International Journal of Scientific Research in Computer Science and Engineering **Research Paper**
Vol.8, Issue.6, pp.01-07, December (2020) **E-ISSN:** 2320-7639
DOI: https://doi.org/10.26438/ijsrcse/v8i6.17

Random Forest Model for Predicting Grayscale Digits on Images

Stanley Ziweritin[1*], Uchenna Chikwendu Anyimukwu Ugboaja[2], Chidiebere Moses Osu[3]

[1]Department of Estate Management and Valuation, Akanu Ibiam Federal Polytechnic, Unwana-Afikpo, Ebonyi State, Nigeria
[2,3]Department of Computer Science, Akanu Ibiam Federal Polytechnic, Unwana-Afikpo, Ebonyi State, Nigeria

Corresponding Author: stanlo4godsluv@yahoo.com, Tel.: +2348033997435

Available online at: www.isroset.org

Received: 10/Dec/2020, Accepted: 15/Dec/2020, Online: 31/Dec/2020

***Abstract*-** The random forest is one of the fastest, powerful and easy to use techniques with the ability of performing regression and classification task recorded in the biblical study of machine learning. It grows into forest with some number of decision trees as a hybrid system. The problem of missing data which results from over-fitting can best be handled by random forest. The process places a very big barrier on how well the existing methods and functions. This also affects the success metrics in the process of training, testing and validating classifiers in order to produce a more generalized and accurate results for predicting grayscale digits on images. This model is usually employed in replacing the continuous variables with the median values while computing the correct weighted proximity average of all the missing data values. Our aim is to develop an efficienct random forest model using the classification and regressor classifiers with some high-quality of input data by introducing weighted average and the best in majority vote. This model has been trained and tested using the MNIST dataset with some adjusted hyper-parameter values which defined and improved the general performance of the combined trees in the forest. The simulation was done using python programming language and resulted to 99% and 90% metrics of accuracy in comparison with the random forest classification and regression approach respectively.

Keywords—Random forest, grayscale images, RF_classification, RF_regression classification, MNIST dataset

I. INTRODUCTION

The random forest(RF) is one of the most widely known and easy-to-use supervised machine learning technique for sovling classification and regressor problems[1],[2]. This help to combine multiple decission trees with samples of observed features in training and testing of models in order to predict data patterns. A single decision-tree and other machine learning classifiers, in most cases tends to overfit training data. The RF can overcome this anomaly by considering the average predicted results from different trees within the RF[3]. The RF has higher rate of prediction accuracy over the use of a single decision-tree and other machine learing techniques with classifiers[4][5]. Accurate predictions and better generalizations can be achieved through the use of RF forest techniques. This solves the problem of overfitting caused by missing datas, eliminates low biasness, reduces high varinace and influences the overall error rate[6],[7].
The existing methods of machine learning classifiers are faced with the problem of high error-rate and missing data problem as caused by over-fitting when adopted as a single or stand-alone model. Also, the training and testing of machine learning classifiers in predicting grayscale digit on images places a very big burden on the existing methods. The RF is recorded to be highly efficient, accurate and exhibits the strength that can not suffer from the difficulties encountered in over-fitting because of the number of n_estimators involved in the process of fitting the model[8][9]**.** Therefore; we intend to develop an efficient RF model to correctly predict grayscale digits on images using the RF classifiation and regresion approach. The model will be used to handle missing values or data problem by replacing the continuous variables with the median values and compute for the proximity weighted average of all the missing values and predictions.

This work is arranged in sections. The first section focuses on the introduction. Section two presents a brief review of some related literatures used and the gap in adopting the proposed model. Section three, introduces the materials and methods used in developing the model. Section four focuses on detailed discussion of the results while section five presents the conclusion of this work.

II. RELATED WORK

An improved RF classification model was developed using parts of a whole disjoint dataset to stimulate the individual decision-trees based on the weighted voting and Q-statistics measures of diversity[10]. The variation and distances between dataset pairs were recorded and ranked. The metrics of accuracy was made to be more effcient using some varying and different number of n_estimators ranging from 5 to 100 with step value set to 5, but required more training time and could not function well as expected with large dataset.

© 2020, IJSRCSE All Rights Reserved

A RF classifier and j48 type of decision tree was adopted[11]. A ten-fold cross validation set of $\frac{80}{100}$ and $\frac{20}{100}$ of the total items were used as training and testing set for ten iterations to overcome the problem of overfitting. The results were accurate ranging from 69.23 to 96.13 percent and pricise in some of the cases when the instances were increased from 286 to 699 and the model attributes set to a constant value. A support vector machine(SVM), K-Nearest Neighbor (KNN) and RF was employed in predicting different types of hand-written digits[12]. The training and test dataset was loaded and fed to the different classifiers in obtaining 97.7% and 97.74% metrics of accuracy for the SVM testing and training set respectively. Therefore the SVM performed quite better than KNN and the RF model. But there was this problem of not been able to aggregate the individual decision trees within the random forest. And this led to low accuracy rate that required some future engineering techniques to be corrected. A logistic regression(LR), RF and SVM model was developed to classify human-handwritten digits using the MNIST dataset[13]. The implementation of the SVM classifier employed the concept of hyper-plain in classifying data patterns into different classes that are equdistant from each other. RF was employed to randomly construct or grow decision-trees from data samples to make prediction from each trees to have the best solution by means of selecting the best votes. Al-Behadili[14], developed a neural network using 4-input, 3-hidden and 1-output layers, decision-tree and KNN models, to predict human handwritten-digits(from 0-9) based on 28 by 28 grayscale image system. This model was trained and tested using the MNIST dataset to ascertain the performance of NN as the best among KNN and decision tree classifiers but required more training time to perform better in terms of digit recognition[15]. Singh, and Sathyaraj[16] combined the RF, Naive Bayes and decision-tree classifiers to match data patterns between attributes in making predictions using small and large scaled dataset. The results revealed that decision tree was the most suitable for large scaled data and Naive Bayes for small scaled dataset but was limited by the problem of over-fitting as a result of missing data. A RF model was used in combination with Naives Bayes and the J48-decision tree classifiers to classify patterns with predictor variables using protein dataset[17]. The random forest produced 85 correctly and 15 incorrectly classified data patterns, correctly classified J48-Decision-tree as 93 and 7 incorrectly classified data patterns. And Naive Bayes with 90% correctly classified and 10 incorrectly classified data patterns to produce 90% overall metrics of accuracy. Chherawala[18] proposed the use of weighted vote collection of RF and RNN classifiers, trained and tested with some feature dataset to predict offline human handwritten digit recognition system. This was done with some grayscale feature extraction technique to provide an optimal solution. An assembly of machine learning model(DT, Euclidean distance, SVM, KNN and NN classifiers was adopted) capable of predicting MODI-script characters[19]. The performance of Decision was 97.68% which is higher in accuracy among others but resulted to over-fitting that can best be handled by random forest.

III. METHODOLOGY

It is so relevant to give credit to efforts made by different authors in adopting random forest model and other ML techniques to effectively predict grayscale digits on images. We therefore proposed to build on the areas that with some lapses. We were able to identify and narrow our study down to the use of random forest classification and regression techniques.

Grayscale images: Gray images contains regular grid of cells called pixels with only two possible values usually displayed to be black and white. The black and white colored pixel values are often set to digit 0 or 1[20]. This can help separate digits or objects displayed on binary images from the background. The white colored pixels are used to represent the digit as object and black pixels are used to represent the background color which depends on the digit generated on the binary image[21]. The color used or surrounded by the digit(object) in the grayscale image is the fore-color while the pixels outside the images are the background color. Pixels are the smallest units in digital image that contain regular grid of cells[22]. The grayscale images used by the proposed system are stored as 8-by-8 bits formation from 0-to-63 possible shades of gray pixels represented using white and black. Grayscale images are very useful because most hardware devices used to capture and simulation languages used to generate images support the 8-bit image display system[23]. Therefore, grayscale images are sufficient enough for the proposed system tasks.

Segmentation: is the act of partitioning grayscale images into meaningful units or structures that are similar to human hand-written characters[24]. Image segmentation is necessary to obtain a reliable and detailed representation of digits as objects on grayscale images.

Figure 1: The design of grayscale digits on images[25]

Dataset used: The proposed system dataset was sourced from the MNIST(Modified National Institute of Standard and Technology) experimental dataset with high quality and easy to use grayscale digits placed on images that ranges from 0-to-9. This contains one of the largest and highly grayscale digits found in Python sklearn library used as the testing and training of machine learning classifiers. And the dataset is divided into training and testing set.

© 2020, IJSRCSE All Rights Reserved

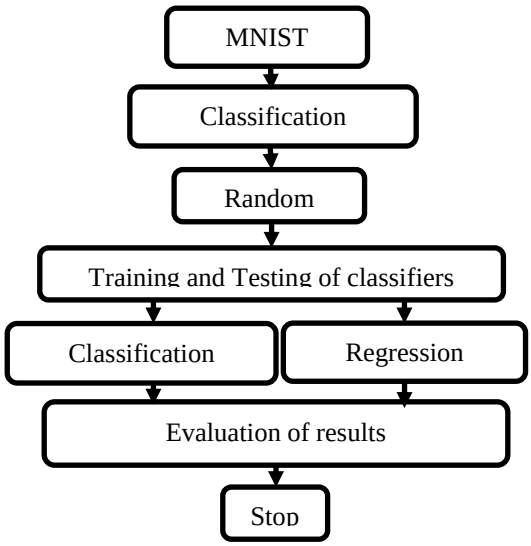

Figure 2: Process flow diagram

Classification technique: is one of the key and useful components involved in decision making process that categorize data based on some observed features or criteria. We employed the feature extraction technique in classification where a feature vector(X) in the process represented by [16] as:

$$X = (f_1, f_2, \dots, f_n) \quad (1)$$

Where "f" represents the features and "n" number of features extracted from the grayscale digits presented on images and efficiently classified to form an appropriate class. We adopted row and class-feature sampling technique for each and every decision tree to reduce bias and high variance. The Change in input dataset causes low variance in the decision tree and output will be very good and accurate with majority votes for the binary classification model[27].

$$\Delta_k = \{(P_1, \dots, P_k): \sum_{k=1}^{N} P_k = 1 \; and \; P_k \geq 0\} \quad (2)$$

Where Δ_k is the set probability distribution over X, We therefore assume that e_k to be a member or element of Δ_k and if a decision-tree(t) predicts that an instance to a class X_k Then we can rewrite as:

$$\hat{Y} = \frac{1}{N} \sum_{t=1}^{T} \hat{Y}_{i,t} \quad (3)$$

Where "N" represents total number of data samples under observation, "T" the overall or total number of DT within the forest, "i" represents the instances of the forest tree predictions that coincides with the class of C_k and $\hat{Y}_i$ is the maximal for all a $\hat{Y} \epsilon \Delta_k$

The RF model: Random forest is the assembly of different decission trees with samples of observed features. We intent to build multiple decision called the Random forest using classification and regression gini approach. A single decision-tree has high variance and the collection of several decision-trees is called the random forest which help solve the problem of high by varience[28]. Because each and every DT within the forest is perfectly trained using the sample data which depend mainly on multiple decion trees but not single tree. The result of random forest classification is always-based on majority voting while the outcome of regression problem is the mean of all the tree predictions as output within the forest[29]. In the random forest most of the trees are producing correct predictions while some are making mistakes. For this reason voting is required to be carried out based on the classifications for the observed result poll and expected the outcome to be closer to the correct classification[30]. We intend to use more high-quality data with adjusted hyper-parameter values for both models which can define the number of trees in the RF to improve and produce a more generalized and better result. A single decision tree will always result to the problem of low bias and High variance. Therefore; we adopted RF tree to convert the low bias and high variance of a single decision-tree to have a low variance.

Training and testing dataset: The uploaded dataset of the proposed system is divided into 80% (1437 items) and 20%(360 items) of the total dataset(1797 items) for training and testing using the xtrain, xtest, ytrain, ytest = train _and_test_split and test_size set to 0.2 command.

Training and testing of classifiers: The sklearn and ensemble libraries are employed to build multiple decision trees in predicting the outcome based on majority vote. To train and evaluate the mode we employed the training(80% or 1437) and testing(20% or 360) of the total dataset(1797 items). We further fine-tuned the hyper-parameter values of the random forest classifier with the following number of decision trees(20, 30, 40, 50, 60 and 70 as estimators) in measuring the general performance of the model.

RF classification model: The ensemble library will be used to predict grayscale digits in the classification model because we are building multiple decision trees to predict the outcome. We propose to fine-tune or adjust the n_estimators in the random forest classifier to train with 10, 20, 30, 40-to-70 random trees. And the seabon visualization library in python to display the confusion matrix and this will help to evaluate the training the model

RF regression model: The training and testing dataset will be scaled before feeding it to the fitted regression model using the regression class of the sklearn ensemble library, trained and tested with specific number of n_estimators(decision-trees) as parameters and random states set to be 0 to obtain an optimal solution. This adjustable parameter defines the random forest.

RF Algorithm for classification and regression [28]

Step 1: Start

Step 2: Assume cases in the training set to be C and randomly select cases with replacement.

Step 3: if there are inputs of M features with a variable representing number m<M being specified such

© 2020, IJSRCSE All Rights Reserved

that at each node, "m" variables are randomly selected out of "M". Take the best split on the node and peg the value of "m" to be constant while we build-up the forest.

Step 4:Grow each decision-tree to have the largest possible size without pruning

Step 5: Predict with n_estimators using majority vote for classification and mean for regression.

Step 6: Stop

Evaluation of algorithms

This is the last and final step to effectively measure the performance our proposed techniques. We proposed the use of mean square error(MSE), mean absolute error(MAE), Root mean square error(MSE), Prediction accuracy and training time complexity to adequately measure the performance of the proposed system for comparison.

III. RESULTS AND DISCUSSION

We discussed about different ML classifiers and adopted the random forest classification and regression classes. And implemented the model with specific number of n_estimators(decision trees) set to 10, 20, 30, 40, 50, 60 and 70 decision trees for performance evaluation.

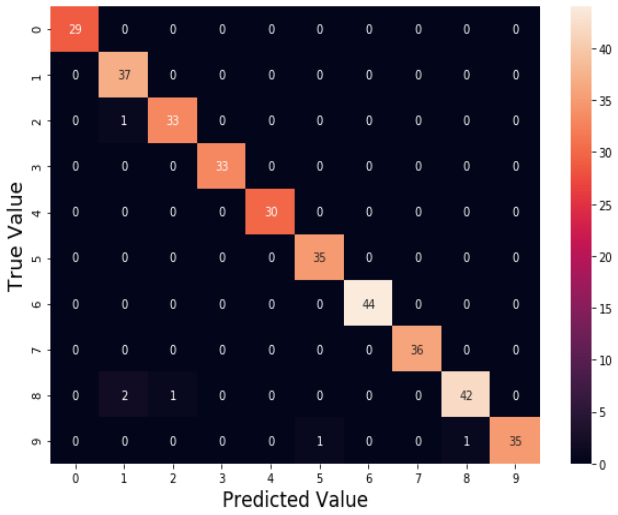

Figure 3: The Confusion Matrix of RF classification

The confusion matrix is employed to evaluate the performance of the proposed system classifiers on set of test data for which the true values are known. The main diagonal elements or values shown in figure 2 represent the total number of points for which the predicted grayscale digits are equal to the actual or true values. The wrongly predicted grayscale digits falls-off the main diagonal called the off-diagonal elements. The higher the diagonal values the better the predictions. The increasing diagonal values of the confusion matrix from 25, 29, 30, 41, 36, 35, 42 to 44 shows evidently the increasing performance of the model in terms of prediction.

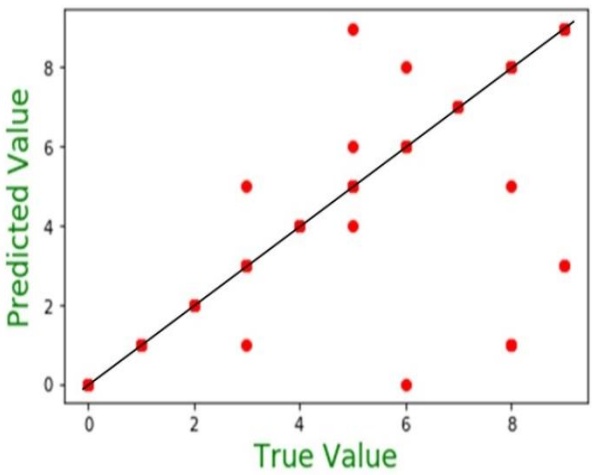

Figure 4: The prediction graph of RF Regression

Figure 5 is the plot r graph of RF predicted values against the true values. The correct predicted points lies within the straight line while the points outside the line are the wrongly predicted points of grayscale digits(4 points above and 6 points below) by the proposed system regression model.

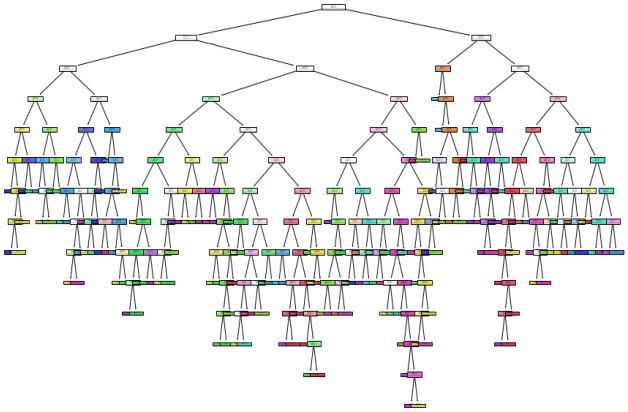

Figure 5: The RF generated from the proposed model

Figure 5 depicts the random forest generated from the proposed system model after training which combined the simplicity of different decision-trees resulting in a vast improvement in accuracy using the MNIST dataset. This was done by choosing samples randomly from the original set with replacement to construct trees using variables to represent random subset at each step. This resulted in a wide variety of forest trees as shown in figure 4.

Table 1: Comparing the accuracy of random forest algorithms

Total No. of Trees	RF_Classifier	RF_Regressor
n_estimator=10	94.72	61.0
n_estimator=20	95.2	80
n_estimator=30	96.0	84.0
n_estimator=40	97.0	85.0
n_estimator=50	98.0	86.0
n_estimator=60	99.17	91.50
n_estimator=70	99.17	91.94

© 2020, IJSRCSE All Rights Reserved

Table 1 shows the change in prediction accuracy with respect to the number of estimators starting from 10, 20, 30-to-70 forest trees. The metrics of prediction accuracy increases as we grow and add more trees to the forest space for both models. The RF_classification model produced 94.72%, 95.2%, ..., 99.7% when the number n_estimators are set to 10, 20, 30,....,70 compared to the RF_regressor model which was not measurable.

Training time variation of RF_classification and Regressor class

The training time variation of the proposed model is the time taken to train the model with specific number of n_estimators in predicting grayscale digits on images shown in table 2 above.

Table 2: Comparing the training time complexity of both models measured in seconds

Total No. of Trees	RF_Classifier	RF_Regressor
n_estimator=10	1.24	1.74
n_estimator=20	1.40	2.73
n_estimator=30	1.77	3.91
n_estimator=40	2.08	4.67
n_estimator=50	2.41	5.57
n_estimator=60	2.57	6.33
n_estimator=70	2.82	6.73

The training time complexity of RF_classification and RF_regressor are recorded to be 1.24 and 1.74 seconds when the trees are set to 10 and 20, 30 to 70 estimators as shown in table 2. Results from table 2 shows that the RF classification performed faster compared to the RF regression classifier at every point. The proposed system training time varies with the number of estimators as it increases when more trees are grown within the forest.

Table 3:Comparing the performance of RF algorithms

RF algorithms	n_estimatos	MAE	MSE	RMSE
RF Classifier		0.191	0.919	0.959
RF Regression	10	0.613	1.208	1.099
RF Classifier		0.187	0.853	1.074
RF Regression	20	0.685	1.273	1.128
RF Classifier		0.144	0.711	0.843
RF Regression	30	0.706	1.357	1.165
RF Classifier		0.154	0.533	0.730
RF Regression	40	0.605	1.09	1.04
RF Classifier		0.142	0.803	0.696
RF Regression	50	0.679	1.389	1.178
RF Classifier		0.047	0.336	0.486
RF Regression	60	0.631	1.096	1.047
RF Classifier		0.041	0.232	0.467
RF Regression	70	0.524	0.748	0.864

The increasing value of n_estimators decreases the MAE, MSE in classification and increases the RMSE value of the regression model but later decreased in value. The RF_Classification recorded the least MAE, MSE and RMSE rates compared to the RF_regression classifier. In comparison the RF_Classification model produced the best result in predicting digits on grayscale images than the RF_regressor classifier.

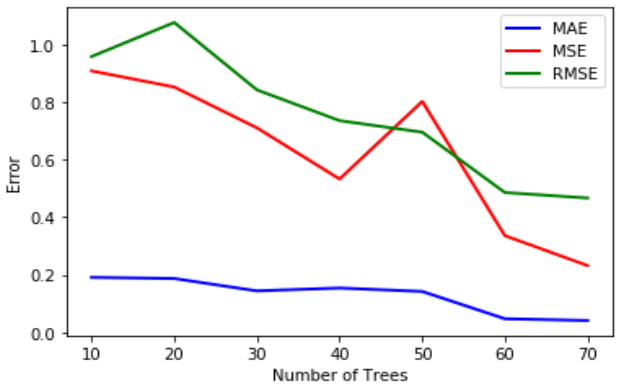

Figure 6: The error terms

Figure 6 shows the MAE, MSE and RMSE used to measure the performance of the system model represented with green, red and blue colored lines. The error terms of proposed model decreases along the horizontal plane based on the number of estimators at different rate. The MSE represented by the red-line decreased from point 10 to 40 and increased from 40 to 50 and converges from point 60. Both error terms tends to be aligned at point 60.

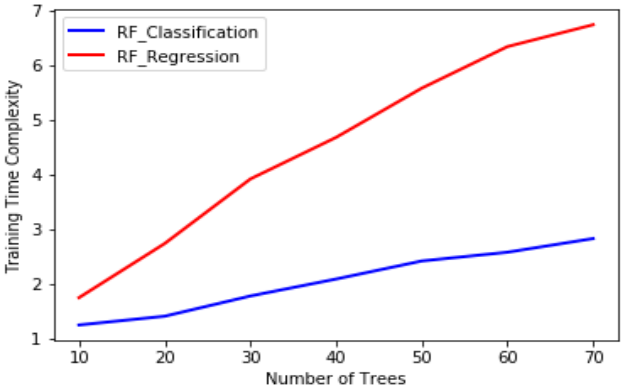

Figure 7: The Training time complexity of RF

Figure 7 shows the training time variation graph for both models with respect to the number of grown decision trees in the RF space. The simulation plot of RF classification model is recorded to be lower and faster compared to the RF_regressor classifier as shown in figure 7. The training time of the proposed model increases along with the number of n_estimators(decision trees).

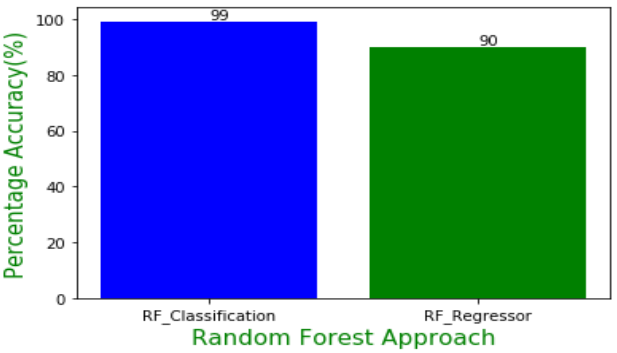

Figure 8: Accuracy Graph of RF classification and regressor model

© 2020, IJSRCSE All Rights Reserved

Figure 8 depicts the recorded success metrics of the RF classification and regression classifiers measured in percentage. The RF_Classification classifier produced 99% metrics of accuracy which was higher compared to the RF_regressor classifier with success rate of 90%.

II. CONCLUSION

The existing methods of practice are inefficient in terms of accuracy, training time complexity and error rates for predicting grayscale digits on images. From the experimental results the performance metrics of the RF classification model increases along with the training time complexity which is higher in accuracy. And the error rate converges faster and tends toward the zero point as we construct and add more trees with an incremented value of ten estimators to the RF space.

The experimental results proved to be highly accurate in prediction and errors converges as required when more trees are grown and added to the forest space. We therefore; evidently conclude that the RF classification tree model performed better as required than the random regression model.

REFERENCES

[1]. H. T. Malazi, and M. David, (2018). "Combining Emerging Patterns with Random Forest for Complex Activity Recognition in Smart Homes," *Applied Intelligence,* ***Vol. 48, Issue. 2, pp.315-330, 2018.***

[2]. Y. K. Chen, and J. F. Wang, "Segmentation of Single- or Multiple-Touching handwritting Numerial String Using Background and Foreground Analysis," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* ***Vol. 22, Issue. 11, pp.1304-1317,*** **2000**

[3]. V. Y. Kulkami, and P. K. Sinha, "random Forest Classifiers: A Survey and Future Research Directions," *International Journal of Advanced Computing* ***Vol. 36, Issue. 1, pp.1144-1153,2013.***

[4]. L. Breiman, "Random Forest," *Machine Learning,* ***Vol. 45, Issue. 1, pp.279-290, 2001.***

[5]. R. Anitha, and S. R. D. Siva, "Development of Computer-aided Approach for Brain Tumor Detection using Random Forest Classifier," *International Journal of Imaging Systems and technology,* ***Vol. 28, Issue. 1, pp.48-53,2018.***

[6]. V. Lepetit, P. Fua, (2006). "Keypoint Recognition Using Randomized trees," *IEEE Trans. Pattern Anaysis Machine Intelligence* ***Vol. 28, Issue. 9, 1465-1479, 2006.***

[7]. 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),* ***pp.1043-1047, 2007.***

[8]. N. Arica, and F. T. Yarman-Vural, "Optical Character Recognition for Cursive Handwritting," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* ***Vol. 24, Issue. 6, pp.801-113, 2002.***

[9]. L. Kuncheva, "Combining Pattern Recognition: Methods and Algorithms," *John Wiley and Sons,* ***pp.60-75, 2004.***

[10]. Y. K. Vrushali, and K. S. Pradeep, "Effective Learning and Classification Using Random Forest Algorithm," *International Journal of Engineering and Innovative Technology(IJEIT)* ***Vol. 3, Issue. 11, pp.267-273, 2014.***

[11]. A. Jehad, K. Rehanulla, A. Nasir, and M. Imran, "Random Forest and Decision Trees," *International Journal of Computer Science(IJCS),* ***Vol. 9, Issue. 5, pp.273-278, 2012***

[12].H. Mishra, S. K. Pathak, and A. Srivastava, "Handwritten Digits Recognition Using Machine Learning Algorithms", *International Research Journal of Engineering and Technology(IRJET),* ***Vol. 07, Issue. 05, pp.8073-8079,2020.***

[13]. V. Garg, "Handwritten Digit Classification Using Machine Learning Models," *International Research Journal of Engineering and Technology(IRJET),* ***Vol. 06, Issue. 11, pp.3593-3597, 2019.***

[14]. K. N. H. Al-Behadili, "Classification Algorithms for Determining Handwritten Digit," *Iraqi Journal for Electrical and Electronic Engineering(IJEEE),* ***Vol. 12, Issue. 1, pp.96-102, 2016.***

[15]. V. P. Dhaka, and M. K. Sharma, "An Efficient Segmentation Technique for Devanagari Offline-Handwritten Scripts using the Feedforward Neural Network," *Neural Computing and Applications,* ***Vol. 26, Issue. 8, pp.1881-1893, 2015.***

[16]. A. Singh, and R. Sathyaraj, "A Comparison Between Classification Algorithms on Different Datasets Methodologies using Rapidminer," *International Journal of Advanced Research in Computer and Communication Engineering(IJARCCE),* ***Vol. 5, Issue 5, pp.560-563, 2016.***

[17]. R. R. Rani, P. Manikadan, and R. D. Chitra, "An Empirical Analysis of Classification Trees Algorithm for Protein Datasets," *International Journal of Advanced Research in Computer Science(IJARCS),* ***Vol. 6, Issue. 6, pp.34-39, 2015.***

[18]. Y. Chherawala, P. P. Roy, and M. Cheriet, "Feature set Evaluation for Offline Handwriting Recognition systems: Application to the Recurrent Neural Network Model," *IEEE transactions on cybernetics,* ***Vol. 46, Issue. 12, pp.2825-2836, 2016.***

[19]. S. Joseph, and J. George, "Feature Extraction and Classification Techniques of MODI Script Character Recognition," *Pertanika Journal of Science and Technology(PJST),* ***Vol. 27, Issue. 4, pp.1649 - 1669, 2019***

[20]. R. Anushri, N. Ashitha, and Mamath, "Digit Recohnition," *International Advanced Research Journal in Science, Engineering and Technology(IARJSET),* ***Vol. 4, Issue. 8, pp.44-50, 2017.***

[21]. G. Pirlo, and D. Impedovo, "Fuzzy Zoning Based Classification for Handwritten Characters," *IEEE Transaction on Pattern Recognition and Machine Intelligence,* ***Vol. 19, Issue. 04, pp.780-785, 2011.***

[22]. S. Vinneet, and P. L. Sunil, "Digits recognition using single layer neural Network with principal component analysis," *Computer Science and Engineering (APWC on CSE), Asia-Pacific World Congress IEEE,* ***pp.4-5, 2014.***

[23]. J. Puzicha, M. Held, J. Ketterer, J. M. Buhmann, and D. W. Fellner, "On Spatial Quantization of Color Images," *IEEE Transactions on Image Processing 9,* ***pp.666–682, 2000.***

[24]. T. Horiuchi, "Grayscale Image Segmentation using Color Space," *Institute of Electronics, Information and Communication Engineers(IEICE) Trans. Infomation System,* ***Vol. E89-D, Issue. 3, pp.1231-1235, 2006.***

[25]. S. Ziweritin, C. C. Ukegbu, and E. U. Ezeorah, Building Data Mining Classification Model for Pixilated Digit Recognition System," *SSRG International Journal of Computer Science and Engineering(IJCSE),* ***Vol. 7, Issue. 10, pp.6-12, 2020.***

[26]. A. Liaw, and M. Wiener, "Classification and Regression by Random Forest," *R News,* ***Vol. 2, Issue. 3, pp.18–22, 2002***

[27]. L. Kuncheva, and C. Whitaker, "Measures of Diversity in Classifier Ensembles and their Relationship with the Ensemble Accuracy," *Machine learning,* ***Vol. 51. Issue. 2, pp.181–207, 2003.***

[28]. 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),* ***Vol. 5, Issue. 02, pp.493- 496, 2019.***

[29]. J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: A New Classifier Ensemble Method," *IEEE Transactions*

© 2020, IJSRCSE All Rights Reserved

on Pattern Analysis and Machine Intelligence(TPAMI), ***Vol. 28, Issue. 10, pp.1619 - 1630, 2006.***

[30]. E. Kremic, and A. Subasi, "Performance of Random Forest and SVM in Face Recognition," *International Arab Journal of Information Technology(IAJI),* ***Vol. 13, Issue. 2, pp.288-292, 2016.***

AUTHORS PROFILE

Mr. Stanley Ziweritin, Completed HND(Computer Science) from Rivers State Polytechnic, Bori. PGD and M.Sc(Computer Science) from University of Portharcourt(UPH). Presently working as Principal Instructor in the Department of Estate Management and Valuation, Akanu Ibiam Federal Polytechnic, Unwana. His research interest include: AI, Data Mining, Fuzzy Logic, Machine Learning(ML), Algorithms, Database System Design and programming, Traditional and Machine Learning Programming. He has several years of teaching and reaearch experience. He has published several International Journals in Machine Learning and Data Mining.

Mr. **C. A. Uchenna Ugboaja,** Completed his HND(Computer Science) from the Institute of Management and Technology(IMT), Enugu. PGD in Project Managemnt, PGD and MSC in Computer Science from Federal University of Technology(FUTO) Owerri and Ebonyi State University(EBSU) respectively. He was the Assistant Co-ordinator in School of Continue Education since 2008-2012 and was appointed as HOD pre-ND Sciences from 2013-2014, HOD Computer Science Department and Dean School of Science since 2014-2016. Presently He is the director of the institute Akanu Ibiam Federal Polytechnic, Unwana ICT Unit. He has published serveral International Journals. He is a memeber of Nigeria Computer Society(NCS), Computer Professionals(Registraton Council) of Nigeria(CPN), British Computer Society(BCS), Nigerian Society of Engineers(NSE). He has several years of research experience. His research interest include: AI and Information Communicatooon Technology(ICT)

Mr. Chidiebere Moses Osu, Completed HND(Computer Science) from Akanu Ibiam Federal Polytechnic, Unwana and B.Sc(Computer Education) from Enugu State University of Science and Technology. He is currently a Senior Technologist, ICT unit in Akanu Ibiam Federal Polytechnic Unwana. He has published many journal papers in Cloud Computing, Internet of Thing IoTs, and Database Design. Research interest include: Programming, Cloud Computing, Hardware Maintenance and Data Mining.

© 2020, IJSRCSE All Rights Reserved