Classification and Segmentation in Satellite Imagery Using Back Propagation Algorithm of ANN and K-Means Algorithm

P. Sathya and L. Malathi

***Abstract*—Now a days unsupervised image classification and segmentation increasingly popular. Existing information system classification tools used the same method for years. These basic classification methods do not provide satisfactory results when it applied on wide database of images.**

This paper describes the implementation of two algorithms, namely Back Propagation Algorithm of ANN and K-Means Algorithm on wide database of images. It provides the tool for segmentation and classification of remote sensing images. This classified image is given to K-Means Algorithm and Back Propagation Algorithm of ANN to calculate the density count. The density count is stored in database for future reference and for other applications. This tool also has the capability to show the comparison of the results of both the algorithms. High resolution basically means that an image is reproduced with a high level of detail. Usually it is referring to an image that is of very high quality, where there is a lot of detail.

***Index Terms*—ANN: Artificial Neural Network**

I. Introduction

Satellite image classification includes Segmentation and Classification. Images may be constructed by classes of image types or natural scene itself may have diverse structures or textures [1].

Segmentation refers to the process of partitioning a digital image into multiple segments. The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

Classification is an important task for all remote sensing applications, which partitions the images into homogenous regions, each of which corresponds to some particular landcover type. The problem of pixel classification is often posed as clustering in the intensity space [2]. Clustering [3] is a popular unsupervised pattern classification technique that partitions a set of n objects into K groups based on some similarity/dissimilarity metric where the value of K may or may not be known a priori.

In this paper, the satellite image segmentation and classification tool is done using K-Means and Back Propagation Algorithms.

A. *Satellite Image Segmentaton and Classification:*

Segmentation of a satellite image into differently textured regions' 'classes' is a difficult problem. Usually, one does not know a priori what types of textures exist in a satellite image, how many textures there are, and what regions have certain textures [4].

The monitoring task can be accomplished by supervised classification techniques, which have proven to be effective categorisation tools [5].

Unfortunately, these techniques require the availability of a suitable training set (classes' numbers for example) for each new image of the considered area to be classified. However, in real applications, it is not possible to rely on suitable ground truth information for each of the available images of the analysed site. Consequently, not all the satellite images acquired on the investigated area at different times can be used for updating the related land-cover maps. In this context, it would be important to develop classification methods capable of analysing the images of the considered site for which no training data would be available, thus increasing the effectiveness of monitoring systems based on the use of remote-sensing images.

Recently, researchers faced this problem by proposing an unsupervised retraining technique for maximum-likelihood (ML) classifiers capable of producing accurate land-cover maps even for images for which ground-truth information is not available [6]. This technique allows the unsupervised parameters' updating of an already trained classifier on the basis of the distribution of the new image to be classified.

B. *Unsupervised and supervised classification principle*

Classification principle could be described as follows: any individual pixel or spatially grouped sets of pixels representing some feature, class, or material is characterized by a (generally small) range of digital numbers for each band monitored by the remote sensor. The digital numbers values (determined by the radiance averaged over each spectral interval) are considered to be clustered sets of data in 2D plotting space. These are analyzed statistically to determine their degree of uniqueness in this spectral response space and some mathematical function(s) is/are chosen to discriminate the resulting clusters. Two methods for unsupervised and supervised classification are commonly used [5]: shown as Figure 1.

Manuscript received March 17, 2011, revised August 11, 2011.

P. Sathya is with II- ME (Computer Science and Engineering) Vivekanandha College of Engineering for WomanTiruchencode, Tamilnadu, India,(e-mai: sathyavenkat81@yahoo.com)

L. Malathi is with Lecturer (Computer Science and Engineering) Vivekanandha College of Engineering for WomanTiruchencode, Tamilnadu, India

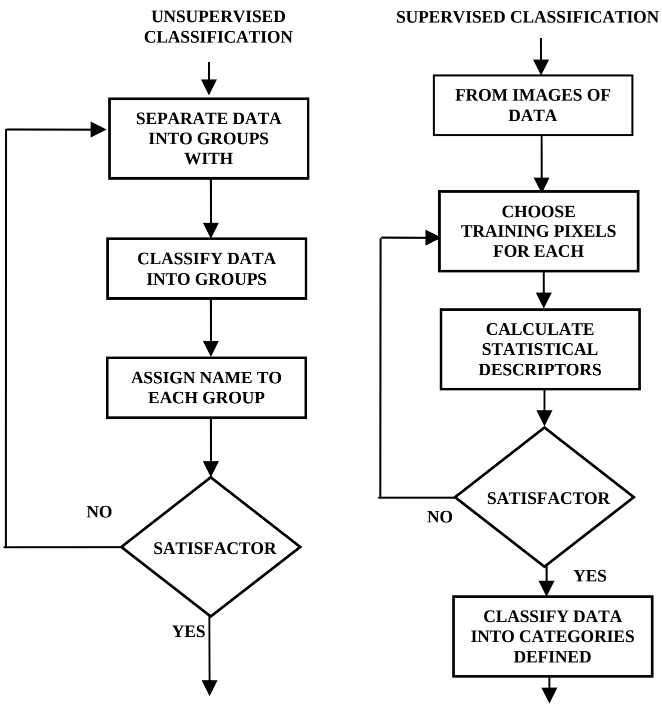


Fig.1. Unsupervised and supervised classification principle.

In supervised classification the interpreter knows beforehand what classes are present and where each is in one or more locations within the scene. These are located on the image, areas containing examples of the class are circumscribed (making them training sites), and the statistical analysis is performed on the multiband data for each such class. Instead of clusters then, one has class groupings with appropriate discriminant functions (it is possible that more than one class will have similar spectral values but unlikely when more than three bands are used because different classes/materials seldom have similar responses over a wide range of wavelengths). All pixels in the image lying outside training sites are then compared with the class discriminates, with each being assigned to the class it is closest to this makes a map of established classes.

In unsupervised classification, every individual pixel is compared to each discrete cluster to see which one it is closest to. A map of all pixels in the image, classified as to which cluster each pixel is most likely to belong, is produced (in black and white or more commonly in colors assigned to each cluster). This, then, must be interpreted by the user as to what the color patterns may mean in terms of classes, which are actually present in the real world scene; this requires some knowledge of the scene's feature/class/material content from general experience or personal familiarity with the area imaged.

The aim of the unsupervised classification methods is to find partitions of individuals set according to proximity criteria's of their attribute vectors in the representation space. The objective is in fact to group multiband spectral response patterns into clusters that are statistically separable.

Thus, a small range of digital numbers (DNs) say three bands, can establish one cluster that is set apart from a specified range combination for another cluster (and so forth).

Separation depends on the parameters we choose to differentiate. However, the unsupervised classification methods are used to do blind classification and so to achieve segmentation without priori knowledge of the image, but some important parameters like the class numbers must be fixed. Indeed, the majority of unsupervised classification algorithms need an initialization step on which parameters like the class numbers must be known. It exists different techniques for the estimation of this parameter.

II. .K-MEANS ALGORITHM

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group age is done.

At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an objective function, in this case a squared error function.

This algorithm starts with some clusters of pixels in the feature space, each of them defined by its center. The first step consists in allocating each pixel to the nearest cluster. In the second step the new centers are computed with the new clusters. These two steps are repeated until convergence [7], [8].The basic step of k-means clustering is simple. In the beginning determine number of clusters K and assume the centroid or center of these clusters. Take any random objects as the initial centroid or the first K objects in sequence can also serve as the initial centroid.

The K-means technique's takes pixels intensities as a basis. One randomly assigns each pixel to a class and one reiterates as follows: Centers of various groups (class) would be recalculated and each pixel would be again affected to the group according to its nearest center. Convergence would be reached when all centers would be fixed.

A. The K means algorithm will do the three steps below until convergence.

Iterate until stable (= no object move group)

a) Determine the centroid coordinate.
b) Determine the distance of each Object to the centroid.
c) Group the object based on minimum distance.

After implementation of K – means, the results of segmentation and classification will be stored and then same images will give to the neural network classifier for segmentation and classification and the result will be stored.

III. Back Propagation Algorithm Of ANN

The back propagation algorithm is a generalization of the least mean square algorithm that modifies network weights to minimize the mean squared error between the desired and actual outputs of the network.

The back propagation algorithm trains a given feed-forward multilayer neural network for a given set of input patterns with known classifications. When each entry of the sample set is presented to the network, the network examines its output response to the sample input pattern. The output response is then compared to the known and desired output and the error value is calculated. Based on the error, the connection weights are adjusted.

The flow diagram for land cover classification using ANN depicted as follows:

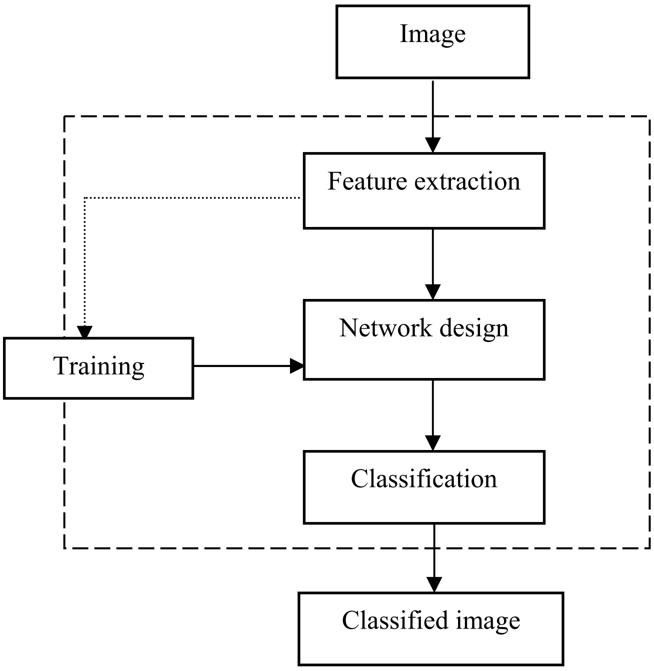


Fig.2.Flowchart For Land Cover Classification Using ANN.

Back propagation uses supervised learning in which the network is trained using data for which inputs as well as desired outputs are known. Once trained, the network weights are frozen and can be used to compute output values for new input samples.

The feed-forward multilayer neural network has been widely used in supervised image classification of remotely sensed data. A backpropagation Feed-forward multilayer network depicted as follows:

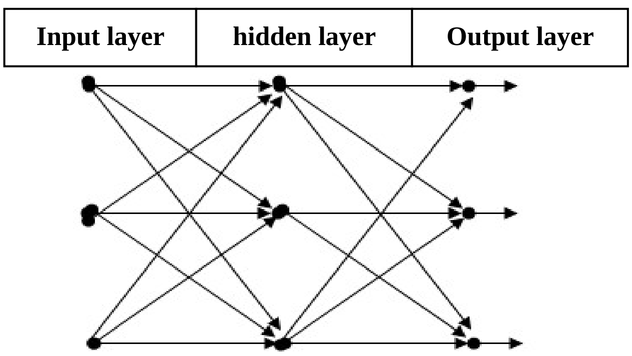


Fig.3. A back propagation feed-forward multilayer neural networks.

A. Back propagation Algorithm:

Start with randomly chosen weights.

While mean squared error (MSE) is unsatisfactory and computational bounds are not exceeded,

Do for each input pattern.

1. Compute hidden node inputs $\left(net_{p,j}^{(1)}\right)$
2. Compute hidden node outputs $\left(x_{p,j}^{(1)}\right)$
3. Compute inputs to the outputs nodes $\left(net_{p,j}^{(2)}\right)$
4. Compute the network outputs $\left(O_{p,k}\right)$
5. Compute the error between $O_{p,k}$ and desired output $d_{p,k}$
6. Modify the weights between hidden and output nodes:

$$\Delta W_{k,j}^{(2,1)} = \eta \,(d_{p,k} - O_{p,k})\ \delta'\left(net_{p,j}^{(2)}\right)\ x_{p,j}^{(1)}$$

7. Modify the weights between input and hidden nodes:

$$\Delta W_{k,j}^{(2,1)} = \eta \sum_k ((d_{p,k} - O_{p,k})\ \delta'\left(net_{p,j}^{(2)}\right) \Delta W_{k,j}^{(2,1)})\ \delta'\left(net_{p,j}^{(1)}\right)\ x_{p,i}$$

End-for

End-while

IV. Convertion from RGB TO YCbCr

A. RGB

The RGB colour model relates very closely to the way we perceive colour with the r, g and b receptors in our retinas. RGB uses additive colour mixing and is the basic colour model used in television or any other medium that projects colour with light. It is the basic colour model used in computers and for web graphics, but it cannot be used for print production.

The secondary colours of RGB – cyan, magenta, and yellow – are formed by mixing two of the primary colours (red, green or blue) and excluding the third colour. Red and green combine to make yellow, green and blue to make cyan, and blue and red form magenta. The combination of red, green, and blue in full intensity makes white. YCbCr or Y'CbCr is a family of color spaces used as a part of the Color image pipeline in video and digital photography systems. Y' is the luma component and Cb and Cr are the blue-difference and red-difference chroma components. The prime (') on the Y is to distinguish the luma from luminance, meaning that light intensity is non-linearly encoded using gamma.

Y'CbCr is not an absolute color space, it is a way of encoding RGB information. The actual color displayed depends on the actual RGB colorants used to display the signal. Therefore a value expressed as Y'CbCr is only predictable if standard RGB colorants or an ICC profile are used.

B. Rationale

Cathode ray tube displays are driven by red, green, and blue voltage signals, but these RGB signals are not efficient as a representation for storage and transmission, since they have a lot of mutual redundancy. YCbCr and Y'CbCr are a practical approximation to color processing and perceptual uniformity, where the Primary colors corresponding roughly to Red, Green and Blue are processed into perceptually meaningful information. By doing this, subsequent image/video processing, transmission and storage can do operations and introduce errors in perceptually meaningful ways.

Y'CbCr is used to separate out a luma signal (Y') that can be stored with high resolution or transmitted at high bandwidth, and two chroma components (Cb and Cr) that can be bandwidth-reduced, sub sampled, compressed, or otherwise treated separately for improved system efficiency.

C. Name

YCbCr is sometimes abbreviated to **YCC**. Y'CbCr is often called YPbPr when used for analog component video, although the term Y'CbCr is commonly used for both systems, with or without the prime.

Y'CbCr is often confused with the YUV color space, and typically the terms YCbCr and YUV are used interchangeably, leading to some confusion; when referring to signals in video or digital form, the term "YUV" mostly means "Y'CbCr".

V. Experimental Analysis

Color based segmentation and classification using K-MEANS and Back Propagation algorithms.

Step 1**:** Read the original image and display it

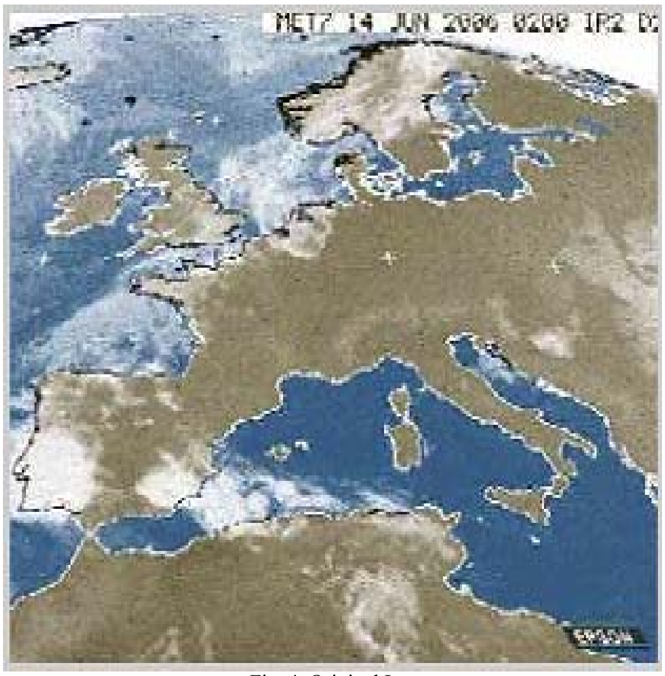


Fig. 4. Original Image

Step 2: Convert image from RGB color space YCBCRcolor space.

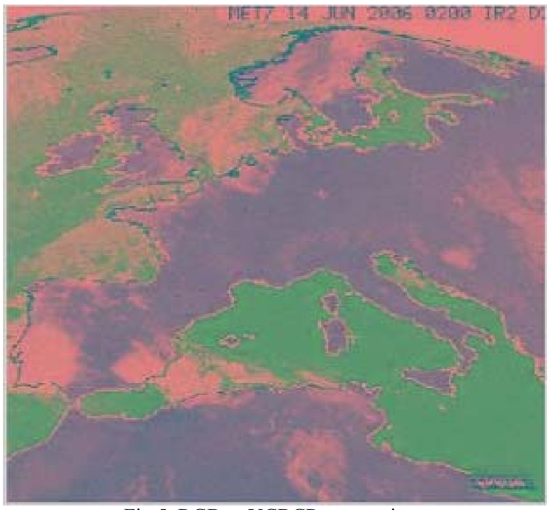


Fig.5. RGB to YCBCR conversion.

Step 3: Above image is given as an input for both the classifiers i.e. K-Means classifiers and Neural Network Classifiers. Hereafter we have compared the results obtained from both the classifiers.

Step 4: Label every pixel in the image using the results from K-Means. For every object in our input, K-means returns an index corresponding to a cluster. Label every pixel in the image with its cluster-index. Output obtained under K-Means Classifier is shown in Fig.3


Fig.6. Output from K-Means Classifier.

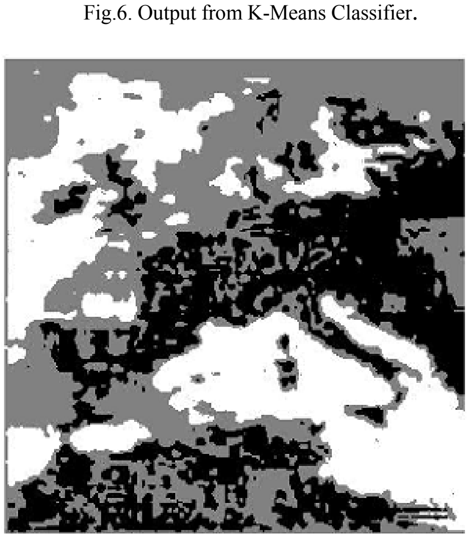
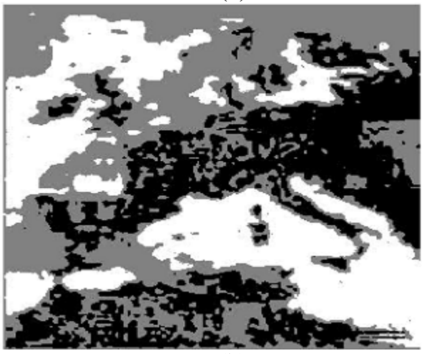


Fig. 7. Output from Back Propagation of ANN Classifier.

Step 5 : Comparison of two classifier's images.

(a)



(b)

Fig.8. Comparison of (a) K-Means Classifier and (b) Back Propagation of ANN Classifier.

Step 6 : Comparison of density count values of K-Means Classifier and Back Propagation of ANN Classifier.

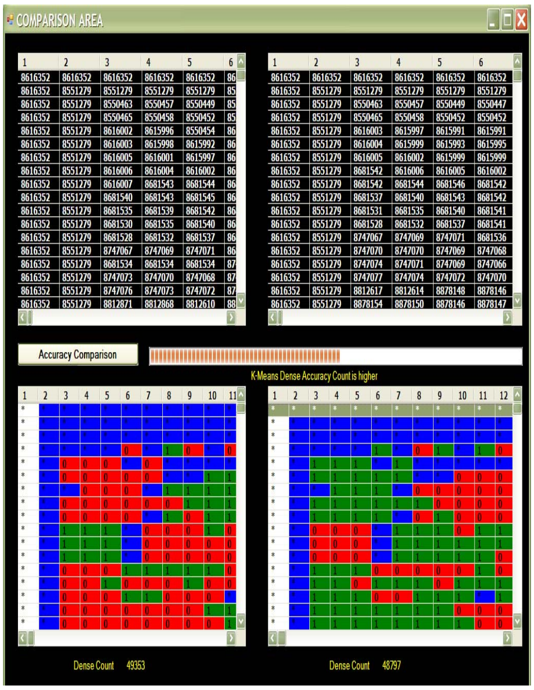


Fig.9. Comparison of dense count values of both classifiers

VI. Conclusions and Future Scope

K-Means algorithm and back propagation algorithm of ANN are implemented on various databases for the segmentation and classification purpose. This paper provides the tool for segmentation and classification of remote sensing images. This classified image is given to K-Means Algorithm and Back Propagation Algorithm of ANN to calculate the density count. The density count is stored in database for future reference and for other applications. This paper also has the capability to show the comparison of the results of both the algorithms.

It is found that K- means algorithm gives very high accuracy, but it is useful for single database at a time. Whereas neural network is useful for multiple databases, once it is trained for it. Neural network also provides good accuracy.

In future different neural network algorithms can be used to classify the satellite images and the classification results of those images will be compared with results of existing classification methods and also in future work, these classification results are used for different purposes. For example, based on dense count the weather conditions are identified and also various algorithms are used to identify the classification problems, object recognition, particularly to larger problems.

References

[1] V.P.Subramanyam Rallabandi and S.K.Sett, "Unsupervised Texture Classification and Segmentation", Proceedings of World Academy of Science, Engineering Technology, vol. 5, pp. 1307-6884, April 2005.

[2] U.Maulik and S. Bandyopadhyay, "Fuzzy partitioning using a real-coded variable-length genetic algorithm for pixel classification," IEEE Trans. Geoscience and Remote Sensing. vol. 41, no. 5, pp. 1075–1081, May 2003.

[3] U.Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," IEEE Transaction on Pattern Analysis Machine Intelligence., vol. 24, no. 12, pp. 1650–1654, Dec. 2002.

[4] T.R. Reed and J.M.H. Du Buf, "A review of recent texture segmentation, feature extraction techniques", CVGIP Image Understanding, vol. 57, pp. 359–372, 1993.

[5] J.A. Richards, "Remote Sensing Digital Image Analysis", second ed., Springer-Verlag, New York, 1993.

[6] L.Bruzzone and D.Fernandez Prieto, "Unsupervised retraining of a maximum-likehood classifier for the analysis of multitemporal remote-sensing images", IEEE Transactions on Geoscience and Remote Sensing, vol. 39 , pp.456-460, 2001.

[7] Xiao yang Tan, Song can Chen and Zhi-zhou, "Recognizing partially. Occluded Expression Variant Faces from Single Training Image per Person with SOM & soft K-NN Ensemble," IEEE Trans. Pattern Anal. Vol.16, No. k4, pp.875-855, July 2005.

[8] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman and Angela Y. Wu, "An Efficient k-Means Clustering Algorithm: Analysis and Implementation", IEEE Transaction on Pattern Analysis Machine Intelligence., vol.24, No.7, pp 881-892, July 2002.