

# Automatic Fruit Classification Using Deep Learning for Industrial Applications

M. Shamim Hossain , Senior Member, IEEE, Muneer Al-Hammadi ,  
and Ghulam Muhammad , Member, IEEE

**Abstract**—Fruit classification is an important task in many industrial applications. A fruit classification system may be used to help a supermarket cashier identify the fruit species and prices. It may also be used to help people decide whether specific fruit species meet their dietary requirements. In this paper, we propose an efficient framework for fruit classification using deep learning. More specifically, the framework is based on two different deep learning architectures. The first is a proposed light model of six convolutional neural network layers, whereas the second is a fine-tuned visual geometry group-16 pretrained deep learning model. Two color image datasets, one of which is publicly available, are used to evaluate the proposed framework. The first dataset (dataset 1) consists of clear fruit images, whereas the second dataset (dataset 2) contains fruit images that are challenging to classify. Classification accuracies of 99.49% and 99.75% were achieved on dataset 1 for the first and second models, respectively. On dataset 2, the first and second models obtained accuracies of 85.43% and 96.75%, respectively.

**Index Terms**—Computer vision, convolutional neural networks (CNNs), deep learning, fruit classification, VGG-16.

## I. INTRODUCTION

PRECISE classification of different kinds of fruit species, or food in general, is an important topic nowadays. It is not only a relevant topic in the field of academic research, but also in industrial applications. Many useful applications can be built based on such a classification system. In one of the main applications, it can be used in supermarkets to help cashiers. Cashiers need to identify not only the species of the fruit bought by the customer, but also its variety to determine the correct price. The list of prices must be stored in a query table. Such classification-based applications can identify the species purchased by the

Manuscript received September 6, 2018; accepted September 17, 2018. Date of publication October 10, 2018; date of current version February 1, 2019. This work was supported by the Deanship of Scientific Research, King Saud University, Riyadh, Saudi Arabia, under the Research Group Project RGP-228. Paper no. TII-18-2331. (*Corresponding author: M. Shamim Hossain*)

M. S. Hossain is with the Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia (e-mail: mshossain@ksu.edu.sa).

M. Al-Hammadi and G. Muhammad are with the Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia (e-mail: eng.muneer2008@gmail.com; ghulam@ksu.edu.sa).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TII.2018.2875149

customer and match it automatically with the correct price. For packaged products, this problem has been solved effectively via barcode reader systems. Unfortunately, this solution is not applicable for fruits and vegetables because when a customer goes to the supermarket to buy fruit, he/she needs to select each fruit by himself/herself [1]. The fruit classification system can also be implemented as a smartphone application. This application can help people, especially those with health issues when deciding whether a specific fruit/vegetable species satisfies their dietary requirements. The application will identify the fruit species and display its matched list of contents, necessary information, and advice. Regrettably, fruit classification based on computer vision remains difficult for the following reasons.

- 1) Shape, color, and texture similarity among numerous species of fruits.
- 2) Extremely high variation in a single class, which depends on the phase of fruit maturity, and the form in which the fruit is presented (e.g., fruits inside plastic bags, sliced on dishes, or unpicked).

Many systems can be found in the literature to automate the inspection of fruits for defects, maturity phase identification, and category recognition. The method in [1] introduced a fuzzy model for ripening level classification of pineapples. Peak hue and a normalized brown area were the features extracted from a hue channel and opponent colors of CIELab\*. Particle swarm optimization was used to tune the parameters of the fuzzy model. An accuracy of 93.11% was obtained by the proposed system on the MUSA (corresponds to Musa Species) database, which has 3108 image samples for pineapple at different ripening stages. The same particle swarm optimization technique was used to parallelize the computing resource to achieve a fast response [2]. Zhang and Wu [3] investigated different multiclass kernel SVMs with appearance descriptors for fruit classification. A combination of color, texture, and shape descriptors was used. A dataset of 18 classes, consisting of 1653 color images, was used to evaluate the performance of the method. The best classification accuracy reported on that dataset was 88.2%. A hybridization classification method proposed in [4] incorporated feedforward neural network classifiers, where the parameters were trained by a fitness-scaled chaotic artificial bee colony optimization technique. A color histogram, in addition to texture and shape descriptors of fruits, was used as a feature. The same dataset of 18 classes in the previous work [3] was used to estimate the performance of the method. The best classification accuracy was 89.1%. The proposed system in

[5] investigated the same appearance descriptors in [3] and [4] with a single hidden layer feedforward neural network. A hybridization of particle swarm optimization and an artificial bee colony was used to optimize the proposed network. An accuracy of 89.5% was achieved by evaluating the proposed method on the same dataset in [3] and [4]. The method in [6] examined quite a few appearances, color, texture, and shape-based image descriptors using several machine learning algorithms such as a support vector machine (SVM), a linear discriminant analysis (LDA), classification trees, ensembles of trees, and K-nearest neighbors (K-NN). The proposed approaches were evaluated on supermarket fruit dataset, which contains 2633 images distributed among 15 different classes. The best accuracy reported was 97%. Nandi *et al.* [7] proposed a method to classify mango fruits into four levels of maturity. The method used an ensemble of seven binary SVM classifiers. The descriptors were statistic calculations based on the RGB color channel values in different regions of the fruit. The method was evaluated on a dataset of 16 400 images distributed among four maturity levels. An average accuracy of 96% was achieved.

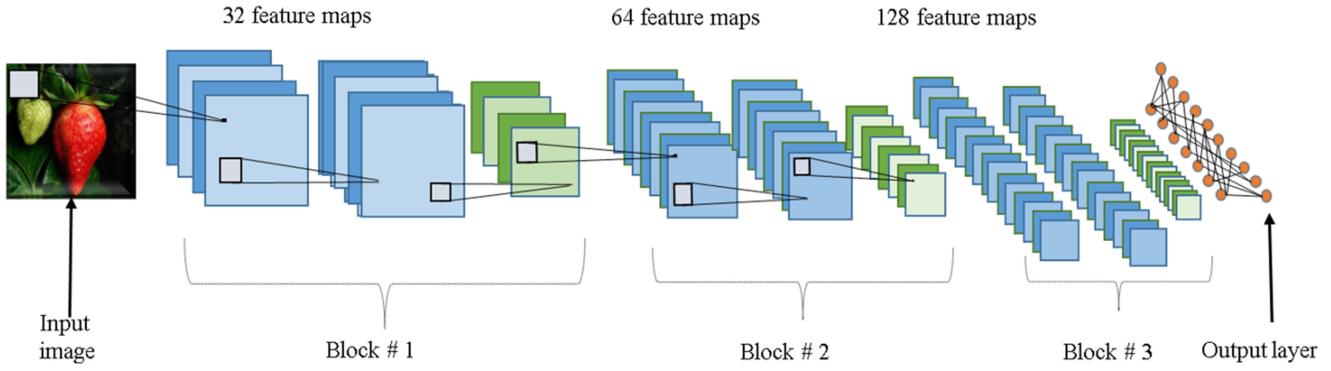
The method in [8] investigated an improved sum and difference histogram of intensity values of the neighborhood as a feature space with the SVM classifier for fruit classification. The reported accuracy, on a supermarket dataset of 2633 images and 15 classes [6], was 99%. A date-fruit classification method using shape features and image descriptors was introduced in [9]. A local binary pattern was used as the image descriptor. The obtained accuracy was as high as 99%; however, the elliptical shape feature was limited to date-fruits. An oil palm fruit bunch maturity classification accuracy of 85% was achieved in [10]. Different classifiers, such as quadratic discriminant analysis, K-NN, and LDA, were investigated on the spectral reflectance data of a multiband sensor system. A parallelepiped classifier with hyperspectral imaging and principal component analysis (PCA) were investigated in [11] for hidden bruise detection in kiwi fruit. A detection accuracy of 85.5% was reported. Different classification techniques, such as partial least square discriminant analysis (PLS-DA) and soft independent modeling of class analogy, were proposed in [12]. Visible spectroscopy was used to classify longan fruit into bruised and nonbruised. An accuracy of 100% was reported in the best case. Different shape descriptors such as area, perimeter, and axis lengths were employed in [13] to classify fruits of seven species. Naïve Bayes, multi-layer perceptron (MLP) neural network, and K-NN were the investigated classifiers. The accuracies on a dataset of 210 images were 95.24%, 91.43%, and 88.57% for the mentioned classifiers, respectively. A chilling injury scale classification in cucumbers was researched in [14]. Different feature selection methods with various classifiers, such as SVM, K-NN, and naïve Bayes, were investigated on hyperspectral cucumber images. A dataset of 334 samples was used to evaluate different combinations of feature selection and classifiers. An accuracy of 100% was stated for the best scenario. Another method to classify cucumbers into desirable (cylindrical) and undesirable (curved or conical) shapes was proposed in [15]. Centroid nonhomogeneity and width nonhomogeneity were investigated in addition to other common shape descrip-

tors such as area, perimeter, and roundness. An artificial neural network with different selected features was evaluated on a dataset of 102 images. An average classification accuracy of 97.1% was obtained. Apple disease classification was presented in [16]. It classified apples into one of the four classes, healthy fruit or infected by blotch, rot, or scab infections. K-means-based segmentation was used to detect the infected regions. Different combinations of appearance-based descriptors, such as local binary pattern, Zernike moment, and color coherence vector, were tested with multiclass SVM. An accuracy of 95.94% was reported for the best case of evaluation on a dataset of 320 samples. A genetic-optimized SVM was proposed in [17] to identify apples and differentiate them from tree leaves and branches for harvesting robotically. A fusion of three-dimensional (3-D) geometric descriptors with color components extracted from point cloud data input was used as features for classification. An accuracy of 92.3% was reported for the best case of apple identification. Zoning and character-edge distance, common descriptors in handwritten image recognition, were employed in [18] for fruit classification. Different combinations of descriptors were used with the discrete Fourier transform. A dataset of 163 fruit images was used to evaluate the performance of MLP and K-NN. The best accuracy reported was 97.5%. Another model was proposed in [19] for quality-attribute estimation and ripeness stage classification in strawberries. Multispectral imaging was investigated in the proposed model with different classifiers such as neural networks, SVM, and PLSs. It was reported that the best accuracy of 100% was obtained by SVM for ripeness stage classification on a dataset of 280 images.

The aforementioned classification techniques have one or more shortcomings as mentioned in the following.

- 1) The classifier may not be robust because dissimilar fruit images may have similar or identical shape and color features.
- 2) Some recognition systems are not appropriate to classify all fruit species; they can simply classify the diversities of the same class such as [1], [7], [9], and [15]. Other systems may only be used to detect infections or the level of maturity in the same fruit species such as [10]–[12], [14], [16], and [19].
- 3) The recognition system may require extra sensors, such as gas, invisible light, or weight sensors, as in [20] and [22].

To overcome these shortcomings, this paper presents an efficient deep learning-based framework for visual fruit classification. More specifically, our framework is based on the convolutional neural networks (CNNs), which are multilayered feedforward neural networks that are able to learn task-specific invariant features in a hierarchical manner [23]. Over the past few years, deep learning models have been intensively utilized for automatic feature engineering. These models showed robust ability in feature representation [24]. They became a common solution to deal with the rapid growth of heterogeneous big data [25]. These models achieved excellent accuracy in different approaches such as image classification, object recognition, and speech recognition [26] [27]. For these reasons, we were encouraged to employ deep learning for fruit classification.



**Fig. 1.** Block diagram of the proposed light architecture. The blue layers represent CNN layers, the green layers represent max pooling layers, and the orange nodes represent a hidden fully connected layer and the output layer.

The proposed framework avoids the problems of other shallow learning-based approaches. Moreover, the need for huge annotated data to fit deep models was solved via the application of transfer learning and augmentation principles.

The presented framework investigates two different deep learning architectures. It is evaluated on two distinct datasets of color fruit images. Our proposed deep learning-based framework nearly overcomes all the aforementioned challenges. The rest of this paper is organized as follows. Section II describes the proposed framework in detail. Section III reports the experimental results. Section IV concludes the paper.

## II. PROPOSED FRAMEWORK

The proposed framework investigates two deep learning architectures. The first architecture is a suggested light model of six CNNs layers and the second one is based on a pretrained Visual Geometry Group (VGG)-16 deep learning model.

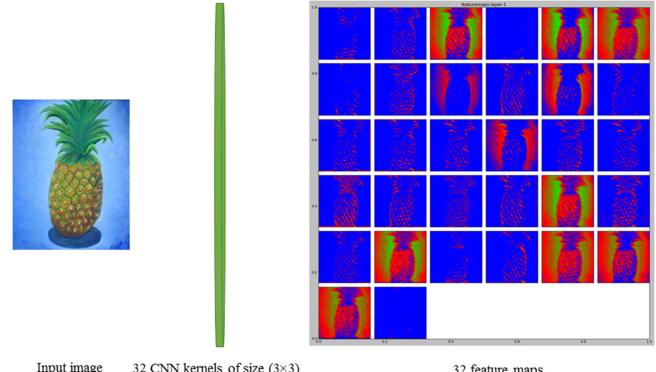
### A. Light Architecture

A block diagram of the light architecture is shown in Fig. 1. It consists of three steps—preprocessing, feature transformation, and classification.

In the preprocessing step, we cropped the image to produce an image with equal height and width, both dimensions equal to the smallest one in the original image. Then all the images were resized to a unified fixed size of  $64 \times 64 \times 3$ , which is the input shape (3 corresponds to three color channels) of the first layer of the model.

The feature transformation step was achieved via repeated convolution and pooling operations. As it is clear in the diagram, this step consists of three consecutive blocks followed by a fully connected layer. Each block contains two successive convolution layers followed by a max pooling layer. All the layers in the model use rectified linear units (ReLU) for activation, which is the simplest nonlinear function to be used for activation. As defined in the following equation, the ReLU function has a simple derivative, which makes it fast for training large networks [28]:

$$\text{ReLU}(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}. \quad (1)$$



**Fig. 2.** Thirty-two feature maps in the first layer of our model.

Fixed-sized kernels of  $3 \times 3$  for convolution and  $2 \times 2$  for max pooling are used in this model. The stacked CNN layers represent the input image at different levels of abstraction. Fig. 2 illustrates a sample input image and its representative 32 feature maps produced by the first CNN layer in this model.

The final step is the classification. In this step, the 2-D feature maps that were learned in the previous step are flattened into a 1-D feature vector. This 1-D feature vector is fed to the output layer. The output layer is a fully connected layer of neurons with the same number of classes. The output layer uses softmax activation because it has to output the probability for each class. We also add dropout layers after each pooling layer and before the output layer. Dropout layers regularize the network and protect it from overfitting. The first model training computation cost is low because there is a limited number of learnable parameters.

### B. VGG-16-Based Architecture

A block diagram of this model is shown in Fig. 3. VGG-16 is a deeper CNN model. It consists of five blocks of convolutional operations. Adjacent blocks are connected via a max pooling layer. Each block contains a series of  $3 \times 3$  convolutional layers. The number of convolution kernels stays the same within each block and increases from 64 in the first block to 512 in the last one [29]. The total number of learnable layers is 16. Fig. 4 illustrates a sample input image and its representative 64 feature

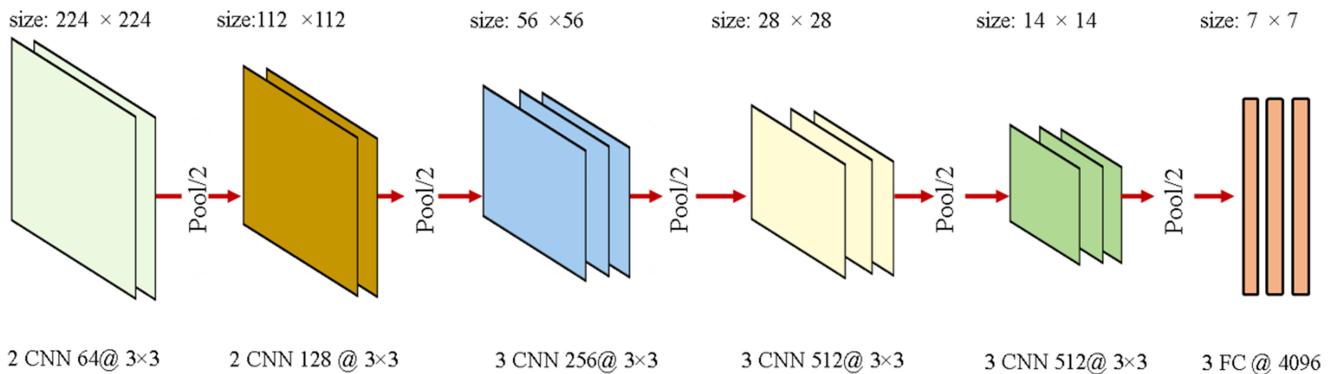


Fig. 3. VGG-16 model. The input size for each block is mentioned at the top of each block, while the structure is described at the bottom.

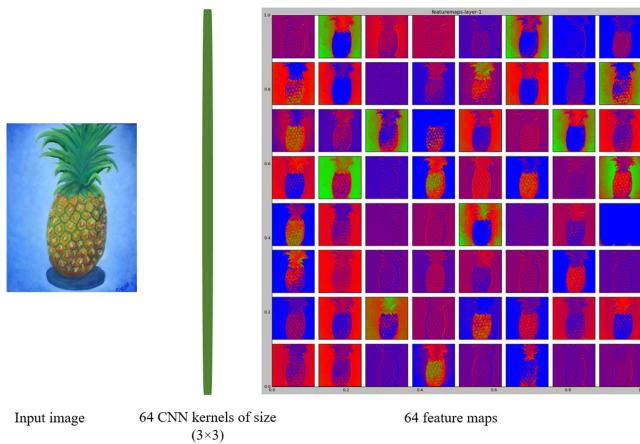


Fig. 4. Sixty-four feature maps in the first layer of the VGG-16.

maps produced by the first CNN layer in the VGG-16 fine-tuned model.

### III. EXPERIMENTAL RESULTS AND DISCUSSION

We carried out many experiments to evaluate the proposed framework and compare its performance with the performance of existing methods for fruit classification. Two datasets were used in our experiments.

*Dataset 1 (supermarket produce dataset):* This is a small public dataset collected a few years ago [6]. It contains 2633 color images.

All the images were collected in JPG format, with a fixed resolution of  $1024 \times 768$ . The images are distributed among 15 classes as illustrated in Table I. The images in this dataset are in RGB, where each color channel contains 8 bits per pixel. The images were captured at different dates and times [6]. Fig. 5 shows sample images from this dataset.

*Dataset 2:* This is our own dataset. We collected the images of this dataset from the Internet. The total number of images in this dataset is 5946, distributed among 10 classes as illustrated in Table II. Fig. 6 shows sample images from this dataset.

It can be noticed from the samples in Figs. 5 and 6 that the images in dataset 1 are simple and easy to be classified because the fruits in each image are of a single class and the background

TABLE I  
DATASET 1 CLASSES AND SAMPLES

Class	Num. of samples	Class	Num. of samples
Agata-potato	201	Nectarine	247
Asterix-potato	182	Onion	75
Cashew	210	Orange	103
Diamond-peach	211	Plum	264
Fuji-apple	212	Spanish-pear	159
Granny-smith-apple	155	Tahiti-lime	106
Honeydew melon	145	Watermelon	192
Kiwi	171		

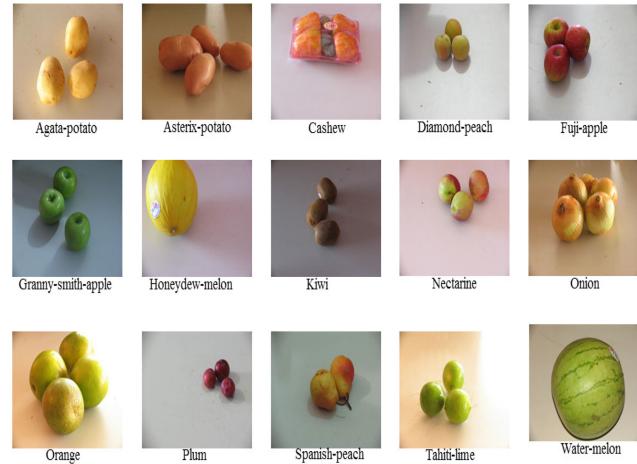


Fig. 5. Sample images from dataset 1.

TABLE II  
DATASET 2 CLASSES AND SAMPLES

Class	Num. of samples	Class	Num. of samples
Pineapple	525	Orange	521
Avocado	525	Papaya	515
Banana	622	Pomegranate	602
Carrot	563	Strawberry	894
Kiwi	724	Watermelon	455

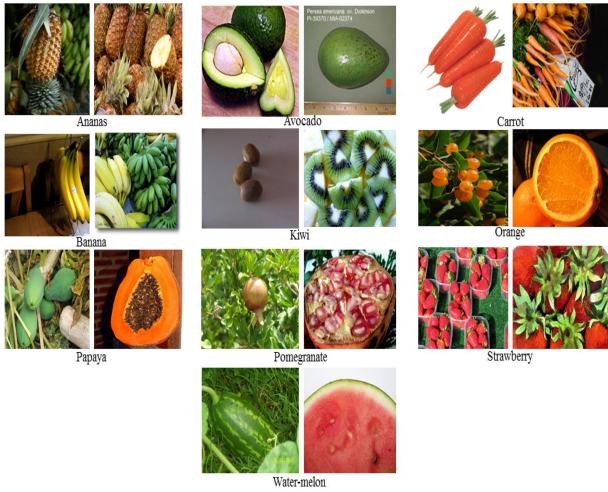


Fig. 6. Sample images from dataset 2.

is homogeneous. Dataset 2, however, is more challenging because, in many cases, the images contain fruits from multiple classes and the background is not homogeneous.

Dataset 2 consists of a variety of poses of same types of images. It also contains images having uncut fruits, fruits cut in half, or partially occluded fruits. In many images, the fruits are inside plastic bags, sliced on dishes, or unpicked in a very challenging environment.

In all experiments, after preprocessing the involved dataset, it was split into training and test datasets, where 85% of the images in the dataset were used for training and the rest 15% were used for testing. Moreover, during fitting, 5% of the training dataset was used for validation. A stochastic gradient descent was our choice for optimization with adaptive learning rate  $\alpha$  in each experiment. The learning rate value changes for each epoch; its value depends on the epoch number as

$$\alpha_n = \alpha_0 \times 0.1^{\frac{ep(n)}{10}} \quad (2)$$

where  $\alpha_n$  is the learning rate in epoch number  $n$ ,  $\alpha_0$  is the initial learning rate value, and  $ep(n)$  is the current epoch number.

#### A. Experiments on Dataset 1

In this part, we used a batch size of 32 samples and 100 epochs to fit the model on the training dataset.

- In the first experiment, we evaluated the performance of the proposed light model on dataset 1. We achieved an accuracy of 88.35% on the test data. The behavior of training and validation accuracy with respect to the epoch number is illustrated in Fig. 7. The confusion matrix in Fig. 8 shows the performance of the model on the test dataset. In the confusion matrices, the row represents the actual fruit class and the column represents the fruit class predicted by the model.

For example, in Fig. 8, four Agata-potato image samples were predicted as watermelon samples. The accuracy achieved on the test dataset is low although the images of this dataset are simple and have a wide interclass variation and a limited intraclass

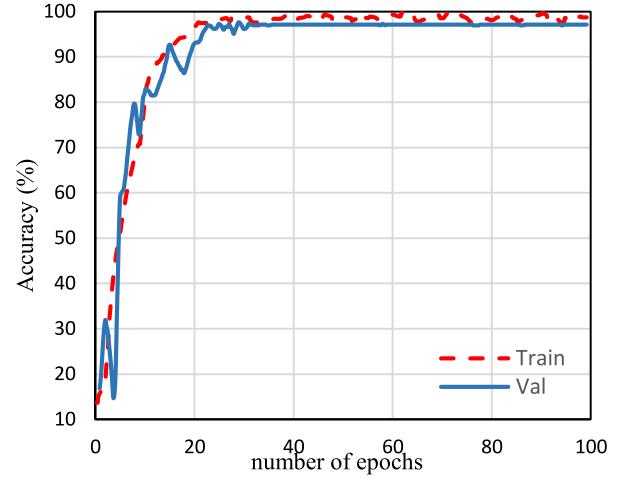


Fig. 7. Training versus validation accuracy for dataset 1.

	Agata-potato	Asterix-potato	Cashew	Diamond-peach	Fuji-apple	Granny-smith-apple	Honeydew-melon	Kiwi	Nectarine	Onion	Orange	Plum	Spanish-pear	Tahiti-lime	Watermelon
Agata-potato	23	3	0	0	0	0	0	0	0	0	0	0	2	0	4
Asterix-potato	0	28	0	0	1	0	0	0	0	0	0	0	0	0	0
Cashew	0	0	36	0	0	0	0	0	0	0	0	0	0	0	0
Diamond-peach	0	0	0	17	8	2	0	0	0	0	0	0	1	0	0
Fuji-apple	0	0	0	2	32	0	0	0	0	0	0	0	0	0	0
Granny-smith-apple	1	0	0	0	0	17	0	1	0	0	0	0	0	0	0
Honeydew-melon	0	0	0	0	0	0	12	0	0	0	0	0	1	1	0
Kiwi	1	0	0	1	0	2	0	20	3	0	0	0	0	0	0
Nectarine	0	0	0	3	0	0	0	0	30	0	0	1	0	0	0
Onion	1	2	0	0	0	0	0	0	0	8	0	0	0	0	0
Orange	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0
Plum	0	0	0	0	0	0	0	0	0	0	0	41	0	0	0
Spanish-pear	2	0	0	0	0	0	0	0	0	0	0	0	25	0	0
Tahiti-lime	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0
Watermelon	0	1	0	0	0	0	0	1	0	0	0	0	0	0	32

Fig. 8. Confusion matrix for dataset 1.

variation. The expected reason for such low classification accuracy is that the training dataset is very small.

- In the second experiment, we tested the effect of data augmentation on the performance of the light model. We performed intensive augmentation on the training dataset. The 32 samples in each batch were randomly subjected to the following operations:
  - horizontal and vertical shift with a random value not more than 0.2 of the original dimension;
  - zooming with a random rate less than 0.2;
  - rotation with a random value between 0° and 90°;
  - randomly flipping some samples horizontally or vertically.

The test accuracy rose to 99.49%. The training and validation accuracy behavior and the performance of the model on the test dataset are shown in Figs. 9 and 10, respectively. There is a clearly significant enhancement in the model accuracy, which can be noticed in the training and validation accuracy curves.

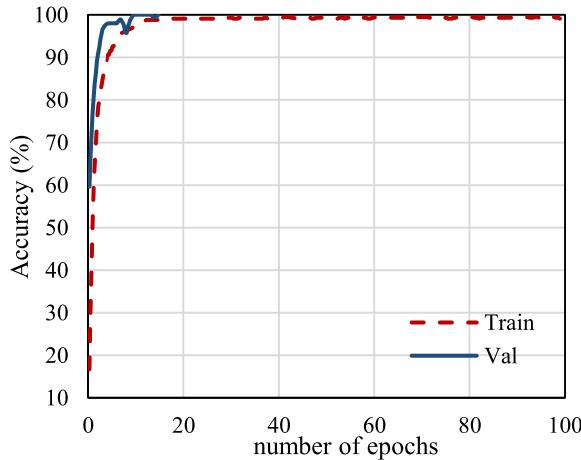


Fig. 9. Training versus validation accuracy for dataset 1 with augmentation.

	Agata-potato	Asterix-potato	Cashew	Diamond-peach	Fuji-apple	Granny-smith-apple	Honeydew-melon	Kiwi	Nectarine	Onion	Orange	Plum	Spanish-pear	Tahiti-lime	Watermelon
Agata-potato	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Asterix-potato	0	28	0	0	0	0	0	0	0	1	0	0	0	0	0
Cashew	0	0	36	0	0	0	0	0	0	0	0	0	0	0	0
Diamond-peach	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0
Fuji-apple	0	0	0	0	34	0	0	0	0	0	0	0	0	0	0
Granny-smith-apple	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0
Honeydew-melon	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0
Kiwi	0	0	0	0	0	0	27	0	0	0	0	0	0	0	0
Nectarine	0	0	0	0	0	0	0	34	0	0	0	0	0	0	0
Onion	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Orange	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0
Plum	0	0	0	0	0	0	0	0	0	0	0	0	41	0	0
Spanish-pear	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0
Tahiti-lime	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0
Watermelon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34

Fig. 10. Confusion matrix for dataset 1 with augmentation.

3) From the confusion matrix, we notice that only two samples were misclassified from a total of 395 samples in the test dataset. In the third experiment, we evaluated the performance of a fine-tuned VGG-16 model on dataset 1. In this case, we only fine-tuned the last layers of this model on dataset 1. A small value of 0.001 was used as an initial learning rate over 100 epochs and the same batch size of 32 samples as in the previous experiments was used. This model achieved a classification accuracy of 99.75%. There is only one test sample misclassified as in the confusion matrix in Fig. 11. The behavior of training and validation accuracy versus epoch number during fine-tuning the model is shown in Fig. 12.

### B. Experiments on Dataset 2

In this part, we used a batch size of 32 samples and 50 epochs to fit the model on the training dataset as dataset 2 is larger than dataset 1.

	Agata-potato	Asterix-potato	Cashew	Diamond-peach	Fuji-apple	Granny-smith-apple	Honeydew-melon	Kiwi	Nectarine	Onion	Orange	Plum	Spanish-pear	Tahiti-lime	Watermelon
Agata-potato	31	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Asterix-potato	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0
Cashew	0	0	36	0	0	0	0	0	0	0	0	0	0	0	0
Diamond-peach	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0
Fuji-apple	0	0	0	0	34	0	0	0	0	0	0	0	0	0	0
Granny-smith-apple	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0
Honeydew-melon	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0
Kiwi	0	0	0	0	0	0	27	0	0	0	0	0	0	0	0
Nectarine	0	0	0	0	0	0	0	34	0	0	0	0	0	0	0
Onion	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0
Orange	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0
Plum	0	0	0	0	0	0	0	0	0	0	41	0	0	0	0
Spanish-pear	0	0	0	0	0	0	0	0	0	0	0	27	0	0	0
Tahiti-lime	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
Watermelon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34

Fig. 11. Confusion matrix of the VGG-16 model for dataset 1.

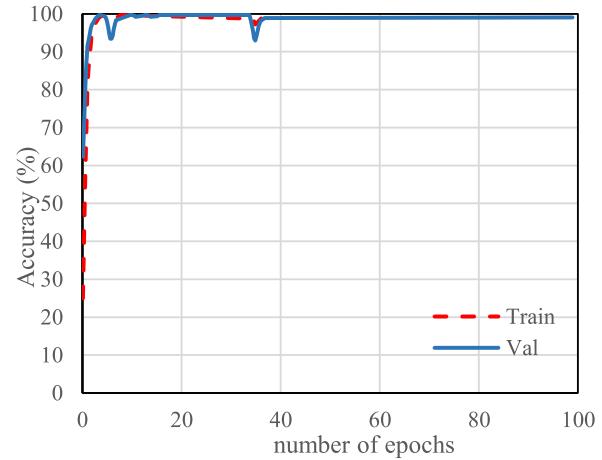


Fig. 12. Training versus validation accuracy for fine-tuning the VGG-16 for dataset 1.

- 1) In the first experiment, we evaluated the performance of the proposed light model on dataset 2. The classification accuracy of the model, in this case, was very bad; it was slightly below 35%. The training and validation accuracy behavior during model fitting is illustrated in Fig. 13. This poor accuracy might be attributed to the relatively small size of the dataset for training such a deep architecture from scratch and the difficulty of differentiating between various classes in this dataset, which is clearly noticeable in Fig. 6.
- 2) In the second experiment on dataset 2, we tested the effect of data augmentation on the performance of the proposed light model. We performed image augmentation on the training dataset by applying the same operations mentioned in the previous part with the same parameters. The test accuracy rose to 85.43%. The training and validation accuracy behavior and the confusion matrix are shown in Figs. 14 and 15, respectively.

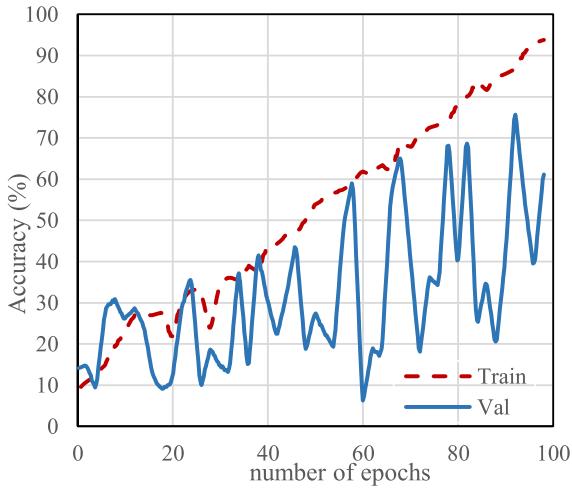


Fig. 13. Training versus validation accuracy for dataset 2.

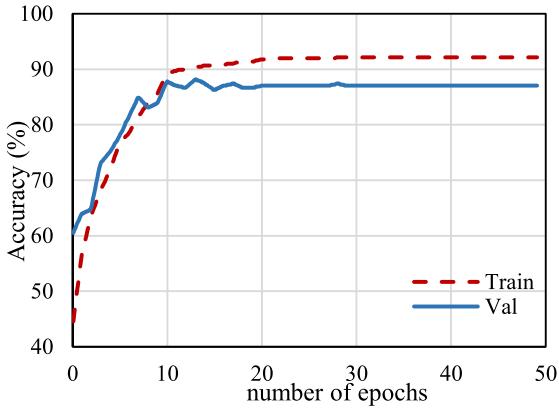


Fig. 14. Training versus validation accuracy for dataset 2 with augmentation.

	Pineapple	Avocado	Banana	Carrot	Kiwi	Orange	Papaya	Pomegranate	Strawberry	Watermelon
Pineapple	76	0	1	1	1	0	1	0	0	2
Avocado	1	59	4	0	1	0	1	1	0	9
Banana	3	0	75	0	1	1	5	1	1	4
Carrot	2	1	1	57	0	3	1	3	4	0
Kiwi	0	3	6	0	97	0	1	1	0	1
Orange	2	0	1	5	0	68	2	0	0	0
Papaya	1	5	7	3	2	2	59	2	0	6
Pomegranate	0	0	0	6	0	2	0	84	4	2
Strawberry	0	0	0	0	0	0	0	2	133	1
Watermelon	3	3	0	0	0	0	0	0	3	54

Fig. 15. Confusion matrix for dataset 2 with augmentation.

- 3) In the third experiment, we evaluated the performance of a fine-tuned VGG-16 model on dataset 2. In this case, we froze the first ten layers of the model and only fine-tuned the last layers. A small value of 0.001 was used for the learning rate over 50 epochs and the same batch

	Pineapple	Avocado	Banana	Carrot	Kiwi	Orange	Papaya	Pomegranate	Strawberry	Watermelon
Pineapple	82	0	0	0	0	0	0	0	0	0
Avocado	0	73	1	0	0	0	2	0	0	0
Banana	1	1	89	0	0	0	0	0	0	0
Carrot	0	0	2	69	0	1	0	0	0	0
Kiwi	0	1	0	0	108	0	0	0	0	0
Orange	1	0	0	2	0	72	2	1	0	0
Papaya	0	1	0	0	0	2	80	1	0	3
Pomegranate	0	0	0	1	1	1	1	94	0	0
Strawberry	0	0	0	1	0	0	0	0	135	0
Watermelon	0	1	0	0	0	0	1	0	0	61

Fig. 16. Confusion matrix of the VGG-16 model for dataset 2.

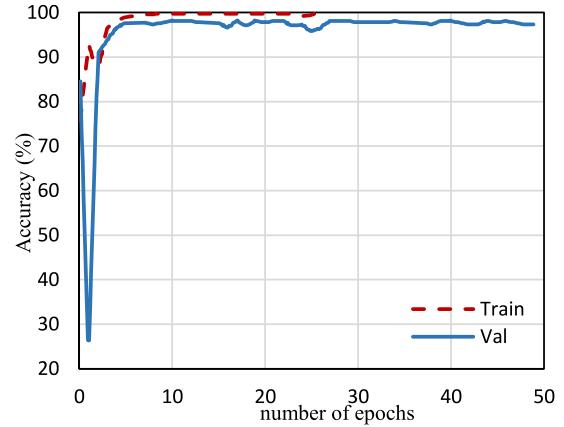


Fig. 17. Training versus validation accuracy for fine-tuning the VGG-16 for dataset 2.

TABLE III  
PERFORMANCE COMPARISON WITH OTHER WORK

Method	Accuracy (%)
The method in [6]	88.2
The method in [8]	98
Proposed with light architecture	99.49
Proposed with VGG-16 fine tuning	99.75

size of 32 as in the previous experiments. The model, in this case, achieved a classification accuracy of 96.75%. The confusion matrix of the model for the test dataset is illustrated in Fig. 16. The behavior of training and validation accuracy versus epoch number during fine-tuning the model is shown in Fig. 17.

### C. Comparison With Other Classification Methods

For comparison with other proposed techniques, we tested our framework on the publicly available supermarket produce dataset (dataset 1). Two reports in the literature were evaluated using this dataset.

The accuracies of our framework and these two works are shown in Table III.

#### IV. CONCLUSION

A deep learning-based framework for fruit classification was proposed in this paper. Two CNN models were investigated in the proposed framework, a small CNN model and a VGG-16 fine-tuned model. Two datasets with different sizes and complexity were used to evaluate the proposed framework. The VGG-16 fine-tuned model achieved excellent accuracy on both datasets. The light CNN model also achieved excellent accuracy on dataset 1 with data augmentation. The performance of the two models has been compared with two other methods in the literature. It was found that the two proposed models outperformed the two existing methods on dataset 1.

As for future work, we will generalize the evaluation of the proposed framework for more classes (using extra fruit and vegetable species). We will also investigate the effect of different parameters such as activation function, pooling function optimization method, and a loss function. The proposed framework can also be deployed into a cloud-based framework.

#### REFERENCES

- [1] S. Marimuthu and S. M. Roomi, "Particle swarm optimized fuzzy model for the classification of banana ripeness," *IEEE Sensors J.*, vol. 17, no. 15, pp. 4903–4915, Aug. 2017.
- [2] M. S. Hossain, M. Moniruzzaman, G. Muhammad, A. Al Ghoneim, and A. Alamri, "Big data-driven service composition using parallel clustered particle swarm optimization in mobile environment," *IEEE Trans. Services Comput.*, vol. 9, no. 5, pp. 806–817, Sep./Oct. 2016.
- [3] Y. Zhang and L. Wu, "Classification of fruits using computer vision and a multiclass support vector machine," *Sensors*, vol. 12, no. 9, pp. 12489–12505, Sep. 2012.
- [4] Y. Zhang, S. Wang, G. Ji, and P. Phillips, "Fruit classification using computer vision and feedforward neural network," *J. Food Eng.*, vol. 143, pp. 167–177, Dec. 2014.
- [5] S. Lu, Z. Lu, P. Phillips, S. Wang, J. Wu, and Y. Zhang, "Fruit classification by HPA-SLFN," in *Proc. 8th IEEE Int. Conf. Wireless Commun. Sig. Process.*, Yangzhou, China, Oct. 2016, pp. 1–5.
- [6] A. Rocha, D. C. Haugage, J. Wainer, and S. Goldenstein, "Automatic fruit and vegetable classification from images," *Comput. Electron. Agriculture*, vol. 70, no. 1, pp. 96–104, Jan. 2010.
- [7] C. S. Nandi, T. Bipan, and C. Koley, "A machine vision-based maturity prediction system for sorting of harvested mangoes," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 7, pp. 1722–1730, Jul. 2014.
- [8] S. R. Dubey and A. S. Jalal, "Robust approach for fruit and vegetable classification," *Procedia Eng.*, vol. 38, pp. 3449–3453, 2012.
- [9] G. Muhammad, "Date fruits classification using texture descriptors and shape-size features," *Eng. Appl. Artif. Intell.*, vol. 37, pp. 361–367, 2015.
- [10] O. Saeed *et al.*, "Classification of oil palm fresh fruit bunches based on their maturity using portable four-band sensor system," *Comput. Electron. Agriculture*, vol. 82, pp. 55–60, Mar. 2012.
- [11] L. Qiang and T. Mingjie, "Detection of hidden bruise on kiwi fruit using hyperspectral imaging and parallelepiped classification," *Procedia Environ. Sci.*, vol. 12, pp. 1172–1179, Jan. 2012.
- [12] T. Pholpho, S. Pathaveerat, and P. Sirisomborn, "Classification of longan fruit bruising using visible spectroscopy," *J. Food Eng.*, vol. 104, no. 1, pp. 169–172, May 2011.
- [13] S. Jana and R. Parekh, "Shape-based fruit recognition and classification," in *Proc. Int. Conf. Comput. Intell., Commun., Bus. Analytics*, Singapore, Mar. 2017, pp. 184–196.
- [14] H. Cen, R. Lu, Q. Zhu, and F. Mendoza, "Nondestructive detection of chilling injury in cucumber fruit using hyperspectral imaging with feature selection and supervised classification," *Postharvest Biol. Technol.*, vol. 111, pp. 352–361, Jan. 2016.
- [15] K. Kheiraliour and A. Pormah, "Introducing new shape features for classification of cucumber fruit based on image processing technique and artificial neural networks," *J. Food Process Eng.*, vol. 40, no. 6, p. e12558, Mar. 2017, doi: [10.1111/jfpe.12558](https://doi.org/10.1111/jfpe.12558).
- [16] S. R. Dubey and A. S. Jalal, "Apple disease classification using color, texture and shape features from images," *Signal, Image Video Process.*, vol. 10, no. 5, pp. 819–826, Jul. 2016.
- [17] Y. Tao and J. Zhou, "Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking," *Comput. Electron. Agriculture*, vol. 142, pp. 388–396, Nov. 2017.
- [18] P. A. Macanhã, D. M. Eler, R. E. Garcia, and W. E. Junior, "Handwritten feature descriptor methods applied to fruit classification," in *Information Technology—New Generations*, vol. 558, S. Latifi, Ed., Cham, Switzerland: Springer, 2017, pp. 699–705.
- [19] C. Liu *et al.*, "Application of multispectral imaging to determine quality attributes and ripeness stage in strawberry fruit," *PLoS ONE*, vol. 9, no. 2, p. e87818, Feb. 2014.
- [20] P. Leekul, S. Chivapreecha, C. Phongcharoenpanich, and M. Krairiksh, "Rician k-factors-based sensor for fruit classification by maturity stage," *IEEE Sensors J.*, vol. 16, no. 17, pp. 6559–6565, Sep. 2016.
- [21] S. Wang, Y. Zhang, G. Ji, J. Yang, J. Wu, and L. Wei, "Fruit classification by wavelet-entropy and feedforward neural network trained by fitness-scaled chaotic ABC and biogeography-based optimization," *Entropy*, vol. 17, no. 8, pp. 5711–5728, 2015.
- [22] S. Ciptohadijoyo, W. S. Litananda, M. Rivai, and M. H. Purnomo, "Electronic nose based on partition column integrated with gas sensor for fruit identification and classification," *Comput. Electron. Agriculture*, vol. 121, pp. 429–435, Feb. 2016.
- [23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [24] X. Yuan *et al.*, "Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE," *IEEE Trans. Ind. Inform.*, vol. 14, no. 7, pp. 3235–3243, Jul. 2018.
- [25] P. Li *et al.*, "Deep convolutional computation model for feature learning on big data in internet of things," *IEEE Trans. Ind. Inform.*, vol. 14, no. 2, pp. 790–798, Feb. 2018.
- [26] Z. Gao, L. Wang, L. Zhou, and J. Zhang, "HEp-2 cell image classification with deep convolutional neural networks," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 2, pp. 416–428, Mar. 2017.
- [27] T. Kooi *et al.*, "Large scale deep learning for computer aided detection of mammographic lesions," *Med. Image Anal.*, vol. 35, pp. 303–312, Jan. 2017.
- [28] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," *IEEE Int. Conf. Acoust., Speech Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 8609–8613.
- [29] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," May 2014. [Online]. Available: arXiv:1405.3531.

**M. Shamim Hossain** (SM'09) received the Ph.D. degree in electrical and computer engineering from the University of Ottawa, Canada, in 2009. He is a Professor with the Department of Software Engineering, King Saud University, Riyadh, Saudi Arabia. His research interests include cloud networking, smart environment (smart city, smart health) social media, IoT, edge computing, multimedia for healthcare, deep learning approach for multimedia processing, and multimedia big data. He is the author or coauthor of more than 200 publications including refereed journals, conference articles, books, and book chapters.

Prof. Hossain is a Member of the editorial boards of *IEEE Network*, *IEEE Multimedia*, *IEEE Wireless Communications*, *IEEE Access*, *Journal of Network and Computer Applications*, and the *International Journal of Multimedia Tools and Applications*.

**Muneer Al-Hammadi** received the Master degree from the Department of Computer Engineering, King Saud University, Riyadh, Saudi Arabia, where he is currently working toward the Ph.D. degree.

His research interests include image and video processing, and deep learning.

**Ghulam Muhammad** (M'10) received the Ph.D. degree in electronics and computer engineering from Toyohashi University of Technology, Japan, in 2006. He is a Professor with the Department of Computer Engineering, King Saud University, Riyadh, Saudi Arabia. He is also affiliated to the Center for Smart Robotics Research at CCIS, King Saud University. His research interests include speech processing, image and video processing, deep learning, and smart healthcare.