Engineering and Technology Journal
Journal homepage: engtechjournal.org

Using Texture Feature in Fruit Classification

Mauj H. Abd al karim [a]*, Abdulamir A. Karim [b]

[a] Department of Computer Science,University of Technology, Baghdad, Iraq, cs.19.69@grad.uotechnology.edu.iq

[b]Department of Computer Science,University of Technology, Baghdad, Iraq, 110004@uotechnology.edu.iq

*Corresponding author.

Submitted: 18/06/2020 Accepted: 02/12/2020 Published: 25/03/2021

K E Y W O R D S

HoG, GLCM, LBP, Decision Tree, Fruit Classification.

A B S T R A C T

Recent advances in computer vision have allowed wide-ranging applications in every area of life. One such area of application is the classification of fresh products, but the classification of fruits and vegetables has proven to be a complex problem and needs further development. In recent years, various machine learning techniques have been exploited with many methods of describing the different features of fruit and vegetable classification in many real-life applications. Classification of fruits and vegetables presents significant challenges due to similarities between layers and irregular characteristics within the class.Hence , in this work, three feature extractor/ descriptor which are local binary pattern (LBP), gray level co-occurrence matrix (GLCM) and, histogram of oriented gradient(HoG) has been proposed to extract fruite features , the extracted features have been saved in three feature vectors , then desicion tree classifier has been proposed to classify the fruit types. fruits 360 datasets is used in this work, where 70% of the dataset were used in the training phase while 30% of it used in the testing phase. The three proposed feature extruction methods plus the tree classifier have been used to classifying fruits 360 images, results show that the the three feature extraction methods give a promising results , while the HoG method yielded a poerfull results in which the accuracy obtained is 96%.

How to cite this article: M. H. Abd al karim and A. A. Karim "Using Texture Feature in Fruit Classification," Engineering and Technology Journal, Vol. 39, Part B, No. 01, pp. 67-79, 2021.
DOI: https://doi.org/10.30684/etj.v39i1B.1741

1. INTRODUCTION

Modern time is a technical age in which to believe in the intellect cantered on intuition. The use of digital images has, therefore, been tremendous in everyday life. Photo analysis plays an important part in science and technology. The digital image processing technology is becoming efficient and cost-effective, so the shape, color, and texture of the objects were investigated using a variety of specific results techniques. It requires the identification, based on the digital vision, of the form, color, and texture of the object residing in the image. High-quality food production depends not only on an effective processing technology but also on the selection of the right raw materials. Well known among food manufacturers [1], this experience. Computer vision offers various solutions for reducing human effort in the agricultural field. Researchers have focused on several topics, such as finding weeds, fruit searching, monitoring the health status of trees, automatic fruit classification, determining fruit maturity, detecting diseases in trees, and fruit [2]. Food is very important to human life and fruits are one of the best natural foods. These fruits have lots of nutrients needed for our daily lives. Photos are the primary source of data and knowledge in the agricultural sector, with the use of image processing techniques having an important and fundamental impact on the study of farm processes. The methodology for image processing was widely used [3].

Fruit production plays a vital economic role in any region. It is part of agro-industrial production (juices, dried fruit, and jam). Fruits are of varying sizes, shapes, and colors, the process of determining fruit quality is very important in the export and packaging of foodstuffs that require a vision machine to check fruit quality and efficiency. The use of photography in the process of obtaining photographs of fruit is crucial since the photos are a major source in the agricultural sector sciences The collection of data and information, as well as the replication of this information and the documentation of the use of image processing, have prominent effects in the analysis of images and digital treatment works to enhance the image range The analysis of images and digital treatments in agricultural sciences has been adapted to differentiate between fruits in terms of quality[4].

There is the importance and need to classify the deceptions and fruits in the evaluation of the agricultural product to meet all quality standards to increase market value. The features that can be derived from fruits and vegetables from any image are shape, color, size, texture features. Hence, the classification process is based on these features [5].

In the classification process, fruits should be selected because they are considered accurate materials, where physical properties are the most important form of the fruit specimen, while color is the property of the visual, these characteristics may be affected by the process of transport or storage and marketing. The computer has been used in the field of image processing and computer vision, and it has become increasingly important in the fruit industry, quality inspection applications, and sorting [6].

Computer vision has emerged recently and has been considered as the technology standard in assessing the quality of non-destructive food where information can be extracted from visual properties. The quality of visual food is usually used. Many applications such as quality control, grades, measurements, defects detection, characterization, and other things. It is therefore best to resort to computer applications and image processing [7].

2. LITERATURE REVIEW

In this paragraph, different schemes have been proposed in fruit classification as shown below:

In 2016, Renold ,M .,and Evans ,M., The main aim of this research was to study the applicability and performance of the Naive Bayes algorithm in the classification of varieties of apple fruit. The technique included the collection, preprocessing, and segmentation of images, analyzing, and classifying apple varieties. Apple's prototype classification system was developed using MATLAB R2015a platform development environment. The results showed that the approximate accuracy, sensitivity, precision, and specificity average values were 91%, 77%, 100%, and 80 % respectively. Comparison of the results of their classification with that of the Naive Bayes technique showed that the accuracy of Naive Bayes was higher than the accuracy of the Naive Bayes was higher than the accuracy of principal components analysis, fuzzy logic, and MLP Neural with 91%, 90%, 89%, and 83% respectively [8].

In 2017, Ali, N. M et.al. present a Form recognition algorithm and Oriented Gradient Histogram theory for detecting and counting the total number of mangoes on the tree using a quadcopter with an attachable webcam. The traditional method of mango harvesting has its limitation which leads to degradation of the quality of the mango harvested. As a result, there will be a dampening of the production rate and tree structure. Thus, the use of the image processing algorithm may be a solution for the pre-harvesting process of a better and more reliable mango. This differentiates the mango and its leaf based on photographs taken on real scenes and therefore estimates the mango tree's growth rate for the time being. The mango tree's height and mango location do not impact the ability of the farmer to examine the mango as the drone hovers according to the intention of the user. The mango grower, the agricultural planner, and the investor are expected to have an alternative analysis. The highest detection rate (80,95 percent) was achieved when only two cascade training stages were used [9].

In 2018 A. Wajid, et al presents the mean for speedily distinguishing orange condition, Fruit image capabilities including RGB color space and BIC-based gray values (Border / Interior Pixel Classification) are extracted. An analysis has been carried out on the applicability and efficiency of various classification algorithms including Naïve Bayes, Artificial Neural Network, and Decision Tree. Comparisons have been drawn between the results of these algorithms, and it has been observed that the technique of classification of the Decision Tree for orange conditions is more effective than other techniques. The results obtained using this technique for accuracy precision, and sensitivity are 93.13%, 93.45%, and 93.24% respectively [10].
In 2019, Kavita. k and Dr. Sonia Process information which is stored in the form of pixels, the processing method of images is used. This process is associated with the nature of the orange fruit or an evaluation of quality. The Naive Bayes classifier is used in the current method, which gives the consistency and nature of the orange fruit evaluation cheap accuracy and time for execution. The Naive Bayes classification approach in this work cycle is changing with the help vector machine for assessing the consistency and nature of the orange fruit. The current approach is compared to the previous approach in the form of precision, time of execution, specificity, and sensitivity. It is observed after the analysis that the execution time is low as it compares with the previous approach and other parameters like- accuracy, specificity, and sensitivity of the current approach are high for the assessment of the nature of the orange fruit. The matrix algorithm with gray level co-occurrence is used for the extraction of the function. The results of the experiments were good and gave a high accuracy have been presented 97% [11].

3. THE PROPOSED METHOD

The framework of the proposed system comprises two main phases: a training phase and a testing phase. These two primary phases share the fundamental modules of the system (pre-processing module and features extraction module), as shown in Figure (1).

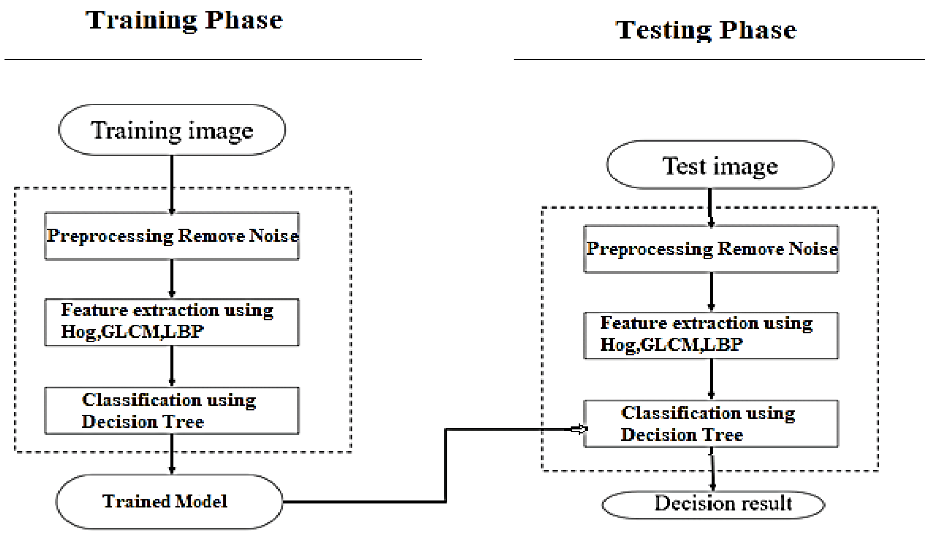

Figure 1: The framework of the proposed fruit classification.

In the training phase, after the pre-processing and feature extraction steps extracted features vector, for each training sample, the features vectors are saved in the system database. While in the classification phase the system should match these feature vectors of input fruit images with all vectors listed in the database and return the file name in the dataset classifier. Matching and decision steps in the proposed system have been done by using a decision tree.

I. Pre-processing stage

The first step is to apply pre-processing images to improve fruit images. This technique is used to improve the image to show the image detail well because sometimes the images may be taken in conditions that are inappropriate in terms of light, noise or the size of the image is very large and does not produce good results.

A. Convert RGB to a gray level

This step works to convert the original input models from the traditional RGB color format into the color format of a gray level. This is done to improve the perception of details in the Models by focusing on the brightness factor which makes it more accurate and specific than the common gray level format.

B. Bilateral Filter

After the process of converting the image to grayscale, improving the image and eliminate noise have been needed while preserving image accuracy and edges, therefore, a bilateral filter was used. This improved Gaussian method was used where the filters were improved by doubling the use of another Gaussian filter due to multiple iterations and different pixel density i.e. meaning that pixels with a density similar to the ones in the middle are included only to calculate an intense density value. As a result, this technique preserves the fruit edges, because the adjacent pixels are placed on the other side of the edge for pixels close to the edges and therefore large differences in density appear in a blur when compared to the central pixel [13].

C. Histogram equalization

In this process, histogram equalization is used for fruit images, to eliminate the effect of different illumination conditions, where images are taken. the color histogram of the grayscale image is equalized, so that, the features are more distinguishable by the classifier, the overall effect of the lighting in the environment is removed.

II. Feature Extraction

The most important stage in the identification system is feature extraction from samples such as fruit classification.

A. Histogram of Oriented Gradient (HOG) feature descriptor

In this system, the Histogram Oriented Gradient (HOG) algorithm is used for fruit classification, and the additional benefit of using this algorithm is that it will help extract image features. This will assist in training and testing the decision tree. HOG extracts the color feature, texture feature, and shapes the context feature. These features are stored in a database and used for decision tree training. Compute a Histogram of Oriented Gradients (HOG) includes five major steps:

The first stage applies an optional equalization of global image normalization designed to reduce the influence of the effects of the illumination. In practice, we use compression of gamma (power-law), either by computing the square root or the log of each color channel. The intensity of the image texture is usually proportional to the local illumination of the object and this compression helps to minimize the effects of local shadowing and lighting variations.

The second stage calculates gradients of the image in the first order. These capture contour, outline, and some detail about the texture, while providing additional resistance to variations in lighting. The locally dominant color channel is used, which in large measure provides color invariance. Variant methods can also include second-order image derivatives, which serve as basic bar detectors-a useful capture function, e.g. bar-like structures in bicycles and human limbs.

The third stage aims to create an encoding that is adaptive to the content of the local image while remaining immune to slight changes in pose or appearance. The adopted approach pools information on gradient orientation locally in the same way as the function on SIFT 2. The image window is split into small areas of space, called "cells." We accumulate a local 1-D histogram of gradient or edge orientations for each cell over all the pixels inside the cell. This combined cell-level 1-D histogram forms the basic representation of an "orientation histogram." Every histogram of orientation divides the angle of gradients into a fixed number of predetermined bins. The gradient of the pixels within the cell is used to vote for the orientation histogram.

The fourth stage measures normalization, which takes local cell groups and standardizes their overall responses before moving on to the next stage. Standardization provides greater invariance to contrasting lighting, shadowing, and ground. It is done by accumulating a measure of local histogram "energy" over local cell groups, which we call "blocks." Uses the result to normalize each block cell. Each cell is typically shared between several blocks, but its normalizations are dependent on blocks and thus different. Thus the cell appears multiple times with different normalizations in the final output vector. This can sound repetitive but performance improves the performance. We refer to the normalized block descriptors as Histogram of Oriented Gradient (HOG) descriptors.

The final step collects the HOG descriptors from all blocks of a dense overlapping grid of blocks covering the detection window into a combined feature vector.

$$D_x = [-1 \quad 0 \quad 1] \tag{1}$$

$$D_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \tag{2}$$

$$I_x = I \times D_x \tag{3}$$

$$I_y = I \times D_y \tag{4}$$

The magnitude of the gradiant

$$|G| = \sqrt{{I_x}^2 \; + \; {I_y}^2} \tag{5}$$

The orientation of the gradient

$$\theta = \tan^{-1} \frac{I_x}{I_y} \tag{6}$$

Below is the proposed algorithm for to HOG feature descriptor

Algorithm name: HOG) feature description **Input: Grey level \ Color image** **Output:** HOG feature vector
Start **Step1:** Read the RGB image. **Step2:** convert RGB image to a grayscale image. **Step3:** apply a bilateral filter to remove noise **Step4:** apply histogram equalization. **Step5:** apply an optional equalization of global image normalization **Step6:** calculates gradients of the image in the first order **Step7:** produce an encoding that is sensitive to local image content **Step8:** apply normalization. **Step9:** obtain HOG descriptors from all blocks of a dense overlapping grid of blocks covering the detection window. **Step10:** Save the obtained HOG feature in the feature vector to be used in the classifier. **End**

B. Local Binary Pattern (LBP)

Given a pixel in the image of the input fruit, LBP is computed by comparing it to its neighbors:

$$LBP_{NR} = \sum_{n=0}^{n-1} s(v_n - v_c)2^n, s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \tag{7}$$

Where v_c is the central pixel value, v_n is the neighborhood value, R is the neighborhood radius, and N is the neighborhood total. Suppose vc's coordinate is (0, 0), then vn's coordinates are (Rcos (2ÿn / N), Rsin (2πn / N)). Interpolation may estimate the values of neighbors which are not present in the image grids. Let image size be I * J. After each pixel's LBP code is calculated a histogram is created to represent the texture image:

$$H(k)=\sum_{i=1}^{I}\sum_{j=1}^{J} f(LBP_{N,R}(i,j),k), k\in[0,K], \quad (8)$$

$$f(x,y)=\begin{cases}1 & ,x=y\\ 0, & otherwise\end{cases} \quad (9)$$

Where K is the value for maximum LBP code.
Below is the proposed algorithm to LBP feature descriptor

Algorithm name : LBP feature description
Input : Grey level image
Output : LBP featur vectore

Start
Step1: Read the RGB image.
Step2: convert RGB image to a grayscale image.
Step3: apply bilateral filter to remove noise
Step4: apply histogram equalization.
Step5: Obtain a region (a window of 3*3 pixels or 3*3 matrix with 0~255 pixels' intensity) from the input image.
Step6: Select the matrix central value to be utilized as a threshold // This threshold will be utilized to find the new values from the eight neighbors.
Step7: Obtain the matrix of new binary values for the neighbors of the threshold
7.1 setting "zero" to the values which are less than the threshold
7.2. setting "one" to the values which are higher or equal to the threshold.
Step8: Concatenate every binary value from every location from the matrix line by line into a new binary value (for example; 10001101).
Step9: Transform this value of binary to the value of decimal and set it to the central value of the matrix it is a pixel from the original image.
Step10: Obtain a new image that includes the preferable characteristics of the original image.
End.

C. GLCM texture feature extraction

Extraction of the GLCM texture function to be checked uses distance (1.0), which means one pixel to the right and zero pixels to the left, or pairs of pixels to be measured are those which are one-pixel apart from 00, as shown in Figure (3) [14]:

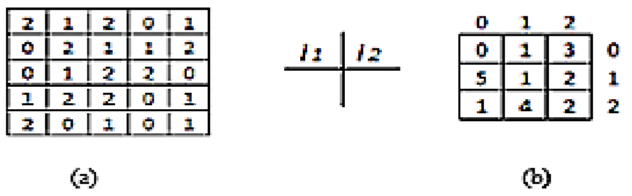

Figure 2: (a) Image measuring 5x5 three intensities (0, 1, 2); (b) GLCM distance (1.0).

The image in Figure 3a is 5x5 in size and has three intensity values, namely 0, 1, and 2, and then the GLCM is 3x3. The distance between the pixels specified is d = (1, 0), meaning one pixel to the right, and zero pixels down. Then count the number of pairs of pixels where the first pixel has an intensity value of i1 and the pair separated d has an intensity value of i2, and is entered on GLCM in column i1 and row. For eg, 2 pairs in the first pixel pair have an intensity value (2, 1), so that the value 2 is entered on the GLCM in column 2 row 1. Figure 3b. Shows the GLCM resulting from the

full pair of pixels calculated. After the matrix forms GLCM, the characteristics of the texture can be calculated based on the matrix.
Below is the proposed algorithm to GLCM feature extractor

Algorithm name : GLCM feature extraction **Input : Grey level image** **Output : GLCM featur vectore**
Start **Step1:** Read the RGB image. **Step2:** convert RGB image to a grayscale image. **Step3:** apply bilateral filter to remove noise Step4: apply histogram equalization. **Step5:** Quantize the image data. **Step6:**Create the GLCM \\ It is a square matrix N x N in size where N is the Number of levels specified under Quantization. **Step7:** Normalize the GLCM, divide each element by the sum of all elements \\ The elements of the GLCM may now be considered probabilities of finding the relationship i, j (or j, i) in W. **Step8:** Calculate the GLCM Feature. **8.1:** Calculate energy **8.2:** Calculate entropy **8.3:** Calculate homogeneity **8.4:** Calculate shade **8.5:** Calculate contrast **8.6:** Calculate correlation **8.7:** Calculate prominence **Step9:** Save the seven features obtain from step 7 in a feature vector. **End.**

III. Classification Using Decision Tree

There are two phases in decision tree classification, first is to generate the decision tree from the given training data and the second is the actual classification where decision rules of the formed decision tree are applied to the transaction having an unknown class label to classify it in one of the classes.

$$V=\sum_{i=1}^{M}\sum_{j=1}^{N}(f_{(i,j)}-\overline{f}_{(i,j)})^2 \qquad (10)$$

Where f is the feature extraction and f is the average value of the pixels in a fruit image.
The algorithm for this classification is given below:

Algorithm name: Decision Tree **Input: feature vector** **Output : Classification**
Start **Step1.** For each transaction to be classified, read one by one the decision rule from the Decision table. **Step2.** Match the fields from the transaction with each decision rule. **Step3.** First, try to find out perfect match and fill the Class field of the transaction with the class of matched rule. **Step4.** If a perfect match is not found then among matched rules, the rule having the highest level is chosen and the class field of the transaction is filled with that class of matched rule. **End**

through it was known that the properties of the HoG method gave high results, by calculating Sensitivity, precision, Accuracy. Table I show the Evaluation performance of fruit classification.

4. THE USED DATASET

This section describes the fruit360 dataset image. The photographs were obtained by recording the fruits as a motor rotates them and then collects frames. Fruits were planted in a low-speed motor shaft (3 rpm), and a 20-second short film was made. A white sheet of paper is set as a backdrop behind the fruits. However, the background was not consistent due to the differences in the lighting conditions, and wrote a dedicated algorithm that extracts the fruit from the background. This algorithm is of type of flood fill: start from each edge of the image and mark all the pixels there, then mark all the pixels found in the neighborhood of the already marked pixels for which the color distance is less than a prescribed amount. Repeat the previous step until you can't mark any more pixels. All marked pixels are considered to be background (which is then filled with white), and the remaining pixels are considered to belong to the object. The maximum distance value between two neighboring pixels is an algorithm parameter and is set (by trial and error) for each film. Fruits were scaled to fit a 100 × 100 pixels image. Other datasets (like NIST) use 28×28 images, but the small size of the dataset is detrimental when you have too similar objects (a red cherry looks very similar to a red apple in small images) [12]. The data set is available on GitHub and Kaggle. Figure (2) shown the sample of Standard Dataset (fruit 360 Database).

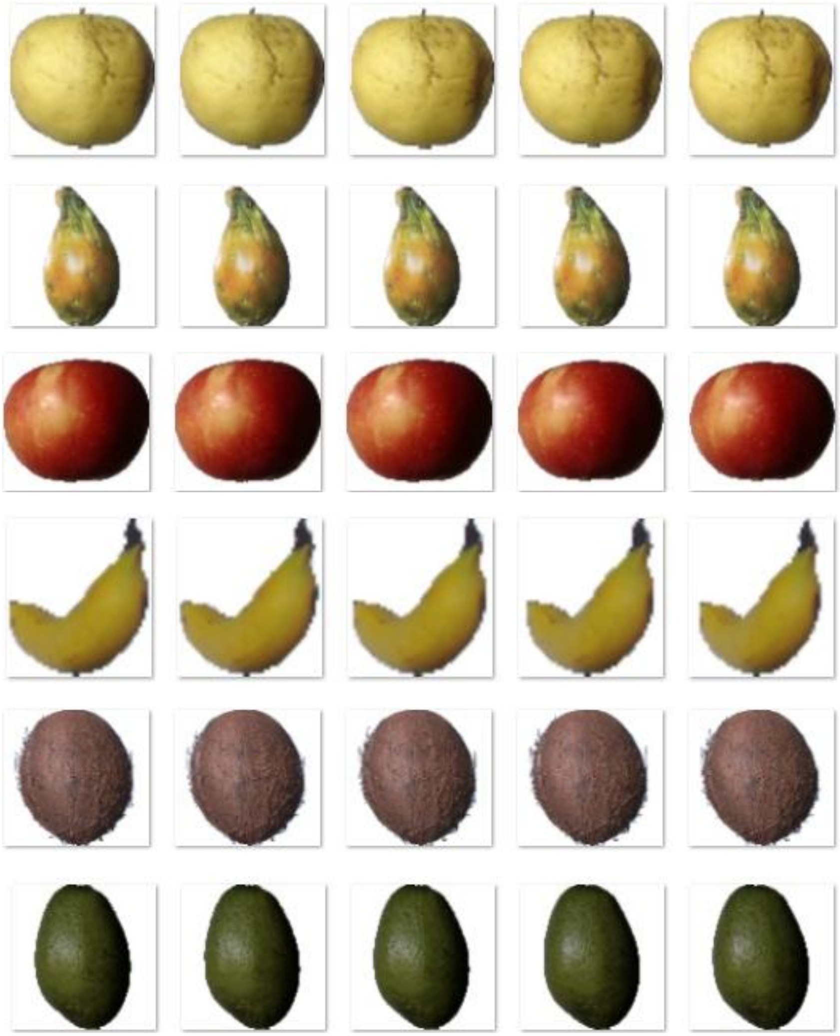

Figure 3: The sample of the fruit dataset

5.Experimental Results

In this paragraph, the results obtained are illustrated to find the best features of the fruit image and its classification.

I.Pre-processing stage: This is the important stage during which the original image is converted to a grayscale image, bilateral filter, and histogram equalization as shown in Figures (4).

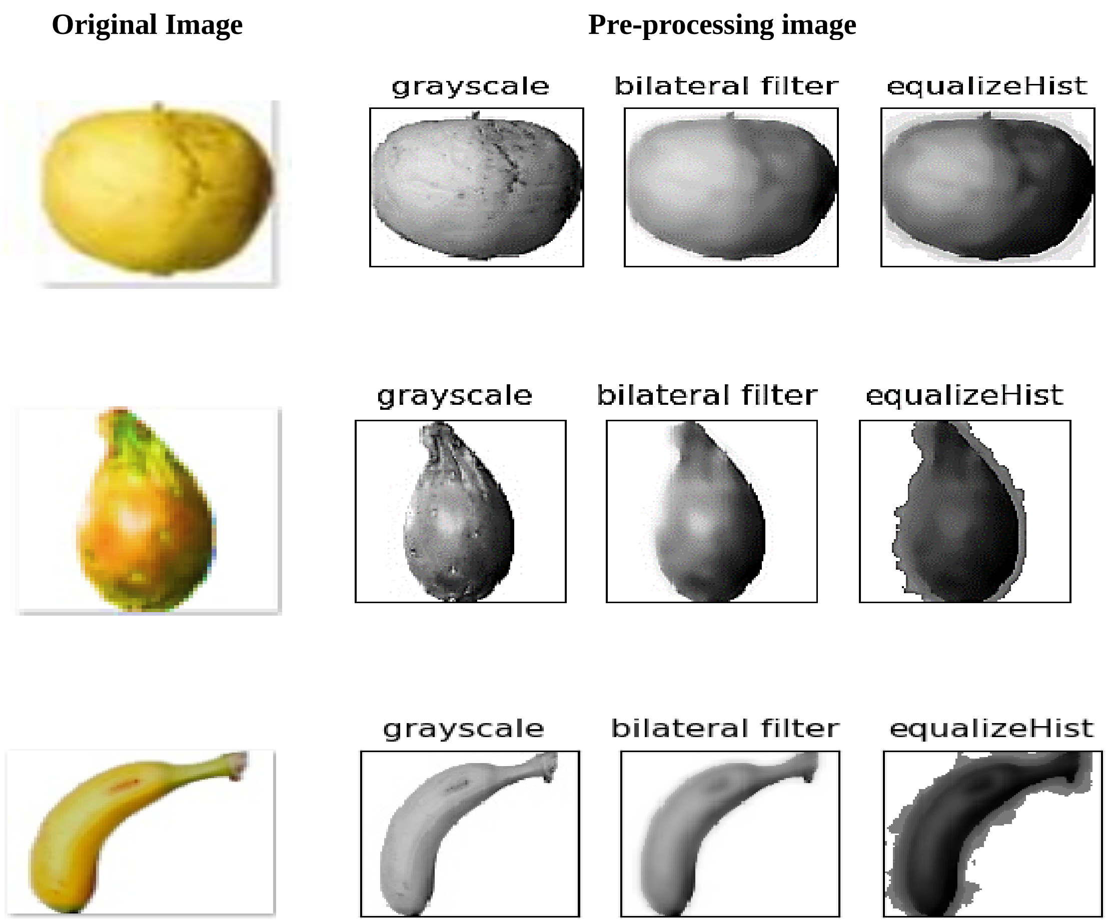

Figure 4: Pre-processing Image.

II.Feature Extraction

A. *HOG feature:* After pre-processing, feature descriptors are described in points of interest in fruit images, and this is the step by which the descriptor is calculated based on the areas cantered around the detected features. This includes converting the local pixel neighborhood into a small vector representation that allows comparison of neighborhoods regardless of changes in fruit orientation or scale as shown in Figure (5).

B. Local Binary Pattern (LBP): The feature extraction process will produce LBP texture feature values. Figure (6) shown the LBP feature extraction.

C. GLCM Feature Extraction: The feature extraction process will produce GLCM texture feature values including energy features, entropy, contrast, homogeneity, ASM, max, dissimilarity, mean, stander deviation. Figure (7) shown GLCM Feature Extraction.

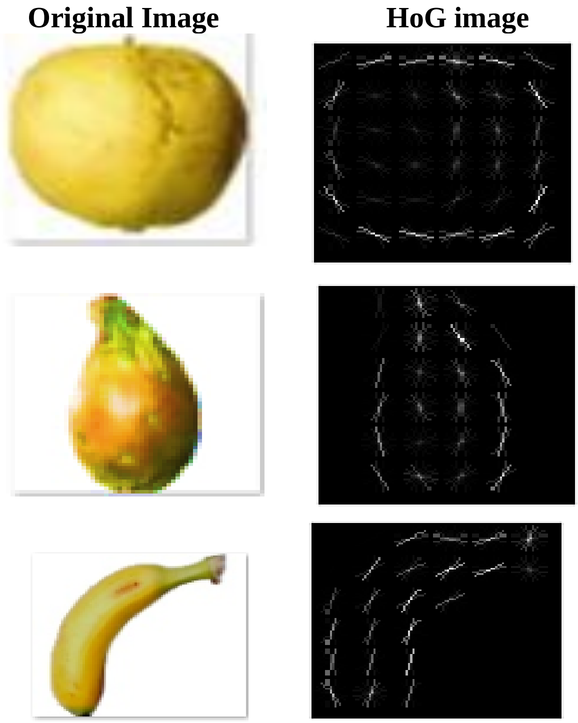

Figure 5: HoG Feature Extraction.

Local Binary Pattern

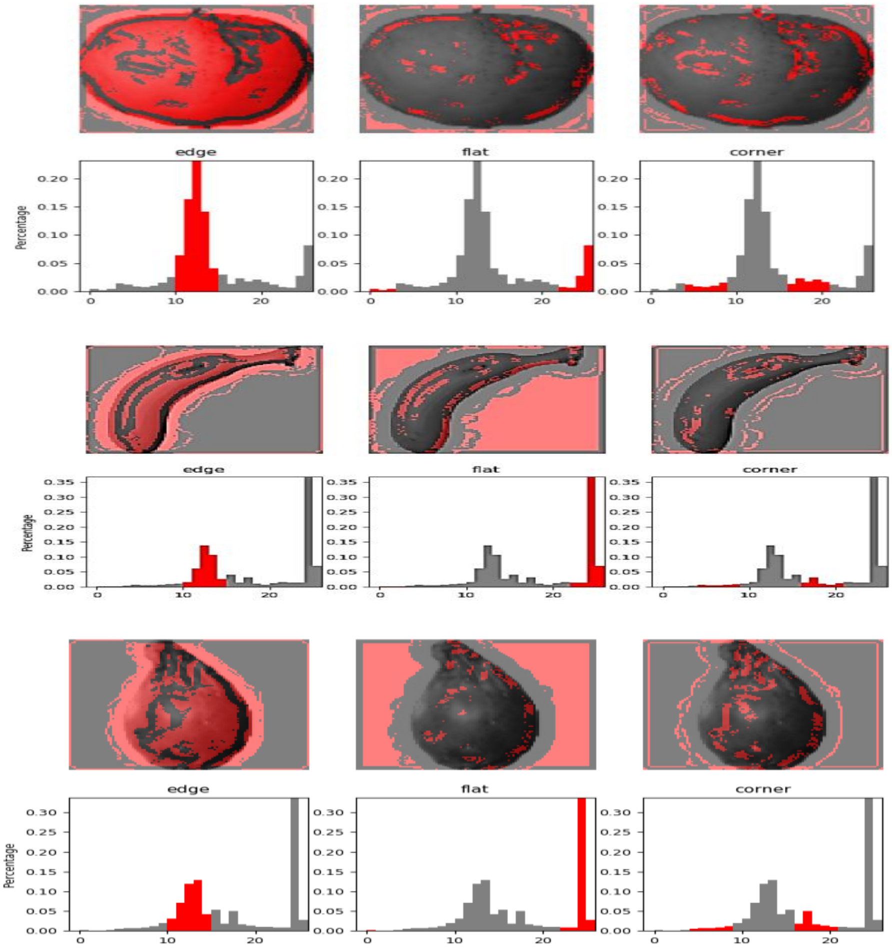

Figure 6: LBP Feature Extraction

GLCM Feature Extraction

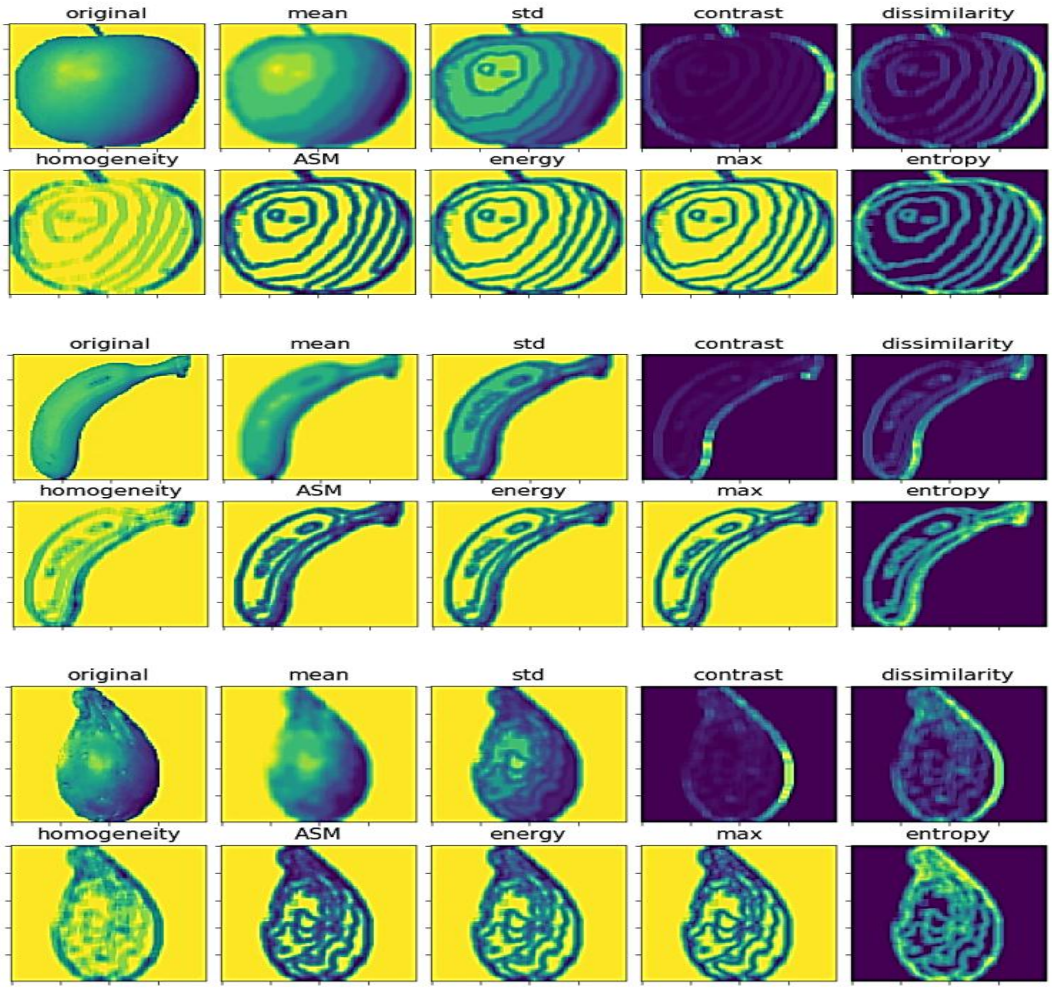

Figure 7: GLCM Feature Extraction.

6.PEFORMACE EVALUATION

The following performance measures have been used to evaluate the proposed sysyem performance**.**

I.Accuracy

Accuracy (ACC) is determined as a number for all predictions of correct (TP + TN) divided by the total number of data sets (P + N). The best accuracy equal to 1.0, while the worst equal to 0.0. It can likewise be determined by 1 - error (ERR) as shown in equation (11).

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FN} + \mathrm{FP}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}} \qquad (11)$$

II.Sensitivity (Recall or True Positive Rate)

Determined the number of predictions of true positive (TP) divided by a total number of the positives (P) this method called Sensitivity (SN) or likewise Recall or the True Positive Rate (TPR)(REC). The sensitivity equal to 1.0 is best, whereas the worst equal 0.0 as shown in equation (12).

$$\mathrm{SN} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}} \qquad (12)$$

II. Precision

Determined the number of predictions of True positive (TP) divided by the total number of negatives (N) as shown in equation (13).

$$\text{Precision} = \frac{TP}{TP + FN} = \frac{TP}{N} \qquad (13)$$

TABLE I: Evaluation performance of Fruit Classification using HoG

Class	samples	FP	TN	TP	FN	Recall	Precision	Accuracy
Class1	90	2	1	84	3	0.99	0.98	0.96
Class2	90	2	1	86	1	0.99	0.97	0.96
Class3	90	2	4	82	2	0.98	0.99	0.98
Class4	90	4	2	80	4	0.99	0.95	0.98
Class5	90	1	1	87	1	0.99	0.97	0.97
Average						0.99	0.97	0.96

TABLE II: Evaluation performance of fruit classification using GLCM

Class	Samples	FP	TN	TP	FN	Recall	Precision	Accuracy
Class1	90	6	7	70	7	0.92	0.97	0.90
Class2	90	4	9	73	4	0.96	0.96	0.92
Class3	90	10	9	65	6	0.91	0.95	0.82
Class4	90	7	11	67	5	0.93	0.97	0.87
Class5	90	4	9	74	3	0.99	0.94	0.92
Average						0.93	0.95	0.90

TABLE III: Evaluation performance of fruit classification using LBP

Class	samples	FP	TN	TP	FN	Recall	Precision	Accuracy
Class1	90	10	2	74	4	0.95	0.89	0.85
Class2	90	6	5	78	1	0.99	0.88	0.92
Class3	90	8	11	62	9	0.97	0.89	0.81
Class4	90	10	7	65	8	0.99	0.87	0.85
Class5	90	4	8	76	2	0.97	0.89	0.95
Average						0.96	0.88	0.87

Three features (LPB, HOG, and GLCM) are used and compared to see which is better, by classifying the fruits using a decision tree after training the system by 70% of the data set and testing it by 30%, then evaluating the results that we obtained through the three methods as shown in Table I, Table II, and Table III, it was proven that the classification accuracy reaches 96% in HoG and 90% GLCM and 87% LBP. where HOG gives promising results comparzim of GLCM and LBP.

7.CONCLUSION

Today, fruit classification plays an important role in marketing. Besides, There have been a lot of difficulties regarding fruit classification. In the proposed system, and to overcome those difficulties, Three feature extractor \ descriptors methods have been employed, which are (LPB, HOG, and GLCM) to extract features and save them in a feature vector, besides decision trees classifier is used to categorize the fruit. In this work fruit360 dataset downloaded from GitHub and Kaggle is used. after training the system by 70% of the data set and testing it by 30%, then evaluating the results that we obtained through the three methods (LPB, HOG, and GLCM) as shown in tables(1),(2),(3) above, it was proven that the classification accuracy reaches 96% in HoG and 90% GLCM and 87% LBP. However, the results obtained by the LBP and GLCM feature extractor have not been satisfactory, HOG gives promising results.

References

[1] S. Ram and A. Singh, " Application of Image Processing in Fruit and Vegetable Analysis: A Review'', Journal of Intelligent Systems, vol. 24, no. 4, pp. 405-424, 2015.

[2] Mohana S. And Prabhakar C., "Automatic Detection of Surface Defects on Citrus Fruit Based On Computer Vision Techniques'', Graphics And Signal Processing, Vol. 9, Pp.11-19,2015.

[3] Hetal N., R.K.Jain, And M.V.Joshi, "Fruit Detection Using Improved Multiple Features Based Algorithm '', International Journal Of Computer Applications, Vol.13, No.2, 2011.

[4] M. Nagata And Q. Cao, "Study On Grade Judgment Of Fruit Vegetables Using Machine Vision'', JARQ, Vol. 32, Pp. 257-265,1998.

[5] D. Mery, F. Pedreschi, And A. Soto, " Automated Design Of A Computer Vision System For Visual Food Quality Evaluation'', Springer Science,2012.

[6] Paini, D. R. "Impact of the introduced honey bee (Apis mellifera)(Hymenoptera: Apidae) on native bees: a review." Austral ecology 29.4 (2004): 399-407.

[7] Zorarpaci, E., Ozel, S.A.: "A hybrid approach of differential evolution and artificial bee colony for feature selection". Expert Syst. Appl. 62, 91–103,2016.

[8] Ronald, M., & Evans, M.. Classification of selected apple fruit varieties using Naive Bayes. Indian Journal of Computer Science and Engineering (IJCSE), 7(1), 13-19.2016,

[9] Ali, N. M., Karis, M. S., Sobran, N. M. M., Bahar, M. B., Ken, O. K., Ibrahim, M. M., & Johan, N. F. Detection of Multiple Mangoes using Histogram of Oriented Gradient Technique in Aerial Monitoring. ARPN Journal of Engineering and Applied Sciences, 12, 2730-2736.2017.

[10] A. Wajid, N. K. Singh, P. Junjun, M. A. Mughal, Recognition of Ripe, Unripe and Scaled Condition of Orange Citrus Based on Decision Tree Classification, in International Conference on Computing, Mathematics and Engineering Technologies, pp. 2–5, 2018.

[11] Kavitakomal, Dr. Sonia," Quality Assessment Of Orange Fruit Using Svm Classifier And Gray Level Co-Occurrence Matrix Algorithm" International Journal Of Scientific & Technology Research Volume 8, Issue 11, pp 2277-8616, November 2019.

[12] https://www.kaggle.com/moltean/fruits

[13] Banterle, F.; Corsini, M.; Cignoni, P.; Scopigno, R. (2011). "A Low-Memory, Straightforward and Fast Bilateral Filter Through Subsampling in Spatial Domain". Computer Graphics Forum. 31 (1): 19–32.

[14] Texture Analysis Using the Gray-Level Co-Occurrence Matrix (GLCM) - MATLAB & Simulink - MathWorks United Kingdom". uk.mathworks.com. Retrieved 2020-06-26