Academic Journal of Nawroz University (AJNU), Vol.13, No.1, 2024

https://doi.org/10.25007/ajnu.v13n1a1514

A Recognition and Classification of Fruit Images Using Texture Feature Extraction and Machine Learning Algorithms

Nohadra Behnam Israel[1], Adnan Ismail Al-Sulaifanie[2] and Ahmed Khorsheed Al-Sulaifanie[2]

1 Energy Engineering Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Kurdistan Region-Iraq.

2 Electrical & Computer Engineering Department, College of Engineering, University of Duhok, Duhok, Kurdistan Region-Iraq.

ABSTRACT: Fruits classification is demanded in some fields, such as industrial agriculture. Automatic fruit classification from their digital image plays a vital role in those fields. The classification encounters several challenges due to capturing fruits' images from different viewing angle, rotation, and illumination pose. In this paper a framework for recognition and classification of fruits from their images have been proposed depending on texture features, the proposed system rely on three phases; firstly, pre-processing, as images need to be resized, filtered, color convert, and threshold in order to create a fruit mask which is used for fruit's region of interest segmentation; followed by two methods for texture features extraction, first method utilize Local Binary Pattern (LBP), while the second method uses Principal Component Analysis (PCA) to generate features vector for each fruit image. Classification is the last phase; two supervised machine learning algorithms; K-Nearest Neighbor (K-NN) and Support Vector Machine (SVM) are utilized to identity and recognize the fruits images classes. Both methods are tested using 1200 fruits images, from 12 classes acquired from Fruits-360 database. The results show that combining LBP with K-NN, and SVM yields the best accuracy up to 100% and 89.44% respectively, while the accuracy of applying PCA with K-NN and SVM reached to 86.38 % and 85.83% respectively.

Keywords: Pattern Recognition, Fruits Classification, LBP, PCA, K-NN, SVM

1. Introduction

In recent years, the improvement in the cameras and sensors fields had led to an increase in intelligent systems, an essential purpose of those systems is to understand and perceive an image as done by human brain (Bhargava & Bansal, 2021). The usage of image processing has been wide increasingly in agricultural field to automate its processes; automation system can be implemented in crop ripeness monitoring, crop disease detection, fruits and vegetables recognition (Al-falluji, 2016). The automatic fruit's image classification has attract wide attention by researchers worldwide cause its offers numerous solutions such as reducing manual effort to a large extent as well as time evolvement (Gill & Khehra, 2021). Fruits recognition systems can be utilized in many real-life implementations, such in store checkout, where it may be utilize rather than manual scanner tags; moreover, for helping eye weakness people as a supportive appliances, an educational tool for small children and Down syndrome patients. Recognizing several fruits species is a repeated chore in supermarkets, where the cashier has to define each item type that will determine its cost, a fruit recognition system, which automates labeling and computing the price, is the right solution for this problem, furthermore, fruit recognition system could be utilized as a mobile application that can help the user to identify nutrition and dietary information (Behera, Rath, Mahapatra, & Sethy, 2020). Fruits image visual characteristics like color, shape, size and texture are usually used for assisting the identification process (Jana & Parekh, 2017; Nosseir & Ahmed, 2018; Saranya, Srinivasan, Pravin Kumar, Rukkumani, & Ramya, 2019; Shukla & Desai, 2016), yet there are major challenges for an accurate imagining system; such as viewpoint variation, illumination pose, inter-class similarities and intra-class diversities. Texture considers an effective characteristic that analysis image surface, every fruit image owns unique and different texture when it compares to others (Indriani, Kusuma, Sari, & Rachmawanto, 2017), additionally texture-based approach has translation, rotation, shape and color dependency.

In this paper a framework for fruits recognition and classification system has been proposed, based on texture feature, machine learning algorithms are utilized to recognize and classify the extracted features. The aim of this study is to investigate the influence of two methods in texture feature extraction and compare their accuracies. As well as to investigate about the performance of Principal Component Analysis (PCA), as texture feature extraction and reduction technique in fruit image classification, in which was used before in this domain as a feature reduction technique only (Ghazal et al., 2021; Zeeshan, Prabhu, Arun, & Rani, 2020). The rest of this paper is organized as follows: section 2 will be on a survey about some related works, section 3 demonstrates the proposed system. The result and discussion are explored in section 4. Performance comparison with other researches is outline in section 5, while conclusion is presented in section 6.

2. Related works

There are numerous researches related to fruits images recognition and classification systems. A fruit recognition system was presented in (Shukla & Desai, 2016) on a database comprise of nine different types of fruits like (apple, banana, Indian lemon, mango, pear, plum, orange, watermelon, and strawberry), features such as in color, shape and texture were extracted from fruit images, for increasing the accuracy, those features were fused and passed through the K-Nearest Neighbors (K-NN) and the Support Vector Machine (SVM) classifiers, the result showed that best accuracy was 91.3% with the K-NN classifier. The authors in (Jana & Parekh, 2017) introduced a new approach for fruits classification by implementing fruit images shape features, their database consisted of seven fruits types such as apple, banana, cucumber, lemon, mango, strawberry, and tomato, recognition approached was relied on several shape features descriptors, like area, perimeter, minor and major axis, width, height, minimum bounding box etc. in classification stage three different classifiers were used like Naïve Bayes(NB), Neural Network(NN) and (K-NN). Those classifiers provide an overall accuracy of 88.57- 95.24%. The proposed fruit recognition model suggested by (Nosseir & Ahmed, 2018) was based on fruits images color and texture features, color features were extracted from histogram color value, while in texture, features were acquired from first order statistical features and gray-level co-occurrence matrix (GLCM), a combination of those features were applied to different types of K-NN classifiers like fine K-NN, medium K-NN, coarse K-NN, cosine K-NN and weighted K-NN. The fine K-NN with k=1 achieved the best accuracy reached to 96.3%. An automatic sorting and classification of different kinds of fruits using their images were proposed by(Saranya et al., 2019), four categories of fruits such as apple, banana, orange, and pomegranate acquired from Fruits-360 database were utilized to investigate the accuracy of the system, different extracted features such as mean of RGB color, size, height and width were used to build and test the model, in classification phase the K-NN and SVM were implemented as they report an accuracy of 93.8% and 100% respectively. A system for recognition strawberry ripeness was presented in (Anraeni, Indra, Adirahmadi, & Pomalingo, 2021) four different strawberry's classes like ripe, unripe, raw, not strawberry was utilized as a database, features were extracted from RGB image, such as RGB color component value, area, roundness, and centroid value for each channel in RGB, K-NN was implemented in the classification phase to recognize each class the accuracy of this system was reached to 85%.

It has been observed that each type of features have its own limitation, most of the studies considers combined approach of color, size, shape, and texture. Less attempted has been made to address texture-based methods and using Local Binary Pattern (LBP) operator as texture feature description. Based on related works, conclusion was made to implement texture features with K-NN and SVM classifiers in order to obtain a higher accuracy.

3. The proposed framework

The fruit recognition and classification system consist of four essential components. First there is a database, which is divided into two parts; training and testing set to represent the training and testing phase respectively, followed by pre-processing, and then texture features extraction. Finally, a classification stage is required through training and testing the classifier to classify fruits images and to predict their classes. Figure 1 below illustrates the proposed framework flowchart.

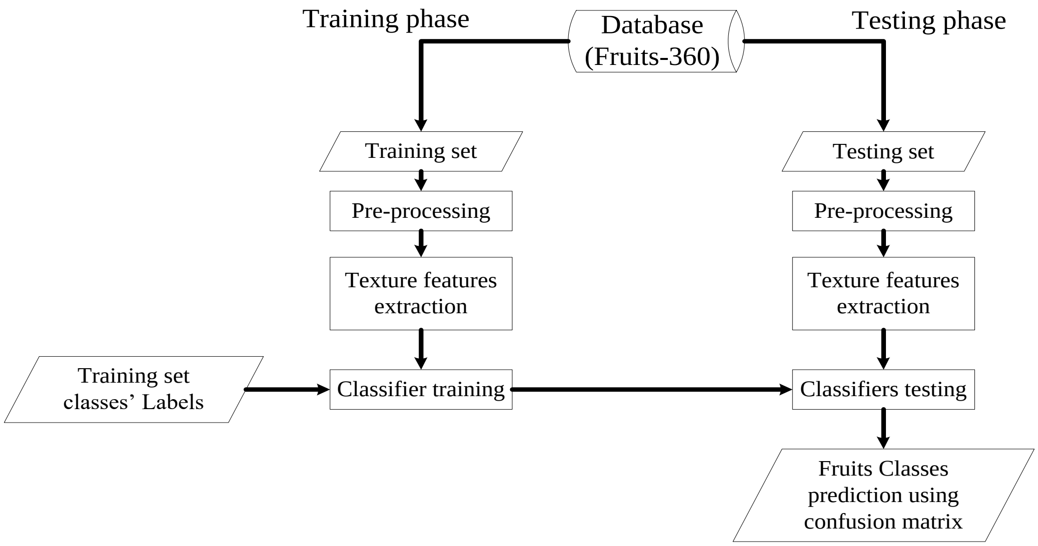


Figure 1: The proposed framework for fruit recognition and classification

3.1 Database:

In this paper, the selected database for the proposed framework is Fruits-360. This large dataset is made freely available at (Oltean, 2018). The Fruits-360 dataset contains 131 fruit and vegetable classes with a total of 90380 images. These images were gathered through rotating a fruit by using a slow speed motor shaft which run at 3 rpm, in front of a white paper sheet placed as a background, fruits images dimensions were scaled to fit 100x100 pixels. In the current study, 12 categories of fruits were acquired for classification purpose. Table 1 illustrates samples of fruit image.

Table 1: Samples of fruit images

Label	**Class**	**Samples of selected images**
1.	Apple	
2.	Apricot	
3.	Banana	
4.	Cherry	
5.	Fig	

6.	Kiwi	
7.	Orange	
8.	Peach	
9.	Pear	
10.	Pomegranate	
11.	Strawberry	
12.	Watermelon	

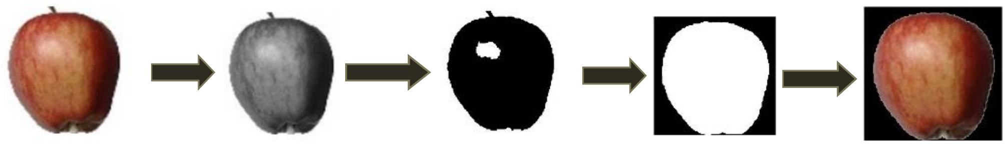
3.2 Pre-processing:

Pre-processing is an essential step for achieving efficient object recognition. In the preprocessing phase, image is enhanced, and any undesirable distortion is removed to improve quality of the image(San, Aung, & Khaing, 2019). Filtering, color conversion, thresholding and morphological operations are used for extracting the fruit's region of interest. Initially, images are resized to 200x200 pixels, then Gaussian smoothing filter with standard deviation of 0.5 is applied on the fruits images for reducing noise and illumination effects, furthermore the colored fruits images are converted to gray scale to keep only the luminance value of red, green, and blue as demonstrated in Equation 1.

$$Gray\ Image = 0.299 * R + 0.587 * G + 0.114 * B \qquad (1)$$

For pixels separation of fruit from that in the background, OTSU thresholding method is used to convert the gray image to binary (Otsu, 1979). After thresholding; operation like complementation, filling region, and removing connected components which have less pixels, are necessary to produce the fruit mask, which will undergo an opening morphological operation (erosion followed by dilation) by using the same structuring element (disk type) in order to smooth the boundary and remove any stalks, the final fruit mask will be utilize for segmentation purpose (cropping fruits' region of interest). Equation 2 illustrates the opening operation where A is the binary image, B is the disk structure element. While erosion and dilation operation represented as $\ominus$, $\oplus$ respectively. Figure 2 clarifies the pre-processing steps.

$$A\text{o}B = (A \ominus B) \oplus B \qquad (2)$$



(a) (b) (c) (d) (e)

Figure 2: Pre-processing steps, (a) resized, filtered color image, (b) corresponding gray image, (c) threshold image, (d) final fruit mask, (e) segmented fruit region of interest

3.3 Texture features extraction:

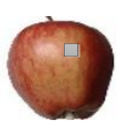
In general, feature extraction from images plays an essential role in any recognition system. It can be expressed as a process of obtaining higher-level information about meaningful objects in an image. The purpose behind using extracted features is to reduce redundant information, save more memory, and enlarge the rate of recognition (Rajasekar & Sharmila, 2019). Texture features contain information regarding spatial arrangements of intensities in the image; the Local Binary Pattern (LBP) has been implemented for feature extraction, its performance is compared with that of Principal Components Analysis (PCA).

3.3.1 Local Binary Pattern (LBP)

The LBP is an efficient texture operator due to its simplicity, robust performance against image rotation, and invariance to grayscale changes. The original LBP was presented by (Ojala, Pietikäinen, & Harwood, 1996). With LBP it is possible to describe texture and shape of digital image. The original form of LBP works with eight surrounding neighbors of pixels (mask of 3x3). Thresholding is applied to those pixels included in the mask by their central pixel. If a neighbor pixel has a higher gray value than the central pixel (or the same gray value) it gets one, otherwise zero is assigned to that pixel, see Figure 3, suppose that the central pixel is Pc and Pi is the surrounding pixels with $i \in \{0, \ldots\ldots, 7\}$, the LBP can be implemented using Equation 3 (Ahonen, Hadid, & Pietikäinen, 2004).

$$LBP = \sum_{i=0}^{i=7} LB(Pi - Pc).\, 2^i \qquad (3)$$

$$\text{Where } LB(Pi - Pc) = \begin{cases} 1 & Pi \geq Pc \\ 0 & Pi < Pc \end{cases}$$



101	96	110
138	90	76
67	56	43

(a)

1	1	1
1		0
0	0	0

(b)

Figure 3: The process of LBP operator, (a) gray value, (b) corresponding binary pattern

The LBP code of the central pixel is generated through multiplying the threshold neighborhoods by their binary weights (power of two) that are given to the corresponding pixels; lastly, the eight pixels' values are summed and replaced the central pixel as shown in Figure 4, thus will yield 2^8 or 256 bins labels or histogram.

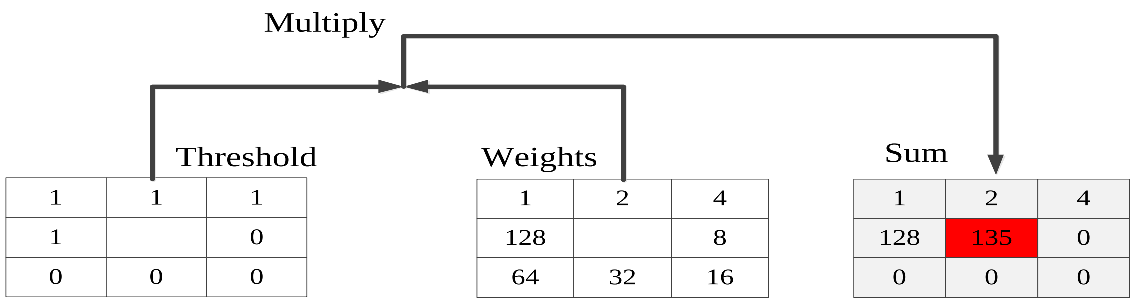


Figure 4: The calculation of LBP

In order to treat textures at several scales, the LBP operator was expanded to make use of neighborhoods at different sampling points and radius. To perform this procedure, suppose a circle is made with radius R from a central pixel value; then by using bilinear interpolation any number of neighbors (P sampling point) on the edge of this circle can be calculated (Ahonen et al., 2004). Figure 5 demonstrates circularly neighbor-set with different values of R, P samples.

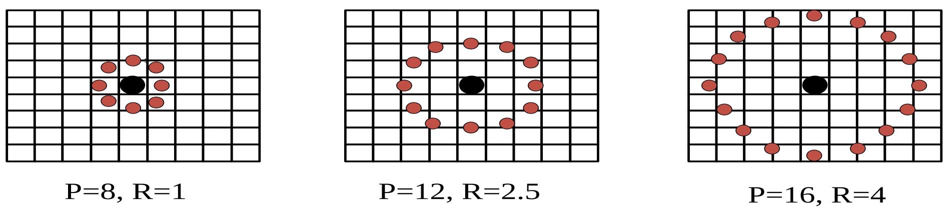


Figure 5: Different values of R with circular neighbor-set P samples point

The Uniform Local Binary Pattern is a fundamental type of LBP; the LBP is called uniform if it comprises at most two bitwise transitions from 1 to 0 or vice versa. For example 00000000, 11111111 are eight bits with zero transition, while 00011111 and 11001111 are examples for one and two transition. The possible combination of uniform pattern can be calculated by $p(p-1)$, which will produce 56 bins labels or histogram for neighborhoods of 8 sampling points. The notion of $LBP_{P,R}^{u2}$ is used to describe the uniform pattern with P sampling points and R radius (Ahonen et al., 2004). The uniform patterns have many advantages over the non-uniform patterns, as it saves memory, reduce the number of bins in the histogram and detect the essential local texture (rotation invariant).

In our study, the segmented fruit's region of interest is converted to gray scale, then the image is divided into several non-overlapped regions, from each region the LBP code is computed to form a histogram with $p(p-1)+3$, which represent 59 bins, 56 bins for uniform pattern, two bins for zero transition patterns, while the non-uniform patterns are accumulated in a single bin. These histograms are concatenated and normalized to form a feature vector for each fruit image, see Figure 6.

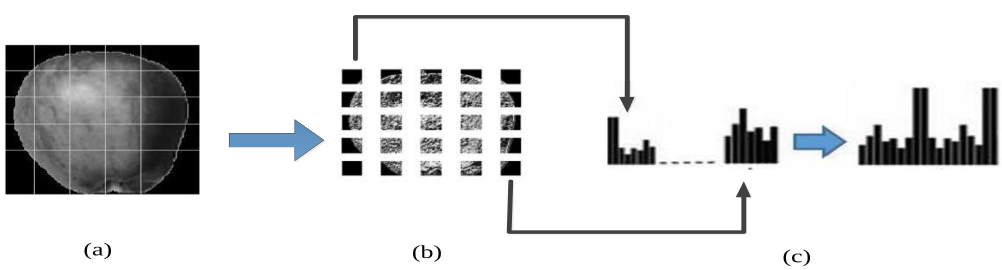


Figure 6: The histogram of non-overlapping regional LBP, (a) segmented gray apple image divided into 5x5 non-overlapping regions, (b) LBP for each region, (c) histogram calculation.

3.3.2 Principal Component Analysis (PCA)

PCA is unsupervised algorithm, and a well- known statistical method that used in feature extraction and dimensionality reduction. PCA implements a procedure of orthogonal transformation for converting a group of observed correlated variables into a group of values that are linearly uncorrelated. It is utilized to preserve the important information and remove the redundant ones. PCA algorithm exemplifies the concept of Eigenvalues and Eigenvectors, which are obtained from the covariance matrix (Abdi & Williams, 2010; Zebari, Abdulazeez, Zeebaree, Zebari, & Saeed, 2020). To explain PCA algorithm, assume X is a matrix in a form of gray level with size of MxN. Each row represents an individual object that is consisting with N observation, while M refer to number of variables.

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & x_{3N} \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix} \quad (4)$$

The following steps summarize the PCA algorithm according to (Kaur & Himanshi, 2015):

1. Compute average of the matrix:

$$\bar{x}=\frac{1}{M}\sum_{i=1}^{M} X_i \quad (5)$$

2. Subtract average from the matrix:

$$\phi_i=X-\bar{x} \quad (6)$$

3. Calculate covariance matrix C from ϕ_i as:

$$C=\frac{1}{N-1}\phi_i\phi_i^T \quad (7)$$

The matrix C's diagonal represents the variance in ϕ_i which contains the most important characteristics (Eigen value) while the off-diagonal elements are the covariance that consider the redundancy.

4. From the Eigen value, calculate the Eigen vector and sort them in a descending order to create the principal components PC of matrix X. Each column contains coefficients for one principal component as:

$$PC=\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix} \quad (8)$$

5. From the principal components PC , the principal component scores S are compute, which consider a representations of matrix X in the principal component space, as given in Equation 9.

$$S=\phi_i PC \quad (9)$$

In our approach, Eigenvectors of the covariance matrix is computed from the training set of fruit images. These eigenvectors are used to create the Eigen space, the principal component scores represent the features that are extracting from the fruit images and used to train the classifiers. Furthermore, fruits images in testing set are projected on the Eigen space, and consider as features in testing phase. In recognition phase, the position of the unknown projected fruit image from testing set will be compared with the place of features or score from the training set. Features extraction by PCA must retain principal components which explain a higher variation, for achieving that a certain number of features will be select which leads to reduce the dimensionality and increase the accuracy.

3.4 Classification:

Categorization of digital image considered an informatics task cause it leads for a model creation to predict the class of categorical variables depending on numerical features (Rajasekar & Sharmila, 2019). In the proposed framework for fruit recognition and classification, two machine learning algorithms, K-Nearest Neighbor (K-NN) and multi-Support Vector Machine (SVM) are utilized for fruits classes' prediction.

3.4.1 K-Nearest Neighbor (K-NN)

Basically, the K-NN is a simple algorithm that has the ability to store all classes in the training set and classifies the new unlabeled classes of testing set depending on a similarity measure. (Taunk, De, Verma, & Swetapadma, 2019). The K-NN algorithm used to search for number of (k) training samples which are close to the unknown sample from the testing set by using the Euclidian distance metric as expressed in equation 10.

$$Dist(x, y) = \sqrt[2]{\sum_{i=1}^{n}(xi - yi)^2} \qquad (10)$$

Where x, y represent two classes want to be learned, and xi, yi refers to its features (Rajasekar & Sharmila, 2019), the number of nearest neighbors, which is indicate by (k) play an important role in the classification process. In this paper, the value of (k) is set to 1. Usually, it is determined experimentally based on predefined error.

3.4.2 Multi-class Support Vector Machine (SVM)

Originally, the SVM is a binary classifier that finds an optimal hyperplane using given training data to classify the unknown data into two classes. The hyperplane is placed at distance in which separate nearest point of both classes with maximum value called margin, finite number of hyperplanes can be founded in SVM. However, SVM selects the maximum one. Suppose that X is given as $(x1, y1), (x2, y2), \ldots\ldots, (xm, ym)$, where $xi,\ i \in \{1,2,3, \ldots, m\}$ refer to training set while, yi represent their labels as $yi \in \{1, -1\}$, Figure 7 demonstrates an optimal SVM. An equation for the separating hyperplane is given by Equation 11.

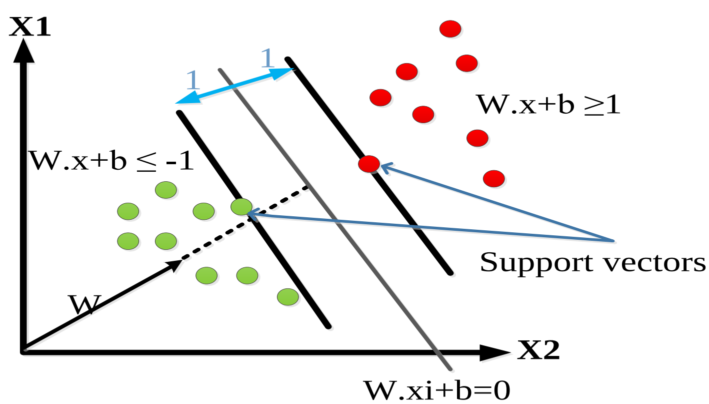


Figure 7:An optimal SVM classifier(Wang & Xue, 2014)

$$W.xi + b = 0 \qquad (11)$$

Where W is a weight vector and equal to $\{w1, w2, w3, \ldots.wn\}$, n refer to the number of features and b is a bias. If an unknown sample falls in $W.x + b \geq 1$, it refers to class's label $y = 1$, otherwise it is belong to class -1, thus SVM search for weights that maximized the margin (Shukla & Desai, 2016). Practically, the multi-class classification problems (more than two classes) are resolved into a series of binary problems in which the standard SVM can be implemented directly (Wang & Xue, 2014). In the proposed framework, multi-class learning is performed by utilizing one versus one (1V1) coding design, while using a linear kernel function.

4. Result and discussion

An experimental investigation using MATLAB 2020a software is implemented to evaluate the performance of the proposed framework of fruit recognition and classification system by using 1200 images of 12 classes acquired from Fruits-360 dataset, in each class 100 images, 70% of those images were used for classifiers training (training phase), while 30% were used to evaluate and test the classifiers performance (testing phase). In the training phase, fruits images from training set are pre-processed, and texture features are extracted using both methods. In First method, the LBP operator, the value of sampling points P or neighbors is maintain at 8, while the value of radius R and the number of non-overlapped regions are varied. For comparison second method, a benchmark algorithm, PCA is used for extracting and reducing features. Features obtained from the two methods are used to train the classifiers individually. Moreover, in testing phase, the extract features from the testing set are used to test the classifier performance. The accuracy of the proposed system and other performance metrics such as sensitivity, recall and F1-score are calculated using the confusion matrix. The following equations describe those metrics.

$$Accuracy = \frac{no.\ of\ sample\ predicted\ correctly}{total\ no.of\ sample} \tag{12}$$

$$Recall = \frac{TP}{(TP+FN)} \tag{13}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{14}$$

$$F1\ Score = 2 \times \frac{(Precision \times Recal)}{(Precision+Recal)} \tag{15}$$

Where TP indicate truly positive predicted class, FN is a false negative predicted class and FP is a false positive predicted class. Using the LBP features, best accuracy was found in block region (5× 5) at R=1, $LBP_{8,1}^{u2}$, as it achieve overall accuracy reach to 100% in K-NN while in multi-SVM is 89.44%, the accuracy of several regional division with R=1 is shown at Figure 8. The confusion matrix of LBP descriptor with multi-SVM classifier at regional division 5x5 blocks is shown in Figure 9.

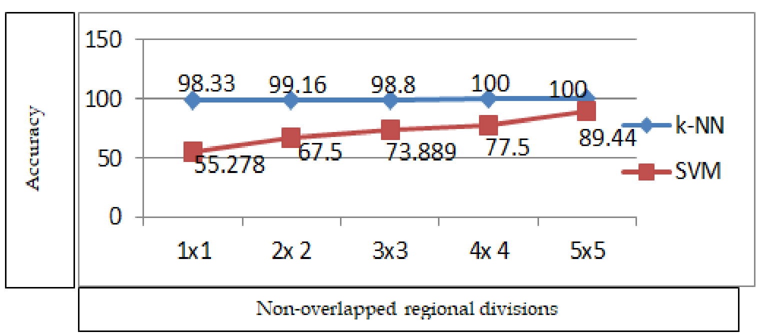


Figure 8: The accuracy rate of several regional divisions with K-NN and multi- SVM at R=1

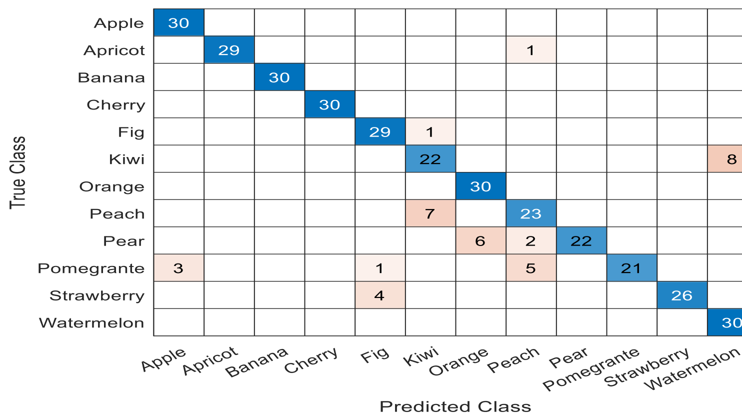


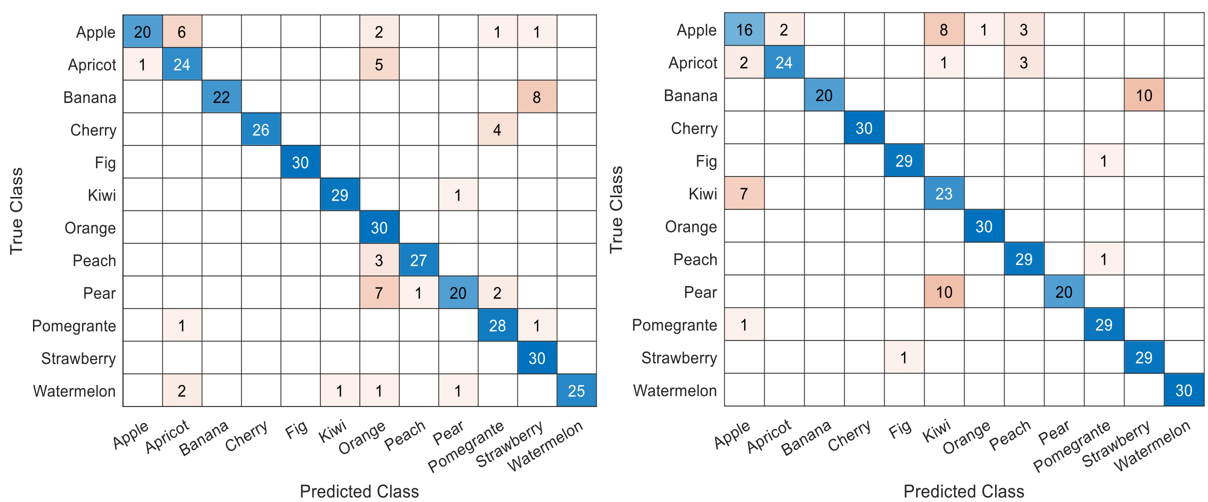
Figure 9: Confusion matrix of LBP descriptor with multi- SVM classifier at regional 5x5 divisions, R=1

The performance evaluation of different metrics in confusion matrix using LBP features with K-NN and multi-SVM classifiers at best accuracy are given in Table 2.

Table 2: Performance of different metrics of LBP operator with K-NN and multi-SVM classifiers

Fruit class	No. of testing samples	K-NN						Multi-SVM					
		Tp	*Fn*	*Fp*	Precision	Recall	F1-score	*Tp*	*Fn*	*Fp*	Precision	Recall	F1-score
Apple	30	30	0	0	1	1	1	30	0	3	0.909	1	0.952
Apricot	30	30	0	0	1	1	1	29	1	0	1	0.966	0.982
Banana	30	30	0	0	1	1	1	30	0	0	1	1	1
Cherry	30	30	0	0	1	1	1	30	0	0	1	1	1
Fig	30	30	0	0	1	1	1	29	1	5	0.852	0.966	0.905
Kiwi	30	30	0	0	1	1	1	22	8	8	0.733	0.733	0.733
Orange	30	30	0	0	1	1	1	30	0	6	0.833	1	0.908
Peach	30	30	0	0	1	1	1	23	7	8	0.741	0.766	0.753
Pear	30	30	0	0	1	1	1	22	8	0	1	0.733	0.845
Pomegranate	30	30	0	0	1	1	1	21	9	0	1	0.7	0.823
Strawberry	30	30	0	0	1	1	1	26	4	0	1	0.866	0.928
Watermelon	30	30	0	0	1	1	1	30	0	8	0.789	1	0.882
Average %					100	100	100				90.5	89.40	89.26

By utilizing PCA algorithm, the principal components which explain at least 80% of variation are selected, thus will lead to reduce features vector from 839 to 28 features and yield the best accuracy as 86.38%, 85.83% in K-NN and multi-SVM respectively. Figure 10 illustrates the confusion matrix of PCA algorithm with both classifiers.



(a) (b)

Figure 10: (a) confusion matrix of PCA algorithm with K-NN classifier (b) confusion matrix of PCA algorithm with multi-SVM classifier

Table 3 demonstrates the performance evaluation of different metrics in confusion matrix using PCA algorithm with K-NN and multi-SVM classifiers.

Table 3: Performance of different metrics of PCA with K-NN and multi-SVM classifier

Fruit class	No. of testing samples	K-NN						Multi-SVM					
		Tp	*Fn*	*Fp*	Precision	Recall	F1-score	*Tp*	*Fn*	*Fp*	Precision	Recall	F1-score
Apple	30	20	10	1	0.952	0.666	0.783	16	14	10	0.615	0.533	0.571
Apricot	30	24	6	9	0.727	0.800	0.761	24	6	2	0.923	0.800	0.857
Banana	30	22	8	0	1.000	0.733	0.846	20	10	0	1.000	0.666	0.799
Cherry	30	26	4	0	1.000	0.866	0.928	30	0	0	1.000	1.000	1.000
Fig	30	30	0	0	1.000	1.000	1.000	29	1	1	0.966	0.966	0.966
Kiwi	30	29	1	1	0.966	0.966	0.966	23	7	19	0.547	0.766	0.638
Orange	30	30	0	18	0.625	1.000	0.769	30	0	1	0.967	1.000	0.983
Peach	30	27	3	1	0.964	0.900	0.931	29	1	6	0.828	0.966	0.891
Pear	30	20	10	2	0.909	0.666	0.768	20	10	0	1.000	0.666	0.799
Pomegranate	30	28	2	7	0.800	0.933	0.861	29	1	2	0.935	0.966	0.950
Strawberry	30	30	0	10	0.750	1.000	0.857	29	1	10	0.743	0.966	0.840
Watermelon	30	25	5	0	1.000	0.833	0.908	30	0	0	1.000	1.000	1.000
Average %					89.10	86.35	86.48				87.7	85.79	85.78

5- Performance comparison

Best accuracies obtained from the proposed framework of fruit recognition and classification system in this paper and those of other in the field of fruit recognition and classification are compared according to feature extraction techniques, fruits images number used in their database, selected types of fruit and K-NN, SVM classification algorthims as shown in Table 4.

Table 4: Comparison of performance

Ref.	Images'database no. and classes	Types of used features	Feature extraction techniques	Classification algorthim	Best accurcy
(Shukla & Desai, 2016)	155 of 9 classes	Color+texture +shape	Mean of LAB color components,color coherent vector(CCV),gray-level co-occurrence matrix (GLCM) , LBP, aspect ratio,roundness ,exent,and othor statistical features	K-NN ,multi-SVM	91.3% ,86.96% respectively
(Jana & Parekh, 2017)	210 of 7 classes	shape	Area,primeter,major and minor axis length,width and hight	K-NN	88.57%
(Saranya et al., 2019)	1707 images of 4 classes	Color + shape	Mean of RGB color components, size, height and width	K-NN and SVM	93.8%,100% respectively
(Anraeni et al., 2021)	50 strawberry images of 4 classes	Color + shape	RGB color components value, area, roundness, and centroid value for each channel in RGB	K-NN	85%
This work	1200 images of 12 classes	Texture	LBP and PCA	K-NN and multi-SVM	100% ,89.44% with LBP operator respectively

6- Conclusion

In this paper a framework for fruits recognition and classifcation is proposed using texture features extration by LBP operator and PCA algorthim to form a feture vector. Two supervised machine learning algorthim are used for classification pupose; K-NN and multi-SVM. The 12 classes of fruits with 1200 images are acquired from Fruits-360 database,70% of those images were used to train the classifiers while 30% for system testing. It can be conclude that the best recognition and classification accuracy achieved through utilizing LBP features along with K-NN algorthim as it achive 100% in overall accuracy and in different metrics such as recall, precision, and F1-score,comparing to multi-SVM,which yield a 89.44% ,90.5%, 89.4%, and 89.26 % in overall accuracy, precision,recall, and F1-score respectively. The result of LBP features with both classifiers are compared with those in PCA, The differences can be explained by the fact that LBP captures the variance between pixel intensities, thus has more accurate information whereas PCA loses part of information during its dimensionality reduction.

References

[1] Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics, 2*(4), 433-459.

[2] Ahonen, T., Hadid, A., & Pietikäinen, M. (2004). *Face recognition with local binary patterns.* Paper presented at the European conference on computer vision.

[3] Al-falluji, R. A. A. (2016). Color, shape and texture based fruit recognition system. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 5*(7).

[4] Anraeni, S., Indra, D., Adirahmadi, D., & Pomalingo, S. (2021). *Strawberry Ripeness Identification Using Feature Extraction of RGB and K-Nearest Neighbor.* Paper presented at the 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT).

[5] Behera, S. K., Rath, A. K., Mahapatra, A., & Sethy, P. K. (2020). Identification, classification & grading of fruits using machine learning & computer intelligence: a review. *Journal of Ambient Intelligence and Humanized Computing*, 1-11.

[6] Bhargava, A., & Bansal, A. (2021). Fruits and vegetables quality evaluation using computer vision: A review. *Journal of King Saud University-Computer and Information Sciences, 33*(3), 243-257.

[7] Ghazal, S., Qureshi, W. S., Khan, U. S., Iqbal, J., Rashid, N., & Tiwana, M. I. (2021). Analysis of visual features and classifiers for Fruit classification problem. *Computers and Electronics in Agriculture, 187*, 106267.

[8] Gill, H. S., & Khehra, B. S. (2021). Hybrid classifier model for fruit classification. *Multimedia Tools and Applications, 80*(18), 27495-27530.

[9] Indriani, O. R., Kusuma, E. J., Sari, C. A., & Rachmawanto, E. H. (2017). *Tomatoes classification using K-NN based on GLCM and HSV color space.* Paper presented at the 2017 international conference on innovative and creative information technology (ICITech).

[10] Jana, S., & Parekh, R. (2017). *Shape-based fruit recognition and classification.* Paper presented at the International Conference on Computational Intelligence, Communications, and Business Analytics.

[11] Kaur, R., & Himanshi, E. (2015). *Face recognition using principal component analysis.* Paper presented at the 2015 IEEE international advance computing conference (IACC).

[12] Nosseir, A., & Ahmed, S. E. A. (2018). *Automatic identification and classifications for fruits using k-NN.* Paper presented at the Proceedings of the 7th International Conference on Software and Information Engineering.

[13] Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition, 29*(1), 51-59.

[14] Oltean, M. (2018). Fruits 360 dataset on kaggle. Retrieved from https://www.kaggle.com/moltean/fruits.

[15] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics, 9*(1), 62-66.

[16] Rajasekar, L., & Sharmila, D. (2019). Performance analysis of soft computing techniques for the automatic classification of fruits dataset. *Soft Computing, 23*(8), 2773-2788.

[17] San, M., Aung, M. M., & Khaing, P. P. (2019). Fruit Recognition Using Color and Morphological Features Fusion. *International Journal of Image, Graphics and Signal Processing, 10*(10), 8.

[18] Saranya, N., Srinivasan, K., Pravin Kumar, S., Rukkumani, V., & Ramya, R. (2019). *Fruit classification using traditional machine learning and deep learning approach.* Paper presented at the International Conference on Computational Vision and Bio Inspired Computing.

[19] Shukla, D., & Desai, A. (2016). *Recognition of fruits using hybrid features and machine learning.* Paper presented at the 2016 International Conference on Computing, Analytics and Security Trends (CAST).

[20] Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019). *A brief review of nearest neighbor algorithm for learning and classification.* Paper presented at the 2019 International Conference on Intelligent Computing and Control Systems (ICCS).

[21] Wang, Z., & Xue, X. (2014). Multi-class support vector machine *Support vector machines applications* (pp. 23-48): Springer.

[22] Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends, 1*(2), 56-70.

[23] Zeeshan, M., Prabhu, A., Arun, C., & Rani, N. S. (2020). *Fruit Classification System Using Multiclass Support Vector Machine Classifier.* Paper presented at the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC).