

Performance Assessment of Neural Network and K-Nearest Neighbour Classification with Random Subwindows

M. Seetha, K. V. N. Sunitha, and G. Malini Devi

Abstract—Satellite image classification plays an important role in remote sensing, where information of an object or phenomenon is acquired from real time sensing devices, such as satellites and spacecrafts. In satellite image classification, the goal is to correctly classify vegetation, agriculture, water bodies, urban and open areas. The features like mean, Euclidean distance, RGB and slope values are extracted for each pixel in the input image. The image is classified using back propagation algorithm which reduces the misclassification that occurs in pixel based classification. This paper emphasizes on the classification of IRS 1-D LISS-III images using neural network, k- nearest neighbor and k- nearest neighbor with subwindows. An experimental comparison of neural network approach with back propagation algorithm was made with other considerations of the k-nearest neighbor and with subwindows. The results show that the k-nearest neighbor with subwindows has better overall accuracy and kappa coefficient when compared to neural networks.

Index Terms—Image classification, neural networks, remote sensing, spatial data, classification accuracy.

I. INTRODUCTION

Data collection using satellite remote sensing offers a variety of advantages compared to other forms of data acquisition, making it possible to measure energy (such as ultra-violet, infrared and microwave) at wavelengths that cannot be reached by human vision. It involves gathering information about the earth's surface using satellites orbiting around the earth. Through this, scientific advancements have resulted in a variety of discipline such as agriculture, forestry, hydrology, geology, cartography and meteorology. Satellite image classification is used to specify the vegetation, agriculture, water bodies, open area and urban lands present in a satellite image, which is used for crop acreage, groundwater identification, disaster management, etc. The neural network has been implemented to accurately perform image segmentation by using various methods to solve the problem of insufficient training sets [9]. A feed forward back propagation network was used for sky modeling and forecasting daylight availability for the tropical climate found in the central region of the north-eastern part of Thailand [6]. An experimental comparison of various approaches together with other considerations has made the k-nearest neighbor to have better accuracy when compared to neural networks [1].

Various learning paradigms are implemented to perform pixel-based classification while comparing them in terms of their accuracy using neural networks [13]. In this classification method, the satellite images are classified into different classes as vegetation, agriculture, water body, open area and urban land. This paper emphasizes on the classification of IRS 1-D LISS-III images using neural network (NN), k- nearest neighbor (KNN) and k- nearest neighbor with subwindows and its performance analysis using the classification parameters overall accuracy and kappa coefficient.

II. BACK PROPAGATION ALGORITHM

In this paper, features like mean, Euclidean distance, RGB, slope values are extracted for each pixel in the input image. Then, these extracted features are given to NN which is designed and trained by training data to classify each pixel as belonging to one of the output classes and a classified image is produced. The same training data is given to the other techniques like KNN to produce the classified outputs.

The BP algorithm is used in layered feed forward NN [11]. This means that the artificial neurons are organized in layers, and send their signals “forward”, and then the errors are propagated backwards. The network receives inputs by neurons in the input layer, and the output of the network is given by the neurons on an output layer. There may be one or more intermediate hidden layers. As mentioned earlier, BP algorithm uses supervised learning, which means that the algorithm is provided with the inputs and outputs that are to be computed by the network and then the error (difference between actual and expected results) is calculated. The idea of the BP algorithm is to reduce this error, until the NN learns the training data [14]. The training begins with random weights, and the goal is to adjust them so that the error will be minimal.

In order to train an NN to perform some task, the weights of each unit must be adjusted in such a way that the error between the desired output and the actual output is reduced. This process requires that the neural network compute the error derivative of the weights (EW). In other words, it must calculate how the error changes as each weight is increased or decreased slightly. The BP algorithm is the most widely used method for determining the EW [12].

The BP algorithm is the easiest to understand if all the units in the network are linear. The algorithm computes each EW by first computing the EA, the rate at which the error changes as the activity level of a unit is changed. For output units, the EA is simply the difference between the actual and the desired output. To compute the EA for a hidden unit in

Manuscript received July 20, revised September 11, 2012.

The authors are with the Dept. of CSE, GNITS, Hyderabad-8, India (e-mail: smaddala2000@yahoo.co.in; k.v.n.sunitha@gmail.com; malini_g12@yahoo.co.in).

the layer just before the output layer, first identify all the weights between that hidden unit and the output units to which it is connected. Then, multiply those weights by the EAs of those output units and add the products. This sum equals the EA for the chosen hidden unit. After calculating all the EAs in the hidden layer just before the output layer, in like fashion the EAs for other layers can be computed, moving from layer to layer in a direction opposite to the way activities propagate through the network. This is what gives BP its name. Once the EA has been computed for a unit, it is straightforward to compute the EW for each incoming connection of the unit. The EW is the product of the EA and the activity through the incoming connection [15].

III. K- NEAREST NEIGHBOR CLASSIFICATION

The KNN method is a very intuitive method that classifies unlabeled examples based on their similarity with examples in the training set [3]. It is a very intuitive method that classifies unlabeled examples based on their similarity with examples in the training set. KNN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification [7]. This algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. The same method can be used for regression, by simply assigning the property value for the object to be the average of the values of its k nearest neighbors [4]. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. A common weighting scheme is to give each neighbor a weight of $1/d$, where d is the distance to the neighbor. This scheme is a generalization of linear interpolation. The neighbors are taken from a set of objects for which the correct classification (or, in the case of regression, the value of the property) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required [15]. The KNN algorithm is sensitive to local structure of the data. KNN rules in effect compute the decision boundary in an implicit manner. In cases such as text classification, another metric such as the overlap metric (or Hamming distance can be used [2]. Another popular approach is to scale features by the mutual information of the training data with the training classes [3].

A. Distance Metric

The three kinds of distances Manhattan Distance (Man), Euclidean Distance (Euld), Minkowski Distance (Mkw) are considered for a KNN algorithm. The Manhattan distance between two items is the sum of the differences of their corresponding components [3].

B. Performance Assessment

Assessing accuracy of a remote sensing output is one of the most important steps in any classification exercise.

Without accuracy assessment, the output or results is of little value [10]. A user of the imagery who is particularly interested in class A, say, might wish to know what proportion of pixels assigned to class A were correctly assigned [10]. Producer's accuracy is a measure of how much of the land in each category was classified correctly [10]. Different people have different interpretations as to what is a good level of agreement [5].

IV. RANDOM SUBWINDOWS

The images are classified using randomly extracted subwindows that are suitably normalized to yield robustness to certain image transformations [8]. In this process, subwindows of sizes 4X4, 8X8, 12X12 and 16X16 are extracted from the training images. As the size of the subwindow increases, the amount of data overlapping increases and the same pixels are considered more than once. The pixels of these windows form the training data for NN and KNN. Finally, accuracy with respect to the various sizes of subwindows are computed and compared.

V. RESULTS AND DISCUSSIONS

When a LISS-III satellite image was given as input to each pixel-based classifier, classified images were produced. These images were dependent on the type of classifier. The producer's, users and overall accuracy along with the kappa coefficient are displayed in the form of a table to facilitate comparison of the various techniques.

The original image for which pixel-based classification is performed is shown in Fig. 1.

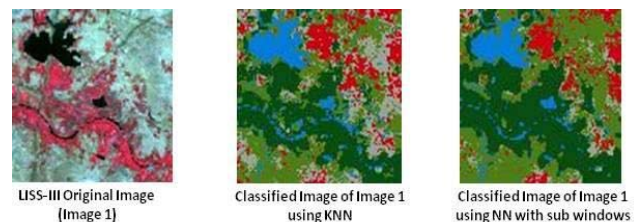


Fig. 1. Original and classified images

The maximum accuracy of 72.29 was obtained for training data of 3000 pixels (Table I). Thus as the training data presented to the neural network increases, the classification results are improved.

TABLE I: ACCURACY OF IMAGE CLASSIFICATION USING NN BASED ON TRAINING DATA DERIVED FROM 3 IMAGES

	Trg data (750p)	Trg data (1500 p)	Trg data (2250 p)	Trg data (3000 p)
OA	42.55	65.87	70.83	72.29

TABLE II: ACCURACY OF IMAGE CLASSIFICATION USING KNN FOR DIFFERENT K VALUES (USING EUCLIDEAN DISTANCE)

	K=5	K=10	K=15	K=20	K=25
Overall Accuracy	79.09	80.66	80.78	80.85	80.58

When the KNN was implemented by varying the k-values, it was observed that the maximum accuracy of 80.85 was obtained at k=20 (Table II).

TABLE III: COMPARISON OF ACCURACY OF IMAGE CLASSIFICATION USING NN AND KNN

	NN	KNN (Man)	KNN (Euld)	KNN (Mkw k=3)	KNN (Mkw k=4)	KNN (Mkw k=5)
OA	70.90	80.19	80.24	80.36	80.35	80.24

The training data that is presented to both NN and KNN is same, where KNN is implemented using various distance metrics. Minkowski distance with $k=3$ gave superior results, that is maximum accuracy of 80.36 compared to Manhattan and Euclidean distances (Table III). The image shown in Fig. 1 is noisy, as it doesn't have all the classes. NN is sensitive to noise but KNN is robust in this aspect. Hence, the accuracy obtained from KNN is more when compared to NN.

TABLE IV: ACCURACY OF IMAGE CLASSIFICATION USING NN AND KNN WITH AND WITHOUT RANDOM SUBWINDOWS

	NN	KNN (Euld ,k=20)
Without subwin	70.9	80.24
With subwin(4*4)	64.65	79.58
With subwin(6*6)	70.63	80.05
With subwin(12*12)	71.18	80.33
With subwin(16*16)	72.73	80.6

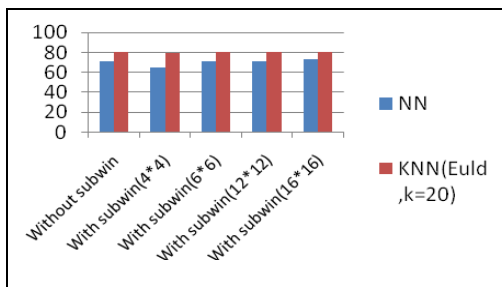


Fig. 2. Overall accuracy of Image 1 using NN and KNN with and without Random Subwindows

As the size of the subwindows which are extracted randomly from the training images increases, the amount of overlapped data increases and hence, the maximum accuracy of 72.73 for NN and 80.60 for KNN were obtained (Table IV and Fig. 2).

Image	KNN		NN		NN with subwins	
	OA	KC	OA	KC	OA	KC
1	61.77	0.47	65.65	0.51	89.16	0.81
2	53.76	0.39	59.11	0.42	74.87	0.69
3	57.88	0.34	58.41	0.30	65.9	0.52
4	60.15	0.45	74.32	0.60	85.76	0.80
5	69.22	0.59	70.36	0.63	72.73	0.62
6	44.83	0.27	52.97	0.30	67.55	0.59
7	60.86	0.45	71.85	0.55	86.41	0.80
8	73.12	0.65	66.00	0.52	81.64	0.76
9	72.6	0.64	57.58	0.44	89.11	0.82
10	60.01	0.49	57.32	0.40	92.61	0.85

The overall accuracy and kappa coefficient obtained by

various techniques for all 10 input images are evaluated (Table V). Neural network with subwindows gave superior results with accuracy of 92.61 for image 10, as some of the pixels were considered more than once in the training data.

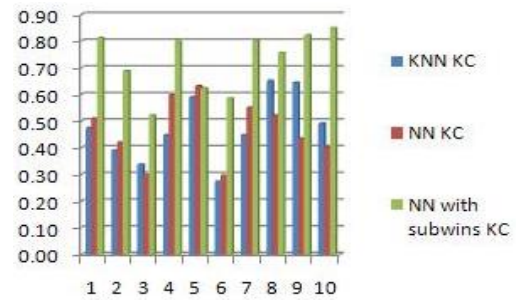


Fig. 3. Kappa Coefficient for the 10 images using KNN and NN with and without Random Subwindows

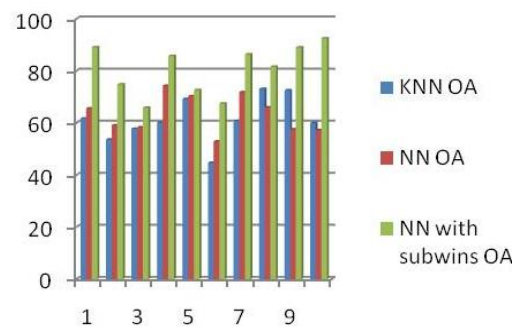


Fig. 4. Overall Accuracy for the 10 images using KNN and NN with and without Random Subwindows

It was ascertained that the overall accuracy and the kappa coefficient are more for all the images when the neural network is accompanied with random subwindows.

VI. CONCLUSIONS

This paper elucidates the potentials of back propagation neural network for performing pixel-based image classification. For this purpose, LISS-III satellite images were used. The main advantage of a k-nearest neighbor over neural networks is it yields smoother decision regions, provides probabilistic information. As a result, the learning capacity of neural network was much improved than before. Thus, it is ascertained that neural network with random subwindows yielded superior results than all the other classification techniques. The pixel-based classification module of the paper can be further extended by implementing several other techniques for image classification like combination of NN and KNN, weighted KNN, Genetic K-means and Neuro Fuzzy algorithms. ID3 algorithm of the DT classifier can be implemented and its results can be compared with those that are obtained from See5 algorithm.

REFERENCES

- [1] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, New York: John Wiley & Sons, 1973.
- [2] Lecture 8: The K Nearest Neighbor Rule." [Online]. Available: http://courses.cs.tamu.edu/rgutier/cs790_w02/l8.pdf.
- [3] Manhattan. Improved outcomes Software. [Online]. Available: http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Clustering_Parameters/Manhattan_Distance_Metric.htm.

- [4] Overview of K-Means Clustering. 16 Sep. 1999. [Online]. Available: <http://www.ece.neu.edu/groups/rpl/papers/kmeans>.
- [5] Simon, Steve. 14 Jul. 2008. [Online]. Available: <http://www.childrensmemory.org/stats/definitions/kappa.htm>.
- [6] S. Pattanasethanon, C. Lertsatitthanakorn, S. Atthajariyakul, and S. Soponronnarit. "An accuracy assessment of an empirical sine model, a novel sine model and an artificial neural network model for forecasting illuminance/irradiance on horizontal plane of all sky types at Mahasarakham, Thailand." 25 Feb. 2008.
- [7] Teknomo, Kardi. *How the K-Mean Clustering Algorithm Works*. Kardi Teknomo's Page. [Online]. Available: <http://people.revoledu.com/kardi/tutorial/kMean/Algorithm>.
- [8] Jakomulska, A. and K. C. Clarke. "Variogram-derived Measures of Textural Image Classification." *geoENV III* 2000.
- [9] Accuracy Assessment, [Online]. Available: [http://www.biology.ualberta.ca/facilities/gis/uploads/instructions/AV/ErrorMatrix.pdf](http://www.biology.ualberta.ca/facilities/gis/uploads/instructions/AV>ErrorMatrix.pdf).
- [10] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing. Exploration of the Microstructure of Cognition*, vol.1: *Foundations*, D. E. Rumelhart and J. L. McClelland, Eds. Cambridge, MA: MIT Press, 1986.
- [11] B. Tso and P. M. Mather "Classification Methods for Remotely Sensed Data," May 2001.
- [12] Weiyang Zhou. "Verification of the Nonparametric Characteristics of Back Propagation Neural Networks for Image Classification," *IEEE*, 1999.
- [13] G. Rogova, "Combining the results of several neural network classifiers," *Neural Networks*, vol. 7(5): pp. 777--781, 1994.
- [14] K. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol.2, pp.183-192, 1989.



Dr. M. Seetha had completed Ph.D in Computer Science and Engineering in the area of image processing in December 2007 from Jawaharlal Nehru Technological University, Hyderabad, India. She has teaching experience of 18 years in various engineering colleges. She is guiding 10 Ph.D scholars and her research interest includes image processing, neural networks, computer networks and data mining. She had published more than 50 papers in refereed journals and in the proceedings of National/International Conferences and Symposiums.



Dr K. V. N. Sunitha did her B.Tech ECE from Nagarjuna University, M.Tech Computer Science from REC Warangal and Ph.D from JNTUH in 2006. She has 21 years of Teaching Experience, working as HOD, CSE Dept in G.N.I.T.S from the inception of the CSE Dept since 2001. She is a recipient of Academic Excellence award by GNITS in 2005. She has received "Best computer Science Engineering Teacher award" by ISTE (Indian society for Technical education) in Feb 2008. She is guiding 12 PhDs & published more than 65 papers in International & National Journals and conferences. She has authored three text books. She has received funding for her research proposal from CSI Computer society of India. She is a fellow of Institution of Engineers, Sr.Member for IEEE & Life member for CSI.