them some of the absorption bands contain little signal but noise. Processing of the original spectral bands not only is inefficient but also tends to create poor results. So the Maximum Noise Fraction transform (MNF) [20] is applied on the image. MNF consists in projecting the original image in a space where the new components are sorted in order of SNR. The method first estimates the noise level in the original image by taking advantage of the spatial correlation between pixels. Then, a first Principal Component Analysis (PCA) is applied to the data, by using the estimated noise covariance matrix, leading to spectrally whitened noise, independent from the data. A second PCA is applied on the projected data, leading to maximize the SNR of the new successive components [21]. This data representation is very interesting because it allows the filtering to be adapted to the SNR of each component in the transformed space. By considering only the first high eignvalues and using an inverse MNF we finally, allow the filtered image to be reprojected in the original space. An example of the resulting transformed image is detailed in figure 2.

5.2. Implementation

Experiments were carried out to evaluate the performance of the proposed method. In this study 30 features were used for each feature subset. The training data are used to train the Bayesian classifier using different feature subsets and the test data are used to evaluate the results. Table 2 shows the number of training and test data for each of the 12 classes. In order to assess the efficiency of using the classification accuracy as an evaluation function the same procedure is conducted using Jeffries-Matusita distance which is an inter-class measure (Table 2). The Jeffries-Matusita distance is as follow:

$$JM = 2\sum_{n=1}^{k}\sum_{m>n}^{k} JM_{mn} \tag{1}$$

$$JM_{mn} = \sqrt{2(1-e^{-bmn})} \tag{2}$$

$$b_{mn} = \frac{1}{8}(M_m - M_n)^T\left(\frac{C_m + C_n}{2}\right)^{-1}(M_m - M_n) + \frac{1}{2}\ln\left(\frac{\left|\frac{C_m + C_n}{2}\right|}{\sqrt{|C_m||C_n|}}\right) \tag{3}$$

where k is the number of classes, b_{ij} is the Bhattacharyya distance between class i and j and M_i and C_i are the mean vector and covariance matrix of the class i respectively.

Table 2. Performance of the proposed approach as compared with the others

Classes	Classification Accuracy(Percent)					
	Random Subspace Method				Jeffries-Matusita	***Proposed Method***
	MAX	MIN	MEAN	PRODUCT		
Corn-notill	31.343	35.323	26.866	27.363	42.786	***59.204***
Corn-min	52.821	54.359	58.974	58.462	60.513	***66.667***
Grass/pasture	81.618	80.882	80.882	80.882	80.882	***83.088***
Grass/trees	92.308	90.95	92.76	92.76	93.213	***93.213***
Grass/pasture-mowed	47.812	57.221	58.534	57.768	61.16	***59.628***
Woods	86.282	76.715	86.282	86.462	91.111	***86.667***
Oats	86.667	86.667	95.556	95.556	70.397	***72.563***
Soy-notill	63.538	67.87	69.314	68.953	91.765	***90.588***
Alfalfa	78.571	78.571	90.476	90.476	33.333	***85.714***
Soy-clean	91.765	90.588	92.941	92.941	81.227	***89.531***
Hay-windrowed	99.206	96.032	99.206	99.206	99.206	***99.206***
Corn	18.265	16.895	12.329	12.329	18.265	***15.982***
Overall Accuracy	63.387	64.774	67.097	66.871	67.74	***70.94***

On the other hand in order to show the efficiency of using class-based feature selection for creating an ensemble of classifiers a random subspace method (RSM) is also applied on the same data. In this method different feature subsets are randomly selected and passed to the same classifier (Bayesian classifier in this study). The outputs of each maximum likelihood classifier are vectors of probabilities for different classes. Four operators including the Max, Min, Mean and Product are then used to combine the outputs of the individual classifiers. The results are illustrated in table 3. Results of this experiment have shown that the use of the proposed method leads to superior performance as compared to other methods and classification accuracy shows considerable increases for most of the classes (Table 2). Although some classes show little increase or even decrease in classification accuracy but the overall accuracy shows a relative increase in the proposed method.

5. Conclusions

Results of this research have shown that using a class-based feature selection can be an effective method for creation of a suitable set of classifiers. Application of this approach, not only can solve the problem of small training samples but it can also serve as a good approach for utilization of the redundant features in hyperspectral data.

Although employment of the classification accuracy of the classes as the evaluation function is computationally burdensome, but it can lead to minimization of the biases inherent in the feature selection algorithm and the classifier. As the feature selection is based on the performance of the classifier that later classifies the selected features, the accuracy level is expected to increase accordingly.

Utilization of the class accuracy as the evaluation function seems to be a good procedure for parallel computing. Because in this approach, the best features for one class can be selected independent from the others. Therefore, the process for each class may be run in different computers and then, the time consuming problem of the wrapper methods can be almost solved.

References

[1] G.F. Hughes. On the mean accuracy of statistical pattern recognizers, IEEE Transactions Information Theory, pp. 55- 63, 1968.

[2] David Landgrebe. On Information Extraction Principles for Hyperspectral Data, School of Electrical and computer engineering, Purdue University, pp.168-173,July 1997.

[3] J. Kittler. Feature set search algorithm. In C.H. Chen, editor, Pattern Recognition and Signal Processing, pages 41–60. Sithof and Noordhoff, Alphen aan den Rjin, The Netherlands, 1978.

[4] F.J. Ferri, P. Pudil, M. Hatef, and J. Kittler. Comparative study of techniques for large-scale feature selection. In E.S. Gelsema and L.N. Kanal, editors, Pattern Recognition in Practice, volume IV, pages 403–413. Elsevier Science B.V., 1994.

[5] P. Pudil, J. Novovicova, and J. Kittler. Floating search methods in feature selection. Pattern Recognition Letters, 15:1119–1125, 1994.

[6] A. Jain and D. Zongker. Feature selection: Evaluation, application, and small sample performance. IEEE Trans. Pattern Analysis and Machine Intelligence, 19(2):153–158, 1997.

[7] P. H. Swain and S. M. Davis, Remote sensing: the quantitative approach. New York: McGraw-Hill, 1978.

[8] K. Fukunaga. Introduction to Statistical Pattern Recognition. Academic Press, San Diego, California, 1990.

[9] P. Narendra and K. Fukunaga. A branch and bound algorithm for feature subset selection.IEEE Trans. Computer, C-26(9):917–922, 1977.

[10] W. Siedlecki and J. Sklansky. A note on genetic algorithms for large-scale feature selection. Pattern Recognition Letters, 10:335–347, 1989.

[11] S. B. Serpico and L. Bruzzone, A New Search Algorithm for Feature Selection in Hyperspectral Remote Sensing Images. IEEE Tr. Geoscience and Remote Sensing, vol. 39,(7), pp. 1360–1367, 2001.

[12] A.L. Blum and P. Langley. Selection of relevant features and examples in machine learning. Artificial Intelligence, 97:245–271, 1997.

[13] G.H. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In Proceedings of the Eleventh International Conference on Machine learning, pages 121–129,
New Brunswick, NJ, 1994. Morgan Kaufmann.

[14] T.G. Dietterich. Ensemble methods in machine learning. In Proc. of MCS 2000, Lecture Notes in Computer Science, pp.1-15, 2000.

[15] L. Breiman. Bagging predictors, Machine Learning, pp.123 140,1996.

[16] R. E. Schapire. The strength of weak learnability, Machine Learning,pp.197-227, 1990.

[17] Ho, T.K. The random subspace method for constructing decision forests. IEEE Transactions, Pattern Analysis and Machine Intelligence,pp.832 844, 1998.

[18] T.G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes, Journal of Artificial Intelligence Research, pp.263 286, 1995.

[19] http://dynamo.ecn.purdue.edu/ ˜biehl/multispec/documentation.html. [13] A.A. Green, M. Berman, P. Switzer, and M.D Craig, “A transformation for ordering multispectral data in terms of image quality with implications for noise removal,” IEEE Tr. Geoscience and Remote Sensing, vol. 26(1), pp. 65–74, 1988.

[20] A.A. Green, M. Berman, P. Switzer, and M.D Craig, “A transformation for ordering multispectral data in terms of image quality with implications for noise removal,” IEEE Tr. Geoscience and Remote Sensing, vol. 26(1), pp. 65–74, 1988.

[21]J.B. Lee, S. Woodyatt, and M. Berman, “Enhancement of high spectral resolution remote sensing data by a noise-adjusted principal components transform,” IEEE Tr. Geoscience and Remote Sensing, vol. 28(3), pp. 295–304, 1990.