EMPIRICAL COMPARISON OF MACHINE LEARNING ALGORITHMS FOR IMAGE TEXTURE CLASSIFICATION WITH APPLICATION TO VEGETATION MANAGEMENT IN POWER LINE CORRIDORS

Zhengrong Li [a, b]*, Yuee Liu [a], Ross Hayward [a], Rodney Walker [b]

[a] School of IT, Queensland University of Technology, George Street, Brisbane, QLD 4001 Australia
[b] Australian Research Centre for Aerospace Automation (ARCAA), George Street, Brisbane, QLD 4001 Australia
(zhengrong.li, yuee.liu, r.hayward, ra.walker)@qut.edu.au

Commission VII

KEY WORDS: Classification, Texture Feature, Machine Learning, Object-based Image Analysis, Vegetation

ABSTRACT:

This paper reports on the empirical comparison of seven machine learning algorithms in texture classification with application to vegetation management in power line corridors. Aiming at classifying tree species in power line corridors, object-based method is employed. Individual tree crowns are segmented as the basic classification units and three classic texture features are extracted as the input to the classification algorithms. Several widely used performance metrics are used to evaluate the classification algorithms. The experimental results demonstrate that the classification performance depends on the performance matrix, the characteristics of datasets and the feature used.

1. INTRODUCTION

Vegetation management activities in power line corridors including tree trimming and vegetation control is a significant cost component of the maintenance of electrical infrastructure. Currently, most vegetation management programs for distribution systems are calendar-based ground patrol (Russell et al., 2007). Unfortunately, calendar-based tree trimming cycles are expensive. It also results in some zones being trimmed more frequently than needed and others not cut often enough. Moreover, it is seldom practicable to measure all the plants around power line corridor by field methods. Satellites and aerial vehicles can pass over more regularly and automatically than the ground patrol. Therefore, remotely sensed data have great potential in assisting vegetation management in power line corridors (Li et al., 2008). Remote sensing image classification is one of the key tasks for extracting useful information to assist power line corridor monitoring.

Texture contains important information for image classification, as it represents the content of many real-world images. Texture feature extraction and classification have been intensively studied for interpreting vegetation properties from remote sensing imagery (Franklin et al., 2000, Coburn and Roberts, 2004). Selection of appropriate texture measurements and classification algorithm are two critical steps in a texture classification problem. However, most previous research focused on how to representing texture in an image, few research verified the discriminatory power of different classification algorithms using these texture features. Lu and Weng reviewed a number of image classification techniques for improving classification performance and suggested that the use of multiple features and selection of suitable classification method are especially significant for improving the classification accuracy. However, no empirical comparison and quantitative results have been presented. It would be interesting to investigate which one have more impact on the classification results, the features or the classifiers?

Machine learning techniques are now widely used in remote sensing classification. A machine learning algorithm is one that can learn from experience (observed examples) with respect to some class of tasks and a performance measure (Mitchell, 1997). Different performance metrics are often used and it is possible for one learning method to perform well on one metric, but be suboptimal on other metrics. For example, SVMs are designed to optimize accuracy, whereas neural networks typically optimize squared error or cross entropy (Caruana and Niculescu-Mizil, 2004). Moreover, in many applications Accuracy are used as the only measure to assess the performance of the built classifier. However, there are many other evaluation methods such as Precision/Recall and ROC analysis. We need to understand the advantage and disadvantage of these measures before using them for evaluation. Sometimes we may need to find tradeoffs on these methods and try to select a model that best suit the problem.

The motivation behind this paper is to develop a better understanding of the machine learning process in object-based image classification, to evaluate the performance of different machine learning algorithms in a specific texture classification application, and to compare the results not only in terms of their classification accuracy but also the benefit and cost and some other properties such as computational cost.

2. METHODOLOGY

2.1 An overview of object-based image classification

Since remote sensing images consist of rows and columns of pixels, conventional land-cover mapping has been based on a per-pixel basis (Mas et al., 2006). Unfortunately, classification

* Corresponding author.

algorithms based on single pixel analysis often are not capable of extracting the information we desire from high spatial resolution images. For example, the spectral complexity of urban land-cover materials results in specific limitations using per-pixel analysis for the separation of human-made materials such as roads and roofs and natural materials such as vegetation, soil, and water (R.Jensen, 2005). We need information about the characteristics of a single pixel but those of the surrounding pixels so that we can identify areas (or segments) of pixels that are homogeneous. Object-based approaches become popular in high spatial resolution remote sensing image classification, which has proven to be an alternative to the pixel-based image analysis and a large number of publications suggest that better results can be expected (Blaschke, 2010).

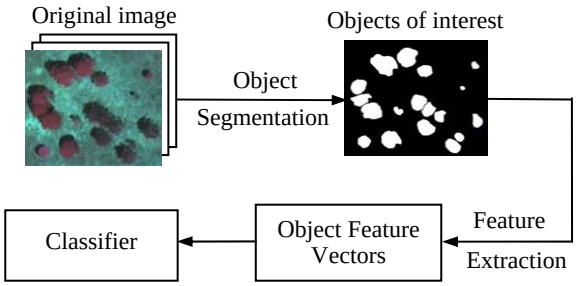


Figure 1 Framework of object-based image classification

A typical object-based image classification consists of a three-stage processing: image segmentation, object feature extraction, and pattern classification. Successful object-based image analysis results largely depend on the performance of image segmentation. Since we are going to classify the species among trees, tree crowns are the only image-objects of interest in our research. The aim of segmentation is, therefore, to detect and delineate all trees from images while eliminating other image regions. We have developed an automatic tree crown detection and delineation algorithm by utilizing spectral features in a pulse coupled neural network followed by post-processing using morphological reconstruction (Li et al., 2009). Although the automatic segmentation is satisfied from visual assessment, decomposition of tree clusters is occasionally poor. Since the main aim of this research is evaluate different machine classifiers, manual segmentation is used to minimize the influence of under-segmentation and over-segmentation. The background is removed and each tree crown is labelled with a unique label to identify the tree which is paired against individual tree species obtained from field surveys.

2.2 Texture Feature Extraction Methods

Texture patterns are defined as the characteristic intensity variations that typically originate from roughness of object surfaces (Davies, 2009). According to a recent review texture feature extraction methods can be divided into three categories: statistical, structural and signal processing based approaches (Xie and Mirmehdi, 2009). In this paper, three widely used texture features are extracted from the segments (polygons) and then input to the classifiers: GLCM, Gabor wavelet features, and Uniform LBP. In this paper, all three texture features are extracted from grey channel which is derived by averaging the four spectral bands of the original image.

GLCM: Grey-level co-occurrence matrices (GLCM) have been successfully used for deriving texture measures from images. This technique uses a spatial co-occurrence matrix that computes the relationships of pixel values and uses these values to compute the second-order statistics (Kubo et al., 2003). In this paper, we use mean and standard deviation of four measures from the grey-level co-occurrence matrices: energy, entropy, contrast, and homogeneity. The GLCM feature vector has 8 dimensions.

Gabor Wavelet Features: 24 Gabor wavelet filters are employed with center frequencies [0.05, 0.4], 4 scaling factors, and 6 orientations at angles of 0 and 180 degrees to achieve optimal coverage in the Fourier domain. The mean and standard deviation of magnitude of each filtered image region are used as feature components. The feature vector has 48 dimensions.

ULBP: Local Binary Pattern (LBP) is first proposed by Ojala et al. to encode the pixel-wise information in the texture images (Ojala et al., 2002). The LBP value for the centre pixel is calculated using the following equation:

$$LBP_{P,R} = \sum_{i=0}^{p-1} u(u_i - u_c) \times 2^i \qquad (1)$$

where P is the total number of neighbouring pixels, R is the radius used to form circularly symmetric set of neighbours. In our experiment, we use the uniform LBP (ULBP) contains at most two bitwise (0 to 1 or 1 to 0) transitions The occurrence histograms of the ULBP are computed using $P = 8, 16, 24$, with $R = 1, 2, 3$ respectively, which is claimed to have the best performance of the local binary patterns in the experiments conducted by Ojala *et al.* (Ojala et al., 2002). The features are obtained by combining the three sets of features together.

2.3 Machine Learning Algorithms

During the past decades, a variety of machine learning algorithms have been proposed for classification tasks. Although the potential advantages and disadvantage of these techniques have been addressed in many published work, most of them are from the theoretical view under some assumption about data distribution, characteristics of the classification task, signal-to-noise-ratio, etc. In reality, these assumptions are often hard to be verified. Therefore, a practical solution for selecting an appropriate model for a given classification task is to experimentally compare these algorithms. In this paper, we compared seven widely used machine classifiers which are implemented in DTREG (Sherrod, 2009): K-Means Clustering, Linear Discriminant Analysis (LDA), Radial Basis Function Networks (RBFN), Multilayer Perceptron Neural Networks (MLPNN), Support Vector Machines (SVM), Single Decision Tree (SDT), and Decision Tree Forest (DTF). Only a brief introduction of these algorithms is presented in this section, and may safely be skipped by readers since they are all well known techniques.

K-Means Clustering (KM): K-Means is a classic unsupervised clustering technique. When used for supervised classification, the model is built by minimizing the classification error (distances between the predicted cluster and the actual cluster membership. In DTREG, the training is done by searching the optimal number of clusters and each category may have several corresponding clusters.

Linear Discriminant Analysis (LDA) The basic idea of Linear Discriminant Analysis (LDA) is to find the linear combination of features (“linear transformation”) which best separate desired classes.

Multilayer Perceptron Neural Networks (MLP): Neural networks are predictive models loosely based on the action of biological neurons. Artificial neural network usually refers to multilayer percetron neural network which is typically full-

connected, three layers, feed forward, perceptron neural network.

Radial Basis Function Networks (RBFN): The basic idea of RBFN is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. A RBFN typically has three layers: an input layer for each predictor variable, a hidden layer that uses Gaussian function as radial activation function and an output layer that implements weighted sum of hidden layer outputs.

Support Vector Machines (SVMs): The basic idea of SVM is to find an optimal decision function (a hyperplane) that separates clusters of vector in such a way that cases with one category of the target variable are on one side of the plane and cases with the other category are on the other size of the plane. The vectors near the hyperplane are the support vectors.

Single Decision Tree (SDT): Decision tree is a binary tree structure whose internal nodes correspond to input patterns and whose leaf nodes are categories of patterns. The tree can be induced by iteratively splitting the dataset into subsets based on classes attributes. The decision tree assigns a pattern category to an input pattern by filtering the pattern from the root to the leaf in the tree.

Decision Tree Forests (DTF): It is also known as Random Forests, which is an ensemble of tree-type classifiers. A decision tree forest grows a number of independent trees in parallel, and they do not interact until after all of them have been built. For classification, each tree in the DTF casts a unit vote for the most popular class at input, while the output of the classifier is determined by a majority vote of the trees.

2.4 Performance Metrics

Given a certain application, more than one method is applicable. This motivates evaluating the performance of these classification methods empirically in a specific application. That is, given several classification algorithms, how can we say one has less error than the others for a given application? Having selected a classification algorithm to train a classifier, can we tell an expected error rate with enough confidence that later on when it is used in a new dataset?

In this section, we consider several most commonly used metrics for evaluating different classification algorithms: overall accuracy, precision/recall, F-measure, ROC analysis, and computational cost. All of these measures are based on the definition of a confusion matrix. An example of confusion matrix for binary classification is described in Table 1. To help the definition that follows, we define the following symbols: TP: True Positive count; FN: False Negative count; FP: False Positive count; TN: True Negative count.

The overall accuracy is the simplest and most intuitive evaluation measure for classifiers. It is defined as

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ samples} = \frac{TP+TN}{P+N}$$

It is worth noting that the overall accuracy does not distinguish between types of errors the classifier makes (i.e. False Positive versus False Negative) (Japkowicz, 2006). For example, two classifiers may obtain the same accuracy but they may behave quite differently on each category. If one classifier obtains 100% accuracy on one category but only 41% on the other category, while another classifier generate 70% for each category, it is hard to claim that the first classifier is better. Therefore, overall accuracy may not be use blindly as the evaluation method for classifiers on a dataset. Precision and Recall can avoid the problem encountered by Accuracy. Precision can be seen as a measure of exactness or fidelity, whereas Recall is a measure of completeness. Their definitions are: $\Pr ecision = TP/(TP+FP)$, $\mathrm{Re}\,call = TP/P$. Usually, Precision and Recall scores are discussed jointly and a single measure can be derived by combing both measures (e.g. F-measure). F-measure is the weighted harmonic mean of precision and recall. In this paper, we use the F_1 measure in which the precision and Recall are evenly weighted. It is defined as:

$$F_1 = 2 \cdot \frac{\Pr ecision \cdot \mathrm{Re}\,call}{\Pr ecision + \mathrm{Re}\,call}$$

The goal of Precision/Recall space is to be in the upper-right-hand corner, which means that the higher value of F_1 measure, the better classifier's performance.

Table 1 A confusion matrix

Predicted Category	Actual Category	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN
	P=TP+FN	N=FP+TN

Precision and Recall do not judge how well a classifier decides that a negative example is, indeed, negative. Receiver Operating Characteristic (ROC) analysis can solve both the problems of Accuracy and Precision/Recall. ROC analysis plots the False Positive Rate (FPR) on the x-axis of a graph and True Positive Rate (TPR) on the y-axis. TPR is equal to Recall and FPR is defined as $FPR = FP/N$. An ROC graph depicts relative trade-offs between true positive (benefits) and false positive (costs), and the goal in ROC space is to be in the upper-left-hand corner (Davis and Goadrich, 2006). The (0,1) point of the ROC space is also called a perfect classification. The diagonal line from the left bottom to the right top corner is also called the random guess line, which can be used to judge the whether it is good or bad classification. Points above the random guess line indicate good classification results, while points below the line are considered as bad classification results. In this paper, we calculate the distance of the each point and the (0,1) point and rank it. The shorter the distance, the better the classification is.

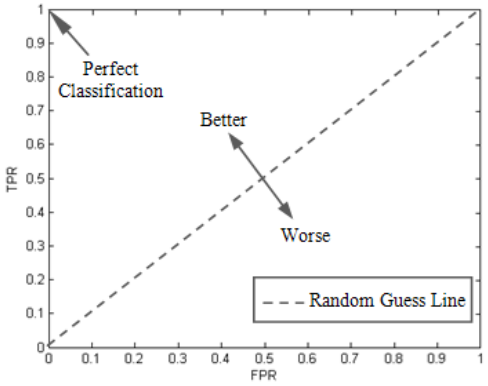


Figure 2 Illustration classifier evaluation in ROC space

Computational costs of the classification algorithms also need to be considered in a real-world problem. Although in most remote sensing image classification tasks real-time processing is not required, it is certainly not unnecessary to choose a computational efficient classification algorithm. In this paper, we compare the computation cost of different machine learning

algorithms by recording the analysis time in both training and testing stages.

3. EXPERIMENT AND DISCUSSION

3.1 Data Collection

The experiment dataset used in this research were collected in rural Queensland Australia in October 2008 for research into vegetation management in power line corridors. The reason why we need species information of individual trees is that vegetation management in power line corridors is based on their potential risks to power lines. Some tree species are of particular interest and are generally categorized into undesirable and desirable species. For example, species with fast growing rates and that also have the potential to reach a mature height of more than four meters are defined as undesirable species. These undesirable species often pose high risks to power lines and therefore should be identified and removed. The images were captured in a 1.8 kilometres corridor by a high resolution 3-CCD digital multi-spectral camera mounted on fixed wing aircraft. Figure 3 shows a mosaic of the test area generated from aerial images acquired from the trial. The spatial resolution of the captured images is about 15cm. The ground truth data of vegetation species were obtained from a field survey with domain experts' participation.

It should be noted that classifying all types of species in power line corridors requires significantly more resources than are currently available, however, classifying species in a given test area as a proof of concept is possible. In this research, we focus on three dominant species in our test field: *Eucalyptus tereticornis*, *Eucalyptus melanophloia*, and *Corymbia tesselaris*. We abbreviate the species names to *Euc-Ter*, *Euc-Mel* and *Cor-Tes*. Through field survey with botanist's participation, 121 trees were selected and labelled for the experiment with 64 Euc-Ter, 30 Euc-Mel and 27 Cor-Tes. The criterion is that tree crowns are big enough so that they can be visually identified from the aerial images. Visual classification of these species often uses features such as leaf shape and bark type which are not available from the data used. However, texture analysis can be very useful to identify these species from digital imagery.

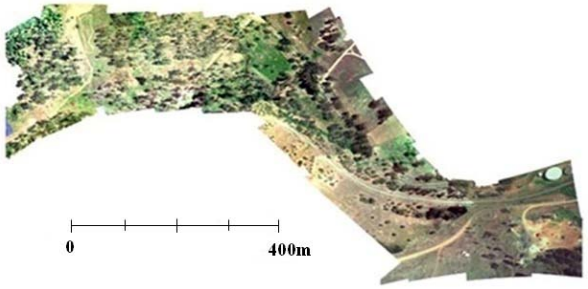


Figure 3 Experiment test site

3.2 Results and Discussion

To evaluate the performance of different machine learning algorithms in texture classification, we use the implementation of these algorithms in DTREG. For all classifiers the default setting of DTREG is used. V-fold cross validation technique is employed in the experiment, and 10 folders were selected for the cross validation. The dataset is partitioned into 10 groups, which is done using stratification methods so that the distributions of categories of the target variable are approximately the same in the partitioned groups. 9 of the 10 partitions are collected into a pseudo-learning dataset and a classification model is built using this pseudo-learning dataset. The rest 10% (1 out of 10 partitions) of the data that was held back and used for testing the built model and the classification error for that data is computed. After that, a different set of 9 partitions is collected for training and the rest 10% is used for testing. This process is repeated 10 times, so that every sample has been used for both training and testing. The classification accuracies of the 10 testing datasets are averaged to obtain the overall classification accuracy.

Table 2 summarizes the overall classification accuracy of each machine classifier on the three feature vectors respectively. As is shown in the experimental results, of the seven methods investigated in this paper, the left three (KM, LDA and RBFN) show relatively low overall classification accuracy, whereas the MLP SVM classifiers generate higher accuracy on all three features. It is also noted that the SDT and DTF methods also give relatively good results when using Gabor and ULBP features, however, the classification accuracy drop off considerably when using GLCM features.

We also compare the average F_1 measure of three categories from different classification algorithms (Figure 4). As discussed in the previous section, a higher value of F_1 measure indicates a better classifier. From the figure it is clear that MLP and SVM generally perform well for all three features, while the performance of other classifiers largely depends on the data used. For example, RBFN obtains reasonable result for Gabor and ULBP features but generates terrible result when using GLCM feature.

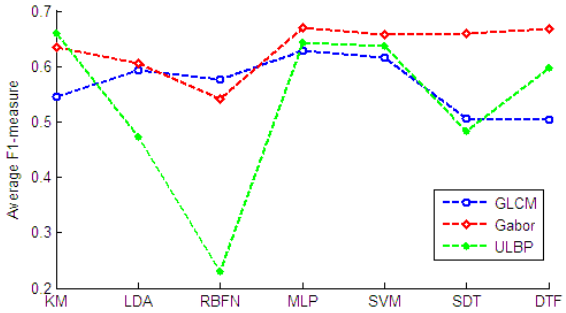


Figure 4 Average F1 measure of different classifiers

Figure 5 presents the analysis results of different classification algorithms for three texture features in ROC space. The plots of different algorithms use different markers specifiers, and within which the three categories are shown as different colours. Different from the analysis results of using overall accuracy and F_1 measure (Precision/Recall), ROC space provide more details of the classifier performance. As we can see from the figures, the performance of the classifier depends on the category and the feature used. By calculating the distance of points to the upper-left-corner point (point (0,1) in ROC space), the performance of the classifiers is ranked. From the experimental results, most classifiers perform the best for classifying *Cor-Tes*. The MLP classifier with ULBP features, the KM classifier with GLCM features, and the KM classifier with ULBP features obtained the best performance for classifying *Euc-Ter*, *Euc-Mel* and *Cor-Tes* respectively. The analysis result from ROC space is different from that derived from overall Accuracy and average F_1 measure, where SVM and MLP are supposed to be superior to other classifiers.

Table 2 Comparison of the overall classification accuracies

Features \ Classifiers	KM	LDA	RBFN	MLP	SVM	SDT	DTF
GLCM	55.37	64.46	62.81	69.42	69.42	58.68	56.20
Gabor	65.29	62.81	57.02	71.90	71.07	71.90	71.07
ULBP	69.42	50.41	52.89	72.73	71.07	66.12	71.07

Table 3 Comparison of the computational costs (in seconds)

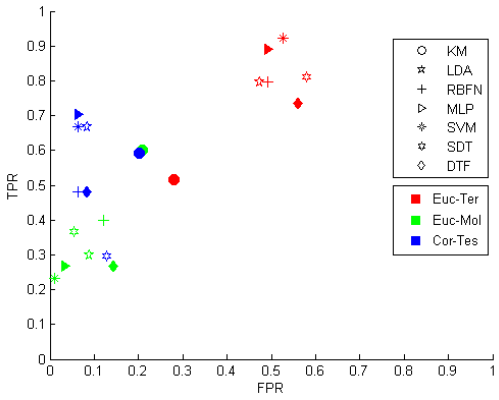
Features \ Classifiers	KM	LDA	RBFN	MLP	SVM	SDT	DTF
GLCM	2.64	0.23	43.53	2.72	22.89	0.3	0.55
Gabor	44.06	0.47	139.14	5.81	15.97	0.56	1.13
ULBP	385.97	7.41	113.19	136.41	230.93	2.53	2.31

Table 3 compares the computational cost of each machine classifier on the three feature vectors respectively. The analysis time is recorded by DTREG software under a desktop PC configuration of core duo 2.66GHz CUP and 2GB memory. From the results, we can see that the analysis time varies a lot for each machine classifiers and feature vectors. Overall, LDA, SDT and DTF are very computational efficient, whereas RBFN, MLP and SVM are computational much more intensive. It should also be mentioned that with the dimensions of feature vectors increase, the computational cost increase considerably (The dimensions of GLCM, Gabor and ULBP are 8, 48 and 607 respectively). For example, the analysis time of KM algorithm increase considerably when using ULBP feature.

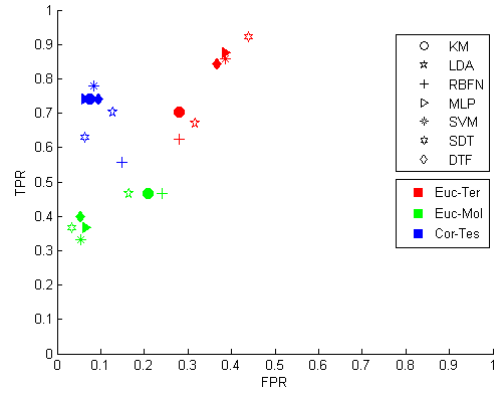
From the evaluation results, it is noticed that: 1) The selection of an approriate performance metrix is critical to evaluate the discriminatory power of different classifiers. Simply choose accuracy as the only measure often cause some misleading evaluation results. ROC analysis provide more details about the benefit and cost of a classifier. 2) The classification performance not only depends on the discriminatory power of classifiers but also the characteristics of datasets and the feature(s) selected. The evaluation results suggest to select approprate feature and classification algorithm for different categories. For example, to classify *Euc-Ter* the MLP classifier and ULBP feature are suggested. 3) Choosing a 'best model' is a complex issue and need to consider many factors such as the tradeoff between discriminatory power and computational cost. 4) Overall, the classification accuracies of all classifiers and texture features are not as good as expected. Trees can often show different apperances in different seasons and even the same tree species may vary due to the their health status. Nevertheless, using texture feature and machine learning techniques has shown the potential in analyzing vegetation in power line corridors by means of digital remote sensing imagery.

4. CONCLUSION

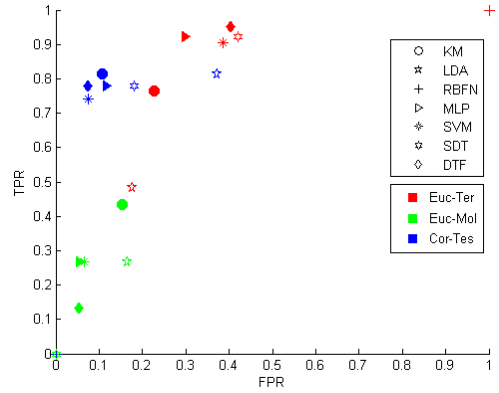
This paper evaluates the capability of seven machine learning algorithms and 3 texture features by means of classifying vegetation species in a power line corridor using high resolution aerial imagery. Object-based method is employed that local texture features are extracted from image-objects (i.e. tree crowns) and the classification is done in object feature space. Several performance matrixes are used to evaluate the performance of classifiers. The experimental results showed that the classification performance depends on the performance matrix, the characteristics of datasets and the feature(s) used.



(a) GLCM feature



(b) Gabor feature



(c) ULBP feature

Figure 5 The analysis results in ROC Space

ACKNOWLEDGEMENT

This work was conducted within the CRC for Spatial Information, established and supported under the Australian Government's Cooperative Research Centers Programme, and in conjunction with the Australian Research Centre for Aerospace Automation (ARCAA). The authors would like to thank Bred Jeffers from Greening Australia for assisting the field survey. The Authors would also like to acknowledge Ray Duplock for his support in high performance computing resources in QUT.

REFERENCES

Blaschke, T., 2010. Object-based image analysis for remote sensing. ISPRS Journal of Photogrammetry & Remote Sensing 65(1), pp. 2-16

Caruana, R. and Niculescu-Mizil, A., 2004. Data mining in metric space: an empirical analysis of supervised learning performance criteria. International Conference on Knowledge Discovery and Data Mining. Seattle, USA pp.

Coburn, C. A. and Roberts, A. C. B., 2004. A multiscale texture analysis procedure for improved forest stand classification. International Journal of Remote Sensing 25(20), pp. 4287-4308

Davies, E. R., 2009. Introduction to Texture Analysis. Handbook of Texture Analysis Mirmehdi, M., Xie, X. and Suri, J. Imperial College Press. 1-31.

Davis, J. and Goadrich, M., 2006. The relationship between Precision-Recall and ROC curve. the 23rd International Conference on Machine Learning. Pittsburgh pp. 233-240

Franklin, S. E., Hall, R. J., Moskal, L. M.et al., 2000. Incorporating texture into classification of forest species composition from airborne multispectral images. International Journal of Remote Sensing 21(1), pp. 61-79

Japkowicz, N., 2006. Why question machine learning evaluation method? AAAI workshop on Evaluation Methods for Machine Learning AAAI Press. 6-11.

Kubo, M., Kanda, F. and Muramoto, k., 2003. Texture feature extraction of tree using co-occurrence matrix from aerial images. SCIE Annual Conference. The Society of Instrument and Control Engineers.

Li, Z., Hayward, R., Zhang, J.et al., 2008. Individual tree crown delineation techniques for vegetation management in power line corridor. Digital Image Computing: Techniques and Applications (DICTA). Canberra pp. 148-154

Li, Z., Hayward, R., Zhang, J.et al., 2009. Towards automatic tree crown detection and delineation in spectral feature space using PCNN and morphological reconstruction. IEEE International Conference on Image Processing.

Mas, G. Y. J. F., Maathuis, B. H. P., Zhang, X.et al., 2006. Comparison of pixel-based and object-oriented image classification approaches - a case study in a coal fire area, Wuda, Inner Mongolia, China. International Journal of Remote Sensing 27(18), pp. 4039-4055

Mitchell, T., 1997. Machine Learning. McGraw Hill.

Ojala, T., Pietikainen, M. and Maenpaa, T., 2002. Multiresolution grey-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), pp. 971-987

R.Jensen, J., 2005. Classification based on object-oriented image segmentation. Introductory Digital Image Processing: A Remote Sensing Perspective (Third Edition). Pearson Education. 393-398.

Russell, B. D., Benner, C. L., Wischkaemper, J.et al., 2007. Reliability based vegetation management through intelligent system monitoring. Power Systems Engineering Research Center. Texas A&M University.

Sherrod, P. H., 2009. DTREG predictive modeling software. Users Manual. www.dtreg.com\dtreg.pdf.

Xie, X. and Mirmehdi, M., 2009. A galaxy of texture features. Handbook of Texture Analysis Mirmehdi, M., Xie, X. and Suri, J. Imperial College Press. 375-406.