© 2021 IJRTI | Volume 6, Issue 6 | ISSN: 2456-3315

Satellite Image Processing Using SVM classifier and ELBP-ML Features

Riya Chourasiya[1], Prof. Shailesh Khaparkar[2]

[1]M.Tech Scholar, Department of Electronics and Communication, Gyan Ganga Institute of Technology and Sciences, Jabalpur, India[1]
[2]Associate Professor, Department of Electronics and Communication, Gyan Ganga Institute of Technology and Sciences, Jabalpur, India

***Abstract*: In this paper, Machine learning model used with components Support vector machine (SVM) Classifier and Extended Local Binary Patterns (ELBP) is used for Image Processing from highly noise corrupted images. The implementation of proposed methodology is being carried out by estimating the noise patterns of wireless image through configuring System Identification with SVM Classifier. The restored images are functioned for further de-noising and LBP Processing techniques. The experimental results demonstrate that SVM Classifier efficiently extract the features and Extended Local Binary Patterns (ELBP) eliminated noise from distorted images and delivered a virtuous evaluation without abundant degradation in performance. Image processing is one of classical problems of concern in image processing. There are various approaches for solving this problem. The aim of this paper is bring together two areas in which are Extended Local binary patterns (ELBP) and Support Vector Machine (SVM) applying for image processing. Firstly, we separate the image into many sub-images based on the features of images. Each sub-image is classified into the responsive class by an ELBP. Finally, SVM has been compiled all the classify result of LBP. Our proposal processing model has brought together many ELBP and one SVM. Let it denote ELBP_SVM. LBP_SVM has been applied for Roman numerals recognition application and the precision rate is 86%. The experimental results show the feasibility of our proposal model.**

***Keywords*: ELBP-Extended Local Binary Patterns, AWGN-Additive White Gaussian Noise, SVM-Support vector machine**

I-INTRODUCTION

Recently, image processing is growing and becoming a trend among technology developers especially with the growth of data in different parts of industry such as e-commerce, automotive, healthcare, and gaming. The most obvious example of this technology is applied to Facebook. Facebook now can detect up to 98% accuracy in order to identify your face with only a few tagged images and classified it into your Facebook's album. The technology itself almost beats the ability of human in image processing or recognition (What is the Working of Image Recognition and How it is Used, 2017). One of the dominant approaches for this technology is deep learning. Deep learning falls under the category of Artificial Intelligence where it can act or think like a human. Normally, the system itself will be set with hundreds or maybe thousands of input data in order to make the 'training' session to be more efficient and fast. It starts by giving some sort of 'training' with all the input data (Faux & Luthon, 2012). Machine learning is also the frequent systems that has been applied towards image processing. However, there are still parts that can be improved within machine learning. Therefore, image processing is going to be occupied with deep learning system. Machine Vision has its own context when it comes with Image Processing. The ability of this technology is to recognize people, objects, places, action and writing in images. The combination of artificial intelligence software and machine vision technologies can achieve the outstanding result of image processing (Haughn M, 2017).

LOCAL BINARY PATTERNS: The original LBP operator labels the pixels of an image with decimal numbers, called Local Binary Patterns or LBP codes, which encode the local structure around each pixel. It proceeds thus, as illustrated in Fig.1: Each pixel is compared with its eight neighbors in a 3x3 neighborhood by subtracting the center pixel value; The resulting strictly negative values are encoded with 0 and the others with 1; A binary number is obtained by concatenating all these binary codes in a clockwise direction starting from the top-left one and its corresponding decimal value is used for labeling. The derived binary numbers are referred to as Local Binary Patterns or LBP codes.

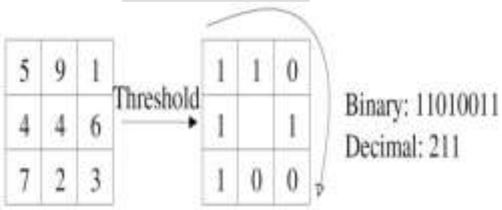

Fig. 1 An example of the basic LBP operator

SUPPORT VECTOR MACHINE: A support vector machine (SVM) is a supervised machine learning model that uses processing algorithms for two-group processing problems. After giving an SVM model sets of labeled training data for each category, they're able to categorize new text. The basics of Support Vector Machines and how it works are best understood with a simple example. Let's imagine we have two tags: red and blue, and our data has two features: x and y. We want a classifier that, given a pair of (x,y) coordinates, outputs if it's either red or blue. We plot our already labeled training data on a plane:

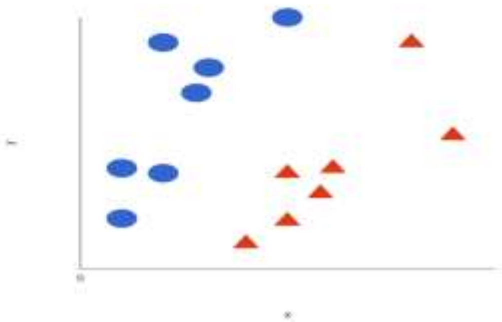

Figure 2: Labelled data in SVM

A support vector machine takes these data points and outputs the hyperplane (which in two dimensions it's simply a line) that best separates the tags. This line is the decision boundary: anything that falls to one side of it we will classify as blue, and anything that falls to the other as red. But, what exactly is the best hyperplane. For SVM, it's the one that maximizes the margins from both tags. In other words: the hyperplane (remember it's a line in this case) whose distance to the nearest element of each tag is the largest.

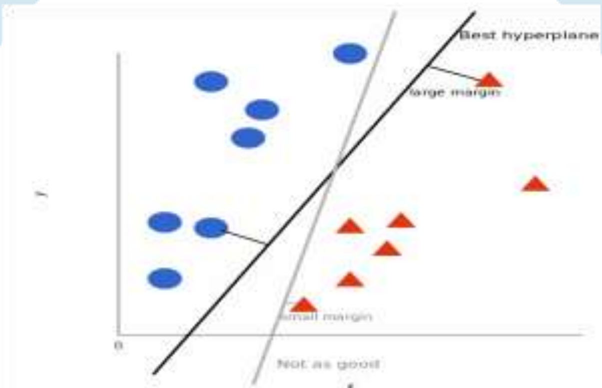

Figure 3: Not all hyper-planes are created equal in SVM

II-DESIGN METHODOLOGY

Figure 4 above shows the flow process the method adopted for this work. Here we have taken four different class of images and train the system with features of those images. Features of those training images are Extended Local Binary Patterns (ELBP) and Linear Kernel base Support Vector Machine (LKSVM) and Radial Kernel based Support Vector Machine (RKSVM). This work use minimum five training images for one class. Next is selection of test image the test image can be any other image, but it must be different then training images. Then features of test image extracted as was extracted of training images. Now compare the ELBP features, LKSVM features and RKSVM features. The processing decision is based on ELBP and any one among LKSVM and RKSVM.

ELBP ALGORITHM: Enhancing the discriminative capability: The LBP operator defines a certain number of patterns for describing the local structures. To enhance their discriminative capability, more patterns or information could be encoded.

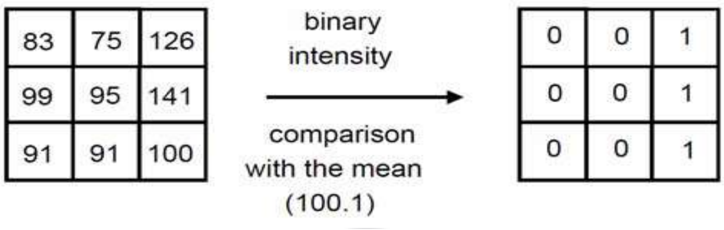

Figure 4. An example of ILBP operator

The Extended LBP (ELBP) is another approach to improve the discriminative capability of LBP. The ELBP operator not only performs binary comparison between the central pixel and its neighbours, but also encodes their exact gray value differences using some additional binary units. Specifically, the ELBP feature consists of several LBP codes at multiple layers which encode the gray-value difference (GD) between the central pixel and its neighbouring pixels. As shown in Fig. 5, the first layer of ELBP is actually the original LBP code encoding the sign of GD. The following layers of ELBP then encode the absolute value of GD. Basically, each absolute GD value is first encoded in its binary representation, and then all the binary values at a given layer result in an additional local binary pattern. For example, in Fig. 5 the first layer is the original LBP code that encodes the sign of GD, thus yielding a decimal number of 211 from its binary form (11010011)2. The absolute values of GD, i.e. 1, 5, 3, 2, 1, 2, 3, 0, are first encoded in their binary numbers: (001)2, (101)2, (011) 2, (010) 2 , …, etc. Using a same weight scheme of LBP on all the binary bits, its ELBP code of the corresponding layer can be generated, e.g., L2 is composed of (01000000)2 and its decimal value is 64; L3 is composed of (00110110)2 and its decimal value is 54; finally L4 is composed of (11101010)2 and its decimal value is 234. As a result, when describing similar local textures, although the first layer LBP is not discriminative enough, the information encoded in the other additional layers can be utilized to distinguish them. Its downside is that ELBP greatly increases feature dimensionality.

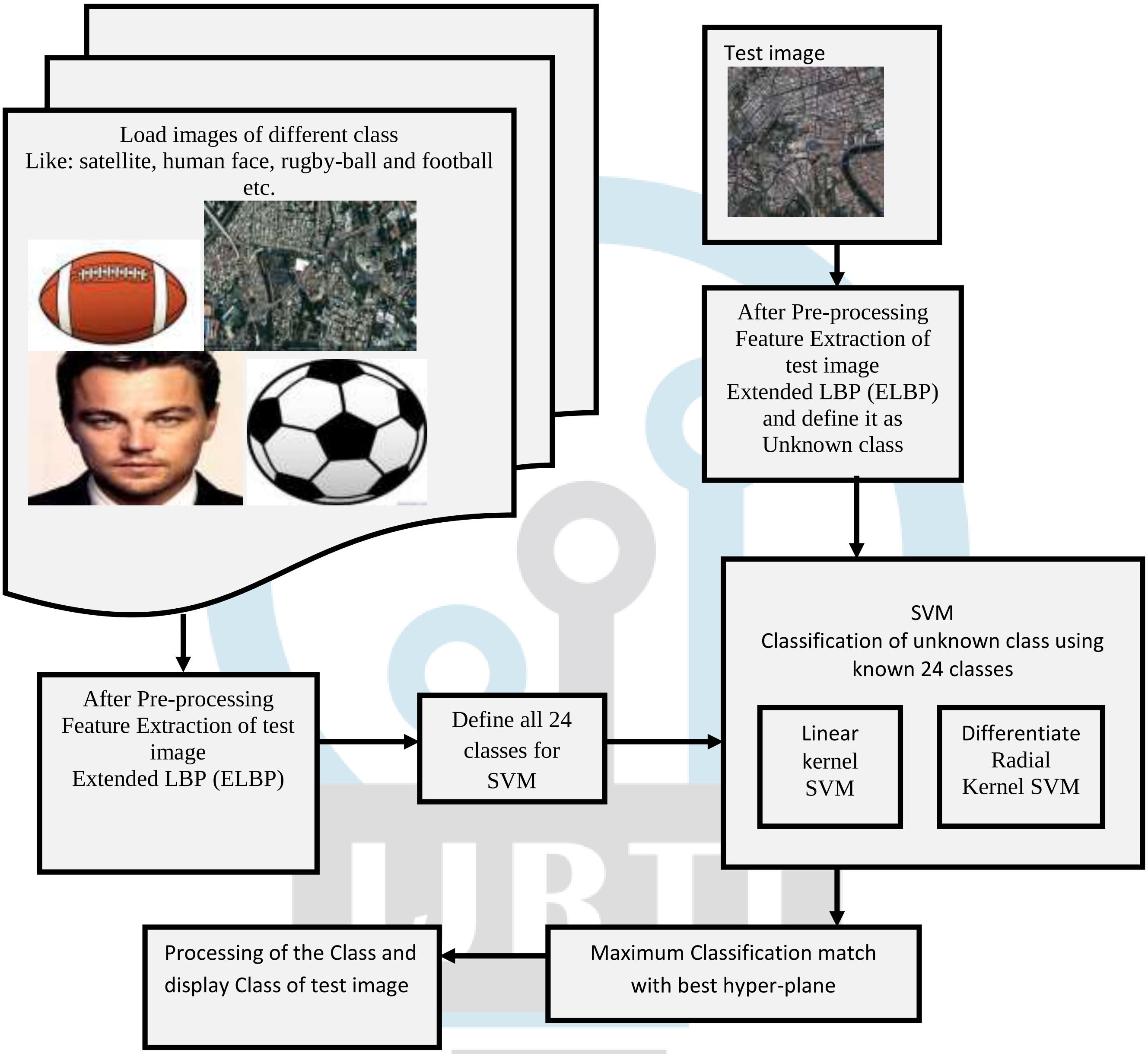

Figure 5 Flow process of method adopted

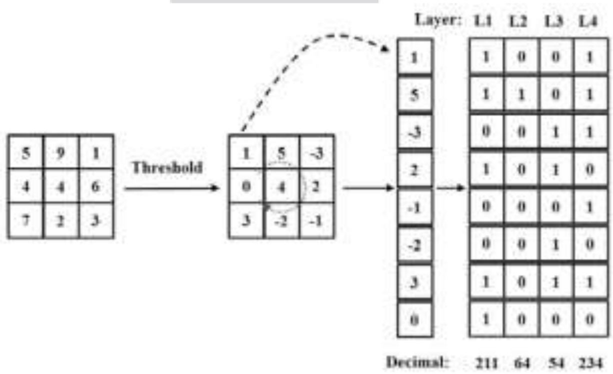

Figure 6 An example of the ELBP operator.

Improving the robustness: LBP is sensitive to noise since the operator thresholds exactly at the value of central pixel. To address this problem, [6] extended the original LBP to a version with 3-value codes, called Local Ternary Patterns (LTP). In LTP, indicator s(x) in (1) is replaced by:

$$s(i_n, i_c, t) = \begin{cases} 1 & i_n \geq i_c + t \\ 0 & |i_n - i_c| < t \\ -1 & i_n \leq i_c - t \end{cases}$$

Choosing the neighbourhood: The choice of an appropriate neighbourhood for LBP-based techniques has a significant impact on the final performance. It involves the number of sampling points, the distribution of the sampling points, the shape of the neighbourhood, and the size of the neighbourhood as well. Extending to 3D LBP: Several researchers have been trying to extend the LBP from 2D plane to 3D volume [3] [8]-[9] [6]; however, it is not so straightforward as it appears at first glance. There are two difficulties: first, equidistant sampling on a sphere is a difficult job, and second, it is also difficult to set an order to those sampling points, which is important to achieve rotation invariance.

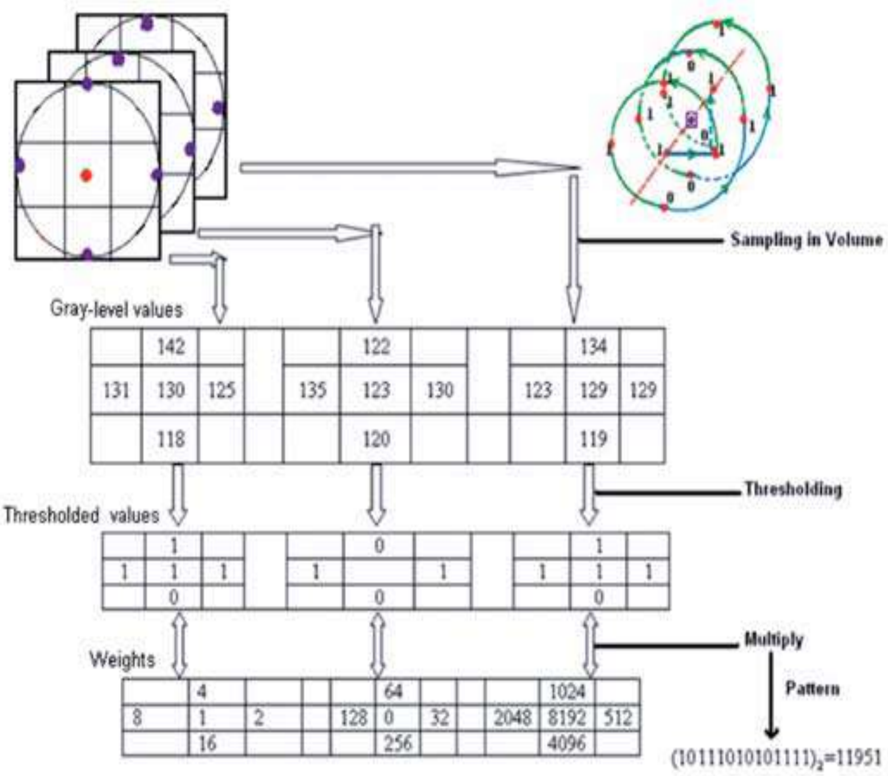

Figure 7 Procedure of VLBP

SUPPORT VECTOR MACHINE: The SVM algorithm is implemented in practice using a kernel. The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra, which is out of the scope of this introduction to SVM. A powerful insight is that the linear SVM can be rephrased using the inner product of any two given observations, rather than the observations themselves. The inner product between two vectors is the sum of the multiplication of each pair of input values.

For example, the inner product of the vectors [2, 3] and [5, 6] is 2*5 + 3*6 or 28.

The equation for making a prediction for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows:

$$f(x) = B_0 + \sum_{i=1}^{N} (a_i x + a_i x_i)$$

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

Linear Kernel SVM: The dot-product is called the kernel and can be re-written as:

$$K(x, x_i) = \sum_{i=0}^{N} x x_i$$

The kernel defines the similarity or a distance measure between new data and the support vectors. The dot product is the similarity measure used for linear SVM or a linear kernel because the distance is a linear combination of the inputs. Other kernels can be used that transform the input space into higher dimensions such as a Polynomial Kernel and a Radial Kernel. This is called the Kernel Trick. It is desirable to use more complex kernels as it allows lines to separate the classes that are curved or even more complex. This in turn can lead to more accurate classifiers.

Radial Kernel SVM: Finally, we can also have a more complex radial kernel. For example:

$$K(x, x_i) = e^{\{-\gamma(\sum_{i=0}^{N}(x-x_i)^2)\}}$$

Where gamma is a parameter that must be specified to the learning algorithm. A good default value for gamma is 0.1, where gamma is often 0 < gamma < 1. The radial kernel is very local and can create complex regions within the feature space, like closed polygons in two-dimensional space.

III-RESULTS

Training: For the processing of satellite image four type of images are taken as classes rugby-ball images, football images, human faces and satellite images.

Figure 8 Training of satellite image

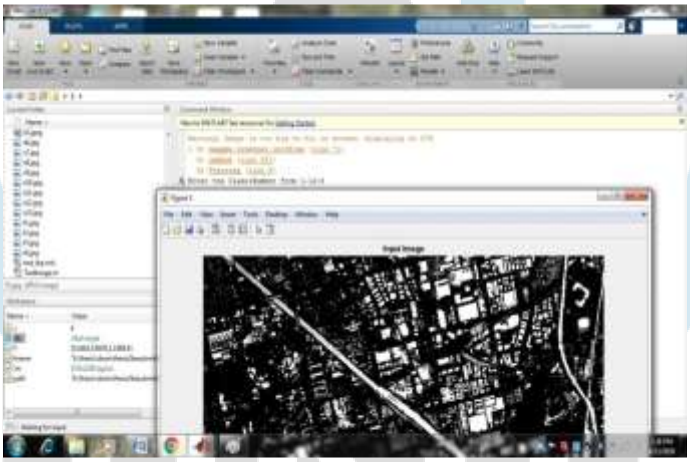

Figure 9 Human face image train as class-4

Figure 8 and 9 shows the training of class-4, with the same process another four images have been trained.

Table 1 below shows the image features extracted and store before code running from training classes.

Table 1 Feature extracted from different classes

LKSVM	RKSVM	ELBP	Class
0.57764	0.17448	1.06842	1
0.51768	0.11867	1.03884	1
0.46114	0.14688	1.05263	1
0.57227	0.10929	1.08986	1
0.66825	0.04125	1.06494	2
0.73745	0.10253	1.04882	2
0.71119	0.0789	1.0761	2
0.82553	0.17686	1.02828	2
0.79449	0.21161	1.01049	3
0.93256	0.18672	1.03398	3
0.58456	0.15736	1.03527	3
0.47472	0.035	1.07194	3
0.1662	0.06764	1.10689	4
0.1662	0.06764	1.10689	4
0.35191	0.04352	1.29283	4
0.19224	0.05501	1.14743	4

Testing Simulation results for ELBP: Using LBP Features to Differentiate Images by Texture Read images that contain different textures.

Figure 10 Test Satellite image

Figure 11 One of train satellite image in class-4

Figure 12 rugby-ball train image in class-1

Extract LBP features from the images to encode their texture information. figure 10 shows Gauge the similarity between the LBP features of by computing the squared error between figure 10 image vs figure 11 & figure 12. Visualize the squared error to compare bricks versus bricks and bricks versus carpet. The squared error is smaller when images have similar texture.

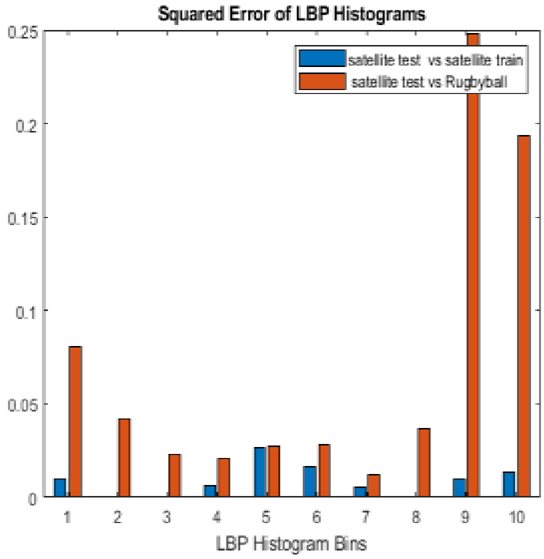

Figure 13 LBP histogram comparison class-4 vs class-1 (orange bar) and class4 vs class4 (blue bar)

Figure 13 shows dissimilarities of LBP observe between test satellite image and train satellite image with blue bars, it also shows the dissimilarities of LBP observe between test satellite image and rugby-ball image. it may clearly observe that test and class four train satellite images have minimum dissimilarities also the test satellite image and train rugby-ball image has maximum dissimilarities. On behalf of that we can classify the satellite image. the rate of correct processing observe is 100%.

Testing Simulation results for SVM: Feature Extraction of test and training images image with Linear Kernel SVM and Radial Kernel SVM comparison done on basic of percentage match with both and that decides the class of the object.

Figure 14 select the test image of class-4

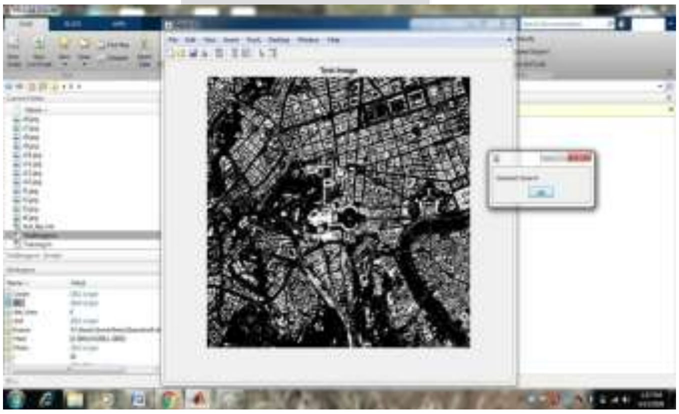

Figure 15 select the test image of class-4

From the above results observation it is observe that with Linear and Radial Kernel SVM the all four type of image classes identify correctly. Accuracy observed for the proposed work when 100 satellite test image classes are simulated and classify. Total 94 times satellite images have been classify correctly with proposed Image Processing Method based on Machine learning features ELBP and SVM classifier. Hence accuracy of this work is 94%.

Table 2 Comparative results

Work	Method	Average Accuracy observe
Proposed	ELBP and LKSVM/RKSVM classifier	94
Anju Asokan [1]	Random Forest (RF) with SVM	88
Sehla Loussaief [2]	Speed Up Robust Features and K-mean clustering	89
Mohd Azlan Abu1 [3]	DNN and Tensorflow	94
Andreas Kolsch [4]	CNN and Extreme Learning Machines	90

From the table 2 it may observe that proposed work has better accuracy then [1],[2] and [4] though accuracy observe for proposed work as compare with [3] is same, but the method of work [3] was Deep Neural network which is more complex then proposed machine learning hence proposed work can be consider better.

IV-CONCLUSION

In this thesis work it relate the different techniques and algorithms used in proposed machine learning framework for satellite image processing. thesis presented machine learning state-of-the-art applied to image processing. This work introduced the Bag of Features paradigm used for input image encoding and highlighted the Extended Local Binary Pattern as its technique for image features extraction. Through experimentations this work proofed that using ELBP local feature extractor method for image vector representation and LKSVM and RKLBP training classifier performs best prediction average accuracy. In test scenarios this focused on satellite image as we project to apply the trained classifier in a general system.presently the researchers are trying to arrive at some solutions by combining various image processing techniques or introducing hybrid models based on spectral and spatial indices for the same to improve the outcome. In near future this type of combination can be implemented for better accuracy. In near future the number of classes also can be improve as his work has limitation of 12 class only, accuracy obtain in this work is 94% it may be improvise with use of Deep learning.

REFERENCES

[1] A. Asokan and J. Anitha, "Machine Learning based Image Processing Techniques for Satellite Image Analysis -A Survey," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 119-124.

[2] Sehla Loussaief, Afef Abdelkrim, Machine Learning framework for image processing , Special issue on Advancement in Engineering Technology Advances in Science, Technology and Engineering Systems Journal Vol. 3, No. 1, 01-10 (2018)

[3] Mohd Azlan Abu, Nurul Hazirah Indra, Abdul Halim Abd Rahman, Nor Amalia Sapiee1 and Izanoordina Ahmad, A study on Image Processing based on Deep Learning and Tensorflow, International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 4 (2019), pp. 563-569

[4] Andreas Kolsch, Muhammad Zeshan Afzal, Markus Ebbecke, Marcus Liwicki, Real-Time Document Image Processing using Deep CNN and Extreme Learning Machines, arXiv:1711.05862v1 [cs.CV] 3 Nov 2017

[5] Bassel Marhaba, Mourad Zribi, The bootstrap Kernel-Diffeomorphism Filter for Satellite Image Processing, 978-1-5386-4615-1/18/2018 IEEE

[6] J. Wang,Y. Li, Y. Zhang,C. Wang, H. Xie, G. Chen, X. Gao, Bag-of-features based medical image retrieval via multiple assignment and visual words weighting. IEEE Trans. Med. Imaging 30(11), 1996–2011, 2011. https://doi.org/10.1109/TMI.2011.2161673

[7] O. Miksik, K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching" in International Conference on Pattern Recognition (ICPR 2012), pp. 2681–2684. Tsukuba, Japan, 2012.

[8] B. Kim, H. Yoo, K. Sohn, Exact order based feature descriptor for illumination robust image matching. Pattern Recognition. 46(12), 3268–3278, 2013.

[9] T T Dhivyaprabha, P. Subashini, M. Krishnaveni, "Computational intelligence based machine learning methods for rule-based reasoning in computer vision applications", in IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 2016. https://doi.org/10.1109/SSCI.2016.7850050

[10] P. Moreels, P. Perona, "Evaluation of features detectors and descriptors based on 3D objects", in Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, China, 2005. https://doi.org/10.1109/ICCV.2005.89

[11] F.M. Campos, L. Correia, J.M.F. Calado, Robot visual localization through local feature fusion: an evaluation of multiple classifiers combination approaches. J. Intell. Rob. Syst. 77(2), 377–390, 2015.

[12] S. Zhang, Q. Tian, Q. Huang, W. Gao, Y. Rui, "USB: ultrashort binary descriptor for fast visual matching and retrieval",in IEEE Transactions on Image Processing, 2014. https://doi.org/ 10.1109/TIP.2014.2330794