



Comparative Analysis of Supervised Classification Algorithms for Urban Sprawl Detection

N. MINALLAH, W. KHAN, A. RASHID, M. N. KHAN, N.AZIZ, M. UZAIR, S. YOUSAF

National Center for Big Data and Cloud Computing (NCBC), University of Engineering and Technology
Peshawar, Pakistan

Received 4th October 2018 and Revised 29th May 2019

Abstract:-As Human population is increasing, a number of small towns are turning into big cities. But human race has developed itself in technological terms smartly, which is helping human kind to act efficiently and consume the resources appropriately. Collection of urban sprawl statistics of an area has become efficient by using remote sensing. In comparison to the traditional methods, the new method of using remote sensing for the detection and classification of urban sprawl has substantially enhanced. Using this data management has become capable of taking suitable measures for its residents. This work compares six supervised classifiers, i.e Maximum Likelihood, Minimum Distance, Support Vector Machines, Mahalanobis, Parallelepiped and Feed Forward Neural Network for urban classification. The data used is of SPOT 5 and criteria for comparison of classifiers is based on accuracy. Due to the absence of blue band in SPOT imagery the collected data samples for training tend to be complex and overlapping. Training data collected and divided into 10 different samples, show 82.74% accuracy for Artificial Neural Network. In comparison with Artificial Neural Networks, the lowest recorded results are of parallelepiped Classifier.

Keywords: Machine Learning, Maximum Likelihood, Minimum Distance, Support Vector Machines, Mahalanobis, Parallelepiped and Feed Forward Neural Network, Remote Sensing, Urban Sprawl detection, SPOT-5

1. INTRODUCTION

The world is changing rapidly and in the past few decades there has been a considerable increase in urban area. Due to the increasing population, urban Planning and Management has become appalling task. For observing and monitoring the urban sprawl, one of the most important and basic tool is remote sensing, so that further actions and planning can be carried out. This work presents a comparison of state of the art supervised classification techniques for Urban Sprawl Change detection. For experimentation the chosen study area is the region of Hayatabad, Peshawar, Pakistan. SPOT-5 Satellite Data for year 2015 has been acquired from SUPARCO, Pakistan.

The rest of the paper resides as: Section 2 discuss study area considered for research. Section 3 describes the data acquired from SUPARCO Pakistan. Section 4 is related to Experimental Setup, following by it Section 5 describes parameters setting for classifiers. While Section 6 discusses impact of classifiers. Section 7 is for Results and Discussions and last section (Section 8) which discusses conclusion of our work.

2. STUDY AREA

The test site for this work is Hayatabad, which is an urban settlement in the valley of Peshawar. Peshawar is the provincial capital of Khyber Pakhtunkhwa (KP), Pakistan, residing at 34.0151° N and 71.5249° E. The

Surroundings of Peshawar is mountainous regions by North, South and West.

3. DATA DESCRIPTION

The satellite data used in this work is SPOT (Satellite Pour l' Observation de la Terre), a series of 7 satellites launched by CNES (French Space Agency). SPOT 5 level 1A, August data of 2014 Hayatabad region has been acquired from SUPARCO (Pakistan Space and Upper Atmosphere Research Commission). (Fig.1 Fig.1) shows the experimental area of Hayatabad. The details of SPOT 5 satellite Images listed in (Table-1).

4. EXPERIMENTAL SETUP

The basic methodology adopted for completion of this work is mentioned in this section of manuscript, which includes preprocessing of the acquired satellite data, collection of training data for classifiers, details of training data used in the study, classification algorithms and methods used for generating results.

Band	Band Width Range(nm)	Code	Maximum Resolution(m)
B1	0.5-0.59	Green	10.0
B2	0.61-0.68	Red	10.0
B3	0.78-0.89	NIR	10.0
SWIR	1.58-1.75	SWIR	20.0

Table 1 SPOT Image Band Characteristics

⁺⁺Corresponding Authors: E-mail: n.minallah@uetpeshawar.edu.pk

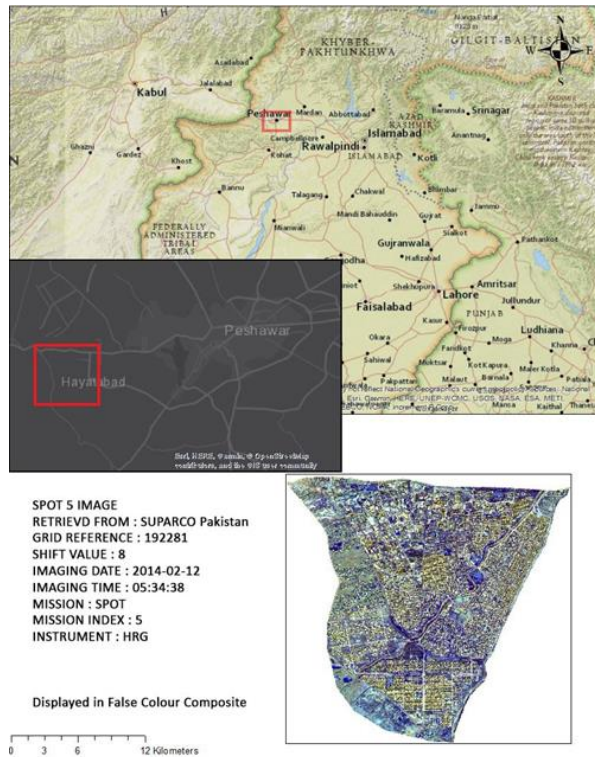
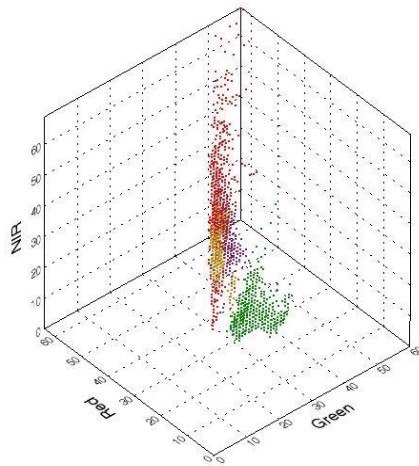


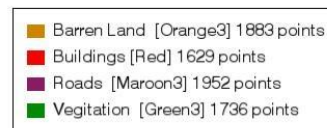
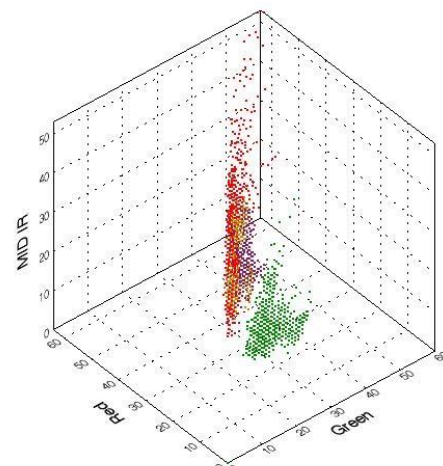
Fig.1 Area of Experimentation

4.1 Pre-Processing

The image that is acquired is contrast stretched by enhancing intensity values of the image. The data has also been georeferenced according to GCS WGS 1984.



(a) Red (x-axis), Green (y-axis) and Near infrared (NIR) (z-axis)



(b) Red (x-axis), Green (y-axis) and MID Near infrared (MIR) (z-axis)

Fig. 2 Representation of Scatter Plot for Training data, in 3d

4.2 Training Data for Classifiers

In our work we have selected 4 class training data to train our classifiers. These include Vegetation, Roads, Buildings and Barren land. Pixel based statistics of these classes are shown in (Table 2). Training data statistics below. Total number of 10 training sample datasets are taken as shown in the (Table 2). These datasets are created in a descending manner, showing percentages of the total ground truth data with the regular intervals of 10%. 3D scattered plot of 100% training dataset is illustrated in (Fig.2). Furthermore collected training or ground truth data means of Barren Land, Buildings, Roads and Vegetation are represented in (Fig.3).

4.3 Classification

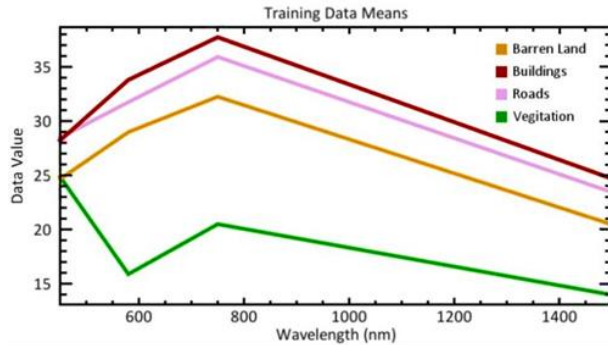
Machine learning has enabled us to process complex data, and provide valuable information in terms of defined classes. We compared the following supervised classifiers in our study;

- Minimum Distance
- Support Vector Machines (SVM)
- Feed Forward Neural Networks (FFNN)
- Mahalanobis Distance Classification
- Parallelepiped Classification

Using the collected training data, we first train the above mentioned classifiers and then test these with validation data. Mentioned classifiers are tested on the basis of their accuracies discussed in results and discussion section.

Table 2 Training data statistics

Classes	Pixels									
	100%	90%	80%	70%	60%	50%	40%	30%	20%	10%
Vegetation	1,736	1,562	1,389	1,215	1,042	868	694	521	347	174
Roads	1,952	1,757	1,562	1,366	1,171	976	781	586	390	195
Buildings	1,629	1,466	1,303	1,140	977	815	652	489	326	163
Barren Land	1,883	1,695	1,506	1,318	1,130	942	753	565	377	188

**Fig 3. Training Data Means.** X-axis represents wavelengths of SPOT bands and y-axis illustrates the DN (Digital)

Maximum Likelihood

Maximum likelihood classification (MLC) is one of the most widely used supervised classifier. MLC is a standout amongst the most famous strategies for image classification in remote sensing. Each pixel in MLC with the most extreme probability is characterized into the relating class. Every pixel is grouped to the class that has the most maximum likelihood (that is, the greatest probability). In the event that maximum probability is smaller than a threshold value you indicate, the pixel stays unclassified. The below equation represents a discriminant function (Pushpendra *et al.*, 2014) of MLC.

$$G_i(X) = \ln P(\omega_i) - 0.5 \ln \left| \sum_i X \right| - 0.5 S(M_i^T) \sum_i^{-1} S(M_i)$$

Where:

i = ith class

X = n (number of bands)-dimensional data

P(ω_i) = Probability of class ω_i and is assumed same for all classes

| L_i | = Determinant of the covariance matrix of the data in class ω_i

L⁻¹ = Inverse of |L|

S(M_i) = X - M_i

M_i = Mean vector

MLC has variety of applications .It is based on Bayes classification (John and Richards. 1999) and Mean vector, covariance metrics are the Key component of MLC that can be regained from training data (Pushpendra *et al.*, 2014). The algorithm depends upon the statistical model of Gaussian probability density function (Fabio *et al.*, 1999). The shortcoming of MLC process is that it cannot be used straightaway when the

number of features is greater than the number of training samples. Thus, the applicability of nearest neighbor or MLC can be restricted when there is minimum number of training samples and high-dimensional feature space of hyperspectral dataset (Chenghai *et al.*, 2009) But comparatively MLC produces nearly better or similar classification accuracy than others classifiers.

Minimum Distance

The Minimum Distance classification is one of the commonly used and computationally simple algorithm having equivalent classification accuracy to that of Maximum Likelihood approach. Maximum Likelihood approach is considered as one of the accurate classifier at the cost of intensive computations. The operation performed using the Minimum Distance classifier is very simple. The only thing it requires from the user is the training samples to be used in classification. The distance between the unknown vector and the mean vector for class i is calculated using Euclidean Distance formula given in equation below (Russell and Green2008).

$$d(m_i, X) = \sqrt{\sum_{i=1}^m (m_i - X)^2}$$

Where m_i is dimensional mean vector for class i and X is n dimensional unknown vector. In matrix form

$$d(X, m_i) = \sqrt{(m_i - X_i)^T - (m_i - X_i)}$$

Where T is the transpose of the Matrix.

A threshold value has to be defined for the classifier so that it can exclude the pixels having lower value than the threshold.

Mahalanobis Distance

Generally, in multivariate analysis, variables (usually two in number) are represented through coordinate system in a Euclidean space (Li *et al.*, 2018). But it is difficult to represent as well measure the variables when greater than two, along the planar coordinates. This is where the Mahalanobis distance (MD) comes into picture (Surekha *et al.*, 2019) It provides a powerful method for measuring the similarity of one set of conditions to ideal set of conditions, and prove useful while comparing the similarity of a test data to training data samples being taken as true data.

The quantitative similarity of entire landscape to ideal conditions is described using Mahalanobis distance (Umair *et al.*, 2015). Moreover, Mahalanobis distances take benefit of covariance among variables, as it depends on the mean and variance of the predicting variables, as well the covariance matrix of all other variables (Sahu, *et al.*, 2015). An ellipse in 2D space (i.e. when only 2 variables are measured), or an ellipsoid or hyper ellipsoid when more variables are used is formed around the mean of the constant Mahalanobis Distance (Rahul *et al.*, 2018).

The MD is a measure of the distance between a point P and a distribution D, introduced by P. C. Mahalanobis in 1936. It is a generalize idea in multi-dimension to measure a point P is how many standard deviations away from the mean of data distribution D. If the point P lies at the mean of Distribution D, the distance is zero, and the distance grows along each principle component axis as P moves away from the mean of D. If each of these principle component axes is re-scaled to have unit variance, then in transformed space the Mahalanobis distance correspond to standard Euclidean distance. Thus the Mahalanobis distance is scale-invariant and unit less, and consider the correlations of the data set (Ajay *et al.*, 2017).

In order to classify a test pixel that belongs to one of N classes, the covariance matrix of each class is usually based on known samples belong to each class (Maryam 2019) Then, for a given test sample, Mahalanobis distance to each class is computed, and for a class for which it is minimum, it classifies the test point to that class (Najam *et al.*, 2016).

Mahalanobis Distance for each pixel to each class in the image as calculated as follow:

$$D^2 = (x - m)^T C^{-1} (x - m)$$

Where:

D2 = Mahalanobis distance

x = Data Vector

m = independent variables Vector of mean values

C⁻¹ = Independent variables Inverse covariance matrix

T = Indicates vector should be transposed.

The following properties are exhibited by Mahalanobis distance Classifier:

- It takes into consideration the difference of variances in each direction.
- It also accounts for the covariance among variables.
 - For uncorrelated variables with unit variance it is reduced to the familiar Euclidean distance (Ajay *et al.*, 2019).

Mahalanobis distance classifier demonstrated to be useful in cases when the axes are correlated in feature

space (Vignajeeth *et al.*, 2018). It finds the center of the sample data and the standard deviation of the sample points from this central value of data. If a test pixel having one standard deviation distance from this center of the data than this test pixel is classified into that class. The further away a pixel is from center of data, higher the chances of pixel not belonging to a class and is placed unclassified.

The shortcoming of this classifier is that it uses statistics for each class. All pixels are characterized to the closest class unless a separation limit is determined.

Support Vector Machine

Proposed nearly 50 years ago, SVM, a binary classifier, has become a proprietary choice for researchers in data analytics. SVM is a kernel based classifier where the training data is first scaled, upon which a discriminant function is applied (Giorgos *et al.*, 2011). A hyper-line is drawn dividing the designated classes according to the feature set provided (Shih-Wei Lin, *et al.*, 2008). The multi classification in SVM is performed in one against one technique (David *et al.*, 2015). Using a designated kernel function, decision boundary called hyper-plane, is created, using training data. (Fig.4) presents the concept of separating hyper-plane in SVM (Varma, *et al.*, 2016). The bold line in is defined by;

$$w \cdot x + b = 0$$

Where b and w are the parameter of the hyper-plane. Vectors which are not in hyper plane are presented by;

$$w \cdot x + b > 0$$

Whereas the two margins or support vectors are presented by;

$$w \cdot x + b = -1 \text{ and } w \cdot x + b = +1$$

Where w is the vector product of two vectors.

This kernel function is based on four characteristics.

1. Linear basis function

$$K(x, y) = x^T y$$

2. Radial basis function

$$k(x, y) = \exp\left\{-\frac{\|x-y\|^2}{\sigma^2}\right\}$$

3. Polynomial degree k

$$k(x, y) = \left(1 + \frac{x^T y}{\sigma^2}\right)^2$$

4. Sigmoid

$$K(x, y) = \left\{-\frac{\sum_{i=1}^N (x-y)^2}{\sigma^2}\right\}$$

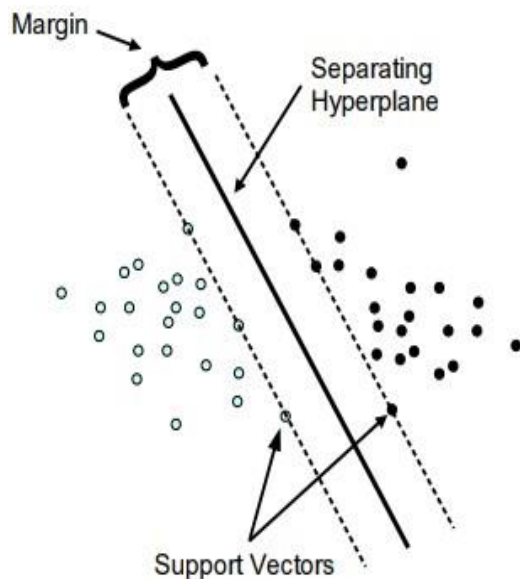


Fig.4 Representation of hyper-plane in kernel function

Neural Net

The motivation behind neural networks is the brain neural system used to perform different decision-making tasks. In brain there are different sections for a task (SitiKhairunniza-Bejo, *et al.*, 2014). Artificial neurons are the building blocks of artificial neural networks. At least three layers are required for construction of Artificial Neural Networks ANN's. First layer is input layer, that distributes input to PE's middle layer is hidden layer that consists of number of Processing Elements PE's the number of PE's is dependent on complexity of problem and the last layer is output layer (Marchantand and Onyango. 2003).

ANNs is usually helpful in the process of recognition (Konstantinos *et al.*, 2019), classification, enhancement (Senthil and Kumar. 2018) analysis (Jean *et al.*, 2008) estimation and prediction (Amin *et al.*, 2016). Having no evident sets of parameters that require estimation, ANN is referred to as non-parametric.

Generally, all of the NN's are trained first, and then compares incoming image pixel values with the available region of interest classes by checking on which side of the linear separating surface they lie (Amin *et al.*, 2011). The necessary processing element model is based on input from the previous layer and weight. An individual neuron or processing element takes a vector of inputs $x = x_1, x_2, \dots, x_N$. w is connection weight is a bias.

Feed Forward neural Network

The feed forward neural network was the simplest type of artificial neural network devised (Edwin and Zagajewski. 2017). A feed forward neural network is also known as multilayer Perceptron's (MLPs) is an

artificial neural network (Olivier *et al.*, 1993). The goal and function of feed forward neural network with n perceptron that we'd like to use to train to solve some problem is to approximate for a classifier. The process of learning take place by the adjustment of node weights so that to minimize the difference between output and node activation output. Approximate mapping of an input pixel in to given set of categories based on the calculated sum of the products of the weights $L_i w_i x$ and the inputs that is calculated at each node. Each pixel is processed to make a class selection for each pixel.

The network is to learn weights and biases so that the output from the network correctly classifies the digit. Feed-forward networks have the following characteristics:

- Processing elements are arranged in layers, input data travels via first input layer and the output layer producing outputs. The intermediate layers have no connection with the external world, and hence are called hidden layers
- In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes
- FFNN is acyclic means that within the network there is no feedback connections among neurons wx to send back the information (Giles 2017).

There are connections, where PE h_1 is connected to input x_1, x_2 and x_3 and h_2 is connected to input x_1, x_2 and x_3 and so here all the weights are given in matrix equation below. The PE computes matrix product of the hidden with those weights plus its own bias then passes to the activation function as well, matrix computation is given.

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} w_{11} * x_1 & w_{12} * x_2 & w_{13} * x_3 \\ w_{21} * x_1 & w_{22} * x_2 & w_{23} * x_3 \end{bmatrix}$$

The math that is needed to process here in the hidden node is the weighted sum of each input multiply by the column of input given together as H_i . The mathematical computation model for neuron is given in equations below, that is the computation that will take place on each artificial neuron. We do all this computation to figure out how to tune all this system to match the output to what we think it should be. Small alteration in weight cause insignificant corresponding change in the out put from neural network, this property will make learning possible. But as it occurs a small alteration in weights leads to a big change in output.

$$H_i = \sigma (w_i j * I_i + B_i)$$

$$O = \sigma (w * H + B)$$

Parallelepiped

Parallelepiped is a supervised statistical classifier that classify multispectral data using simple decision rule (Ana *et al.*, 2018). It categorize a pixel to a class while using the class limit and signature stored in it. A pixel is classified and assigned to a class, if its value lies between the upper and lower threshold value for all n bands being classified, for that particular class. While pixels that do not fall within any of the parallelepiped classes are designated as unclassified (Mohammed *et al.*, 2018). It takes into account the variance in training classes, so is computationally simple but due to correlation amongst classes problems may arise from parallelepiped overlap. Which results in large number of classified pixels as tied and result in poor accuracy. Being a fast classifier having simple decision rules parallelepiped classifier is usually used when there is a need for speed (John and Richards 1999).

In this classifier, a pixel membership is tested through multidimensional boxes which are constructed for each class using class mean and standard deviation. A pixel X vector is allocated to class j if the following equation is satisfied:

$$\mu_{ij} - K\sigma_{ij}X_{ij} \leq K\sigma_{ij}$$

Where μ_{ij} is mean value for class j in band i , K is a constant, and σ_{ij} is standard deviation for class j in band i . and this hold true for all $i=1$ to n , where n is number of spectral bands,

5. RESULTS AND DISCUSSION

This section discusses performance of classifiers, based on its accuracies.

Impact of NN

In this case an accuracy of 82.74% was achieved by using 100% data as training. While an accuracy of 63.68% was attained using 20% training data.

Kappa coefficient a quantitative measure is used to assess the classification accuracy and get the reliable accuracy estimation. The analysis of an individual region of interest ROI class or several classes for classified images will contain an amount of errors. This helps us to detect the sources of errors and find a way to improve the classification accuracy.

Impact of SVM

SVM is a state of the art classifier, used in many domains throughout research and development. The basic functionality of SVM depends upon the kernel function used in the computation of boundaries. This kernel function is setup after observation of ground truth or training data of the classifier. Different kernels viz linear, sigmoid, polynomial degree and radial basis function has been applied during our testing of SVM. Due to the complexity of our training data illustrated in

(Fig 3), we chose polynomial degree 5, based kernel with gamma being 0.4 and the probability of error to be 0.9. Testing upon these setting gave us good results in terms of accuracy for 4 different classes.

The accuracy 77.201 was achieved by using 40% of data as training and 81.8586% was attained for 100% training data as shown in (Table 3).

Impact of Minimum Distance

Minimum distance provides an accuracy of 62.7692 for 60% training data while 65.1332% for 10% of training data. Using minimum distance classifier the results we achieve are ambiguous and unreliable.

Impact of Mahalanobis

For Mahalanobis Distance multiple threshold values are used for each class for correct classification of all pixels to their respective classes. These threshold values are selected from the statistics of each class of all bands. The Maximum standard deviation values in all bands for each class is picked as threshold for that class. Pixel that lie at a distance greater than this threshold value are labeled as unclassified. The Classifier has highest probability of detection when threshold value is 0, as the threshold increases, so as the number of unclassified pixels and the probability of Detection decreases.

This behavior of Mahalanobis classifier indicates that our classification on the retrieved ground truth data is not up to the mark. An accuracy of 70.051 was attained after training the classifier with 70% data and 66.74% accuracy was recorded for 10% training data depicted in (Table 3).

Impact of Parallelepiped

For Parallelepiped Classification different threshold for each class are taken into account based on the maximum standard deviation of each class for all the bands.

As parallelepiped classifier work best in the areas where there is minimum overlap. And as the classes in ground truth imagery overlap in the underlying problem so the results of the said classifier is poor.

Accuracy and Kappa Coefficient

(Table 3 -4) shows results of Mahalanobis, SVM, ANN, Maximum Likelihood, Minimum Distance and Parallelepiped in terms of their kappa and classifier accuracy. Judging from information plotted in (Fig.5). Artificial Neural Network has a higher accuracy of 82.74% and kappa coefficient of 0.7688 than other five classifiers. Second standing in terms of good results is SVM, a binary classifier. SVM attained an accuracy of 81.85% while a kappa value of 0.757 has been noted accordingly. It has been noted that with the decrease of

training data in ANN both accuracy and kappa decreases. The lowest recorded accuracy and kappa coefficient is recorded by training 20% of samples and testing the remaining 80% as testing for the classifier.

Poor results have been found by using parallelepiped classifier tends to be 51.23% recorded accuracy for using all the samples as training data.

Table 3 Accuracy Comparisons of Mahalanobis, SVM and ANN in terms of Kappa and Accuracy

Training Set Data (%)	Mahalanobis Distance		SVM (Polynomial Degree)		ANN	
	Overall Accuracy (%)	Kappa Coefficient	Overall Accuracy	Kappa Coefficient	Overall Accuracy	Kappa Coefficient
100	69.2735	0.5894	81.8586	0.757	82.74	0.7688
90	69.4541	0.5919	81.7058	0.755	81.2	0.7492
80	70.051	0.5999	81.5947	0.7535	82.08	0.7604
70	69.2597	0.596	81.6363	0.754	79.22	0.7213
60	69.2041	0.5885	81.2891	0.7494	81.09	0.7479
50	69.3152	0.5899	80.9557	0.7449	74.28	0.6583
40	69.7597	0.5959	77.201	0.7354	74.34	0.6542
30	68.0928	0.5743	78.2436	0.7235	78.39	0.7098
20	67.7872	0.5703	77.2436	0.6988	63.68	0.522
10	66.7454	0.5565	78.65	0.6935	71.81	0.6192

Table 4 Accuracy Comparisons of MLH, Minimum Distance and Parallelepiped in terms of Kappa and Accuracy

Training Set Data (%)	Maximum Likelihood		Minimum Distance		Parallelepiped	
	Overall Accuracy (%)	Kappa Coefficient	Overall Accuracy	Kappa	Overall Accuracy (%)	Kappa
100	80.1086	0.7605	63.1199	0.5096	51.2356	0.3241
90	81.8336	0.7568	63.3431	0.5126	51.236	0.3243
80	81.8719	0.7574	62.945	0.5074	51.2402	0.3251
70	81.6594	0.7545	63.1997	0.5107	50.8127	0.3195
60	82.1718	0.7613	62.7692	0.505	51.0546	0.3215
50	82.4215	0.7648	63.7878	0.5186	51.0531	0.3208
40	81.9697	0.7532	63.529	0.5747	49.9103	0.3121
30	81.3512	0.7585	63.529	0.5147	51.4526	0.3289
20	82.7083	0.7686	63.8194	0.5188	51.3025	0.3252
10	81.7201	0.7512	65.1323	0.5346	51.6294	0.3314

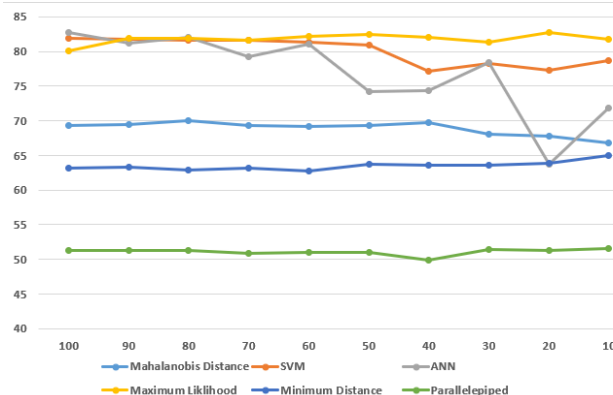


Fig.5 Classifier Statistics

6. CONCLUSION

Urbanization is not simply a new method, however a speedy transformation of human social roots on a worldwide scale, wherever village culture is being replaced by the urban culture. In Peshawar, the complexes of urban development is thus speedy that it demands fast response and perspective planning and management of it. Therefore, it is necessary and basic for policy manufacturers to include remote sensing into urban management. As Remote Sensing (RS) is a field that can be used to extract many type of information of an object, using Satellite imagery, without in physical contact of that object.

This paper, combined remote sensing concepts with Machine Learning Algorithms (MLAs) to give most feasible way of doing the task of urban settlement calculations. The output of our work can have substantial effect in organization and planning of Peshawar. For the stated goal, SPOT 5 imagery of August 2014 OF Hayatabad region has been acquired from SUPARCO (Pakistan Space and Upper Atmosphere Research Commission). The imagery is classified individually by six supervised classifiers. Maximum Likelihood, Minimum Distance, Support Vector Machines, Mahalanobis, Parallelepiped and Feed Forward Neural Networks (FFNN) for urban classification in Hayatabad Peshawar. For the assessments of classifiers, accuracy and kappa coefficients are used. Ground Truth data, for training of classifiers, has been collected and divided into 10 different samples as shown in (Table 2). Statistics of classifiers accuracy and kappa coefficients on 10% to 100% are tabulated in (Table 3 and Table 4). These statistics are plotted in Statistics which shows the behavior of classifier on validation. k-fold cross validation is a technique to evaluate predictive models by partitioning the original sample into a training set to train the model, and a test set to evaluate it. In our work $k = 10-90$. (Table 3) clearly highlights that Artificial Neural Network has a higher accuracy of 82.74% and kappa coefficient of 0.7688 than other five classifiers.

Second standing in terms of good results is SVM, a binary classifier. SVM attained an accuracy of 81.85% while a kappa value of 0.757 has been noted accordingly. It has been noted that with the decrease of training data in ANN both accuracy and kappa decreases. The lowest recorded accuracy and kappa coefficient is recorded by training 20% of samples and testing the remaining 80% as testing for the classifier. Poor results have been found by using parallelepiped classifier tends to be 51.23% recorded accuracy for using all the samples as training data.

REFERENCES:

- Amin T., B. C. Pijanowski, and A. H. Tayyebi. (2011) An urban growth boundary model using neural networks, gis and radial parameterization: An application to tehran, iran. *Landscape and Urban Planning*, 100(1-2): 35–44.
- Ajay D N., A. D. Vibhute, R. K. Dhumal, K. V. Kale, and S. C. Mehrotra. (2019) Urban land use change detection and mapping spatial variations of aurangabad city using IRS LISS-III temporal datasets and supervised classification approach. In *Data Analytics and Learning*, 369–386.
- Amin H. M., M. H. Moghaddam, and M. Esfandiyari. (2016) Stock market index prediction using artificial neural network. *Journal of Economics, Finance and Administrative Science*, 21(41):89–93.
- Ajay D N., R. K. Dhumal, A. D. Vibhute, Y. D. Rajendra, S. Gaikwad, K. V. Kale, and S. C. Mehrotra. (2017) Performance evaluation of urban areas land use classification from hyperspectral data by using Mahalanobis classifier. In *2017 11th International Conference on Intelligent Systems and Control (ISCO)*, 388–392. IEEE.
- Ana C. T., F. Gutierrez, P. Gomes, and J. Rocha. (2018) Remote sensing data and image classification algorithms in the identification of beach patterns. In *Beach Management Tools-Concepts, Methodologies and Case Studies*, pages 579–587. Springer.
- Chenghai Y., J. H. Everitt, R. S. Fletcher, R. R. Jensen, and P. W. Mousel. (2009) Evaluating AISA+hyperspectral imagery for mapping black mangrove along the south Texas Gulf coast. *Photogrammetric Engineering & Remote Sensing*, 75(4):425–435.
- David M and F. H. Technikum Wien. (2015) Support vector machines. *The Interface to libsvm in package e1071*, 28.
- Daniel C., de Oliveira Duarte, J. Zanetti, J. Gripp Junior, and N. das Graças Medeiros. (2018) Comparison of supervised classification methods of maximum likelihood, minimum distance, parallelepiped and neural network in images of unmanned air vehicle (uav) in Viçosa-MG. *Revista Brasileira de Cartografia*, 70(2): 437–452.
- Edwin R and B. Zagajewski. (2017) Comparison of support vector machine, random forest and neural network classifiers for tree species classification on airborne hyperspectral apex images. *European Journal of Remote Sensing*, 50(1):144–154.
- Fabio M., C. Conese, L. Petkov, and R. Resti. (1992) Inclusion of prior probabilities derived from a nonparametric process into the maximum-likelihood classifier. *Photogrammetric Engineering and Remote Sensing*, 58(2): 201–207.
- Giorgos M., J. Im, and C. Ogole. (2011) Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3): 247–259.
- Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, (1999) Extreme learning machine: a new learning scheme of feedforward neural networks. *Neural networks*, 2: 985–990.
- Giles F. (2017) Impacts of sample design for validation data on the accuracy of feed forward neural network classification. *Applied Sciences*, 7(9): 888Pp.
- Ganchimeg G and S. Chasia. (2017) Comparison between possibilistic c-means (pcm) and artificial neural network (ann) classification algorithms in land use/land cover classification. *International Journal of Knowledge Content Development & Technology*, 7(1): 57Pp.
- Jean F. M. and J. J. Flores. (2008) The application of artificial neural networks to the analysis of remotely sensed data. *International Journal of Remote Sensing*, 29(3):617–663.
- Jie D., H. Yamagishi, H. R. Pourghasemi, A. P. Yunus, X. Song, Y. Xu, and Z. Zhu. (2015) An integrated artificial neural network model for the landslide susceptibility assessment of Osado Island, Japan. *Natural Hazards*, 78(3):1749–1776.
- John A. R. and J. A. Richards. (1999) Remote sensing digital image analysis, vol. 3. Springer.
- John A. R. and J. A. Richards. (1999) Remote sensing digital image analysis, volume 3. Springer.

- Konstantinos T., V. Karathanassi, P. Pavlakis, and D. Rokos. (2019) Potentiality of feed- forward neural networks for classifying dark formations to oil spills and look-alikes. *Geocarto International*, 24(3): 179–191.
- Li C. Sun., L. Lin, J. Li, and S. Jiang. (2018) A mahalanobis metric learning-based polynomial kernel for classification of hyperspectral images. *Neural Computing and Applications*, 29(4):1103–1113.
- Marchant J. A. and C. M. Onyango. (2003) Comparison of a bayesian classifier with a multilayer feed-forward neural network using the example of plant/weed/soil discrimination. *Computers and Electronics in Agriculture*, 39(1):3–22.
- Maryam I., (2019) Difference-based target detection using Mahalanobis distance and spectral angle. *International Journal of Remote Sensing*, 40(3): 811–831.
- Mohammed, N S., S. D Al-Kofahi, Y. A Othman, and A. F. Cibils (2018) Assessing rangeland cover conversion in jordan after the arab spring using a remote sensing approach. *Journal of arid environments*, 157: 97–102.
- Najam A., N. Minallah, and A. Junaid. (2016) Land cover classification and crop estimation using remote sensing techniques. *International Journal of Computer Applications*, 155(2).
- Olivier N., P. Roussel-Ragot, L. Personnaz, G. Dreyfus, and S. Marcos. (1993) Neural networks and nonlinear adaptive filtering: unifying concepts and new algorithms. *Neural computation*, 5(2):165–199.
- Pushpendra S. S., V. Tiwari, and A. Kumar.(2014) Analysis of supervised maximum likelihood classification for remote sensing image. In *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, 1–4. IEEE.
- Russell G C. and K. Green. (2008) *Assessing the accuracy of remotely sensed data: principles and practices*. CRC press.
- Rashmi A. and K T. Swamy. (2000) Training feed forward neural network with back-propagation algorithm. *International Journal of Engineering And Computer Science*, ISSN, 2319–7242.
- Rahul N. and A. Khan. (2018) Survey on classification techniques used in remote sensing for satellite images. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 1860–1863. IEEE
- Surekha B., R. Thanki, and N. Dey. (2019) *Satellite Image Analysis: Clustering and Classification*.
- Mbnv S. K., Prasad, K. K. Barik, and B. K. Tripathy. (2015) Study on remote sensing data classification using statistical techniques. *Universe of Emerging Technologies and Science*, 2(5):1–6.
- Shih-Wei Lin, Kuo-Ching Ying, Shih-Chieh Chen, and Zne-Jung Lee. Particle (2008) swarm optimization for parameter determination and feature selection of support vector machines. *Expert systems with applications*, 35(4):1817–1824.
- SitiKhairunniza-Bejo, S. Mustaffha, and W. Ishak W. Ismail. (2014) Application of artificial neural network in predicting crop yield: A review. *Journal of Food Science and Engineering*, 4(1):01Pp.
- Senthil L.S. L and S. S. Kumar. (2018) Classification and mapping of land use land cover change in kanyakumari district with remote sensing and gis techniques. *International Journal of Applied Engineering Research*, 13(1):158–166.
- Umair K., N. Minallah, A. Junaid, K. Gul, and N. Ahmad. (2015) Parallelepiped and mahalanobis distance based classification for forestry identification in Pakistan. In *2015 International Conference on Emerging Technologies (ICET)*, 1–6. IEEE.
- Varma, M. K. S., N.K.K. Rao, K. K. Raju, and G.P.S. Varma. (2016) Pixel-based classification using support vector machine classifier. In *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, 51–55. IEEE.
- Vassilis P P. and Michael N Vrahatis. (2000) Training neural networks with threshold activation functions and constrained integer weights. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, vol. 5, 161–166. IEEE.
- Vignajeeth N., L A. Bharadwaj, K. H. Krishna, and J. Aravinth. (2018) Urban objects classification from hsr- htir data using Gaussian and Mahalanobis distance classifiers. In *2018 International Conference on Communication and Signal Processing (ICCSP)*, pages 1041–1045. IEEE