International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 02 | Feb 2020 www.irjet.net p-ISSN: 2395-0072

Effective Workflow for High-Performance Recognition of Fruits using Machine Learning Approaches

Abdul Kader[1], Samiha Sharif[2], Pranta Bhowmick[3], Fahmida Haque Mim[4], Azmain Yakin Srizon[5]

[1,2,3,4]Dept. of CSE, Bangladesh Army University of Engineering & Technology, Bangladesh
[5]azmainsrizon@gmail.com, Lecturer, Dept. of CSE, Bangladesh Army University of Engineering & Technology, Bangladesh

---***---

Abstract - *Recognition system is an essential combat of computer science. Fruit recognition is one of them. Researcher have been roaming around this area for a decade now. Previously, many traditional machine learning techniques and deep learning methods have been employed for successful recognition of fruits and in some cases, a high accuracy have been achieved. However, in our study we showed that our proposed working diagram outperformed all the previous studies. In this paper, we started with Fruit-360 dataset of 109 distinct classes and applied three feature extraction methods (hu moments, haralick texture and color histogram). After that we applied several machine learning approaches (Decision Tree, K-Nearest Neighbors, Linear Discriminant Analysis, Logistic Regression, Na�e Bayes, Random Forest and Support Vector Machine) to train the models. Finally, the test results were calculated and K-Nearest Neighbor along with Random Forest classifiers produced best results with a false positive rate of 0%, hence, achieving a high accuracy.*

Key Words: **Image Recognition, HU Moments, Haralick Texture, Color Histogram, Decision Tree, K-Nearest Neighbors, Na�e Bayes, Linear Discriminant Analysis, Logistic Regression, Random Forest, Support Vector Machine, ROC Curve Analysis**

1. INTRODUCTION

Recognition practice has risen as a 'grand challenge' for computer vision, with the longer-term intention of being capable to produce a near-human level of recognition for tens of thousands of classes beneath a broad diversity of circumstances like visual perception, sound realization, voice identification, handwriting verification, text recognition and so on [1]. The face identification system automatically identifies the input face pictures from digital image processing. Text classification is practiced to recognize the different types of text for example spam or non-spam email [2]. However, in this research, our main purpose was the recognition of fruits.

Color and texture are the primary aspects of natural pictures and perform an essential part in visual understanding. It is frequently beneficial to explain monochrome strain by developing contrast or detachment. The manner of color categorization includes the extraction of valuable information concerning the spectral characteristics of object exteriors and determining the most suitable match from a set of associated records or class models to perform the recognition task [3]. The texture is one of the most dynamic subjects in machine intelligence and pattern investigation ever since the 1950s which sought to distinguish different decorations of images by removing the dependency of intensity between pixels and their adjacent pixels [4] or by capturing the variance of intensity across pixels.

Previously, several studies have been performed for the successful recognition of fruits. A recent study on fruit recognition from 38,409 images of 60 fruits using deep learning has achieved an overall accuracy of 96.3% [5]. A framework called Pure-CNN achieved a classification accuracy of 98.88% for 81 categories of fruit-360 data recently [6]. With the help of the K-Nearest algorithm technique, a research study achieved an accuracy of 95% [7]. Another research suggested an overall accuracy of 98.33% using neural classifiers [8]. However, in our research, we studied a total of 109 fruit classes with 74,572 images. In this study, we've shown that our proposed working model has outperformed all previous studies.

2. MATERIALS & METHODS

This section starts with the dataset description. After that, the proposed model has been described. Next, several machine learning techniques have been described briefly. Finally, performance measurement has been discussed in terms of different performance measurements.

2.1 Dataset Description

The fruit-360 dataset is one of the most traditional datasets that includes 74,572 pictures of 109 assorted fruits [9]. Fruits were located in the shaft of a low-speed motor of 3 rpm and a small documentary of 20 seconds was videotaped. A Logitech C920 camera was utilized for recording the outcomes. This is one of the most trustworthy webcams available. Behind the fruits, a white layer of paper was installed as a background. Nevertheless, due to the fluctuations in the lighting circumstances, the background was not consistent and the dataset creators composed a dedicated algorithm that extracts the fruit from the background. For further processing, the dataset was divided into train and test set. The train set contained 90% of total data having a total of 67,114 images. The test set had a total of 7,458 images.

2.2 Proposed Recognition Workflow

Firstly, the whole dataset was taken as input and several feature extraction techniques had been performed i.e., Hu Moments, Haralick Texture and Color Histogram. Secondly, the features and labels of the images were saved in a numpy array for further processing [10]. Thirdly, the dataset was split into train and test set. The train set contained 90% of the data where the test set contained 10% of the whole dataset. After that, we applied several machine learning techniques (Decision Tree, K-Nearest Neighbors, Linear Discriminant Analysis, Logistic Regression, Na�e Ba yes, Random Forest, and Support Vector Machine) to train the model. With the help of 10-fold cross-validation, we validated the overall accuracy of the trained model. Finally, the models were applied to the test set for further performance analysis. Our proposed working diagram has been shown in Figure-1.

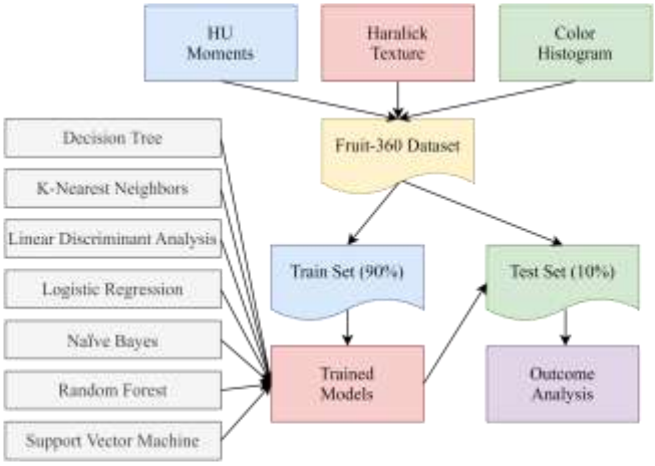

Figure-1: Proposed workflow diagram

2.3 Feature Extraction

Feature extraction starts from an initial set of measured data and builds derived values intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. In this study, we've used three feature extraction techniques: HU Moments, Haralick Texture and Color Histogram.

2.3.1 HU Moments

Hu Moments are a collection of 7 estimates determined using central moments that are invariant to image alterations. The first 6 moments have been determined to be invariant to translation, scale, and rotation, and reflection while the 7th moment's sign varies for image reflection [11].

2.3.2 Haralick Texture

Texture defines the persistence of decorations and colors in an object/image such as woods, water, rocks, sands, etc. To distinguish targets in an image based on texture, one has to examine the uniform spread of decorations and colors in the objects covering. Rough-Smooth, Hard-Soft, Fine-Coarse are some of the texture combinations one can imagine of, although there are several such combinations. Haralick texture is applied to quantify an image based on texture. It was developed by Haralick in 1973 [12]. The primary thought associated with measuring Haralick Texture characteristics is the Gray Level Co-occurrence Matrix or GLCM.

2.3.3 Color Histogram

A histogram is recognized as a diagram or design which is related to the regularity of pixels in a grayscale picture with pixel values differing from 0 to 255. A grayscale photograph is an image in which the value of an individual pixel is a unique sample, that is, it brings only depth information where pixel value differs from 0 to 255. Pictures of this kind, also recognized as black-and-white, are formed solely of shades of gray, ranging from black at the lowest intensity to white at the strongest where a pixel can be viewed as a point in an image.

2.4 Machine Learning Techniques

Several machine learning techniques were applied in this study. All these approaches have been described in this section.

2.4.1 Decision Tree

Decision Tree algorithm is a sorting algorithm for constructing a decision [13]. This makes occasionally a variance which means that in CART the conclusions on how to split values based on an attribute are delayed.

2.4.2 K-Nearest Neighbors

The K-Nearest Neighbors algorithm is a simple, easy-to-implement exercised machine learning algorithm that can be utilized to solve both classification and regression matter [14]. The algorithm is handy, humble and informal to implement. There's no requirement to form a model, tune numerous parameters, or make additional assumptions.

2.4.3 Linear Discriminant Analysis

Linear Discriminant Analysis or LDA is a dimensionality shortening procedure [15]. It is operated as a pre-processing stage in Machine Learning and applications of pattern grouping. The purpose of LDA is to project the characteristic in higher dimensional place onto a lower-dimensional place to avoid the damnation of dimensionality and also reduce support and dimensional spending.

2.4.4 Logistic regression

Logistic regression is an adjustment algorithm employed to produce measurements to a discrete set of groups [16]. Opposite linear regression that outputs continuous number worth, logistic regression changes its output approaching the logistic sigmoid function to proceed back a possible value which can then be outlined to two or more discrete groups.

2.4.5 Na�e Bayes

Na�e Bayes is a probabilistic machine learning algorithm that can be applied in a wide-ranging variety of classification tasks [17]. It is very simple, relaxed to implement and fast, required fewer training data. If the Na�e Bayes con ditional individuality statement holds, then it will appear quicker than discriminative models like logistic regression.

2.4.6 Random Forest

Random Forest is grounded on decision trees [18]. In machine learning, decision trees are a fetch for creating predictive models. It is practiced to resolve equally classification as well as regression problems. Random Forest can habitually handle lost values. It is fairly less impacted by noise. Random Forest algorithm is right stable. Even if a new data point is announced in the dataset, the total algorithm is not affected much since the new data may affect one tree.

2.4.7 Support Vector Machine

Support Vector Machine or SVM algorithm is an easy yet strong supervised machine learning algorithm that can be exercised for constructing both regression and arrangement models [19]. SVM algorithm can act really well with both linearly dividable and non-linearly separable datasets.

2.5 Performance Measurement

For measuring the performance of the classifiers, we've applied the measurements of sensitivity, specificity, precision, f-score, false-positive rate, and overall accuracy. All of these phenomena have been described below. Here, TP, TN, FP, and FN correspond to true positive, true negative, false positive and false negative respectively. ROC curve analysis was also performed in our study.

2.5.1 Sensitivity

Sensitivity is described as the probability of accurately recognizing some conditions. For example, the sensitivity might be employed in a medical investigation to report that a particular test has an 80% probability of identifying anabolic steroid utilized by an acrobat. Sensitivity is calculated with the following formula:

Sensitivity = TP/(TP+FN)

2.5.2 Specificity

Specificity contains a test's strength to precisely produce a negative result for characters who don't hold the condition that's being tested for. A high-specificity test will perfectly rule out nearly everyone who doesn't have the condition and won't produce many false-positive results. Specificity is measured with the following formula:

Specificity = TN/(TN+FP)

2.5.3 Precision

Precision points to how familiar estimations from separate samples are to each other. For example, the standard error is an example of precision. When the standard error is little, estimations from different samples will be alike in value; and vice versa. Precision is measured as follows:

Precision = TP/(TP+FP)

2.5.4 F-Score

The F score is determined as the weighted harmonic mean of the test's precision and recall. This score is measured according to the precision and recall of a test practiced into account. F-Score is estimated with the help of the following formula:

F-Score = 2TP/(2TP+FP+FN)

2.5.5 False Positive Rate

In statistics, when conducting various comparisons, a false positive ratio is the probability of incorrectly discarding the null hypothesis for a distinct test. The false-positive rate is determined as the ratio between the number of negative results incorrectly classified as positive (false positives) and the total amount of actual negative results.

False Positive Rate = FP/(FP+TN)

2.5.6 Overall Accuracy

Overall accuracy is the possibility that a sample will be accurately matched by a test; that is, the total of the true positives and true negatives divided by the total number of individuals examined that is the sum of true positive, true negative, false positive and false negative. However, overall accuracy doesn't show the actual performance as sensitivity and specificity may differ despite having a higher accuracy. Overall accuracy can be estimated as follows:

Overall Accuracy = (TP+TN)/(TP+TN+FP+FN)

Table-1: Sensitivity, specificity, precision, f-score, false-positive rate (FPR) and accuracy measurement for Decision Tree (DT), K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Logistic Regression (LR), Naïve Bayes (NB), Random Forest (RF) and Support Vector Machine (SVM) where the range of the measurements is [0,1]

Approach	Sensitivity	Specificity	Precision	F-Score	FPR	Accuracy
DT	0.99990	0.99038	0.99990	0.99990	0.00961	0.99021
KNN	**1.00000**	**1.00000**	**1.00000**	**1.00000**	**0.00000**	**1.00000**
LDA	0.99985	0.98418	0.99985	0.99985	0.01581	0.98417
LR	0.99997	0.99757	0.99997	0.99997	0.00242	0.99758
NB	0.99967	0.96984	0.99967	0.99967	0.03015	0.96486
RF	**1.00000**	**1.00000**	**1.00000**	**1.00000**	**0.00000**	**1.00000**
SVM	0.99817	0.82869	0.99816	0.99815	0.13461	0.80155

2.5.7 ROC Curve Analysis

ROC curves are regularly utilized to present graphically the association/trade-off among sensitivity and specificity for every reasonable cut-off for a test or a sequence of tests [20]. Besides, the section under the ROC curve provides an opinion about the advantage of practicing the test in question. As the field under a ROC curve is a measure of the usefulness of a test in general, where a larger area indicates a more valuable test, the areas under ROC curves are adopted to distinguish the usefulness of tests. There are two types of ROC curves in terms of multi-class ROC curve analysis: macro-average ROC curve and micro-average ROC curve. A macro-average ROC curve measures the metric freely for each class and then takes the average, whereas a micro-average ROC curve aggregates the participation of all classes to calculate the average metric. Although in a multi-class classification setup, micro-average is favored, in our study, we took both of them under consideration.

3. Experimental Analysis

The experiment started by taking the fruit-360 dataset. At first, three feature extraction processes were applied on the whole dataset: hu moments, haralick texture and color histogram. All these results were then stored in numpy array for further processing. Next, the dataset was split into train and test sets. The train set contained 90% of the data and the test set contained the rest 10%. After that, several machine learning techniques were performed on the train set for fitting the models. Scikit learn was used for training these models. While implementing the models, all the parameters were set to default. 10-fold cross-validation was used for validating the train set results. After the validation, trained models were performed on test data. Sensitivity, specificity, precision, f-score and false positive rate were measured for each class and the mean of these measurements was considered as the overall sensitivity, specificity, precision, f-score, and false-positive rate respectively for a specific model. Overall accuracy was also measured for each of the models. Table-1 shows all the measurements for all the models.

Figure-2 shows the comparison of accuracy for the machine learning classifiers: decision tree, k-nearest neighbors, linear discriminant analysis, logistic regression, naïve bayes, random forest and support vector machine.

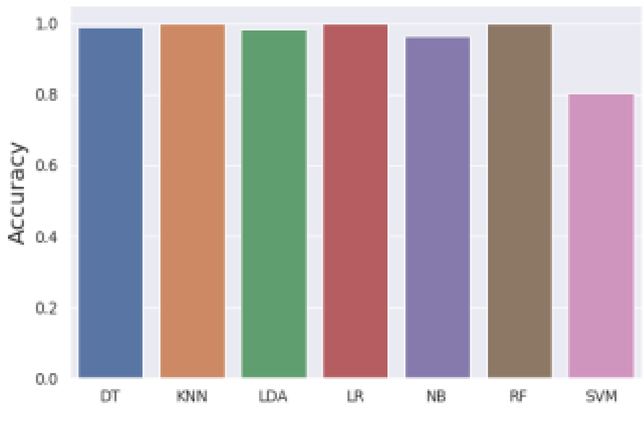

Figure-2: Comparison of accuracy among all seven machine learning classifiers

From Table-1 and Figure-2, it can be assured that in our proposed model K-Nearest Neighbor and Random Forest algorithms outperforms other methods. Moreover, our results also outperformed previous studies. However, for further evidence, the ROC curve analysis was performed as well. We generated both macro-average and micro-average ROC curves. Figure-3 shows the macro-average ROC curve analysis for all seven classifiers.

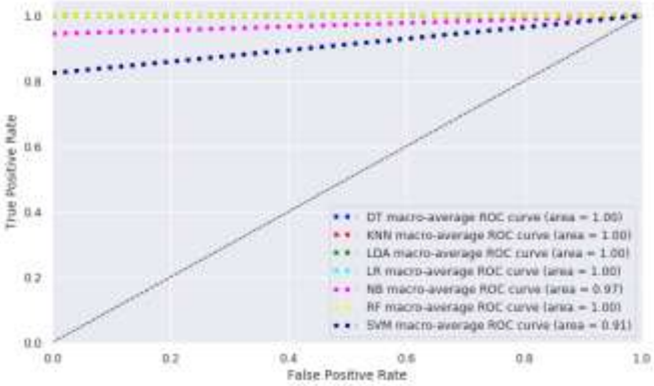

Figure-3: Macro-average ROC curve analysis for seven classifiers

In our research, we also found out the micro-average ROC curve. Figure-4 shows the micro-average ROC curve analysis for all seven classifiers.

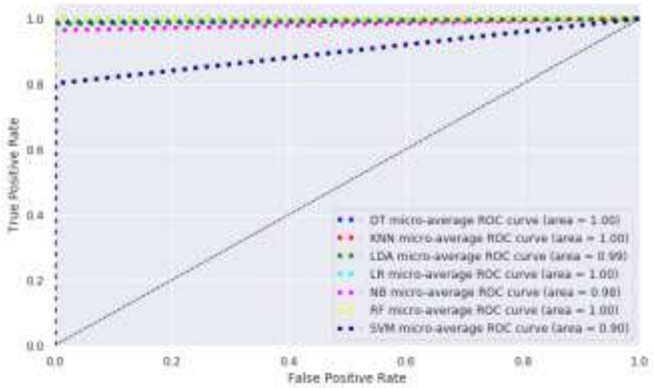

Figure-4: Micro-average ROC curve analysis for seven classifiers

Hence, from all these experimental results, we concluded that our proposed model outperformed all other studies of fruit recognition for fruit-360 datasets for the K-Nearest Neighbors and Random Forest classifiers.

4. CONCLUSION

Researches for fruit recognition on fruit-360 dataset has been circulating through the research community for a long time now. So far, many studies have been performed and a higher accuracy has been achieved on some of the cases. But in our study, we proposed a working diagram that produced an extremely well accuracy for K-Nearest and Random Forest classifiers. Fruit-360 is updated regularly. Hence, there's always a chance for improvement in this sector. But for the time being our proposed method outperformed all previous studies.

REFERENCES

[1] Duda RO, Hart PE, Stork DG. Pattern classification. John Wiley & Sons; 2012 Nov 9.

[2] Bruce V, Young A. Understanding face recognition. British journal of psychology. 1986 Aug;77(3):305-27.

[3] Kartikeyan B, Sarkar A. An identification approach for 2-D autoregressive models in describing textures. CVGIP: Graphical Models and Image Processing. 1991 Mar 1;53(2):121-31.

[4] Sahin, F., 1997. A radial basis function approach to a color image classification problem in a real time industrial application (Doctoral dissertation, Virginia Tech).

[5] Mureşan H, Oltean M. Fruit recognition from images using deep learning. Acta Universitatis Sapientiae, Informatica. 2018 Aug 1;10(1):26-42.

[6] Kausar, A., Sharif, M., Park, J. and Shin, D.R., 2018, December. Pure-CNN: A Framework for Fruit Images Classification. In 2018 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 404-408). IEEE.

[7] Ninawe P, Pandey S. A completion on fruit recognition system using k-nearest neighbors algorithm. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET). 2014 Jul;3(7):2352-6.

[8] Priya MM, Sujatha N. RECOGNITION OF FRUITS USING MULTIPLE FEATURES WITH NEURAL CLASSIFIERS. Asian Journal of Computer Science And Information Technology. 2017 Jul 3;7:37-9.

[9] Mureşan H, Oltean M. Fruit recognition from images using deep learning. Acta Universitatis Sapientiae, Informatica. 2018 Aug 1;10(1):26-42.

[10] Walt SV, Colbert SC, Varoquaux G. The NumPy array: a structure for efficient numerical computation. Computing in Science & Engineering. 2011 Mar; 13(2):22-30.

[11] Huang Z, Leng J. Analysis of Hu's moment invariants on image scaling and rotation. In2010 2nd International Conference on Computer Engineering and Technology 2010 Apr 16 (Vol. 7, pp. V7-476). IEEE

[12] Haralick RM, Shanmugam K, Dinstein IH. Textural features for image classification. IEEE Transactions on systems, man, and cybernetics. 1973 Nov(6):610-21.

[13] Safavian SR, Landgrebe D. A survey of decision tree classifier methodology. IEEE transactions on systems, man, and cybernetics. 1991 May; 21(3):660-74.

[14] Dudani SA. The distance-weighted k-nearest-neighbor rule. IEEE Transactions on Systems, Man, and Cybernetics. 1976 Apr(4):325-7.

[15] Balakrishnama S, Ganapathiraju A. Linear discriminant analysis-a brief tutorial. Institute for Signal and information Processing. 1998 Mar 2;18:1-8.

[16] Kleinbaum DG, Dietz K, Gail M, Klein M, Klein M. Logistic regression. New York: Springer-Verlag; 2002 Aug.

[17] Zhang H. The optimality of naive Bayes. AA. 2004;1(2):3.

[18] Liaw A, Wiener M. Classification and regression by randomForest. R news. 2002 Dec 3;2(3):18-22.

[19] Cortes C, Vapnik V. Support-vector networks. Machine learning. 1995 Sep 1; 20(3):273-97.

[20] Fawcett T. An introduction to ROC analysis. Pattern recognition letters. 2006 Jun 1; 27(8):861-74.