IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

MULTI-LANGUAGE DIGIT CLASSIFICATION USING RANDOM FOREST CLASSIFIER

[1]Gajendra Sharma, [2] Jaswanth, [3] Hemanath Reddy,[4] Hithesh

[1] Professor, Department of ECE, Madanapalle Institute of Technology & Sciences, Madanapalle, Andhra Pradesh, India.

[234] Students, Department of ECE, Madanapalle Institute of Technology & Science, Madanapalle, Andhra Pradesh, India.

ABSTRACT

Handwritten digit classification is the system's ability to predict and detect handwritten digits. Now a days machine learning has more significant role in classification problems. The created model is Random Forest classifier. The dataset used for the project is prepared by own where each dataset consists of 2000 samples of Telugu, English and Hindi language digit images. The aim of the paper is to observe the working of random forest classifier created by using Sklearn python package in terms of training accuracy, testing accuracy and confusion matrix. TheMultilanguage refers to combination of Telugu, English and Hindi language digits into a single dataset. The testing accuracy is found to be 99% for 50 decision trees.

Keywords: Sklearn, Random Forest, HOG and NumPy.

I. INTRODUCTION

Now a days there is an increase in the growth of computer visions and machine learning. Mainly, machine learning is used in Image classification, Image recognition and Handwritten digit classification etc. Handwritten digit classification problem solving using random forest algorithm. Random Forest is a well-known AI calculation that has a place with the managed learning procedure. In ML, it is commonly used for both classification and regression problems. It depends on the idea of group realizing, which is a course of joining ensemble classifiers to take care of a complex issue and to work on the improving the performance of the model.

II. METHODOLOGY

Dataset

We have used digits dataset which is prepared by own. The dataset contains 6000-digit images belonging to ten different classes of different languages like Telugu, English and Hindi. For each language we have 2000 images. We have combined all the digit images into single dataset. We have split the entire dataset into two sections of 8:2 ratio. We have used 80% (i.e., 4800 images) for training and remaining 20% (i.e., 1200 images) for validation purpose.

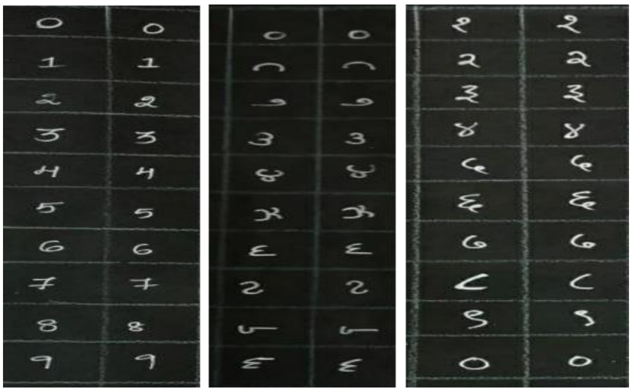


Fig1. Sample digit images

Histogram of Oriented Gradients

After splitting the images into training and testing samples, the HOG features of the images are extracted using Skimage package which consists of command "hog". The HOG gives us the outline of the image and images are formed in terms of HOG feature descriptor. The HOG feature vector holds 3x3 cells with 4 orientations. After HOG feature extraction, the features used for training Random Forest classifier.

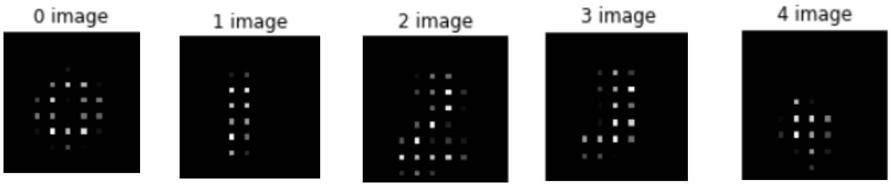


Fig.2 HOG feature images

III. MODELING AND ANALYSIS

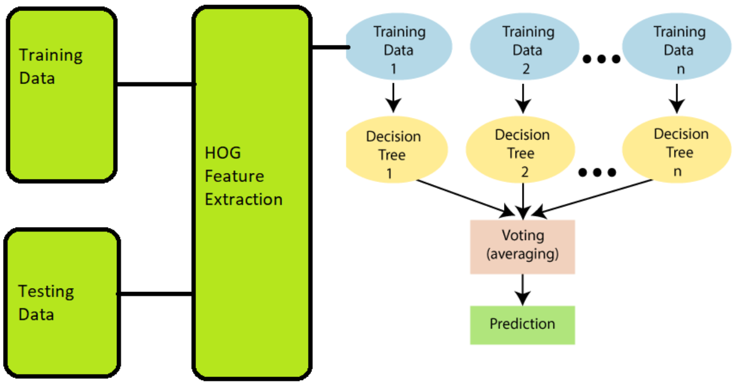


Fig.3. Random Forest Model

The model is created using Sklearn package which contains ensemble command which has n_estimators which defines the number of the decision trees. The HOG features data are used for training. First it splits the training data into n samples. After that the decision trees were created for each sample. For each image it gives an output, the majority output will be considered as the final output of the image.

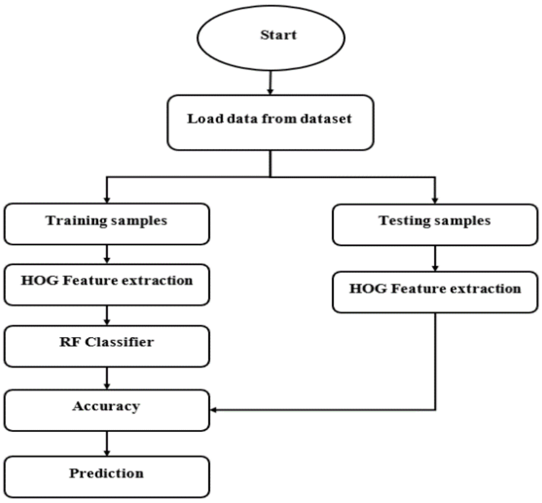


Figure 4: Flow chart of Random Forest classifier.

IV. RESULTS AND DISCUSSION

N_Estimators(Number of trees)	Training Accuracy	Testing Accuracy
10	0.999	0.98
20	1.0	0.986
30	1.0	0.985
40	1.0	0.9875
50	1.0	0.9908

Table – 1 Results for various N_estimators

In Table.1 the n_estimators tells the number of decision trees. For 10,20,30,40 and 50 decision trees the training and testing accuracy were found on 6000 samples.

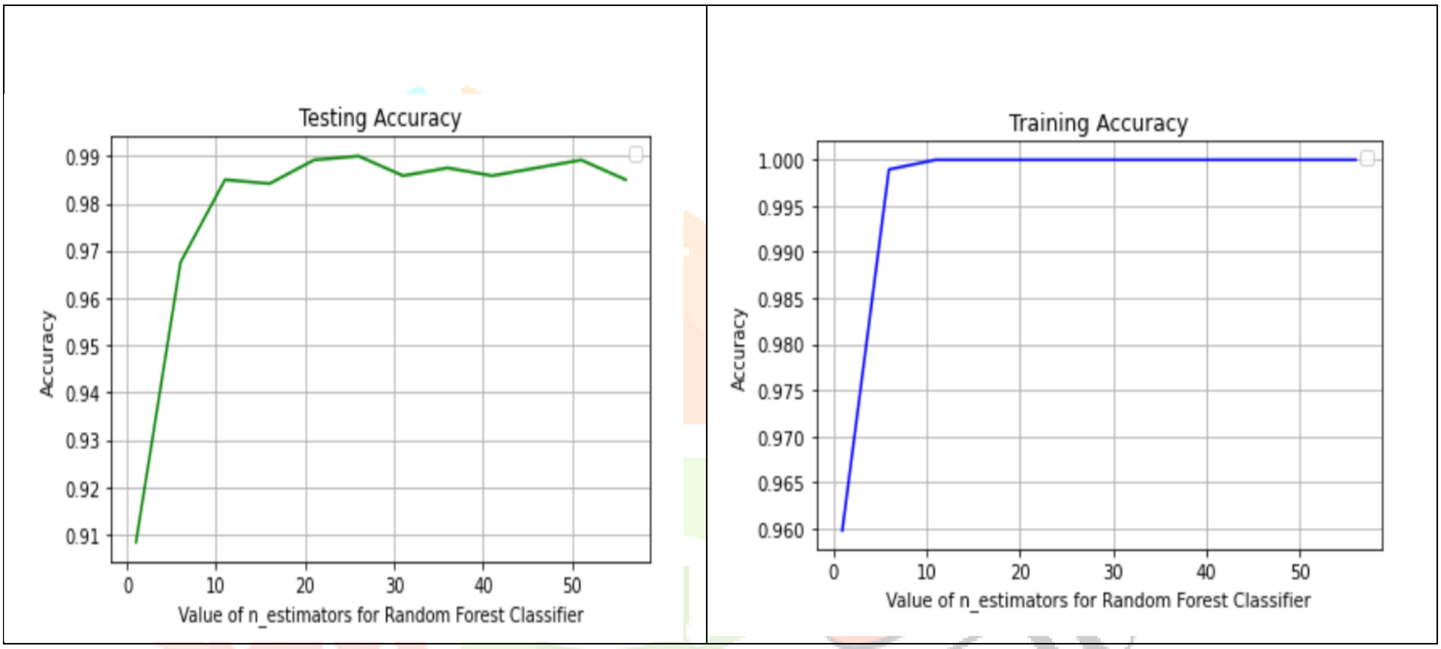


Figure 4: Training and Testing Accuracy of Random Forest classifier

In the above Figure 4 the training accuracy vs number of decision trees and testing accuracy vs number of decision trees. In testing accuracy curve the maximum accuracy is found at 21,22,23,24,25 and 50 decision trees. In training curve below 10 decision tree the accuracy is below 100 %and after 10 decision trees the accuracy is constant as 100%.

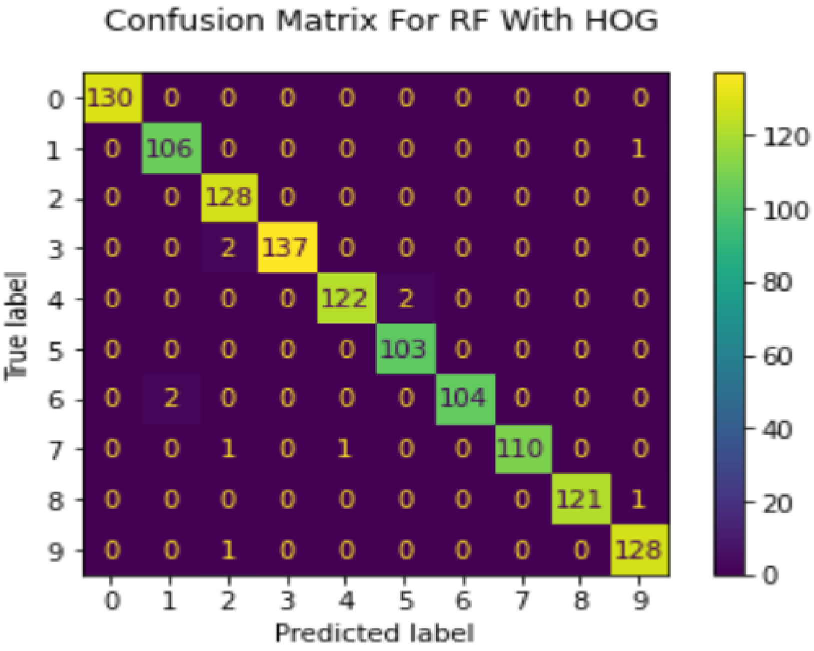
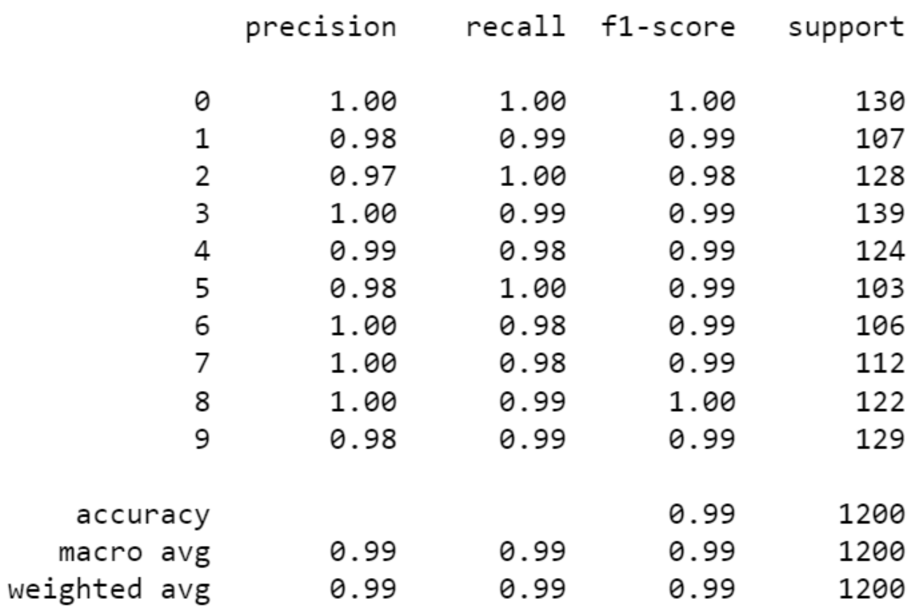


Figure 5: Confusion Matrix on test data

The above Figure 5 defines the true and predicted label of the test data. The size of the test data is 1200. For each class the size of test data is different. For class 0, the accuracy is 100%. For other classes have less than 100% and greater than 99%.



```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       130
           1       0.98      0.99      0.99       107
           2       0.97      1.00      0.98       128
           3       1.00      0.99      0.99       139
           4       0.99      0.98      0.99       124
           5       0.98      1.00      0.99       103
           6       1.00      0.98      0.99       106
           7       1.00      0.98      0.99       112
           8       1.00      0.99      1.00       122
           9       0.98      0.99      0.99       129

    accuracy                           0.99      1200
   macro avg       0.99      0.99      0.99      1200
weighted avg       0.99      0.99      0.99      1200
```

Figure 6: Classification report

The precision is good for class 0,3,6,7,8, and recall is good for class 0, class 8. The harmonic mean of precision and recall is used to calculate the f1 score. Support defines the number of samples in the dataset used for testing.

V. CONCLUSION

The use of Random Forest classifier in Handwritten digit classification can give better results in terms of the training and testing accuracy of the model. The dataset used contains 600 samples and each have 784 feature values. The maximum decision tree value used in this random forest model is 50. The training accuracy is 100% at 50 decision trees. The testing accuracy is 99% at 50 decision trees. We used hog features as feature vector and classifier as Random Forest. Finally conclude that further increase of decision trees after 50 the testing accuracy is not stable and gives accuracy in between 98% and 99% whereas the training is stable at 100% for any number of trees after 10 trees.

VI. REFERENCES

[1] Maduhansi Thenuwara, Harshani R. K. Nagahamulla, Offline Handwritten Signature Verification System Using Random Forest Classifier, Department of Computing and information Systems, Faculty of Applied Sciences, Wayamba University of Sri Lanka Kuliyapitiya, Sri Lanka, September 2017.

[2] Abhishek, Athmiya, Handwritten Digits Recognition using Random Forest Classifier, St Aloysius Institute of Management and Information Technology, Kotekar, Ullal, Karnataka India 575022, January 2020.

[3] T. Dash, T. Nayak, S. Chattopadhyay, Handwritten Signature Verification (Offline) using Neural Network Approaches: A Comparative Study, International Journal of Computer Applications Volume 57– No.7, November 2012.

[4] Simon Bernard, Laurent Heutte and Sebastien Adam,Using Random Forests for Handwritten Digit Recognition, Laboratoire LITIS EA 4108 UFR des Sciences, Universite de Rouen, France.

[5] D. Joon, S. Kikon. An Offline Handwritten Signature Verification System - A Comprehensive Review. International Journal of Enhanced Research in Science Technology & Engineering, Vol. 4 Issue 6, June 2015.

[6] Stanley Ziweritin1, Uchenna Chikwendu Anyimukwu Ugboaja and Chidiebere Moses Osu, Random Forest Model for Predicting Grayscale Digits on Images, Department of Computer Science, Akanu Ibiam Federal Polytechnic, Unwana-Afikpo, Ebonyi State, Nigeria,December 2020

[7] S. Talla, P. Venigalla, A. Shaik, and M. Vuyyuru, "Multiclass Classification Using Random Forest Classifier," International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), 2019.

[8] S. Bernard, S. Adam, and L. Heutte, " Using Random Forests for Handwritten Digit Recognition," Proceedings of the 9th IAPR/IEEE International Conference on Document Analysis and Recognition (ICDAR),2007.