Decision Tree Classification of Remotely Sensed Satellite Data using Spectral Separability Matrix

M. K. Ghose , Ratika Pradhan, Sucheta Sushan Ghose
Department of Computer Science and Engineering
Sikkim Manipal Institute of Technology,
Sikkim, INDIA
mkghose2000@yahoo.com, ratika_pradhan@yahoo.co.in, suchetaghose@gmail.com

***Abstract*— In this paper an attempt has been made to develop a decision tree classification algorithm for remotely sensed satellite data using the separability matrix of the spectral distributions of probable classes in respective bands. The spectral distance between any two classes is calculated from the difference between the minimum spectral value of a class and maximum spectral value of its preceding class for a particular band. The decision tree is then constructed by recursively partitioning the spectral distribution in a Top-Down manner. Using the separability matrix, a threshold and a band will be chosen in order to partition the training set in an optimal manner. The classified image is compared with the image classified by using classical method Maximum Likelihood Classifier (MLC). The overall accuracy was found to be 98% using the Decision Tree method and 95% using the Maximum Likelihood method with kappa values 97% and 94 % respectively.**

Keywords- Decision Tree Classifier (DTC), Separability Matrix, Maximum Likelihood Classifier (MLC), Stopping Criteria.

I. Introduction

Image classification is one of the primary tasks in geo-computation, that being used to categorize for further analysis such as land management, potential mapping, forecast analysis and soil assessment etc. Image classification is method by which labels or class identifiers are attached to individual pixels on basis of their characteristics. These characteristics are generally measurements of their spectral response in various bands. Traditionally, classification tasks are based on statistical methodologies such as Minimum Distance-to-Mean (MDM), Maximum Likelihood Classification (MLC) and Linear Discrimination Analysis (LDA). These classifiers are generally characterized by having an explicit underlying probability model, which provides a probability of being in each class rather than simply a classification. The performance of this type of classifier depends on how well the data match the pre-defined model. If the data are complex in structure, then to model the data in an appropriate way can become a real problem.

In order to overcome this problem, non-parametric classification techniques such as Artificial Neural Network (ANN) and Rule-based classifiers are increasingly being used. Decision Tree classifiers have, however, not been used widely by the remote sensing community for land use classification despite their non-parametric nature and their attractive properties of simplicity, flexibility, and computational efficiency [1] in handling the non-normal, non-homogeneous and noisy data, as well as non-linear relations between features and classes, missing values, and both numeric and categorical inputs [2].

In this paper, an attempt has been made to develop a decision tree classification algorithm specifically for the classification of remotely sensed satellite data using the separability matrix of spectral distributions of probable classes. The computational efficiency is measured in terms of computational complexity measure. The proposed algorithm is coded in Visual C++ 6.0 language to develop user-friendly software for decision tree classification that requires a bitmap image of the area of interest as the basic input. For the classification of the image, the training sets are chosen for different classes and accordingly spectral separability matrix is obtained. To evaluate the accuracy of proposed method, a confusion matrix analysis was employed and kappa coefficient along with errors of omission and commission were also determined. Lastly, the classified image is compared with the image classified by using classical method MLC.

II. Related work

The idea of using decision trees to identify and classify objects was first reported by Hunt et al. [3]. Morgan and Sonquist [4] developed the AID (Automatic Interaction Detection) program followed by the THAID developed by Morgan and Messenger [5]. Breiman et al. [6] proposed the CART (Classification and Regression Trees) to solve classification problems. Quinlan [7] developed a decision tree software package called ID3 (Induction of Decision Tree) based on the recursive partitioning greedy algorithm and information theory, followed by the improved version C4.5 addressed in [2]. Buntine [8] developed IND package using the standard algorithms from Brieman's CART and Quinlan's ID3 and C4. It also introduces the use of Bayesian and minimum length encoding methods for growing trees and graphs. Sreerama Murthy [9] reported the decision tree package OC1

(Oblique Classifier 1) which is designed for applications where instances have numeric continuous feature values. Friedl and Brodley [10] have used decision tree classification method to classify land cover using univariate, multivariate and hybrid decision tree as the base classifier and found that hybrid decision tree outperform the other two. Mahesh Pal and Paul M. Mather [11], [12] have suggested boosting techniques for the base classifier to classify remotely sensed data to improve the overall accuracy. Min Xu et al. [13] have suggested decision tree regression approach to determine class proportion within the pixels so as to produce soft classification for remote sensing data. Michael Zambon et al. [14] have used rule based classification using CTA (Classification Tree Analysis) for classifying remotely sensed data and they have found that Gini and class probability splitting rules performs well compared to towing and entropy splitting rules. Mahesh pal [15] have suggested ensemble approaches that includes boosting, bagging, DECORATE and random subspace with univariate decision tree as the base classifier and found that later three approaches works even well compared to boosting. Mingxiang Huang et al. [16] have suggested Genetic Algorithm (GA) based decision tree classifier for remote sensing data SPOT – 5 and found that GA- based decision tree classifier outperform all the classical classification method used in Remote sensing. Xingping Wen et al. [17] have used CART (Classification and Regression Trees) and C5.0 decision tree algorithms for remotely sensed data. Abdelhamid A. Elnaggar and Jay S. Noller [18] have also found that decision tree analysis is a promising approach for mapping soil salinity in more productive and accurate ways compared to classical classification approach.

III. Decision tree approach

A decision tree is defined as a connected, acyclic, undirected graph, with a root node, zero or more internal nodes (all nodes except the root and the leaves), and one or more leaf nodes (terminal nodes with no children), which will be termed as an ordered tree if the children of each node are ordered (normally from left to right) (Coreman et al. [19]). A tree is termed as univariate, if it splits the node using a single attribute or a multivariate, if it uses several attributes. A binary tree is an ordered tree such that each child of a node is distinguished either as a left child or a right child and no node has more than one left child or more than one right child. For a binary decision tree, the root node and all internal nodes have two child nodes. All non-terminal nodes contain splits.

A Decision Tree is built from a training set, which consists of objects, each of which is completely described by a set of attributes and a class label. Attributes are a collection of properties containing all the information about one object. Unlike class, each attribute may have either ordered (integer or a real value) or unordered values (Boolean value).

Several methods (Breiman et.al [6], Quinlan [2] and [7]) have been proposed to construct decision trees. These algorithms generally use the recursive-partitioning algorithm, and its input requires a set of training examples, a splitting rule, and a stopping rule. Partitioning of the tree is determined by the splitting rule and the stopping rule determines if the examples in the training set can be split further. If a split is still possible, the examples in the training set are divided into subsets by performing a set of statistical tests defined by the splitting rule. The test that results in the best split is selected and applied to the training set, which divides the training set into subsets. This procedure is recursively repeated for each subset until no more splitting is possible.

Stopping rules vary from application to application but multiple stopping rules can be used across different applications. One stopping rule is to test for the purity of a node. For instance, if all the examples in the training set in a node belong to the same class, the node is considered to be pure (Breinman et.al [6]) and no more splitting is performed. Another stopping rule is by looking at the depth of the node, defined by the length of the path from the root to that node (Aho et.al. [20]). If the splitting of the current node will produce a tree with a depth greater than a pre-defined threshold, no more splitting is allowed. Another common stopping rule is the example size. If the number of examples at a node is below a certain threshold, then splitting is not allowed. Four widely used splitting rules that segregates data includes: gini, twoing, entropy and class probability. The gini index is defined as:

$$gini(t) = \sum p_i (1 - p_i) \qquad (1)$$

where pi is the relative frequency of class i at node t, and node t represents any node (parent or child) at which a given split of the data is performed (Apte and Weiss [21]). The gini splitting rule attempts to find the largest homogeneous category within the dataset and isolate it from the remainder of the data. Subsequent nodes are then segregated in the same manner until further divisions are not possible. An alternative measure of node impurity is the towing index:

$$twoing\,(t) = \frac{p_L p_R}{4}\left(\sum_i \left| p(i|t_L) - p(i|t_R) \right|\right)^2 \qquad (2)$$

where L and R refer to the left and right sides of a given split respectively, and p(i|t) is the relative frequency of class i at node t (Breiman [22]). Twoing attempts to segregate data more evenly than the gini rule, separating whole groups of data and identifying groups that make up 50 percent of the remaining data at each successive node. Entropy is a measure of homogeneity of a node and is defined as:

$$entropy(t) = -\sum_i p_i \log p_i \qquad (3)$$

where p_i is the relative frequency of class i at node t (Apte and Weiss [21]). The entropy rule attempts to identify splits where as many groups as possible are divided as precisely as possible and forms groups by minimizing the within group diversity (De'ath and Fabricius [23]). Class probability is also based on the gini equation but the results are focused on the probability structure of the tree rather than the classification structure or prediction success. The rule attempts to segregate the data based on probabilities of response and uses class probability trees to perform class assignment (Venables and Ripley [24]).

From the above discussions, it is evident that a decision tree can be used to classify a pixel by starting at the root of the tree and moving through it until a leaf is encountered. At each non-leaf decision node, the outcome for the test at the node is determined and attention shifts to the root of the sub-tree corresponding to this outcome. This process proceeds until a leaf is encountered. The class that is associated with the leaf is the output of the tree. A class is one of the categories, to which pixels are to be assigned at each leaf node. The number of classes is finite and their values must be established beforehand. The class values must be discrete. A tree misclassifies a pixel if the class label output by the tree does not match the class label. The proportion of pixels correctly classified by the tree is called accuracy and the proportion of pixels incorrectly classified by the tree is called error (Coreman et.al., 1989).

IV. THE PROPOSED CRITERIA

To construct a classification tree, it is assumed that spectral distributions of each class are available. The decision tree is then constructed by recursively partitioning the spectral distribution into purer, more homogenous subsets on the basis of the tests applied to feature values at each node of the tree, by employing a recursive divide and conquer strategy. This approach to decision tree construction thus corresponds to a top-down greedy algorithm that makes locally optimal decisions at each node. Steps involved in this process can be summarized as below:

- Optimal band selection: By using the separability matrix, a threshold and band is chosen in order to partition the training set in an optimal manner.
- Based on a partitioning strategy, the current training set is divided into two training subsets by taking into account the values of the threshold.
- When the stopping criterion is satisfied, the training subset is declared as a leaf.

The separability matrices are obtained in the respective bands, by calculating the spectral distance between pairs of classes in row and column. The spectral distance between any two classes is calculated from the difference between the minimum spectral value of a class and maximum spectral value of its preceding class for a particular band. More the spectral distance, maximum is the separability. For spectral distance less than zero there will be overlapping between classes.

A. Splitting Criteria

An attempt has been made to define splitting rules by calculating the separability matrix to split the given set of classes into two subsets where the separability is maximum or the amount of spectral overlapping is minimum between two subsets. That is, the split group together a number of classes that are similar in some characteristic near the top of the tree and isolate single class in the bottom of the tree. First the spectral classes are arranged in ascending order based on the value of their midpoints. Lower the value of the midpoint lowers the class orders. Here the midpoint of a class is calculated by (Min+ Max)/2. This will result in reduction of the matrix computation/memory allocation, wherein only the upper diagonal elements are to be considered for finding the threshold. The threshold value is calculated, as the midpoint between the spectral distributions of the classes in the band where, the separability is maximum or overlapping is minimum. Once the split is found, this is used to find the subsets at each node.

B. Determining the Terminal Node or Stopping Criteria

When a subset of classes becomes pure, create a node and label it by the class number of the pure subset having only one class. If a subset having more than one class, apply the splitting criteria till it becomes purer.

V. THE PROPOSED ALGORITHM

Notations and Assumptions:

$\mathrm{S} - \textit{the set of spectral distributions of all classes with class label.}$

$n - \textit{the number of classes.}$

$K - \textit{the number of bands.}$

$M_k = \left(m_{ij}^k\right), (1 \le i \le n, 1 \le j \le n) - \textit{the separability matrix in band } k.$

$C_{kl}^1 - \textit{the range } \mathrm{minimum} \textit{ of the spectral distribution in band } k \textit{ for the class } l.$

$C_{kl}^2 - \textit{the range } \mathrm{maximum} \textit{ of the spectral distribution in band } k \textit{ for the class } l.$

Initialize the root by all the classes.

$\mathrm{DTree}\,(N, S, n)$

Step 1: For each band $1 \le k \le K$, sort the spectral distributions with respect to the midpoint.

If $n \le 2$ and overlapping, then construct the tree as in special case.

else

goto step2.

Step 2:

$$For\, 1 \le k \le K\ ,\, 1 \le i \le n\ ,\, 1 \le j \le n$$

$$M_k = \left(m_{ij}^k\right) = \begin{cases} 0 & , \textit{if } i = j \\ C_{kj}^1 - C_{ki}^2 & , \textit{otherwise} \end{cases}$$

Step 3: Find the threshold value, which will divide the set of classes S into two subsets of classes, say S_L and S_R, such that the separability between a pair of classes is maximum. The procedure is as follows.

Case 1: In all the K matrices, consider only the rows having all the elements to the right of the diagonal element are positive. If no such row exists in all the matrices go to case 2.

Find the minimum element from each such row. Find a maximum from all such minimum elements. Let this element be m_{rc}^{b} which is at the r^{th} row and c^{th} column of the matrix M_b. That is,

$$If\ \exists\ b,r,c\ such\,that\ m_{rc}^{b} = \max_{1\le k\le K}\left\{\max_{1\le i<n}\left\{\min_{i+1\le j\le n}\left\{m_{ij}^{k} \,\middle/\, m_{ij}^{k}\ge 0\ \ for\,all\ j\right\}\right\}\right\}$$

then the split is in between the classes represented by the row(r) and column(c) in band b.

$$\text{Threshold(T)} = \frac{1}{2}\left[\left\{\begin{array}{l}\text{range maximum of the } class\\ distribution\ represented\\ by\ the\ row\ in\ band\ b\end{array}\right\} + \left\{\begin{array}{l}\text{range minimum of the } class\\ distribution\ represented\\ by\ the\ column\ in\ band\ b\end{array}\right\}\right]$$

$$= \frac{1}{2}\left\{C_{br}^{2} + C_{bc}^{1}\right\}$$

$\text{BAND} = b$, go to step 4.

Case 2: In all K matrices, consider only the rows having at least one positive element. Find the minimum element from each such row. Find a maximum from all such minimum elements. Let this element be m_{rc}^{b} which is at the r^{th} row and c^{th} column of the matrix M_b. That is,

$$If\ \exists\ b,r,c\ such\,that\ m_{rc}^{b} = \max_{1\le k\le K}\left\{\max_{1\le i<n}\left\{\min_{i+1\le j\le n}\left\{\ m_{ij}^{k} \,\middle/\, m_{ij}^{k}\ge 0\ \right\}\right\}\right\}$$

then the split is in between the classes represented by the row(r) and column(c) in band b. Find the threshold (T=T2) as in case 1. Let BAND=BAND2=b.

Check whether the threshold T lies in any of the spectral range in the band b, except for the class distributions represented by r and c.

If yes, compute $\min\left\{\left|\ \text{T} - C_{bi}^{1}\right|, \left|\ \text{T} - C_{bi}^{2}\right|\right\}$ for each class i, $i \ne r$ and $i \ne c$, such that $T \in (C_{bi}^{1}, C_{bi}^{2})$. Find a maximum of all such minima. Call it as EF2 and then go to case 3.

i) If no, go to step 4.

Case 3: From all the upper triangular elements of all K matrices, find the minimum negative element; say m_{rc}^{b} which is at the r^{th} row and c^{th} column of the matrix M_b.

$$\text{That is, find } m_{rc}^{b} = \max_{1\le k\le K}\left\{\max_{i<j}\left\{m_{ij}^{k}\right\}\right\}$$

Calculate the threshold (T = T3) as in above cases. Let BAND=BAND3=b.

Compute $\min\left\{\left|\ \text{T} - C_{bi}^{1}\right|, \left|\ \text{T} - C_{bi}^{2}\right|\right\}$ for each class i, $i \ne r$ and $i \ne c$, such that $T \in (C_{bi}^{1}, C_{bi}^{2})$. Find the maximum of all such minima. Call it as EF3.

If EF2 ≤ EF3, let $T = T_2$ and BAND=BAND2, go to step 4.

Step 4: Assign the threshold (T) & BAND to the node N.

Step 5: Find the subsets of classes; say left subset (S_L) and the right subset (S_R) as follows.

S_L = Set of classes having distributions with range maximum or mid-point $\le T$ in band BAND and let n_L be the number of classes in S_L.

S_R = Set of classes having distributions with range minimum or mid-point $>$T in band BAND and let n_R be the number of classes in S_R.

Step 6: Initialize the left node (N_L) of N by all the classes in S_L and the right node (N_R) of N by all the classes in S_R.

Step 7: If $n_L \le 1$, terminate splitting and return a node with the corresponding class Label. Else

$$\text{DTree}\,(N_L, S_L, n_L)\,.$$

If $n_R \le 1$, terminate splitting and return a node with the corresponding class Label. Else

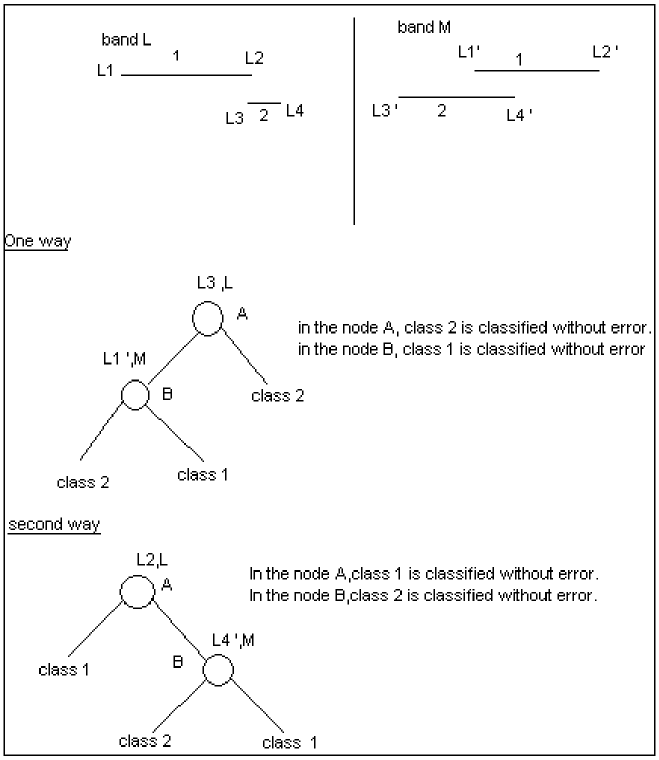
$$\text{DTree}\,(N_R, S_R, n_R)$$

A. *Special Case*

When n=2, and these two classes are overlapping in all the bands, we can construct a decision tree in the following ways:

Consider the following distributions:

We choose way 1, when the node A is the left child of its parent and way 2, when it is the right child of its parent. Because, when the node A is on the left branch and the threshold in node A and the threshold in its parent are chosen from the same band, then it will reduce the error in classification and the same reason applies for the other one.


band L
1
L1
L2
L3
2
L4
band M
L1'
1
L2 '
L3 '
2
L4 '
One way
L3 ,L
A
in the node A, class 2 is classified without error.
in the node B, class 1 is classified without error
L1 ',M
B
class 2
class 2
class 1
second way
L2,L
A
In the node A,class 1 is classified without error.
In the node B,class 2 is classified without error.
L4 ',M
B
class 1
class 2
class 1

VI. Application and Case Study

To validate the applicability of the proposed decision tree algorithm, a case study is presented in this section, which is carried out on IRS-1C/LISS III sample image with 23m resolution. The FCC (False Color Composite) of the input image (Figure 1) belongs to the Mayurakshi reservoir (Latitude: 240 09’ 47.27’’ N and Longitude: 87017’49.41” E) of Jharkhand state (INDIA) and band used to prepare FCC includes – Red (R), Green (G), Near Infrared (NIR). The IRS-IC/LISS III data of Mayurakshi reservoir was acquired on October, 2004. The input image is first converted into a bitmap, and then used as input for the DTCUSM software. As per the software requirement the training set along with test set were selected. For the present study, eleven classes viz., Turbid water, Clear water, Forest, Dense Forest, Upland Fallow, Lateritic Cappings, River Sand, Sand, Drainage, Fallow, and Wetland are considered. The training set for these eleven classes is chosen considering the prior knowledge of the hue, tone and texture of these classes, in addition, physical verification on ground were also done for each of these class. The spectral class distributions for the training set taken from the input image are shown in Table I.

Once the training for all the classes is set, the proposed decision tree algorithm is applied to classify the image. The decision tree construction steps for the spectral class distributions given in Table 1 are shown in Figure 2. The output image after applying the proposed Decision Tree Classification method is shown in Figure 3.

To assess accuracy of the proposed technique, the confusion matrix along with the errors of omission and commission and overall accuracy, kappa coefficient (Jensen, 1996) are obtained and shown in Table II. For the comparison purpose, the same image is classified again by using Maximum Likelihood Classifier (Jensen, 1996) with same training sets as used in Decision tree classification. The classified image by Maximum Likelihood method is shown in Figure 4. The percentage of pixels in each class is given for both the cases in Figure 5.

TABLE I. SPECTRAL CLASS DISTRIBUTIONS FOR THE TRAINING SET TAKEN FROM THE IMAGE IN FIGURE 1.

	Turbid Water	**Clear Water**	**Forest**	**Dense Forest**	**Upland Fallow**	**Lateritic Cappings**	**River Sand**	**Sand**	**Drainage**	**Fallow**	**Wetland**
Band 1	**0-11**	**0-31**	**140-226**	**83-136**	**48-110**	**49-81**	**223-255**	**158-238**	**131-208**	**113-195**	**56-164**
Band 2	**17-64**	**2-57**	**6-103**	**8-52**	**82-131**	**32-76**	**230-255**	**220-255**	**105-177**	**149-230**	**82-222**
Band 3	**67-162**	**0-70**	**0-104**	**0-42**	**44-88**	**4-40**	**236-255**	**223-255**	**127-195**	**123-197**	**117-255**

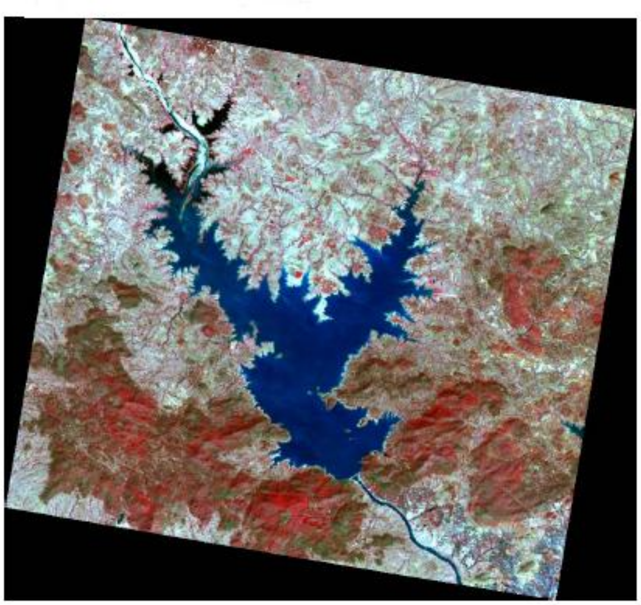


Figure 1. FCC (IRS LISS III Image – OCT 2004) of study area – Mayurakshi Reservoir, Jharkhand, INDIA, Band used – Red (R), Green (G), Near Infrared (NIR).

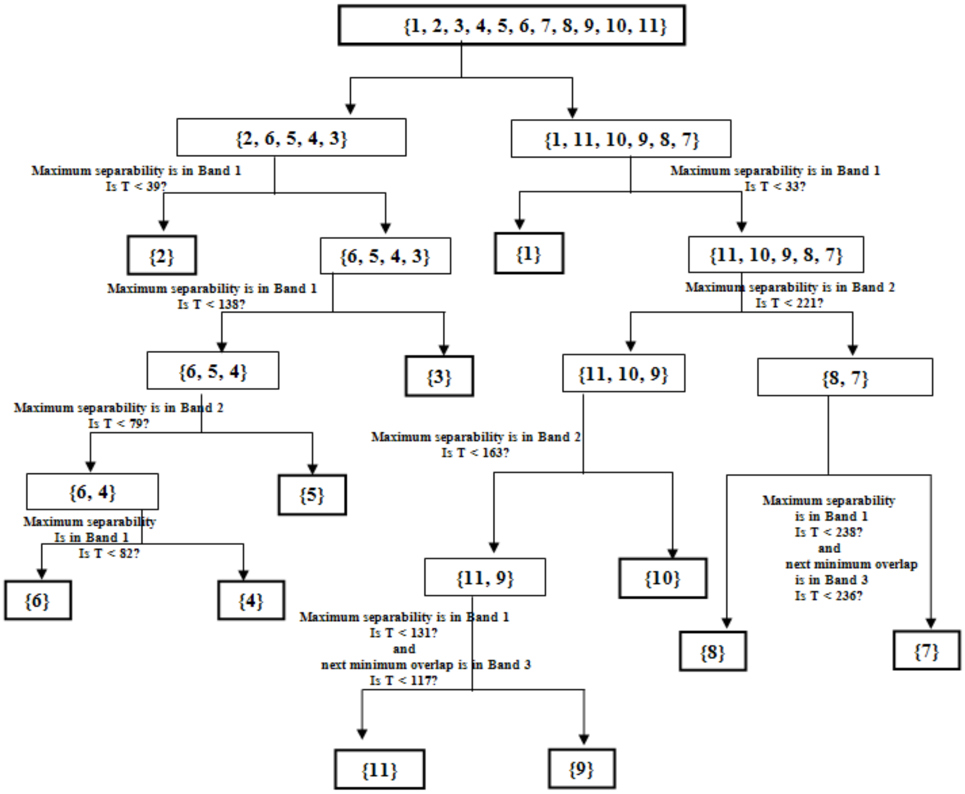


Figure 2. Decision tree construction steps for distribution given in table 1.

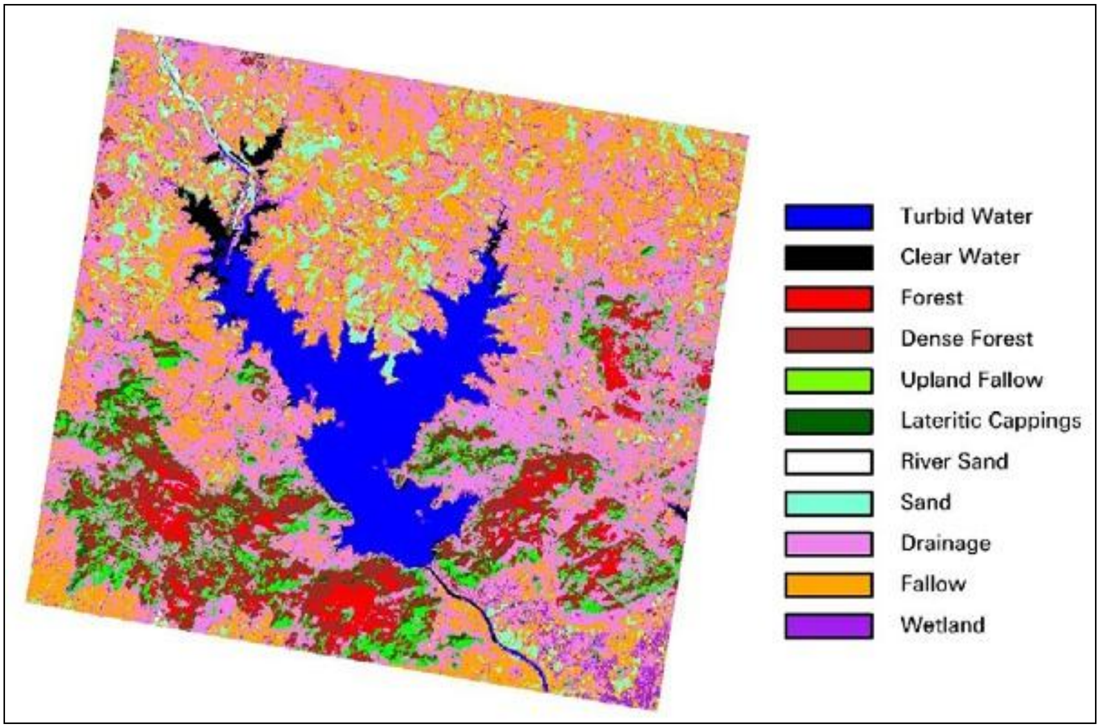


Figure 3. Classified image using the proposed Decision Tree classifier.

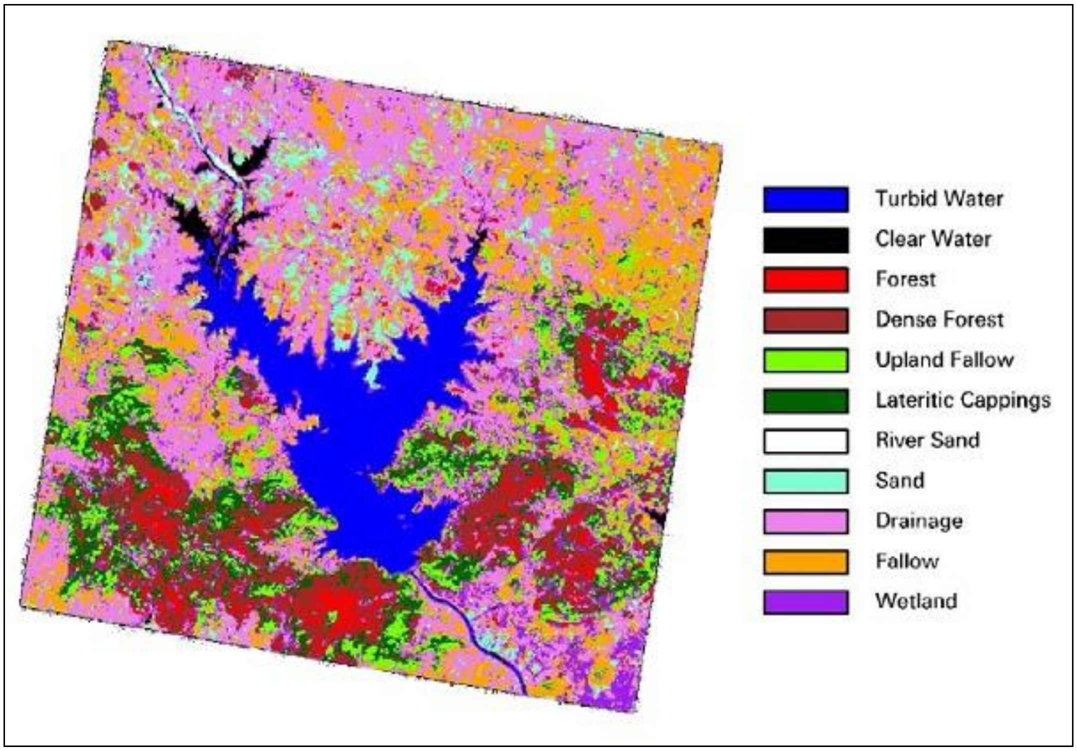


Figure 4. Classified image using standard Maximum Likelihood classifier.

TABLE II. CONFUSION MATRIX AND ACCURACY CALCULATIONS FOR DECISION TREE METHOD

	Turbid Water	**Clear Water**	**Forest**	**Dense Forest**	**Upland Fallow**	**Lateritic Cappings**	**River Sand**	**Sand**	**Drainage**	**Fallow**	**Wetland**	**Error of omission (in %)**
Turbid Water	323	0	0	0	0	0	0	0	0	0	0	**0.00**
Clear Water	1	34	0	0	0	0	0	0	0	0	0	**2.86**
Forest	0	0	42	0	0	0	0	0	0	0	0	**0.00**
Dense Forest	0	0	0	35	0	0	0	0	0	0	0	**0.00**
Upland Fallow	0	0	0	0	15	0	0	0	1	0	0	**6.25**
Lateritic Cappings	0	0	0	0	0	9	0	0	0	0	0	**0.00**
River Sand	0	0	0	0	0	0	12	3	0	0	0	**20.00**
Sand	0	0	0	0	0	0	0	16	0	0	0	**0.00**
Drainage	0	0	0	0	0	0	0	0	15	0	0	**0.00**
Fallow	0	0	0	0	0	0	0	0	0	12	0	**0.00**
Wetland	0	0	0	0	0	0	0	0	0	2	4	**33.33**
Error of commission (in %)	**0.31**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**15.79**	**6.25**	**14.29**	**0.00**	

Overall Accuracy = 98.66% Kappa = 97.77%

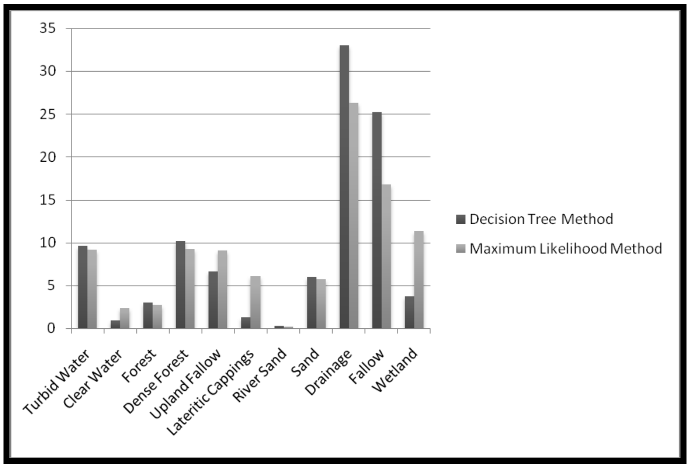


Figure 5. Class wise percentage of pixels for Decision Tree and Maximum likelihood methods

VII. SUMMARY AND CONCLUSION

In this paper, a decision tree classification algorithm for remotely sensed satellite data using separability matrix of spectral distributions of probable classes has been developed. To test and validate the proposed Decision Tree algorithm, the sample image taken into consideration is multi-spectral IRS-1C/LISS III of Mayurakshi reservoir of Jharkhand state. The proposed Decision Tree classifier can also be used for hyper-spectral remote sensing data considering the best bands as input for preparing spectral class distribution. The sample image is classified by both Decision Tree method and Maximum Likelihood method and then the overall accuracy, kappa coefficients were calculated. The overall accuracy for the sample test image was found to be 98% using the Decision Tree method and 95% using the Maximum Likelihood method with kappa values 97% and 94 % respectively. The reason for high accuracy may be to some extent attributed for the reason that the part of the training set is being considered as ground truths instead of actual data. Since the accuracy of the results depends only upon the test set chosen, the efficiency of any algorithm shall not be considered on the accuracy measure alone. Out of eleven classes considered for the sample image, many classes were found to be closely matching in both the methods. However, differences are observed in certain classes in both the methods. The classified images shall also be compared with the input image (FCC) and collecting ground truth information physically. From the comparison, it is found that both the methods are equally efficient, but the decision tree algorithm will have an edge over its statistical counterpart because of its simplicity, flexibility and computational efficiency.

REFERENCES

[1] Friedman, J. H., 1977. A Recursive Partitioning Decision Rule for Nonparametric Classification, IEEE Trans. Computers, Vol C-26, pp 404-408.

[2] Quinlan, J.R., 1993. C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, California, 302 p.

[3] Hunt, E.B, Marin, J., and Stone, P. J., 1966. Experiments in Induction, Academic Press.

[4] Morgan, J. N., and Sonquist, J. A, 1963. Problems in the Analysis of Survey Data and A Proposal, Journal of American Statistics Association, Vol. 58, pp 415-434.

[5] Morgan, J. N., and Messenger, R. C., 1973. THAID- A Sequential Search Program for the Analysis of Nominal Scale Dependent Variables, Institute for Social Research, University of Michigan, Ann Arbor.

[6] Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J., 1984. Classification and Regression Trees, Wadsworth International Group, Belmont, California, 358 p.

[7] Quinlan, J. R., 1975. Machine Learning, Vol. 1, No. 1, University of Sydney.

[8] Buntine, W., 1992. Tree Classification Software, Third National Technology Transfer Conference and Exposition, Baltimore, MD.

[9] Murthy, S.K., Kasif, S., and Salburg, S. 1994. A System for Induction of Oblique Decision Tree, Vol.2, pp. 1-32.

[10] Friedl, M.A., and Brodley, C.E., 1997. Decision tree classification of land cover from remotely sensed data, Remote Sensing of Environment, Vol. 61, pp 399–409.

[11] Mahesh Pal, and Paul M. Mathers, 2001. Decision Tree Based Classification of Remotely Sensed Data, 22nd Asian Conference on Remote Sensing.

[12] Mahesh Pal, and Paul M. Mathers, 2003. An assessment of the effectiveness of decision tree methods for land cover classification, Remote Sensing of Environment, Vol. 86, pp 554–565.

[13] Min Xu, Pakorn Watanachaturaporn, Pramod K. Varshney, and Manoj K. Arora, 2005. Decision tree regression for soft classification of remote sensing data, Remote Sensing of Environment, Vol. 97, No. 3, pp 322-336.

[14] Michael Zamban, Rick Lawrence, Andrew Bunn, and Scott Powell, 2006. Effect of Alternative Splitting Rules on Image Processing Using Classification Tree Analysis, Photogrammetric Engineering and Remote Sensing, Vol. 72, No. 1, January 2006, pp 25-30.

[15] Mahesh Pal, 2007. Ensemble Learning with Decision Tree for Remote Sensing Classification, World Academy of Science, Engineering and Technology, pp 258-260.

[16] Mingxiang Huang, Jianghua Gong, Zhou Shi, Chunbo Liu, and Lihui Zhang, 2007. Genetic algorithm-based decision tree classifier for remote sensing mapping with SPOT-5 data in the HongShiMao watershed of the loess plateau, China, Neural Computing and Applications archive, Vol. 16, No. 6 pp 513 – 517.

[17] Xingping Wen, Guangdao Hu and Xiaofeng Yang, 2008. CBERS-02 remote sensing data mining using decision tree algorithm, International Conference on Forensic Applications and Techniques in Telecommunications, Information, and Multimedia and Workshop, Article No. 58.

[18] Abdelhamid A. Elnaggar and Jay S. Noller, 2010. Application of Remote-sensing Data and Decision-Tree Analysis to Mapping Salt-Affected Soils over Large Areas, Remote Sensing, Vol. 2, pp 151-165.

[19] Cormen, T. H., Leiserson, C. E. and Rivest, R. L. 1989. Introduction to Algorithms, MIT Press, Cambridge, MA.

[20] Aho, A. V., Hopcroft, J. E., and Ullman, J. D., 1974. The Design and Analysis of Computer Algorithms, Addison-Wesley Publishing Company, Reading, MA.

[21] Apte, C., and Weiss, S., 1997. Data mining with decision trees and decision rules, Future Generation Computer Systems, Vol. 13, pp 197–210.

[22] Breiman, L., 1978. Description of Chlorine Tree Development and Use, Technical Report, Technology Service Corporation, Santa Monica, CA.

[23] De'ath, G., and Fabricius, K., 2000. Classification and regression trees: a powerful yet simple technique for ecological data analysis, Ecology, Vol. 81-11, pp 3178–3192.

[24] Venables, W.N., and B.D. Ripley, 1999. Modern Applied Statistics with S-Plus, Third Edition, Springer-Verlag, New York, 501 p.

AUTHORS PROFILE

Dr. M. K. Ghose is the Professor and Head of the Department of Computer Science & Engineering at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India since June, 2006. Prior to this, Dr. Ghose worked in the internationally reputed R & D organization ISRO – during 1981 to 1994 at Vikram Sarabhai Space Centre, ISRO, Trivandrum in the areas of Mission simulation and Quality & Reliability Analysis of ISRO Launch vehicles and Satellite systems and during 1995 to 2006 at Regional Remote Sensing Service Centre, ISRO, IIT Campus, Kharagpur in the areas of RS & GIS techniques for the natural resources management. His area of research interest are Data Mining, Simulation & Modeling, Network & Information Security, Optimization & Genetic Algorithm, Digital Image processing, Remote Sensing & GIS and Software Engineering. Dr. Ghose has conducted quite a number of Seminars, Workshop and Training programmes in the above areas and published 100 technical papers in various national and international journals in addition to presentation/ publication in several international/ national conferences. Presently 9 scholars are pursuing Ph.D work under his guidance and he is having 8 sponsored projects.

Mrs. Ratika Pradhan is the Associate Professor of the Department of Computer Science & Engineering at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India. She completed her Bachelor of Engineering in Computer Science and Engineering from Adhiparasakthi Engineering College, Madras University, Tamilnadu in the year 1998. She completed her Master of Engineering in Computer Science and Engineering from Jadavpur University in the year 2004. Her research interest is mainly in Digital Image Processing and Remote Sensing & GIS.

Ms. Sucheta Sushan Ghose is the Assitant Professor of the Department of Computer Science & Engineering at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India. She completed her Bachelor of Technology in Information Technology, Sikkim Manipal University, Sikkim in the year 2009.