Classifying hyperspectral airborne imagery for vegetation survey along coastlines

Pieter Kempeneers
Bart Deronde
Luc Bertels
Walter Debruyn
Vito (Flemisch Institute for Technological Research)
Boeretang 200, B-2400 Mol, Belgium
Email: pieter.kempeneers@vito.be
Telephone: +32 14 336820

Steve De Backer
Paul Scheunders
University of Antwerp
Groenenborgerlaan 171, B-2020 Antwerpen, Belgium

***Abstract*— This paper studies the potential of airborne hyperspectral imagery for classifying vegetation along the Belgian coastlines. Here, the aim is to build vegetation maps using automatic classification. Besides a general linear multi-class classifier (Linear Discriminant Analysis), several strategies for combining binary classifiers are proposed: one based on a hierarchical decision tree, one based on the Hamming distance between the codewords obtained by binary classifiers and one based on the coupling of posterior probabilities. In addition, a new procedure is proposed for spatial classification smoothing. This procedure takes into account spatial information by letting the decision for classification of a pixel depend on the classification probabilities of neighboring pixels. This is shown to render smoother classification images.**

I. Introduction

Vegetation along coastlines is important to survey because of its biological value with respect to the conservation of nature, but also for security reasons. Some of the vegetation tend to fix the natural seawall, while others do not. Erosion accompanied by the rough conditions along coastlines, reinforce the dynamic process of the existing vegetation. In order to monitor this process, vegetation maps are required on a regular basis. Therefore, an automatic classification is aimed for.

The objective is to monitor a large variety of vegetation types for the entire Belgian coastline (about 30 square kilometers). This, together with the requirements on spatial resolution and, not in the least, the cloudy weather conditions, urge on airborne hyperspectral imagery. A test area at the west coast of Belgium has been selected for which data cubes are obtained from the Compact Airborne Spectrographic Imager (CASI-2) sensor. The data has been acquired in October 2002 with 48 spectral bands and a spatial resolution of 1.3 meters.

In this paper, classification of this test site is performed. 13 vegetation classes were selected. For such high number of classes, multi-class classification becomes very complex. In this work, we investigate the use of combination of binary classifiers. Besides the standard technique of maximum voting, three methods of combination are proposed: one using a hierarchical decision tree approach, one based on the Hamming distance between codewords, obtained by binary classifiers and one based on the coupling of posterior probabilities of binary classifiers. Using the latter approach, we also introduce a spatial smoothing procedure of the classification result. This procedure combines posterior classification probabilities of neighboring pixels, to render smoother classification maps. In the next section, the binary linear classifier is introduced. In section III, the multi-class classifier is presented and the three different combinations of binary classifiers are proposed. In section IV, the classification smoothing procedure is elaborated and in section V, the experiments are conducted.

II. Binary Classification

For the binary classifier, we adopted a simple linear discriminant classifier (LDA) [1]. Assuming equal covariance matrices Σ for both classes, this classifier finds the optimal linear decision boundary. A projection weight vector $\vec{\beta}$ and bias β_0 are the parameters to estimate in the two class problem, and are calculated by :

$$\vec{\beta} = \Sigma^{-1}(\vec{\mu}_2 - \vec{\mu}_1) \qquad \beta_0 = -\frac{\vec{\beta}^T}{2}(\vec{\mu}_1 + \vec{\mu}_2) \tag{1}$$

where $\vec{\mu}_1$ and $\vec{\mu}_2$ are the means of each class, and Σ is the estimated class covariance matrix (we assume equal prior probability for both classes). Test samples ($\vec{x}$) are then classified by the simple rule

$$\vec{\beta}^T\vec{x} + \beta_0 \quad \begin{cases} \leq 0 & : \text{sample assigned to class 1} \\ > 0 & : \text{sample assigned to class 2.} \end{cases} \tag{2}$$

This method is very fast to train and to calculate the classification. In case the training set is not sufficiently large Σ can become singular. In these cases a pseudo-inverse approach can be used to find $\vec{\beta}$ and β_0 [2].

In this work, we are not only interested in the assigned class, but in the posterior probabilities for both classes, which

are estimated by:

$$p(\text{class } i|\vec{x}) = \frac{p(\vec{x}|\text{class } i)p(\text{class } i)}{\sum_{j=1,2} p(\vec{x}|\text{class } j)p(\text{class } j)} \qquad i = 1,2$$

where (3)

$$p(\vec{x}|\text{class } i) = \frac{1}{\sqrt{2\pi\vec{\beta}^T\Sigma_i\vec{\beta}}} \exp\left(-\frac{\left(\vec{\beta}^T(\vec{\mu}_i - \vec{x})\right)^2}{2(\vec{\beta}^T\Sigma_i\vec{\beta})}\right).$$

being the probability of the projected point.

III. Multi-class Classification

A. Linear Multi-class Classifier

The most widely used classifier for multi-class problems is based on the normal distributed conditional probabilities. Here, all classes are described by a normal distribution with mean μ_i, and covariance Σ_i. It is easy to show [1] that this assumption results in quadratic discriminant functions g_i:

$$g_i(\vec{x}) = \log\left(p(\text{class } i)\right) - \frac{1}{2}\log(|\Sigma_i|) - \frac{1}{2}(\vec{\mu}_i - \vec{x})^T\Sigma_i^{-1}(\vec{\mu}_i - \vec{x}) \tag{4}$$

When using equal covariances for all classes, linear discriminant functions are obtained:

$$g_i(\vec{x}) = \log\left(p(\text{class } i)\right) - \frac{1}{2}\vec{\mu}_i\Sigma^{-1}\vec{\mu}_i - \frac{1}{2}\vec{x}^T\Sigma_i^{-1}\vec{\mu}_i \tag{5}$$

A point $\vec{x}$ is assigned to class j for which $g_j(\vec{x}) \geq g_i(\vec{x})\forall i$.

B. Combining Binary Classifiers

Due to the complexity of multi-class classifiers, a common approach is to combine the output of several binary ones. Mostly, one-against-all or one-against-one [3], [4] approaches are used. With the one-against-all strategy, each classifier is trained to differentiate one class from all the others, which requires a number of classifiers equal to the number of classes K. In the one-against-one approach, all possible pairs of classes are compared, requiring $\frac{K(K-1)}{2}$ classifiers. Different methods defining other codings of the classes were also suggested [5], [6]. Here, we will apply the one-against-one scheme. To combine these one-against-one binary classifiers several approaches are proposed.

1) Maximum Voting: Often, a maximum voting mechanism is used [4]. For each binary classification, a vote is given to the winning class. The class with the maximum number of votes is assigned to the test sample.

2) Hierarchical decision tree: The binary classifiers are ordered according to their discriminating ability on the training samples. The Bayes error, derived from equation 3, can be used for this purpose. To classify a test sample, the most discriminating binary classifier is applied first. Suppose this is a classifier for classes j and k. The most likely class, for example k, is retained as a candidate for the class decision. From all the remaining binary classifiers, those that test for class j can be discarded for this test sample. As a result, the number of classifiers is reduced from $\frac{K(K-1)}{2}$ to $K-1$. The procedure ends when a single candidate class is left. This explains the importance of the ordering. A wrong decision in the beginning of the decision tree is disastrous. However, if well ordered, this scheme has a distinct advantage over maximum voting, in particular for heterogeneous classes with diverse discriminating abilities.

3) Codewords: Another method is based on the Hamming distance between codewords, built, by the binary classifiers. Each binary classifier represents one bit in the codeword. For example, if the result of the classifier for class j and k, is class j, then the corresponding bit is set to 0. Else, it is set to 1. As a result, a codeword of $\frac{K(K-1)}{2}$ bits is obtained. A codeword is created for each training sample during training. Likewise, the classifier builds a codeword for a test sample. The class containing the training sample with the nearest codeword in Hamming distance is then assigned to the test sample.

4) Coupling Probabilities: So far, all presented combinations of binary classifiers into a multiclass procedure use the binary classification result to come to a class decision. But, for each of the binary classifiers a posterior probability can be obtained from (3). We will follow [7] to obtain a combined posterior probability for the multiclass case. Define $r_{ij}(\vec{x})$ as the probability for obtaining class i as calculated by (3) for the binary classifier comparing class i against class j. For the K-class case we have to look for K p_i's ($i = 1,\ldots,K$) which satisfy

$$r_{ij} = \frac{p_i}{p_i + p_j} \text{ and } \sum_{i=1}^{K} p_i = 1, \tag{6}$$

This set of equations, to be solved for p_i, has $K-1$ free parameters and $\frac{K(K-1)}{2}$ constraints, so it is generally impossible to find $\hat{p}_i$'s that will meet all equations. In [7], the authors opt to find the best approximation $\hat{r}_{ij} = \frac{\hat{p}_i}{\hat{p}_i + \hat{p}_j}$ by minimizing the Kullback-Leibler distance between r_{ij} and $\hat{r}_{ij}$

$$l(\vec{p}) = -\sum_{i\neq j} n_{ij}\left[r_{ij}\log\frac{r_{ij}}{\hat{r}_{ij}} + (1-r_{ij})\log\frac{1-r_{ij}}{1-\hat{r}_{ij}}\right] \tag{7}$$

where n_{ij} is the sum of the number of training points in class i and j. They also suggest an iterative scheme to minimize this distance:

start with and initial guess for the $\hat{p}_i$, and calculating $\hat{r}_{ij}$
repeat until convergence
  loop over $i = 1,\ldots,K$
    $\hat{p}_i \leftarrow \hat{p}_i \frac{\sum_{j\neq i} n_{ij} r_{ij}}{\sum_{j\neq i} n_{ij}\hat{r}_{ij}}$
    normalize p_i, and calculate $\hat{r}_{ij}$
$\hat{\vec{p}} \leftarrow \frac{\hat{\vec{p}}}{\sum \hat{p}_i}$

For this algorithm Hastie and Tibshirani proved that the distance between r_{ij} and $\hat{r}_{ij}$ decreases at each step, and since the distance is bound above zero, the procedure converges. This procedure is repeated for all points in the test set. Now, classification is obtain by selecting the class with maximum posterior probability.

IV. Spatial Classification Smoothing

Up to now we have considered the pixels as spatially independent. No contextual information has been used to make

a decision on the class label. Building a classification image based only on the spectral information often results in poorer classification performance [8] and in class images with a noisy appearance, containing many single pixel classes. We propose a simple post-processing technique for spatial classification smoothing, requiring little extra computational effort.

As shown in in section III-B.4, we can calculate the posterior probability for a pixel. We call $p_i(k,l)$, the posterior probability for class i calculated for the pixel at location (k,l) in the image. Normally, to assign a label to the pixel, the label of the class with maximum posterior probability is taken. Define $c(k,l)$ as the class with the maximum posterior probability at location (k,l):

$$c(k,l) = \max_{\arg i} p_i(k,l). \tag{8}$$

One can assume neighboring pixels to have similar posterior probabilities. This information can be used as prior knowledge for defining a new prior probability for a pixel, based on the posterior probability from classification in the neighborhood of the pixel. Define this new prior probability of a pixel as the average over the posterior probabilities of neighborhood Ω

$$p_i^{\text{prior}}(k,l) = \frac{1}{N} \sum_{(a,b)\in\Omega} p_i(a,b) \tag{9}$$

where N is the number of points in Ω. When looking at $p_i(k,l)$ as an image, the new prior $p_i^{\text{prior}}(k,l)$ is in fact a smoothed version of this image. A new posterior probability is obtained by using Bayes' rule:

$$p_i^{\text{post}}(k,l) = \frac{p_i^{\text{prior}}(k,l)p_i(k,l)}{\sum_j p_j^{\text{prior}}(k,l)p_j(k,l)} \tag{10}$$

Classifying using these p_i^{post} will result in smoother classification image maps containing less single pixel classes.

V. Experiments and Discussion

A. Data

A test area at the west coast of Belgium has been selected for which data cubes are obtained from the Compact Airborne Spectrographic Imager (CASI-2) sensor (see Fig. 1). The data has been acquired in October 2002 with 48 spectral bands and a spatial resolution of 1.3 meters. Ground truth is available through field work in 148 regions. Using a differential GPS in the field, the ground truth is mapped on the geocorrected image, obtaining over 2000 pixels to train and validate the presented classification procedure.

The vegetation classes to be discriminated are listed in table I. Some of the classes consist of different vegetation types (4) and even combinations of other classes (11 combines 8 and 10). They correspond to patches that occur as homogeneous mixtures and cannot be distinguished by the available sensor resolution. Another observation is that the number of samples (second column) is diverse. They correspond to the availability of the species in the field. However, they are not representative for the occurrence at the entire coastline and thus can not be used as prior probabilities as such.

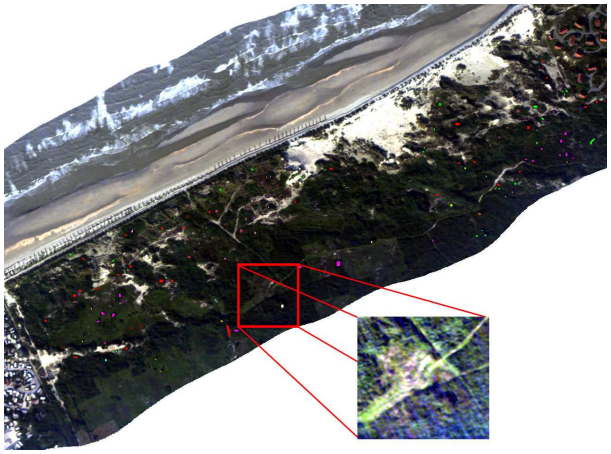

Fig. 1. Image showing part of coastline. Extracted area was used for demonstrating the spatial classification in fig. 2

B. Classification

We will now classify the image for all classes defined in Table I. The regions of interests are randomly split in equally sized sets for training and testing. In the first column, classification results are shown using the multiclass approach. The remaining columns correspond to the 4 different combinations of binary classifiers.

One can observe that all binary classifiers outperform the multiclass approach, which justifies our motivation of choosing for a less complex classifier.

Obviously, all classifiers have difficulties classifying the mixed class "Creeping Willow / Dewberry". Codewords perform best for the large Sea Buckthorn class. However, combined with Wood small-reed, codewords perform worst. The combined "Sea Buckthorn / Wood small-reed" class was observed to be misclassified as Sea Buckthorn. This reveals one of the flaws of this method. In general, codewords are expected to have large Hamming distances between classes. However, a single badly oriented codeword in a large class can jeopardize the entire classifier. A single misjudgment in assigning the labels during fieldwork can be disastrous for the codeword approach but will be of little harm for the other methods.

In Fig. 2, we show part of the color coded classification result. The left image shows the result of the standard maximum posterior classification. The right image shows the results after including the extra prior probability step with $r = 3$. One can immediately see that many single pixel classes and other small structures have vanished. This smoothing property is of importance when interpreting the classification image, when the user is not interested in finely detailed class information.

Acknowledgment

This research is financed by the Belgian Science Policy and by the Flemish Government - Coastal Waterways Division. The authors gratefully thank Dr. Carine Petit, Dr. Joost Vandenabeele, Ir. Peter De Wolf and Ir. Toon Verwaest for their support.

TABLE I

CLASSIFICATION RESULTS

Class	N Samples	Multiclass	Max. Voting	Hier. tree	Codewords	Coupling p
European Alder	26	85	62	49	77	56
European Beachgrass	185	80	94	91	79	90
Silver Berch	40	79	87	79	89	88
Sea Buckthorn / Wood small-reed	191	67	79	74	46	77
Sea Buckthorn	705	54	69	70	81	65
European Privet	381	72	91	89	87	90
Maigold	44	90	90	78	72	91
Dewberry	43	64	59	58	78	67
Grey Willow	42	63	51	43	51	62
Creeping Willow	219	62	77	77	69	78
Creeping Willow / Dewberry	46	36	27	28	34	35
Blue Elderberry	89	75	64	69	58	69
Wall Moss	87	79	76	78	79	83
Total average (weighted)	2098	59	76	75	75	75

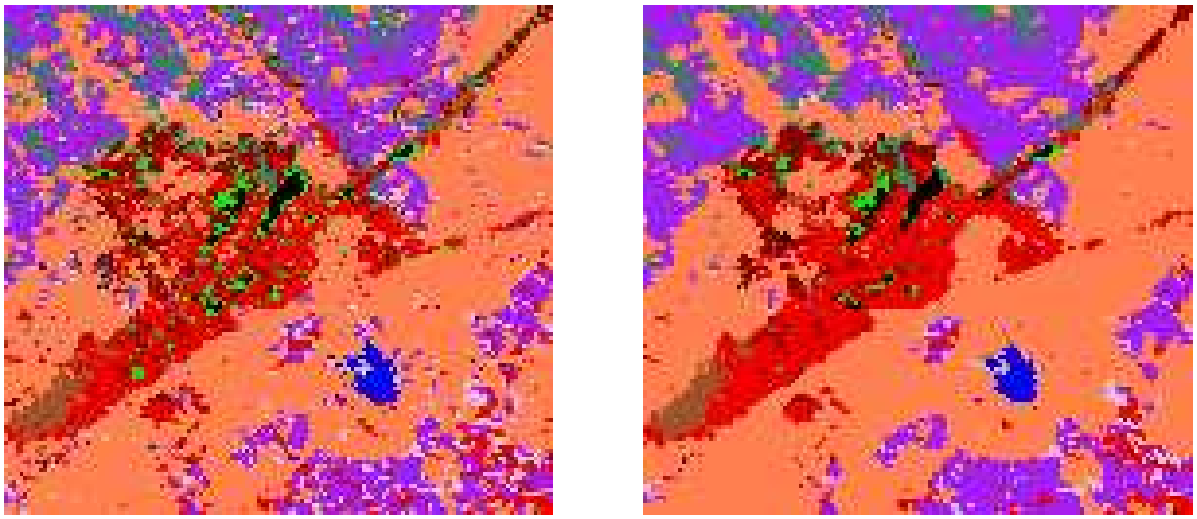

Fig. 2. Classification images from fig. 1. The left image shows the result of maximum posterior classification obtain by coupling probabilities. The right image shows the improvement using the proposed spatial classification smoothing.

REFERENCES

[1] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd Edition*. Wiley, 2001.

[2] S. Raudys and R. P. W. Duin, "Expected classification error of the fisher linear classifier with pseudo-inverse covariance matrix," *Pattern Recognition Letters*, vol. 19, no. 5-6, pp. 385–392, 1998.

[3] S. Knerr, L. Personnaz, and G. Dreyfus, "Single layer learning revisited: a stepwise procedure for building and training a neural network," in *Neurocomputing: Algorithms, Architectures and Applications*, ser. NATO ASI, F. Fogelman-Soulié and J. Hérault, Eds. Springer, 1990, vol. F68, pp. 41–50.

[4] J. Friedman, "Another approach to polychotomous classification," Department of Statistics, Stanford University, Tech. Rep., 1996.

[5] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 1995.

[6] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *Journal of Machine Learning Research*, vol. 1, pp. 113–141, 2000.

[7] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," in *Proceedings of the 1997 conference on Advances in neural information processing systems*, M. I. Jordan, M. J. Kearns, and S. A. Solla, Eds., vol. 10. The MIT Press, 1998, pp. 507–513.

[8] E. Mohn, N. L. Hjort, and G. O. Storvik, "A simulation study of some contextual classification methods for remotely sensed data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 25, no. 6, pp. 796–804, 1987.