A new methodology to improve contextual classifications.*

F.J. CORTIJO and N. PEREZ DE LA BLANCA

Depto. Ciencias de la Computación e I. A. (DECSAI)
E.T.S. Ingeniería Informática
Universidad de Granada, 18071 Granada, Spain

Corresponding author: Francisco J. Cortijo
Depto. Ciencias de la Computación e I. A.
E.T.S. Ingeniería Informática
Universidad de Granada, 18071 Granada, Spain
e-mail: cb@robinson.ugr.es
phone: +58 244078
fax: +58 243317

Abstract

This paper shows some combinations of classifiers that achieve high accuracy classifications. Traditionally it is used the maximum likelihood classification as the initial classification for the contextual correction. We will show that using different non-parametric spectral classifiers to obtain the initial classification we can improve the accuracy of the classification significatively with a reasonable computational cost. More specifically we propose to apply the contextual correction performed by the ICM algorithm to some particular non-parametric spectral classifications.

*This work has been supported by the Spanish “Dirección General de Ciencia y Tecnología” (DGCYT) under grant PB-92-0925-C02-01
1 Introduction

Supervised classifiers assume the existence of a training set $\mathcal{T}$ composed by $n$ labeled training samples, where the labels represent informational classes (labels). This information is used for learning -construction of the classifier- and usually for testing too. We will note by $\Omega = \{\omega_1, \omega_2, \ldots, \omega_J\}$ to the set of informational classes and by $X$ to the samples used for learning and classifying. We assume they are $d$-dimensional random variables.

The main objective of a classification process is to obtain a thematic map. Spectral classifiers use only the spectral information of the pixel to be classified. It is well known, however, that in real images there is spatial correlation between neighbor pixels. It means that spatially neighboring pixels are likely to belong to the same class. If it is only considered the spectral information of the pixel to classify then many pixels will be misclassified and the resulting thematic map will the overall impression of a “noisy” classification. With this term we mean a classification in which there is a high variability in the spatial distribution of labels, that is, we can not find extended patches of labels and many isolated labeled pixels are spread along the map and located inside patches of another classes. This effect is more evident when there is overlapping among the training sets in the spectral space (Cortijo et al., 1995). In this case it is necessary a post-processing over the initial classification because it is expected to find homogeneous regions in the map as they can be found in the Nature. The straightforward solution consists in incorporating additional information into the classifier related to the spatial neighborhood -its context- of the pixel to classify. That information may be the spectral values of the spatially neighboring pixels, their labels or both kinds of information combined in some way. When this kind of information is used for classification it is expected to improve the accuracy of spectral classifications and the classifier is known as a contextual classifier.

From a general point of view a contextual classifier can be seen as a smoothing process over an initial image of labels which is usually a spectral classification. It is well known that some contextual classifiers achieve a local optimum (Besag, 1986) determined by the initial classification so contextual classification should use a high accuracy spectral classification as initial classification. It is used traditionally the maximum likelihood (ML) classification as the starting point for the smoothing process. In this sense, Besag explicitly adopts the conventional ML classification as staring point for the ICM
correction (Besag, 1986) and this is the usual strategy in the rest of contextual classifiers (Sæbø et al., 1985). We have shown (Cortijo & Pérez de la Blanca, 1996a) that the ML classifier is not the best choice when the training sets are high-overlapping. In this work we propose the use of different spectral classifications as initial classifications to a contextual classifier in order to obtain some interesting combinations of spectral-contextual classifiers for Remote Sensing image classification with an acceptable trade-off between the accuracy of the final classification and the required computational effort.

This paper is organized as follows. In section 2 we discuss about a wide number of spectral classifiers families which are of interest in Remote Sensing classification and in section 3 we discuss about some high-performance contextual classifiers. The proposed methodology is tested in section 4 where we describe the datasets used in this paper, and the experimental results obtained. Finally, the main conclusions we have achieved are summarized in section 5.

2 Spectral classifiers

Spectral classifiers are partitioned in two main categories: a) parametric classifiers, if they assume the existence of an underlying probability distribution of the data and b) non-parametric classifiers, if they do not assume anything about the probability distribution. The structure of the Bayes classifier is determined, basically, by the probability density functions (pdf’s) \( p(X \mid \omega_i) \).

The objective in the construction of a supervised parametric classification rule is to characterize the pattern of each class in terms of its pdf which form is assumed to be known in advance so it is only needed to estimate some parameters. The pdf’s are usually multivariate Gaussian functions so it is only needed to estimate the mean vector, \( \mu_i \), and the covariance matrix, \( \Sigma_i \), for \( i = 1, 2, \ldots, J \). In most classification applications, however, the assumption that the forms of the underlying density functions is suspect. The common parametric forms rarely fit the densities actually encountered in practice, for instance, the parametric models assume unimodal densities whereas many practical problems present multimodal densities. The only available information is the training set and the classification rules must be built just from it with no additional assumptions.
2.1 Parametric classifiers

A general framework for parametric classifiers is provided by the regularized discriminant analysis (RDA) proposed by Friedman (Friedman, 1989). RDA allows a wide family of parametric classifiers including the quadratic maximum likelihood classifier (ML) and the linear classifiers as particular cases. The estimation of the covariance matrices is performed by a regularization process determined by two parameters, the values of which are customized to individual situations by jointly minimizing a sample-based estimate of future misclassification risk. The joint optimization of these parameters minimize the cross-validation error so this technique is very accurate for problems in which the training set size is low (Friedman, 1989).

When adopting a quadratic or a linear classifier we are imposing an extreme degree of adjustment of the decision boundaries to the training samples. If the quadratic classifier is adopted several problems arise related to a) the true statistical structure of the training set -the pdf’s may not be Gaussian- (Lachenbruch, 1979) and b) the Hughes effect (Hughes, 1968). Linear classifiers are less sensitive to these problems but they impose a high degree of regularization. It is likely to allow a wider set of possible decision boundaries than the election between quadratic or linear classifier which might be estimated from the data at hand. This is what RDA provides. RDA has been shown useful for classifying both low-dimensional (Cortijo & Pérez de la Blanca, 1996a) and high-dimensional images (Cortijo, 1995), but its computational effort may be sometimes prohibitive for high-dimensional data. To circumvent this problem Friedman proposes a computationally fast method to estimate the optimum pair (Friedman, 1989).

2.2 Non-parametric classifiers

We can find a wide variety of spectral non-parametric classifiers. They can be summarized in three main categories. The first approximation consists in non-parametric statistical techniques to estimate \( p(X|\omega_i) \) (nearest neighbor estimation techniques (Duda & Hart, 1973) and kernel estimation techniques (e.g. (Parzen, 1962) and (Devijver & Kittler, 1982))). The second consists in estimate directly the a posterior probability \( p(\omega_i|X) \) (nearest neighbor classification rules (Devijver & Kittler, 1982)) and the third consists in splitting recursively the representation space by means of binaries questions related
to the values of the variables involved (classification trees (Breiman et al., 1984)).

The most interesting approaches are, from the computational point of view, the classification trees and the nearest neighbors classifiers. In (Cortijo & Pérez de la Blanca, 1996a) we show in detail the application of these techniques to multispectral image classification.

As classification tree technique we have used CART (Breiman et al., 1984) due to its wide use and knowledge. Learning consists, basically, in splitting hierarchically the representation space using the samples in $\mathcal{T}$. This splitting strategy yields to a binary tree in which the terminal nodes are labeled with labels taken from $\Omega$ and the remainder nodes have associated a binary question related to any the values of the variables $X^i$, for $i = 1, 2, \ldots, J$. This classification tree is used in the classification of a new and independent sample, $X$. It is performed by answering to the questions associated to the non-terminal nodes, from the root node until a labeled node is achieved. Then that label is assigned to $X$.

One of the most popular and widely used non-parametric classification rule is the $k$ nearest neighbor rule ($k$-NN) which assigns to $X$ the most populated class in the $k$ nearest spectral neighbors. If $k = 1$ the $k$-NN rule is known as the nearest-neighbor-rule or 1-NN rule.

The requirement of a large training set to assure the convergence of the $k$-NN (Devijver & Kittler, 1982) is the main drawback of the nearest neighbor rules in practical problems. There are, however, two additional drawbacks in the application of these rules: firstly, they are very influenced by incorrectly labeled training samples ("noisy" samples or outliers) and secondly, the computational complexity associated to the search of the nearest neighbor(s) in $\mathcal{T}$.

Many different solutions have been proposed to circumvent these problems (e.g. (Vidal, 1986),(Short & Fukunaga, 1981)) but the most popular approximations are based in to obtain a reduced and representative reference set, $\mathcal{R}$, from $\mathcal{T}$ with the final objective of searching the nearest neighbor(s) in $\mathcal{R}$ with an acceptable trade-off between the accuracy of the classification and the required computational effort. This can be done in two ways: a) by editing-condensing techniques; in this case $\mathcal{R} \subseteq \mathcal{T}$ or b) by adaptive learning techniques; in this case there is not an explicit relation between both sets.

The aim of editing-condensing techniques is two-fold: improving the ac-
curacy of the classification by removing samples located in overlapping acceptance surfaces (editing techniques) and decreasing the computational effort required to find the nearest neighbor(s) (condensing techniques). The joint application of these techniques improve the trade-off between the accuracy of the 1-NN classification and the computational effort required for that classification (Ferri & Vidal, 1992).

The most widely used editing algorithm is the multiedit algorithm (Devijver & Kittler, 1982). It gives as output homogeneous clusters of samples discarding samples located inside or close to clusters of different classes. Each cluster is spanning the Bayes acceptance region corresponding to its class. As it was shown in (Cortijo & Perez de la Blanca, 1996a) it discards numerous samples when the training sets are high-overlapping. We note by \( \mathcal{T}_M \) to the multiedited training set. Now it is possible to classify a new and independent sample using 1-NN with \( \mathcal{R} = \mathcal{T}_M \) as reference set. Let 1-NN \( (M) \) be that classifier. The 1-NN \( (M) \) classifier implements a piecewise-linear decision boundary which is the sample based approximation to the Bayes-optimal decision boundary.

The Hart’s condensing algorithm is the most popular and simple condensing algorithm (Hart, 1968). It requires that the input set be previously edited. This algorithm selects for each class a set of samples that approximate the Bayes-optimal decision boundary discarding those samples which do not contribute to define the decision boundary. If we assume that the we use the multiedit algorithm for editing, we note by \( \mathcal{T}_{MC} \) to the multiedited and condensed training set. \( \mathcal{T}_{MC} \) is not optimal in the sense that it is not the minimal set but the Hart’s condensing algorithm is considered as a good approximation to the minimal set of samples (Devijver & Kittler, 1982). Now it is possible to classify a new and independent sample using 1-NN with \( \mathcal{R} = \mathcal{T}_{MC} \) as reference set. Let 1-NN \( (MC) \) be that classifier.

A different approach consists in adaptive learning techniques. Adaptive learning is a powerful alternative to classical editing-condensing techniques as it allows to fix the reference set size prior to learning (Kohonen, 1992). Adaptive learning algorithms can be tuned by means of a set of parameters in such a way that it is possible to directly supervise the learning process. The training samples are used to tune a fixed number of codebooks or prototypes and the reference set is called the codebooks set or the prototypes set. Adaptive learning is performed in two sequential phases: initialization and learning. The prototypes set is initially a subset of the training set and
the values of the prototypes are updated in a iterative learning process.

The initial values of the prototypes are determined in the initialization phase. When this phase concludes it is verified that \( \mathcal{R} \subseteq \mathcal{T} \) and \( |\mathcal{R}| = p \), where \( p \) is the specified number of prototypes. \( p \) is the only involved parameter in this phase. Let \( p_i \) be the number of prototypes associated to class \( i \), then \( \sum_{i=1}^{d} p_i = p = |\mathcal{R}| \). In the learning phase the values of the prototypes are updated according to the training samples used for learning. Two different adaptative learning approaches can be adopted: the first is based in the use of learning vector quantization (LVQ) methods proposed by Kohonen (Kohonen, 1995) and the second is the decision surface mapping (DSM) algorithm proposed by Geva and Sitte (Geva & Sitte, 1991).

Basically, in LVQ learning the location of the prototypes in the representation space approximate the underlying probability densities using a fixed number of prototypes. Kohonen proposes (Kohonen, 1995) some updating strategies called LVQ-1, LVQ-2 and LVQ-3 but it has been pointed (e.g. (Kohonen et al., 1992) and (Ferri & Vidal, 1992)) that the results obtained by the different LVQ learning strategies are similar and it is difficult to predict which will be the best for a given problem. Considering that the LVQ-1 is the simplest learning strategy in terms of the number of parameters involved we have used LVQ-1 learning. LVQ-1 acts only on the closest prototype to the training sample in each learning step by a punishment-reward criterion (Kohonen, 1995). This updating strategy tends to reduce the density of the prototypes around the decision boundaries, in other words, it tends to approximate the underlying probability distributions of the classes. In the practice this method is very sensitive to the values of the parameters involved so they must be specified very accurately. In that sense we have proposed in (Cortijo & Pérez de la Blanca, 1996b) two algorithms to estimate the parameters involved in LVQ-1.

The aim of DSM (Geva & Sitte, 1991) is placing the prototypes to approximate the decision boundaries in the representation space. In this case it is required that the training set be previously edited. DSM updating acts on pairs of prototypes located in both sides of the boundaries only if there is a misclassification. The probability of misclassification is high in locations close to the decision boundaries.

In both LVQ-1 and DSM it is used the 1-NN classifier using the codebooks set as the reference set. We note by \( \mathcal{T}_{DSM} \) the DSM-learned training set and by \( \mathcal{T}_{LVQ-1} \) to the LVQ-1-learned training set. With this notation we say
that 1-NN (DSM) is the 1-NN classifier that uses \( T_{DSM} \) as the reference set \( (R = T_{DSM}) \) and 1-NN (LVQ-1) is the 1-NN classifier that uses \( T_{LVQ-1} \) as the reference set \( (R = T_{LVQ-1}) \).

3 Contextual classifiers

Contextual classifiers are likely to be classified in two main categories according to the theoretical assumption they are based: a) based in smoothing techniques and b) based in the assumption of the existence of a Markov random field on the map.

Classifiers based in smoothing techniques can also be categorized into pre-smoothing classifiers and post-smoothing classifiers. Pre-smoothing classifiers incorporate contextual information before classification. This is done usually by increasing the dimensionality of data with additional bands in which contextual information is present in some way (McLachlan, 1992). Then, classification is performed with any spectral classifier. Post-smoothing classifiers act on previously classified images. They are usually more or less sophisticated smoothing filters (Townsend, 1986).

Stochastic models and random fields (RF) in particular represent accurately the information a priori on the map in such a way that the Bayes decision theory can be applied. A random field is a joint probability distribution imposed on a set of \( M \) random variables that imposes statistical dependence in a spatially meaningful way. Let \( L = \{ L_1, \ldots, L_M \} \) be the set of \( M \) random variables. It is verified that \( L_i \in \Omega \) in image classification. The spatial dependence can be specified by a global model such as the Gibbs random field (GRF). A GRF describes the global properties of an image in terms of the joint distribution of labels for all pixels (Dubes & Jain, 1989). A Markov random field (MRF) is defined in terms of local properties. It is needed to fix a neighborhood system in which the spatial dependence is relevant. Two neighborhood systems are mainly used, the first order neighborhood which includes the four-nearest-spatial-neighbors, and the second order neighborhood which includes the eight-nearest-spatial-neighbors.

Among different MRF models (Dubes & Jain, 1989) the most accurate for contextual classification are the discrete MRF’s. A RF, with respect to a neighborhood system, is a discrete MRF if its probability function satisfies (Dubes & Jain, 1989):
1. **Positivity.** $P(L = l) > 0 \ \forall l$. Any label’s configuration is possible to occur.

2. **Markov property.** $P(L_i = l_i \mid L_{S_i} = l_{S_i}) = P(L_i = l_i \mid L_{\partial i} = l_{\partial i})$

3. **Homogeneity.** $P(L_i = l_i \mid L_{\partial i} = l_{\partial i})$ is the same for all sites $i$.

Property 2 establishes a local model. The detailed formulation of many local models can be found in the literature (e.g. (Besag, 1986) and (Derin & Elliot, 1987)). Basically they are based in a count within the neighborhood system for each label to be considered pondered by some directional parameters that can be reduced to only one in the simplest case (Besag, 1974).

Given a set of observations, $X = x$, and the contextual information modeled as a MRF, $P(L = l)$, in a Bayesian context the objective is find the estimator $\hat{l}$ which maximizes the a posteriori probability of $L = \hat{l}$, given $X = x$.

$$P(L = \hat{l} \mid X = x) = \frac{P(X = x \mid L = \hat{l}) \ P(\hat{l})}{P(X = x)} \quad (1)$$

This is known as the MAP (maximum a posteriori) method. The model relating observation $x$ to labeling $l$ is chosen to ensure that the posterior distribution of $L$, given $X = x$, is also a MRF. If we require conditional independence of the observed random variables, given the true labels, it is enough to ensure that the posterior distribution is also a MRF. Thus we assume that

$$P(X = x \mid L = l) = \prod_{i=1}^{M} P(X_i = x_i \mid L_i = l_i) \quad (2)$$

If both $P(X = x \mid L = l)$ and $P(L = l)$ are known we can compute $L$ which maximizes the MAP by applying equation 1. In the practice it is clear that even if $M$ and $J$ are low it is not possible to calculate directly the MAP as given in equation 1 (Dubes & Jain, 1989). To circumvent this problem some estimates have been proposed in the literature to estimate the MAP. Geman & Geman have proposed the *simulated annealing algorithm* (Geman & Geman, 1984) to find MAP estimates for all pixels simultaneously. Its computational demands are excessive but it seeks a global maximum (Dubes & Jain, 1989). Another computationally feasible approximations to the MAP estimate are the *ICM algorithm* (*iterated conditional modes*) and the *MPM*
algorithm (maximizer of posterior marginals). A detailed discussion on these methods can be found in (Dubes & Jain, 1989) and references therein. We will center our interest in section 3.1 in the ICM algorithm (Besag, 1986) which has been demonstrate to have an excellent trade-off between the accuracy of the contextual correction and the required computational effort (Cortijo, 1995).

Another approximation to contextual correction using a MRF consists in point-to-point contextual correction methods. They are based in complex conditioned-probability models which are extensions of the MAP expression given in equation 1 by adding an additional term, the contextual correction factor, into the denominator of the MAP expression (Sæbø et al., 1985) and will be discussed in section 3.2.

3.1 The ICM algorithm

The ICM algorithm (iterated conditional modes) is a computationally feasible approximation to the MAP estimate (Besag, 1986) retaining the MRF as a model of prior information and avoiding the tendency of a MRF to degenerate in a plain -single color- image (Dubes & Jain, 1989). This algorithm is based on two basic assumptions: 1) Random variables $X_i$ are conditionally independent with known and equal probability density, $P(X_i = x_i \mid L_i = l_i)$, depending only on $L_i$. 2) The true values of the labels in the map is a realization of a MRF with probability distribution $P(L = l)$.

In the following we will note by $S \setminus i$ to the set of all $M$ sites excluding site $i$ and by $\mathcal{P}i$ to all sites in the neighborhood of site $i$ (Dubes & Jain, 1989). The key to the ICM method is the following proportionality equation:

$$P(L_i = l_i \mid X = x, L_{\mathcal{P}i} = l_{\mathcal{P}i}) \propto P(X_i = x_i \mid L_i = l_i) \cdot P(L_i = l_i \mid L_{\mathcal{P}i} = l_{\mathcal{P}i})$$

(3)

which relates the probability of the label at pixel $i$ given the observed image $x$ and the current estimates of the labels of pixels in the neighborhood of pixel $i$. The ICM algorithm can be written as follows (Dubes & Jain, 1989):
**ICM algorithm**

1. Choose a MRF for the true labels \( L \).
2. Init \( \hat{l} \) by choosing the class which maximize
    \[
    P(X_i = x_i \mid L_i = l_i)
    \]
    for each pixel \( i \),
3. For \( i = 1, 2, \ldots, M \), update \( \hat{l}_i \) by the value of \( l_i \) which maximize
    \[
    P(X_i = x_i \mid L_i = l_i) \cdot P(L_i = l_i \mid L_{\hat{q}_i} = \hat{l}_{\hat{q}_i})
    \]
4. Repeat step 3 \( N_{\text{iter}} \) times.

The critical election is made at step 1 when the prior model is selected (e.g. (Besag, 1986) and (Derin & Elliot, 1987)). Besag suggests (Besag, 1974) a second order neighborhood system and a pairwise-interaction model in which there is only one parameter involved, \( \beta \). If \( \beta = 0 \) then ICM gives the ML classifier. The larger \( \beta \), the greater is the influence of neighboring pixels and the lesser is the detail as a consequence. Besag suggests (Besag, 1986) that \( \beta \) may vary with the iteration. In this case \( \beta \) should increase by equal increments.

The ICM algorithm requires an initial solution (step 2). Besag explicitly adopts the conventional ML classification as staring point for the ICM correction (Besag, 1986). It could be used, however, any map as initial solution. ICM converges to a local maximum so the adopted initial classification is very important for the final result. We will show that when starting ICM with a high accuracy spectral classification then we will obtain a final high accuracy contextual classification.

The model for \( P(X_i = x_i \mid L_i = l_i) \) we assume is a multivariate Gaussian function. The formulation of \( P(L_i = l_i \mid L_{\hat{q}_i} = \hat{l}_{\hat{q}_i}) \) depends on the adopted model. If we assume the Besag’s model, it can be written as

\[
P(L_i = l_i \mid L_{\hat{q}_i} = \hat{l}_{\hat{q}_i}) \propto \exp \{ \beta \times \# \text{ neighbors with label } l_i \} \quad (4)
\]

Finally, experience has demonstrate that 5 or 6 raster scans of an image are sufficient to assure the convergence (Besag, 1986).
3.2 Point-to-point contextual correction methods

The main objective of point-to-point contextual correction methods is to implement the spatial correlation by using conditional probabilities in an extension of the Bayes rule. This extension is implicitly formulated by the inclusion of the called contextual correction factor (Sæbø et al., 1985). This factor is composed by two functions that model the label’s distribution and the spectral responses’ distribution in a neighborhood of the pixel to classify.

Let us introduce the notation used to formulate this extension. If we consider a neighborhood of pixel \( i \) consisting in \( n \) neighbors let \( V_i = \{ i, i_1, i_2, \ldots, i_n \} \) be the set of indexes which references that neighborhood. With this notation let \( D_i = \{ X_i, X_{i_1}, X_{i_2}, \ldots, X_{i_n} \} \) be the spectral values at neighborhood \( V_i \) and let \( C_i = \{ c_i, c_{i_1}, c_{i_2}, \ldots, c_{i_n} \} \) be the labels at neighborhood \( V_i \). If we note \( V_i^* = V_i - \{ i \} \), then \( D_i^* = \{ X_{i_1}, X_{i_2}, \ldots, X_{i_n} \} \) and \( C_i^* = \{ c_{i_1}, c_{i_2}, \ldots, c_{i_n} \} \).

With this notation the a posteriori probability of a pixel \( i \) to be of class \( k \) considering the spectral values in a neighborhood of \( i \) will be noted by \( p_i (k|D_i) \) and can be written (Kittler & Föglein, 1984) as

\[
p_i (k|D_i) = \frac{p_i (D_i|k) \pi_k R_k(X_{i_1}, X_{i_2}, \ldots, X_{i_n})}{p_i (D_i)}
\]

(5)

where \( R_k \) is the contextual correction factor for class \( k \). Once formulated the a posteriori probabilities which incorporate the contextual information we can apply the MAP classification rule.

The contextual correction factor for class \( k \), \( R_k(X_{i_1}, X_{i_2}, \ldots, X_{i_n}) \), or simply \( R_k(D_i^*) \) is defined as

\[
R_k(D_i^*) = \sum_{c_{i_1}} \cdots \sum_{c_{i_n}} h(X_{i_1}, \ldots, X_{i_n}|X_i, k, c_{i_1}, \ldots, c_{i_n}) g(c_{i_1}, \ldots, c_{i_n}|k)
\]

(6)

where \( k \) is the label of pixel \( i \) and \( c_{i_j} \) is the label of a neighbor pixel, refered by \( i_j \), for \( j = 1, 2, \ldots, n \). \( R_k \) is composed by two terms which introduce different contextual information. The first term, \( h \), acts on spectral values and the second, \( g \), acts on labels. More detailed,

- \( h(X_{i_1}, \ldots, X_{i_n}|X_i, k, c_{i_1}, \ldots, c_{i_n}) \) is the simultaneous density of the distribution of the spectral values of the neighbors given the spectral value of the center pixel, the labels at \( V_i \) and given that the label of pixel \( i \) is \( k \). This function considers the spatial correlation of the observations. We will note it by \( h(D_i^*|k, C_i^*) \).
\* \( g(c_i, \ldots, c_n|k) \) is the probability of a configuration of labels in a neighborhood \( V_i^* \) given that the label of pixel \( i \) is \( k \). This function controls the possible configurations that can be found in a neighborhood. We will note it by \( g(C_i^*|k) \).

Now it is required to establish a statistic model for \( h \) and \( g \). Different models yield to different classifiers (Sæbø et al., 1985).

Function \( h \) should incorporate the spatial correlation in a neighborhood. It has been usual to assume that the feature vectors for pixels belonging to the same class are independent of each other. This assumption is more or less realistic depending on the spatial resolution of the scene, being less credible in high-resolution images. In these cases it is preferable to adopt a model for \( h \) which incorporates the spatial dependency (e.g. (Hjort & Mohn, 1984), (Khazeine & Crawford, 1990), (Mohn, Hjort & Storvik, 1987) and (Wilson, 1992)). Hjort and Mohn (Hjort & Mohn, 1984) point that for low resolution images it looks reasonable to assume that the feature vectors are conditionally independent. This assumption can be found in several relevant papers on contextual classifiers (e.g. Welch & Salter, 1971), (Swain, Vardeman & Tilton, 1981), (Owen & Switzer, 1982), (Haslett, 1985) and (Haralick & Joo, 1986).

\( R_k \) is a big sum of products, one sum for each neighbor. The summation is done over all possible configurations of labels at \( V_i^* \) thus the number of terms is \( J^n \). This enormous amount of computation can be reduced by two facts: a) considering a reduced neighborhood and by b) introducing a model describing the distribution of labels in a scene as the result of a stochastic process.

If we are attempting to reduce the neighborhood size, it is generally accepted a first-order neighborhood centered at pixel \( i \). Let \( i_N, i_s, i_E \) and \( i_W \) be the pixels located at north, south, east and west from pixel \( i \). Now \( V_i = \{i, i_N, i_S, i_E, i_W\} \). Any configuration of spectral values at this neighborhood will be noted by \( D_i = \{X_i, X_{i_N}, X_{i_S}, X_{i_E}, X_{i_W}\} \) and any configuration of labels will be noted by \( C_i = \{c_i, c_{i_N}, c_{i_S}, c_{i_E}, c_{i_W}\} \). For simplicity \( C_i = \{k, a, b, c, d\} \) or simply \( (k, a, b, c, d) \) in where \( k \) is the central pixel’s label and \( a, b, c \) and \( d \) are the labels of pixels located at north, south, east and west from pixel \( i \). Now the contextual correction factor can be written as

\[
R_k(D_i^*) = \sum_{a,b,c,d} h(X_{i_N}, X_{i_S}, X_{i_E}, X_{i_W}, X_i, k, a, b, c, d) \ g(a, b, c, d|k)
\]  

(7)
It can be seen that equation 7 is a particular case of equation 6 when \( n = 4 \).

Though the computational effort has decreased by adopting a first-order neighborhood it can be sometimes prohibitive, i.e., if \( J = 10 \) we have \( J^n = 10^4 \) terms in the sum given in equation 7. It is needed a second step which consists in introducing a model for function \( g \) describing the distribution of labels in a scene as the result of a stochastic process (Haslett, 1985). Assuming that the feature vectors are conditionally independent, that is, function \( h \) can be written as

\[
h(D_i^* | X_i, k, a, b, c, d) = p(X_i_N | a) \cdot p(X_i_E | b) \cdot p(X_i_S | c) \cdot p(X_i_W | d)
\]  

(8)

and the contextual correction factor for a first-order neighborhood can be written as

\[
R_k(D_i^*) = \sum_{a,b,c,d} p(X_i_N | a) p(X_i_E | b) p(X_i_S | c) p(X_i_W | d) \cdot g(a, b, c, d | k)
\]  

(9)

It is assumed that the pdf's are multivariate Gaussian functions so it only lefts to specify a functional form for \( g \). Two models of contextual classifiers are frequently used (Sæbø et al., 1985) depending on the particular formulation for function \( g \) adopted: a) the Welch & Salter, Haslett’s model, and b) the Owen & Switzer’s model.

### 3.2.1 Model 1. Welch and Salter, Haslett

The simplest model consists in assuming that the configuration of labels in a neighborhood can be considered as the composition of four configurations which have the central pixel in common (Sæbø et al., 1985). This model was proposed by Welch y Salter Welch & Salter, 1971) and Haslett (Haslett, 1985) from slightly different points of departure.

Let \( i \) and \( j \) be two immediate neighbors pixels, that is, they have one side in common. The conditional probability that one of the pixels has class \( l \) given that the other pixel has class \( k \) is denoted by

\[
\pi(l | k) = Pr (c_j = l | c_i = k)
\]  

(10)

where \( \pi(l | k) \) is called the transition probability between classes \( l \) and \( k \).

This notation assume stationarity and isotropy in the occurrences of the classes. Function \( g \) can be written as

\[
g(a, b, c, d | k) = \pi(a | k) \cdot \pi(b | k) \cdot \pi(c | k) \cdot \pi(d | k)
\]  

(11)
This model assumes conditional independence: given the class of the central pixel, the classes of the four neighbors are independent. This assumption however is not always fulfilled in practice but it takes care of some degree of spatial dependency of the classes and can be considered as a reasonable approximation to reality. Moreover it reduces the number of terms in equation 7 from \( J^4 \) to \( 4 \times J \) terms. The final formulation for the a posteriori probabilities can be found in appendix 1.

### 3.2.2 Model 2. Owen and Switzer

More complex models than the Welch & Salter, Haslett’s model consider the joint distribution of the labels in a neighborhood. In this sense, some authors (Swain, Vardeman & Tilton, 1981) propose to consider all possible configurations in a first-order-neighborhood for low resolution images, that is, \( R_k \) consists in \( J^4 \) terms. In the practice, however, many configurations are unusual, in other words, their probability are low. This is the reason why another authors (e.g. (Owen & Switzer, 1982), (Hjort, Mohn & Storvik, 1985), (Khazeine & Crawford, 1990)) propose to consider only three kinds of possible configurations. In this case \( R_k \) consists in only \( 8 \times J - 7 \) terms. We will center in the second proposal as the first is straightforward.

This model was proposed originally by Owen and Switzer (Owen & Switzer, 1982) and slightly generalized by Hjort and Mohn (Hjort & Mohn, 1984). When using a first-order-neighborhood in most cases all pixels will be of the same class, in some cases we will find two different classes and in few cases we will find three or more classes. This model assumes that the probability of finding a configuration with more than two classes is zero and also that only particular patterns are possible. With these assumptions \( R_k \) consists in only \( 8 \times J - 7 \) terms. More precisely, there are only three admissible patterns of two classes. They are called the \( X \), \( L \) and \( T \) patterns (see figure 1).

In any first-order-neighborhood it is possible to find one ‘X’ pattern and four ‘L’ and ‘T’ patterns so we assume that these 9 patterns are the only ones with probability greater than zero. Let \( p \) be the probability of an ‘X’ configuration, let \( q/4 \) be the probability of one of the four ‘L’ configurations, and let \( r/4 \) be the probability of one of the four ‘T’ configurations. Hjort and Mohn (Hjort & Mohn, 1984) obtain that

\[
g(k, k, k, k | k) = p + (q + r) \pi_k,
\]
\[
g(k, k, m, m | k) = \frac{1}{4} q \pi_m, \]
\[
g(k, k, m, k | k) = \frac{1}{4} r \pi_m.
\]

with $m \neq k$. Equation 12 is a generalization of the formulae proposed by Owen and Switzer (Owen & Switzer, 1982). Only these $8 \times J - 7$ terms are positive, reducing significantly the computational requirements for computing the sum given in equation 7. The values for $p, q$ and $r$ are estimated from an initial classification by point sampling procedures (Sæbø et al., 1985). The final formulation for the a posteriori probabilities can be found in appendix 2.

4 Experimental results

As it was pointed in section 1 the aim of this paper is to show that contextual classifiers will obtain better classifications when the given initial classifications are high accuracy spectral classifications. Moreover we try to show some interesting combination of classifiers which give as result high accuracy classifications with a reasonable computational effort. At this point we have presented many spectral classifiers in section 2 and also some contextual classifiers in section 3. In this section we will show the application of the proposed methodology on two real classification problems.

4.1 Data

The data used to test the performance of the classifiers are two LANDSAT images, landscapes from Greenland, Denmark. The first image is a LANDSAT-2 MSS image of the Igaliko region (figure 2) and the second is a LANDSAT-5 TM image of the Ymer Ø region (figure 4). Both images are $512 \times 512$ pixels in size. The training sets have been selected by experts (Conradsen et al., 1987) and their spectral distribution represent different problematics (Cortijo, 1995).

In Igaliko we have five classes to discriminate and the training set size is 42796 samples (see table 1). In figure 3 we show the spectral distribution of the training samples considering bands 2 and 4 (MSS-5 and MSS-7 bands). This is the two-band combination that presents the maximum
Jeffreys-Matusita average distance (Swain, 1978). We can see that there is a high overlapping among the five ideal clusters.

In Ymer Ô we have twenty classes to discriminate and the training set size is 12574 samples (see table 2). In figure 5 we show the spectral distribution of the training samples considering bands 4 and 6 (TM-4 and TM-7 bands). As above this is the two-band combination that presents the maximum Jeffreys-Matusita average distance (Swain, 1978). We can see that there is again a high overlapping among the twenty ideal clusters.

4.2 Spectral and contextual classifications

The accuracy of the classifications is measured by the test sample estimator thus the training set, $T$, is split into $T^l$ (learning set) and $T^t$ (test set) where $T^l \cup T^t = T$, and $T^l \cap T^t = \emptyset$. Samples in $T^l$ are randomly selected from $T$ and verify that $|T^l| = 2/3 |T|$. Classifiers are constructed from data into $T^l$ and tested with data into $T^t$. In tables 3 and 4 we show the learning and test set sizes for each datasets.

The estimates of the parameters involved in the parametric classifiers (ML and RDA) have been estimated from $T^l$. We have assumed a Gaussian distribution for the density probability functions. For RDA learning (Friedman, 1989) we have sampled a grid consisting in 36 pairs $(\lambda, \gamma)$. See (Cortijo & Pérez de la Blanca, 1996a) for more details.

We have used the standard parameters for learning in CART (Breiman et al., 1984), that is, the Gini diversity index and cost-complexity prune using cross-validation error measure with 10 sets.

The 1-NN classifiers used in this work differ in the reference set used to search the nearest neighbor. 1-NN classifier is the simplest one because it uses the training set as the reference set. In order to reduce the training set size we have experimented with the application of editing-condensing and adaptative learning algorithms.

Among editing-condensing algorithms we have used the multedit algorithm (Devijver & Kittler, 1982) for editing and the Hart’s condensing algorithm (Hart, 1968) for condensing. The adaptative learning algorithms adopted in this paper are the LVQ-1 learning (Kohonen, 1995) and the DSM learning (Geva & Sitte, 1991). We have estimate accurate values for the parameters involved in the LVQ-1 learning with two algorithms proposed by the authors (Cortijo & Pérez de la Blanca, 1996b).
Among the variety of contextual classifiers we have adopted: a) the ICM algorithm, b) the Welch & Salter, Haslett’s model and c) the Owen & Switzer’s model. Once the spectral classifications have been performed we use them as initial maps to the three contextual classifiers we have tested. Given that we have tested 8 spectral classifiers we have performed 24 additional classifications for both datasets.

The Welch & Salter, Haslett’s model and the the Owen & Switzer’s model need to estimate the mean vectors and the covariance matrices for each class. The estimates are computed from $\mathcal{T}_i$. The specific parameters to the models proposed by Welch & Salter, Haslett - the transition probabilities- and by Owen & Switzer - the values for $p$, $q$ and $r$ - are estimated from the initial map (Cortijo, 1995).

We use as prior to the ICM algorithm the pairwise-interaction model proposed by Besag (Besag, 1974) and a second order neighborhood. We have imposed that $N_{\text{iter}} = 8$. We must note, however, that the two last iterations did not modify more than 0.2 % of the pixels so 6 iterations are enough to assure the convergence. We allow $\beta$ to vary with the iteration in fixed increments from 0.5 to 2.5. It means that the neighborhood influence is greater in the last iterations than in the initial ones.

### 4.3 Results

In table 5 we show the accuracy of the classifications performed on the Igaliko image and in table 6 we show the accuracy of the classifications performed on the Ymer Ø image. We show in the first column the name of the spectral classifier used to get the initial map, and the accuracy of that classification, in the second column. The remainder columns show the accuracies of the contextual corrections made over the initial map by using the three models adopted in this paper.

From these tables we must note that the accuracy of the spectral classifications can be improved -sometimes drastically- if they are used as input to a contextual classifier independently of the nature of the spectral classifier. This is true for the contextual classifiers tested in this work.
5 Concluding remarks

We can conclude that among the contextual classifiers, ICM gives the best results and we must note that the required computational effort is lower than the others. The computational effort is identical for any initial classification so the global computational effort is determined by the spectral classification cost (Cortijo, 1995).

We must note that in both problems the accuracies got with the combinations: a) CART + ICM and b) 1-NN (LVQ-1) + ICM are very high. The computational effort associated to CART is mainly influenced by the learning step and it is a function of the training set size but we must note that it is a relatively low cost step. LVQ-1 learning is a quick process and as a additional advantage we can select the training set size and the parameters involved (Kohonen, 1995) or they can be automatically estimated by using the estimation algorithm proposed by the authors (Cortijo & Pérez de la Blanca, 1996b). These combinations have also been tested on synthetic high-spectral images (Cortijo, 1995) and the results obtained do extend and confirm the accuracy of these combinations.
Appendix 1. Welch and Salter, Haslett’s model

If we substitute the expression for \( g \) given in equation 11 into the contextual correction factor expression given in equation 9 we obtain (Sæbø et al., 1985) that

\[
R_k(X_{i_N}, X_{i_S}, X_{i_E}, X_{i_W}) = T_k(X_{i_N}) T_k(X_{i_S}) T_k(X_{i_E}) T_k(X_{i_W}) \quad (13)
\]

where

\[
T_k(X) = \sum_{m=1}^{J} \pi(m|k) p(X|m) \quad (14)
\]

Finally, the a posteriori probability for class \( k \), given the neighborhood of pixel \( i \) can be written as

\[
p_i(k|D_i) \propto \pi_k p(X_i|k) T_k(X_{i_N}) T_k(X_{i_S}) T_k(X_{i_E}) T_k(X_{i_W}) \quad (15)
\]

thus the classification rule for pixel \( i \) consists in to assign the class which maximizes equation 15.

The transition probabilities \( \pi(l|k) \) are estimated from an initial classification by point sampling procedures (Sæbø et al., 1985).

Appendix 2. Owen and Switzer’s model

If we substitute the expression for \( g \) given in equation 12 into the contextual correction factor expression given in equation 9 the contextual correction factor can be written (Hjort, 1984) as

\[
R_k(D_i^*) = \{p + (q + r) \pi_k \} p(X_{i_N}|k)p(X_{i_S}|k)p(X_{i_E}|k)p(X_{i_W}|k) + \\
+ \frac{1}{4} q \sum_{m \neq k} \pi_m p(X_{i_N}|k)p(X_{i_S}|k)p(X_{i_E}|m)p(X_{i_W}|m) + \\
+ \ldots \\
+ \frac{1}{4} r \sum_{m \neq k} \pi_m p(X_{i_N}|k)p(X_{i_E}|m)p(X_{i_S}|k)p(X_{i_W}|k) \quad (16)
\]

which can be written as

\[
R_k(D_i^*) = pA_k(D_i^*) + qB_k(D_i^*) + rC_k(D_i^*) \quad (17)
\]
after some manipulation (Hjort, 1984), where

\[
A_k(D_i^* ) = p(X_{i_N} | k) p(X_{i_E} | k) p(X_{i_S} | k) p(X_{i_W} | k),
\]

\[
B_k(D_i^* ) = \frac{1}{4} \left\{ p(X_{i_N} | k) p(X_{i_E} | k) b(X_{i_S}, X_{i_W}) + p(X_{i_E} | k) p(X_{i_S} | k) b(X_{i_N}, X_{i_W}) + p(X_{i_S} | k) p(X_{i_N} | k) b(X_{i_W}, X_{i_S}) + p(X_{i_W} | k) p(X_{i_S} | k) b(X_{i_N}, X_{i_E}) \right\},
\]

\[
C_k(D_i^* ) = \frac{1}{4} \left\{ p(X_{i_N} | k) p(X_{i_E} | k) p(X_{i_S} | k) a(X_{i_W}) + p(X_{i_E} | k) p(X_{i_S} | k) a(X_{i_N}) p(X_{i_W} | k) + p(X_{i_S} | k) a(X_{i_E}) p(X_{i_N} | k) p(X_{i_W} | k) + a(X_{i_N}) p(X_{i_E} | k) p(X_{i_S} | k) p(X_{i_W} | k) \right\},
\]

and where

\[
a(X) = \sum_{m=1}^{J} \pi_m p(X|m)
\]

\[
b(X, Y) = \sum_{m=1}^{J} \pi_m p(X|m) p(Y|m)
\]

Finally, the a posteriori probability for class $k$, given the neighborhood of pixel $i$ can be written as

\[
p_i (k|D_i) \propto \pi_k p(X_i | k) R_k(D_i^*)
\]

where $R_k(D_i^*)$ is given by equation 17. The classification rule for pixel $i$ consists in to assign the class which maximizes equation 20.
Acknowledgments

We must thank to the IMM, Denmak University of Technology (Lyngby, Denmark) for providing the LANDSAT images used in this work.
References


List of Captions

Table 1: Training set information. Igaliko.

Table 2: Training set information. Ymer Ø.

Table 3: Learning and Test set information. Igaliko.

Table 4: Learning and Test set information. Ymer Ø.

Table 5: Accuracy of the classifications. Igaliko.

Table 6: Accuracy of the classifications. Ymer Ø.

Figure 1: Admissible configurations in the first-order-neighborhood, Owen & Switzer model.

Figure 2: LANDSAT-2 MSS image. Igaliko, Greenland. MSS-2 Bands 4 to 7.

Figure 3: The training set for Igaliko. a) Spatial distribution. b) Spectral distribution.

Figure 4: LANDSAT-5 TM image. Ymer Ø, Greenland. TM-5 bands 1 to 5 and 7.

Figure 5: The training set for Ymer Ø. a) Spatial distribution. b) Spectral distribution.

Figure 6: Igaliko classifications. a) CART classification. b) CART classification + ICM.

Figure 7: Igaliko classifications. a) 1-NN (LVQ-I) classification. b) 1-NN (LVQ-I) classification + ICM.

Figure 8: Ymer Ø classifications. a) CART classification. b) CART classification + ICM.
Figure 9: Ymer Ő classifications. a) 1-NN (LVQ-I) classification. b) 1-NN (LVQ-I) classification + ICM.
<table>
<thead>
<tr>
<th>Class</th>
<th>Informational Class</th>
<th>Training Areas</th>
<th>Training Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intrusives</td>
<td>1</td>
<td>5725</td>
</tr>
<tr>
<td>2</td>
<td>Water</td>
<td>1</td>
<td>11372</td>
</tr>
<tr>
<td>3</td>
<td>Barren Granite</td>
<td>1</td>
<td>8231</td>
</tr>
<tr>
<td>4</td>
<td>Plain Granite</td>
<td>8</td>
<td>4191</td>
</tr>
<tr>
<td>5</td>
<td>Dolerite</td>
<td>1</td>
<td>13277</td>
</tr>
<tr>
<td>Sum</td>
<td></td>
<td>12</td>
<td>$</td>
</tr>
</tbody>
</table>

Table 1:
<table>
<thead>
<tr>
<th>Class</th>
<th>Informational Class</th>
<th>Training Areas</th>
<th>Training Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Moraine</td>
<td>3</td>
<td>3698</td>
</tr>
<tr>
<td>2</td>
<td>Delta + young alluvial fans</td>
<td>2</td>
<td>1235</td>
</tr>
<tr>
<td>3</td>
<td>Old alluvial fans</td>
<td>1</td>
<td>6071</td>
</tr>
<tr>
<td>4</td>
<td>Quartzites/shales</td>
<td>1</td>
<td>279</td>
</tr>
<tr>
<td>5</td>
<td>Quartzites</td>
<td>1</td>
<td>714</td>
</tr>
<tr>
<td>6</td>
<td>Shales (black)</td>
<td>1</td>
<td>709</td>
</tr>
<tr>
<td>7</td>
<td>Quartzites (white)</td>
<td>1</td>
<td>255</td>
</tr>
<tr>
<td>8</td>
<td>Shale (red/brown), free of veget.</td>
<td>1</td>
<td>493</td>
</tr>
<tr>
<td>9</td>
<td>Quartzites (red only)</td>
<td>1</td>
<td>73</td>
</tr>
<tr>
<td>10</td>
<td>Shales (red)</td>
<td>2</td>
<td>266</td>
</tr>
<tr>
<td>11</td>
<td>Dolomite (white/yellow)</td>
<td>2</td>
<td>127</td>
</tr>
<tr>
<td>12</td>
<td>Limestone (black)</td>
<td>2</td>
<td>969</td>
</tr>
<tr>
<td>13</td>
<td>Shale (red/yellow), sunlit</td>
<td>1</td>
<td>208</td>
</tr>
<tr>
<td>14</td>
<td>Shale (red/yellow), partly cov. by veg.</td>
<td>1</td>
<td>536</td>
</tr>
<tr>
<td>15</td>
<td>Dolomite (white)</td>
<td>1</td>
<td>323</td>
</tr>
<tr>
<td>16</td>
<td>Quartz. (yellow/brown), sunlit</td>
<td>1</td>
<td>273</td>
</tr>
<tr>
<td>17</td>
<td>Quartz. (yellow/brown), shaded areas</td>
<td>1</td>
<td>393</td>
</tr>
<tr>
<td>18</td>
<td>Limestone (black)</td>
<td>2</td>
<td>673</td>
</tr>
<tr>
<td>19</td>
<td>Limestone (black/grey)</td>
<td>2</td>
<td>389</td>
</tr>
<tr>
<td>20</td>
<td>Glacial deposits</td>
<td>2</td>
<td>354</td>
</tr>
<tr>
<td>Sum</td>
<td></td>
<td>29</td>
<td>$</td>
</tr>
</tbody>
</table>

Table 2:
<table>
<thead>
<tr>
<th>Class</th>
<th>$T^i$</th>
<th>$T^j$</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3806</td>
<td>1919</td>
<td>5725</td>
</tr>
<tr>
<td>2</td>
<td>7542</td>
<td>3830</td>
<td>11372</td>
</tr>
<tr>
<td>3</td>
<td>5463</td>
<td>2768</td>
<td>8231</td>
</tr>
<tr>
<td>4</td>
<td>2796</td>
<td>1395</td>
<td>4191</td>
</tr>
<tr>
<td>5</td>
<td>8834</td>
<td>4443</td>
<td>13277</td>
</tr>
<tr>
<td>Sum</td>
<td>28441</td>
<td>14355</td>
<td>42796</td>
</tr>
</tbody>
</table>

Table 3:
<table>
<thead>
<tr>
<th>Class</th>
<th>$T^l$</th>
<th>$T^r$</th>
<th>Sum</th>
<th>Class</th>
<th>$T^l$</th>
<th>$T^r$</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2464</td>
<td>1234</td>
<td>3698</td>
<td>11</td>
<td>94</td>
<td>33</td>
<td>127</td>
</tr>
<tr>
<td>2</td>
<td>843</td>
<td>392</td>
<td>1235</td>
<td>12</td>
<td>656</td>
<td>313</td>
<td>969</td>
</tr>
<tr>
<td>3</td>
<td>413</td>
<td>194</td>
<td>6071</td>
<td>13</td>
<td>144</td>
<td>64</td>
<td>208</td>
</tr>
<tr>
<td>4</td>
<td>196</td>
<td>83</td>
<td>279</td>
<td>14</td>
<td>369</td>
<td>167</td>
<td>536</td>
</tr>
<tr>
<td>5</td>
<td>480</td>
<td>234</td>
<td>714</td>
<td>15</td>
<td>227</td>
<td>96</td>
<td>323</td>
</tr>
<tr>
<td>6</td>
<td>476</td>
<td>233</td>
<td>709</td>
<td>16</td>
<td>192</td>
<td>81</td>
<td>273</td>
</tr>
<tr>
<td>7</td>
<td>178</td>
<td>77</td>
<td>255</td>
<td>17</td>
<td>274</td>
<td>119</td>
<td>393</td>
</tr>
<tr>
<td>8</td>
<td>344</td>
<td>149</td>
<td>493</td>
<td>18</td>
<td>453</td>
<td>220</td>
<td>673</td>
</tr>
<tr>
<td>9</td>
<td>52</td>
<td>21</td>
<td>73</td>
<td>19</td>
<td>271</td>
<td>118</td>
<td>389</td>
</tr>
<tr>
<td>10</td>
<td>187</td>
<td>79</td>
<td>266</td>
<td>20</td>
<td>247</td>
<td>107</td>
<td>354</td>
</tr>
<tr>
<td>Sum</td>
<td></td>
<td></td>
<td>8560</td>
<td>4014</td>
<td>12574</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4:
<table>
<thead>
<tr>
<th>Spectral Classifier</th>
<th>Init.</th>
<th>ICM</th>
<th>Welch</th>
<th>Owen</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>73.51 %</td>
<td>81.33 %</td>
<td>79.83 %</td>
<td>80.21 %</td>
</tr>
<tr>
<td>RDA</td>
<td>78.97 %</td>
<td>89.37 %</td>
<td>85.46 %</td>
<td>85.68 %</td>
</tr>
<tr>
<td>CART</td>
<td>80.66 %</td>
<td>92.30 %</td>
<td>86.66 %</td>
<td>86.55 %</td>
</tr>
<tr>
<td>1-NN</td>
<td>74.61 %</td>
<td>86.94 %</td>
<td>85.87 %</td>
<td>85.70 %</td>
</tr>
<tr>
<td>1-NN (M)</td>
<td>77.76 %</td>
<td>83.02 %</td>
<td>84.63 %</td>
<td>84.66 %</td>
</tr>
<tr>
<td>1-NN (MC)</td>
<td>77.08 %</td>
<td>82.83 %</td>
<td>84.83 %</td>
<td>84.85 %</td>
</tr>
<tr>
<td>1-NN (DSM)</td>
<td>77.50 %</td>
<td>85.32 %</td>
<td>84.12 %</td>
<td>84.52 %</td>
</tr>
<tr>
<td>1-NN (LVQ-i)</td>
<td>79.07 %</td>
<td>90.80 %</td>
<td>86.42 %</td>
<td>86.44 %</td>
</tr>
</tbody>
</table>

Table 5:
<table>
<thead>
<tr>
<th>Spectral Classifier</th>
<th>Init.</th>
<th>ICM</th>
<th>Welch</th>
<th>Owen</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>61.92 %</td>
<td>91.37 %</td>
<td>85.11 %</td>
<td>85.33 %</td>
</tr>
<tr>
<td>RDA</td>
<td>64.29 %</td>
<td>85.55 %</td>
<td>69.36 %</td>
<td>69.57 %</td>
</tr>
<tr>
<td>CART</td>
<td>62.35 %</td>
<td>95.58 %</td>
<td>86.73 %</td>
<td>87.16 %</td>
</tr>
<tr>
<td>1-NN</td>
<td>78.50 %</td>
<td>97.98 %</td>
<td>86.50 %</td>
<td>87.08 %</td>
</tr>
<tr>
<td>1-NN (M)</td>
<td>65.67 %</td>
<td>90.07 %</td>
<td>82.96 %</td>
<td>83.60 %</td>
</tr>
<tr>
<td>1-NN (MC)</td>
<td>63.23 %</td>
<td>81.09 %</td>
<td>70.12 %</td>
<td>70.35 %</td>
</tr>
<tr>
<td>1-NN (DSM)</td>
<td>64.55 %</td>
<td>80.97 %</td>
<td>72.66 %</td>
<td>73.22 %</td>
</tr>
<tr>
<td>1-NN (LVQ-f)</td>
<td>68.18 %</td>
<td>93.64 %</td>
<td>85.55 %</td>
<td>86.41 %</td>
</tr>
</tbody>
</table>

Table 6:
Figure 1:
Figure 2:
Figure 3:
Figure 4:
Figure 5:
Figure 9: