# AUTOMATIC RECOGNITION AND CLASSIFICATION OF LAND COVER TYPES FROM SATELLITE IMAGES USING QUADRATIC DISCRIMINANT ANALYSIS AND THE MULTILAYER PERCEPTRON

Hillary Kipsang Choge[1], Scott E. Umbaugh[2]
hchoge@ieee.org sumbaug@siue.edu

[1]Department of Electrical and Electronic Engineering
Jomo Kenyatta University of Agriculture and Technology
P. O. Box 62000 - 00200
Nairobi, Kenya

[2]Department of Electrical and Computer Engineering
Southern Illinois University at Edwardsville,
Edwardsville, IL 62026 - 1801
USA

**Abstract.** In this paper, two schemes are applied to the recognition and classification of satellite images: Bayesian or Quadratic Discriminant Analysis and Artificial Neural Networks, specifically the Multilayer Perceptron (MLP). They are used to recognize and classify four land cover types: Healthy vegetation, Sparse vegetation, Water, and Urban areas or Roads. The two methods are used in order to compare the performance of the MLP to that of statistical methods in terms accuracy and speed. The training set consists of uncompressed, segmented JPEG satellite images of uniform size. Objects on each image in this set are manually assigned a class to train the Perceptron until convergence. The network is then used to classify the test set, consisting of similarly sized and processed images. The MLP is initially trained using both histogram and binary object features. Experiments show that histogram features are sufficient, and that the Multilayer Perceptron is able to classify land cover types from satellite images with a high degree of accuracy.

**1 INTRODUCTION.** More and more civilian applications such as communication and remote sensing demand the launching of satellites into space. This has been accompanied by a rising demand for better and more flexible tools for interpreting and processing the massive amounts of information relayed back to earth by these satellites. Particularly, for those applications that use the images relayed back by the satellites, such as weather and other geographic images, a lot of processing is needed before the images can be of any use. This is because the cameras and scanners mounted on these satellites sense and capture images in the invisible portions of the electromagnetic spectrum, especially the near, middle and far infrared portions. Infrared rays can penetrate, the hazy atmosphere and ionosphere that envelop the earth a lot better than visible light. Different land covers reflect, absorb and transmit different levels of the infrared spectrum, as shown in Figure 1.
Classification of different land covers can be performed by measuring the amount of near infrared reflected by objects in the image, after proper segmentation has been done. In this project, both the training and test image sets were acquired from LANDSAT image bank used by the Geography department at Southern Illinois University, Edwardsville. Minimal compression with JPEG was used to reduce data, and bigger images were split into manageable 512 by 512 pixel sub images. Segmentation and feature extraction were then performed manually, with the aid of the CVIPtools suite of tools developed by the Computer Vision and Image Processing research group, led by Scott Umbaugh[2]. Objects in the images were manually assigned one of four classes: water, healthy vegetation, sparse vegetation, and urban areas, which included roads and other infrastructure.

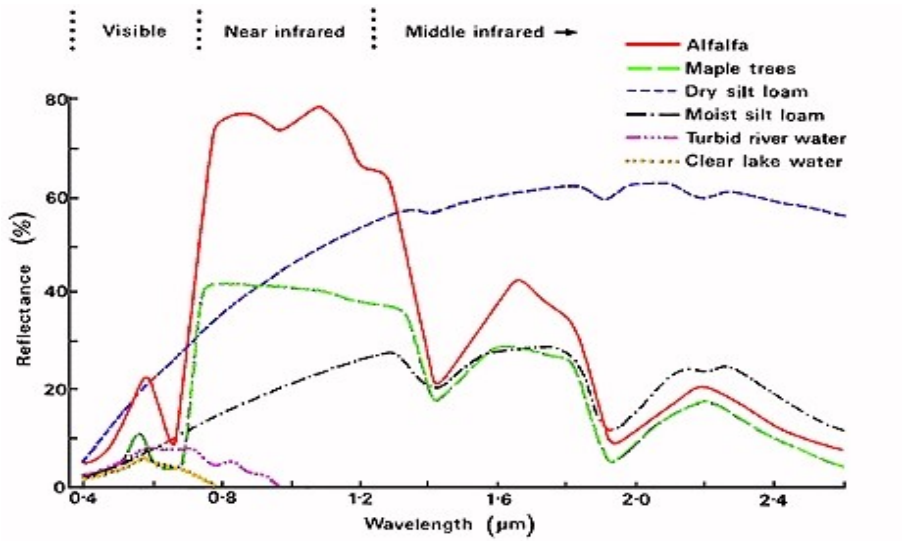



**Figure 1**:Typical spectral curves produced by different plants and soils. From the distinct shapes of each curve, it is possible to distinguish them on satellite images. This distinction is most in the near-infrared portion of the spectrum, Pearson (3**)**

In Section 2, we demonstrate how the Spherical Coordinates Transform or SCT is used for the segmentation stage, prior to feature extraction. We also show that only color or histogram features can be used for the classification without considering texture features. Campbell et al (2) used both texture and color or histogram features, and they also used artificial neural networks for both segmentation and classification. In the last part of the section, we explain how the features were chosen for extraction, and how we manually extracted these and other features before reducing the list to just histogram features. Discriminant analysis is then used for variable selection, where different combinations of features are ranked according to how well they could model the data and perform the classification. This step helped to drastically reduce the computational load by providing an advance view of how accurately the classification would go if certain choices of variables were made. The variables with the lowest correlation between the different classes were ranked highest.
In Section 3, we show how the features extracted, including the histogram mean, variance, entropy among others were used both to train a Multilayer Perceptron and to perform Bayesian or Quadratic Discriminant Analysis. We then compare the results from both classification schemes, with the percentage of correct classifications used as the yardstick. Confusion matrices are also shown and a discussion of the results given at the end of the section. Section 4 provides a brief conclusion and suggestion for further work.

## 2 DATA REDUCTION AND FEATURE EXTRACTION.

**2.1 Segmentation.** The methodology used was arrived at heuristically, by applying the segmentation and observing the results. The Spherical Coordinates Transform Center or SCT/Center algorithm, described in Umbaugh (2), was chosen because it produced a segmented image that retained most of the color properties of the original, which made it easier to perform manual classification of objects. This algorithm was initially developed for the identification of variegated coloring in skin tumor images, a feature believed to be highly predictive in the diagnosis of skin cancer. Unlike histogram thresholding techniques, the SCT/Center algorithm decouples the color information from the brightness information, which may vary with lighting conditions. It basically converts the RGB values of the original image into spherical coordinates L, angle A and

Angle B. L is a one-dimensional brightness space while angles A and B form a two-dimensional color space with A representing the angle between the blue coordinate and L, and B representing the angle between the red and green coordinates, as shown in Figure 2.

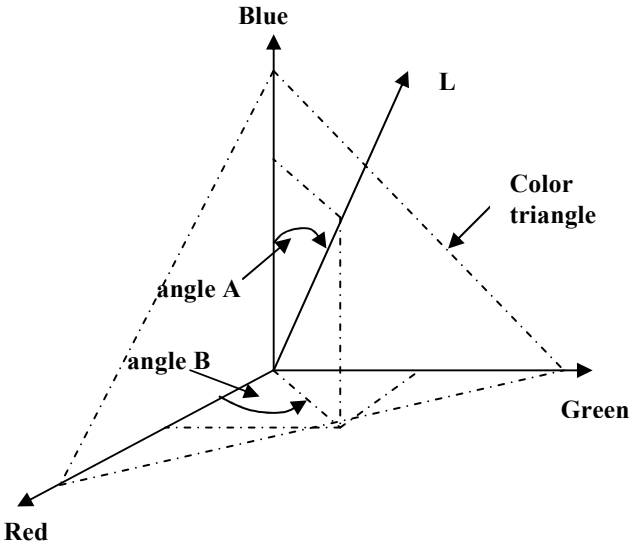
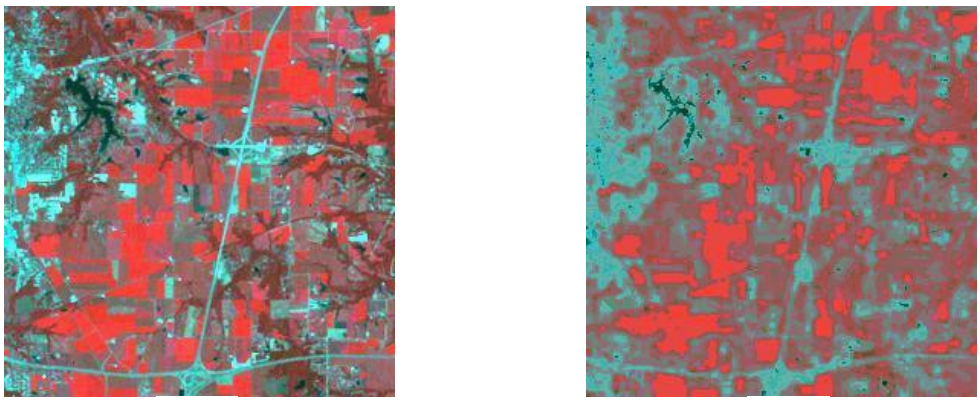



**Figure 2**: The SCT/Center algorithm separates the RGB information into the 2-D color space formed by angles A and B, and 1-D space formed by brightness L. The center of the color triangle is white, with the sides being the repective color vertices.

The segmented images contained much less data, and cut the memory requirements for the training and test images by half, without much loss of visual information, as seen in Figure 3. After trying out different numbers of colors for the A and B axes, an optimal number of 7 was arrived at after observing that more colors created contours in the segmented image while fewer colors caused different classes to merge into each other.



(a) (b)

**Figure 3**: (a) The original, minimally compressed JPEG image and (b) the image after SCT/Center segmentation with 7 colors each along A and B axes**.**

**2.2 Feature Selection and Extraction.** At the start, it was expected that a combination of geometric, texture as well as color or histogram features would be the optimal means of classifying land cover types. But after experimenting with each set separately, we noticed that only histogram

features were sufficient for accurate classification, especially after noting that the two geometric features included in one experiment, namely the Area of the object and its Euler number contributed a significant amount of the total variance when Principal Components Analysis is performed, which could cause misclassification if used in training the Perceptron. Table 1 below shows the features used, together with their respective columns in the Features spreadsheet.

| Feature # | Feature or Variable name |
|---|---|
| 1 | Histogram mean of band 1 |
| 2 | Histogram mean of band 2 |
| 3 | Histogram mean of band 3 |
| 4 | Standard deviation – band 1 |
| 5 | Standard deviation – band 2 |
| 6 | Standard deviation – band 3 |
| 7 | Skew – band 1 |
| 8 | Skew – band 2 |
| 9 | Skew – band 3 |
| 10 | Energy – band 1 |
| 11 | Energy – band 2 |
| 12 | Energy – band 3 |
| 13 | Entropy – band 1 |
| 14 | Entropy – band 2 |
| 15 | Entropy – band 3 |

**Table 1**:The features used. The same feature was evaluated for each of the three bands.

Feature extraction was then performed using CVIPtools. The features listed above were extracted for an average of 15 objects per training and test image, each of whose class was visually determined and assigned. About 1000 such observations were extracted, and exported as Comma Separated Values (CSV) into a spreadsheet that was then used for variable selection, training of the Perceptron and for Bayesian or Quadratic Discriminant analysis.

**3 OBJECT CLASSIFICATION.** In this Section, we describe how the data obtained from the previous section was analyzed visually, and also using the Principal Coordinates Transform, in order to determine which if any of the features would be redundant in the classification, and also which features were most likely to give the best results. In the second part of the section we demonstrate how both classification schemes were performed and compared.

**3.1 Visual and Principal Component Analysis.** Again, this part of the project was performed to provide an advance view of what the results were expected to be. In visual analysis, a histogram per class for each feature was generated and its shape compared to those of the other classes. If each class possessed a uniquely shaped histogram with distinct peaks, then the feature in question would most likely be useful for classification. Other measures used here included the mean, variance and skew of the histogram obtained. From this analysis, it was found that features #1 through #6, and #13 to #15 from Table 1 would be the best suited features as all possessed distinctly shaped histograms for the 4 classes.

This was confirmed when Principal Component Analysis (PCA) was performed. PCA converts a set of interrelated variables into a new uncorrelated set while still accounting for all the variance in the original variables. The correlation matrix was used as the dispersion matrix for computing the principal components, and the results dumped into a spreadsheet as a projection table, which takes every value, computes its projection and replaces the original value with the projection. After these

observations were made, a choice was made to reduce the number of variables from 15 to just 9, as the energy and skew features would introduce unnecessary redundancy and a reduction in speed of classification, especially for the MLP

**3.2 Classification With The Multilayer Perceptron (MLP).** The MLP was trained and optimized using the spreadsheet that contained the projection table obtained above. It consisted of an input layer with 9 inputs corresponding to each of the nine variables chosen during visual analysis, and 4 label outputs corresponding to the 4 classes of land covers. Only 1 hidden layer with 10 nodes was used, after noting that an increase in the number of hidden layers did not improve the results, while almost doubling processing time for every additional layer. The sigmoid neuron type was used for both the output and input layers, with steepness of 1.0, momentum of 0.9 and a learning rate of 0.05. The neuron weight or bias was set with an active range scale factor of 0.9. This implied that the sigmoid neurons would produce a minimum of 0.05 and a maximum of 0.95, or 90% of the full range from 0 to 1, thus avoiding saturation. The network parameters to be optimized were the weight and the input bias, using the Hestenes-Stiefel training method for a maximum of 10,000 iterations. Hestenes-Stiefel's method of conjugate gradients, requires less computation while providing faster convergence unlike other finite methods like Gaussian elimination, according to Lide (3). The results from this and Quadratic discriminant analysis are compared and discussed in the last part of this section.

**3.3 Quadratic (Bayesian) Classifier.** The Bayesian Classifier attempts to classify by determining the most economic way of distinguishing the different groups in the data, like by discarding variables which are little related to group distinctions. It maximally separates groups of observations and deals with two or more groups. An attempt is made to delineate based upon maximizing between-group variance while minimizing within-group variance thus building a model for optimal group prediction. Variable selection for this part was done using backward elimination, in which all 15 features were searched, and the best 14 out of these determined. The remaining 14 are searched again and the best 13 selected, and continued until the best 9 were obtained. The biggest advantage of this method is that when determining the best variables to use, the expected misclassification rate as a result of using these variables is also predicted, and this way it was determined that using the same 9 variables used for the MLP would produce optimal results with an expected misclassification rate of only 0.7%. The classification results are given below

**4 RESULTS.** Table 2 and 3 show the classification summary and confusion matrix respectively after creating and training the MLP with 10,000 iterations.

| Class | # per class | # classified correctly | # classified wrong | % correct | % error |
|---|---|---|---|---|---|
| Healthy Vegetation | 155 | 152 | 3 | 98.1 | 1.9 |
| Sparse Vegetation | 169 | 158 | 11 | 93.5 | 6.5 |
| Urban areas and Roads | 171 | 168 | 3 | 98.3 | 1.7 |
| Water | 106 | 100 | 6 | 94.3 | 5.7 |
| Total | 601 | 578 | 23 | 96.2 | 3.8 |
| Normalized | | | | **96.1** | **3.9** |

**Table 2**: Classification summary from the Perceptron showing a normalized error of only 3.9%.

| | Healthy vegetation | Sparse vegetation | Urban + roads | Water |
|---|---|---|---|---|
| Healthy vegetation | 152 | 3 | 0 | 0 |
| Sparse vegetation | 8 | 158 | 1 | 2 |
| Urban + roads | 0 | 0 | 168 | 3 |
| Water | 0 | 3 | 3 | 100 |

**Table 3**: Confusion matrix, from MLP.

From Table 2, it is seen that water and sparse vegetation were the most misclassified, and the confusion matrix in Table 3 shows that the wrongly classified sparse vegetation was classified as healthy vegetation, which can be understood, since pixels at the boundaries of the segmented image would most likely mistake the two. Table 4 and 5 show the results from running the Bayesian classifier, using the test set for validation. Cross validation was also tried, where the same data used for training was split into 10 partitions. Classification was then performed using 9 of these, leaving one partition out to be used for testing the results. This is a faster method, but does not adequately represent the results since it makes no use of the test data. However, applying this method to both sets of data produced nearly similar results, and hence the inclusion here of these results

| | **# per class** | **# classified correct** | **# Classified wrong** | **% correct** | **% error** |
|---|---|---|---|---|---|
| Healthy Vegetation | 155 | 153 | 2 | 98.71 | 1.29 |
| Sparse vegetation | 169 | 162 | 7 | 95.86 | 4.14 |
| Urban + Roads | 171 | 166 | 5 | 97.08 | 2.92 |
| Water | 106 | 104 | 2 | 98.11 | 1.89 |
| Total | 601 | 585 | 16 | 97.34 | 2.66 |
| **Normalized** | | | | **97.44** | **2.56** |

**Table 4**: Results from the Bayesian classifier slightly better than the MLP

| Class | Healthy Vegetation | Sparse vegetation | Urban + roads | Water |
|---|---|---|---|---|
| Healthy Vegetation | 153 | 2 | 0 | 0 |
| Sparse vegetation | 2 | 162 | 4 | 1 |
| Urban + roads | 0 | 1 | 166 | 4 |
| Water | 0 | 0 | 2 | 104 |

**Table 5**: Confusion matrix, showing that the Bayesian classifier did not suffer the same setbacks seen in the MLP, when it comes to borderline objects.

| Class | **# per class** | **# correct** | **# error** | **% correct** | **% error** |
|---|---|---|---|---|---|
| Healthy Vegetation | 155 | 153 | 2 | 98.71 | 1.29 |
| Sparse vegetation | 169 | 163 | 6 | 96.45 | 3.55 |
| Urban + Roads | 171 | 166 | 5 | 97.08 | 2.92 |
| Water | 106 | 104 | 2 | 98.11 | 1.89 |
| Total | 601 | 586 | 15 | 97.50 | 2.50 |
| **Normalized** | | | | **97.59** | **2.41** |

**Table 6**: Results from the Bayesian classifier, but with a 10-partition Cross validation used instead of the test data**.**

**5 CONCLUSION.** This paper shows that artificial neural networks can be used to classify land cover types almost as accurately as statistical methods like the Bayesian Classifier, though it is much slower because of the amount of computation that needs to be done. We have also shown that

for classification of land cover types from satellite images, it is possible to attain above 96% for the MLP and 97% correct classification for the Bayesian classifier using just the histogram or color features, without the need to include texture and geometrical information. Only 1 hidden layer was used for the Perceptron, and using Cross Validation, it is possible to check the validity of the classification. Future work will involve the application of the perceptron in classification of a larger data set, with extracted features not just limited to histogram properties. The system can also be adapted to perform finger or palm-print recognition for biometric applications by including geometric and other features that do not vary with rotation, scaling and translation. Different training, optimization and stopping criteria could also be applied in training the network, and more layers would then be necessary.

This paper has also shown that care has to be taken in selecting which variables to use for the input layer of the perceptron because a large number of features that may be interrelated across classes results in a reduction of the accuracy, while at the same increasing computational time drastically. In this experiment, we tried using all the extracted features without any kind of selection, and the accuracy went down to 49%. On the other hand, after running the backward elimination method of variable selection and choosing the best two features, the classification rate was still a healthy 80% from the Bayesian classifier and 75% for the MLP. This underlines the importance of visual and Principal Component Analysis in any image processing application.

## 6 REFERENCES.


1. Campbell, N. W., Thomas, B. T., Troscianko, T., Automatic Segmentation and Classification of Outdoor Images Using Neural Networks, Special Issue on Neural Networks for Computer Vision Applications,

2. Umbaugh, S. E., Computer Vision and Image Processing – A Practical Approach Using CVIPtools, Prentice Hall, NJ, 1998.

3. Lide, D.R., Methods of Conjugate Gradients For Solving Linear Systems, A Century of Excellence in Measurements, Standards, and Technology**,** NIST Special publication (SP 958), http://nv1.nist.gov/pub/nistpubs/sp 958-lide/081-085.pdf, 2001.

4. Kou-Yuan, H., Neural network for robust recognition of seismic patterns, Proceedings of the International Joint Conference on Neural Networks, vol. 4, 2930-2935, 2002.

5. Schalkoff, R. J., Pattern Recognition: Statistical, Structural, And Neural Network Approaches**,** John Wiley & Sons, New York, 1992.

6. Gurney, K., An Introduction to Neural Networks, UCL Press, UK, 1996.

7. Jiang, X., Harvey, A., Wah, K. S., Constructing And Training Feed-forward Neural Networks For Pattern Classification, Pattern Recognition Journal, vol. 36, 4, 853 – 867, 2003.

8. Irvin, P. D S., Wilkinson, A. J., An Integrated Image Segmentation and Analysis System, 4th International Conference on Pattern Recognition, Editor: J. Kittler, 26 – 33, 1998.

9. Pearson, R. S., Spectral Nature of Remote Sensing, Department of Geography, Southern Illinois University at Edwardsville, IL, 2002.

10. You, W. L. J., Zhang, D., Texture-based Palm-print retrieval Using a Layered Search scheme For Personal Identification, IEEE Transactions on Multimedia, vol. 7, 5, 891-898, 2005.