IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 30, 2019, accepted August 18, 2019, date of publication August 20, 2019, date of current version September 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2936536

Date Fruit Classification for Robotic Harvesting in a Natural Environment Using Deep Learning

HAMDI ALTAHERI[1,2], (Member, IEEE), MANSOUR ALSULAIMAN[1,2], (Member, IEEE), AND GHULAM MUHAMMAD[1,2], (Senior Member, IEEE)

[1]Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[2]Center of Smart Robotics Research, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Corresponding author: Hamdi Altaheri (altaheri@ieee.org)

The authors would like to thank Deanship of Scientific Research for funding and supporting this research through the initiative of DSR Graduate Students Research Support (GSR).

ABSTRACT An accurate vision system to classify and analyze fruits in real time is critical for harvesting robots to be cost-effective and efficient. However, practical success in this area is still limited, and to the best of our knowledge, there is no research in the area of machine vision for date fruits in an orchard environment. In this work, we propose an efficient machine vision framework for date fruit harvesting robots. The framework consists of three classification models used to classify date fruit images in real time according to their type, maturity, and harvesting decision. In the classification models, deep convolutional neural networks are utilized with transfer learning and fine-tuning on pre-trained models. To build a robust vision system, we create a rich image dataset of date fruit bunches in an orchard that consists of more than 8000 images of five date types in different pre-maturity and maturity stages. The dataset has a large degree of variations that reflects the challenges in the date orchard environment including variations in angles, scales, illumination conditions, and date bunches covered by bags. The proposed date fruit classification models achieve accuracies of 99.01%, 97.25%, and 98.59% with classification times of 20.6, 20.7, and 35.9 msec for the type, maturity, and harvesting decision classification tasks, respectively.

INDEX TERMS Dates classification, maturity analysis, automated harvesting, deep learning, convolutional neural networks.

I. INTRODUCTION

Global date fruit production was 8.5 million tons in 2016 according to the Food and Agriculture Organization [1]. Date fruit cultivation is a major strategic agricultural industry in Middle East and North Africa countries, which produce 91% of the world's dates [1]. **FIGURE 1** shows the top 10 producer countries in the world. Dates are harvested and marketed as fresh ripe fruits at three stages of their development, Khalal, Rutab, and Tamar, which makes them one of the most suitable fruits for exportation and consumption throughout the year [2].

In date cultivation, manual harvesting is the dominant method used, which requires skilled workers to climb palm trees to reach date bunches. However, manual harvesting is dangerous and highly labor-intensive as well as inefficient in terms of both time and the economy. Such methods are the major cause of delays in the date production cycle and account for more than 45% of the date production cost [3]. Recently, due to the increase in date palm cultivation and shortage of skilled workers, the cost of date harvesting has increased significantly, necessitating a change to automated harvesting. Indeed, advanced agricultural automation such as robotic harvesting can significantly increase quality and yield as well as reduce production costs and delays [4].

One of the most important aspects of harvesting robots is their ability to interpret and analyze visual data. Using an accurate vision system to classify and analyze fruits in real time is critical for the harvesting robot to be cost-effective and efficient. However, practical success in this area remains limited due to the difficulties caused by unstructured and unconstrained agricultural environments [4]. Furthermore, research on machine vision for date fruits in the pre-harvesting and harvesting stages is scarce. Hence, in this work, we investigate the machine vision problem of date fruit classification based on type, maturity, and harvesting decision in an orchard

The associate editor coordinating the review of this article and approving it for publication was el-Hadi M. Aggoune.

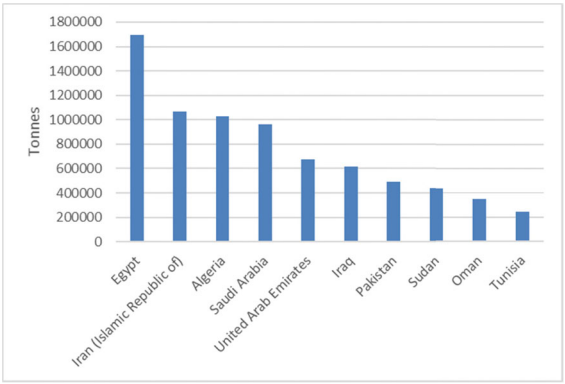


FIGURE 1. Date fruit production of the top 10 producing countries in 2016 [1].

environment. The decision to harvest date fruit depends on both fruit type (variety) and maturity stage. Date type is a key factor in determining the stage at which dates are harvested, unlike most fruits that only depend on the maturity stage of an individual fruit (i.e. ripe or not). For instance, some types of dates are harvested in the first maturity stage (Khalal) such as Barhi, whereas other date types are usually harvested later in the Rutab or Tamar stages such as Sullaj and Khalas. Therefore, an accurate, reliable, and real-time classification system for date fruits based on both type and maturity in a natural environment is essential to establish robotic harvesting.

Many studies have been performed to classify and harvest fruits other than dates such as berries [5], citrus fruits [6], and apples [7]. Compared with the research on other fruits, especially in the area of automated harvesting, research on date fruits is negligible. Most previous research has focused on dates after the harvesting phase, with research on date fruit classification or maturity analysis in an orchard environment lacking. Current studies of date fruits can be categorized into three groups: type classification [8]–[14], maturity analysis [15]–[17], and quality grading [18]–[26].

Most research on dates focuses on grading fruit quality. Date quality can be evaluated by several factors such as moisture and sugar content, hardness, and surface defects. For instance, in [19], the authors proposed an electronic sensor to measure the moisture content of dates to classify them into moist, semi-moist, and dry. In [20], the researchers adopted the hardness of dates as a quality indicator, using stepwise and linear discriminate analysis to grade dates into soft, semi-hard, and hard. Grading dates according to surface defects has been investigated in many studies. For example, [21], [22] proposed techniques using a co-occurrence matrix and color machine vision for date grading. The study in [23] sorted dates as defected or good using image analysis techniques. In [24], a back-propagation neural network was used to classify dates into three quality grades using size, shape, intensity, and surface defect features. In the more recent automatic date grading approach proposed by [26], fuzzy inference was used to measure the quality of dates based on length and freshness features.

Unlike quality grading, few studies have analyzed the maturity of date fruits. The study by [15] investigated date maturity using infrared spectrometry. In [16], the authors used a color distribution analysis and back projection to classify one date type into four maturity classes. In another study [17], a taxonomy classification method with RGB color and texture features including contrast, entropy, and homogeneity was used to classify one date type into four maturity classes.

The classification of date fruits according to their type also has limited research. In [8], the researchers developed a date fruit classification system using neural networks, and in [9] they used probabilistic neural networks for a similar task. In [11], 15 size, shape, color, and texture features were used to classify single date images into seven classes according to their type. In another study [10], a technique was proposed based on shape and size features and local texture descriptors to classify four classes of dates using 800 single date images. In a more recent study [12], the authors classified single date images based on their types using a dataset containing 5000 images of 10 date types. They used an RGB color histogram, a gray-level co-occurrence matrix, and four shape features including area and eccentricity, combined with a Gaussian mixture model. All these studies used single date images with a uniform background. However, the handcrafted feature-based approaches used in previous research are unsuitable for constructing a robust vision system that works in unstructured natural environments such as that of date orchards. In the most recent approach [14], the researchers reported an accuracy of 99.2% using a deep learning-based technique for date fruit classification. They built a dataset of four date types by acquiring date images from the Google search engine.

None of these previous works has investigated the problem of date fruit classification or maturity analysis in an orchard environment. All previous approaches have used datasets with images from a limited perspective. Further, most such images were for individual dates in the post-harvesting or post-production stages, so they were solving simpler problems than this work, which solves a real-life problem that has many difficulties.

Automatic fruit classification and maturity analysis in a natural environment are challenging machine vision tasks due to the difference in size, shape, color, and texture properties of various fruits; the large degree of uncertainty with unstructured orchard scenes; harsh occlusions; and highly variable illumination and shadow states. However, these tasks are more complicated in date fruits due to several aspects. First, date orchards usually have many types (varieties) of dates with numerous similarities in their visual appearance. Second, different date varieties in the same orchard are harvested in different maturity stages. Third, individual dates in one bunch do not mature uniformly (i.e. a date bunch usually has individual dates at different maturity stages), which complicates labeling or classifying date bunches into a specific maturity class. Fourth, date bunches can be covered with net bags, which distort their visual features. To overcome

these challenges, we propose an efficient deep learning-based machine vision framework for date fruit harvesting robots in an orchard environment. The framework consists of three classification models to classify date fruit images in real time according to their type, maturity, and harvesting decision. In the classification models, deep convolutional neural networks (CNNs) are utilized due to their robust ability in automatic feature representation for challenging classification tasks compared with handcrafted feature-based approaches. We employ transfer learning with fine-tuning using two pre-trained CNN models: AlexNet and VGGNet. Transfer learning is an effective technique used to obtain high accuracy with less training time using a relatively small dataset.

To build a robust vision system, a rich image dataset of date fruit bunches is created. The dataset includes a wide degree of variation to reflect the challenges in natural environments and date orchards. The dataset consists of more than 8000 images of date bunches of five date types in different pre-maturity and maturity stages. It also has large variations in scales, angles, and illumination with bagged and unbagged date bunches. The dataset is fully labeled according to type, maturity, and harvesting decision, and it is publicly available with its associated files to the research community in the IEEE DataPort repository [27] (http://dx.doi.org/10.21227/x46j-sk98).

The remainder of this paper is organized as follows. Section II describes the proposed framework in detail. Section III presents the dataset and labeling method. Section IV reports the experimental results. Finally, we conclude in Section V.

II. PROPOSED FRAMEWORK

The proposed framework consists of three classification models for type, maturity, and harvesting decision. The input to the framework is a stream of images (frames) from an RGB video camera in a date orchard. The framework has three outputs to determine the type, maturity stage, and harvesting decision of dates in each image. Type and maturity models are multiclass classifiers that use transfer learning with fine-tuning based on pre-trained CNN models. We investigate two popular CNN architectures that differ in size and depth: AlexNet [28] and VGGNet [29]. AlexNet has a light architecture with a small size, while VGGNet has a deeper architecture and a larger size. The harvesting decision model, on the contrary, is a binary classifier that uses the output of the two previous models with the harvesting rules, which are entered manually by a user, and then suggests the appropriate harvesting decision based on all the previous information. The block diagram in FIGURE 2 provides an overview of the proposed framework in the training phase (a) and real-time implementation (b).

A. CNN DEEP LEARNING

Deep learning has become the dominant approach in many computer vision tasks including object detection, recognition, and classification [30], [31]. Deep learning achieves high levels of success in these tasks due to the availability of a large number of labeled images, such as ImageNet [32], and substantial computing power devices such as GPUs or distributed large-scale clusters using cloud computing [33]. The success of deep learning also goes beyond images and video to speech and audio. Earlier, its success was restricted by its need for large databases and long training times. However, these problems have been solved using transfer learning and fine-tuning techniques.

A CNN is one of the most successful types of deep learning. It uses deep convolutional networks and non-linearity to learn local and spatial features and patterns directly from raw data such as images, video, text, and sound. A CNN thus learns the features from data automatically, eliminating the need to extract them manually. It can build complex features as an integration of simple features through a series of successive convolutional layers. The earlier layers learn low-level features such as edges and curves and the deeper layers learn to recognize complex high-level features such as entire objects in an image [30].

CNN models can be trained from scratch (with random initialization) or using transfer learning. Training from scratch needs a large dataset to learn millions of parameters. In many tasks such as fruit classification, it is rare for CNNs to learn from scratch due to the need for a labeled dataset of sufficient size. Instead, it is common to pre-train a CNN using a large-scale dataset (e.g. ImageNet, which has 1.2 million images with 1000 classes), and then use the CNN model either as a fixed feature extractor or as an initialization for other specific tasks. In transfer learning, the CNN learns generic mid- and low-level features from a large dataset that can be fine-tuned by other target datasets [34]. Therefore, in this work, two CNN architectures, well trained over the ImageNet dataset [32], are utilized and fine-tuned for the date fruit dataset. This approach learns features automatically with high accuracy and short training times.

B. ALEXNET ARCHITECTURE

The AlexNet model was the first deep architecture that popularized convolutional networks in computer vision and significantly improved the classification accuracy of the ImageNet ILSVRC2012 challenge compared with traditional methods. The AlexNet architecture (FIGURE 3-**b**) consists of eight learnable layers, five convolutional (conv) layers followed by three fully connected (FC) layers. Every convolutional and fully connected layer is attached by rectified linear unit (ReLU) as non-linear activation. Adding non-linearity using ReLUs helps CNNs train much faster. The first and second convolutional layers are followed by both local response normalization (LRN) and max-pooling layers, but only a max-pooling layer is used after the fifth convolutional layer. The first convolutional layer has 96 kernels (filters) of dimension 11×11 with a step (stride) of four pixels. The strides of the remaining convolutional layers are set to one pixel. The second layer has 256 kernels of dimension 5×5. The third, fourth, and fifth layers have 384,

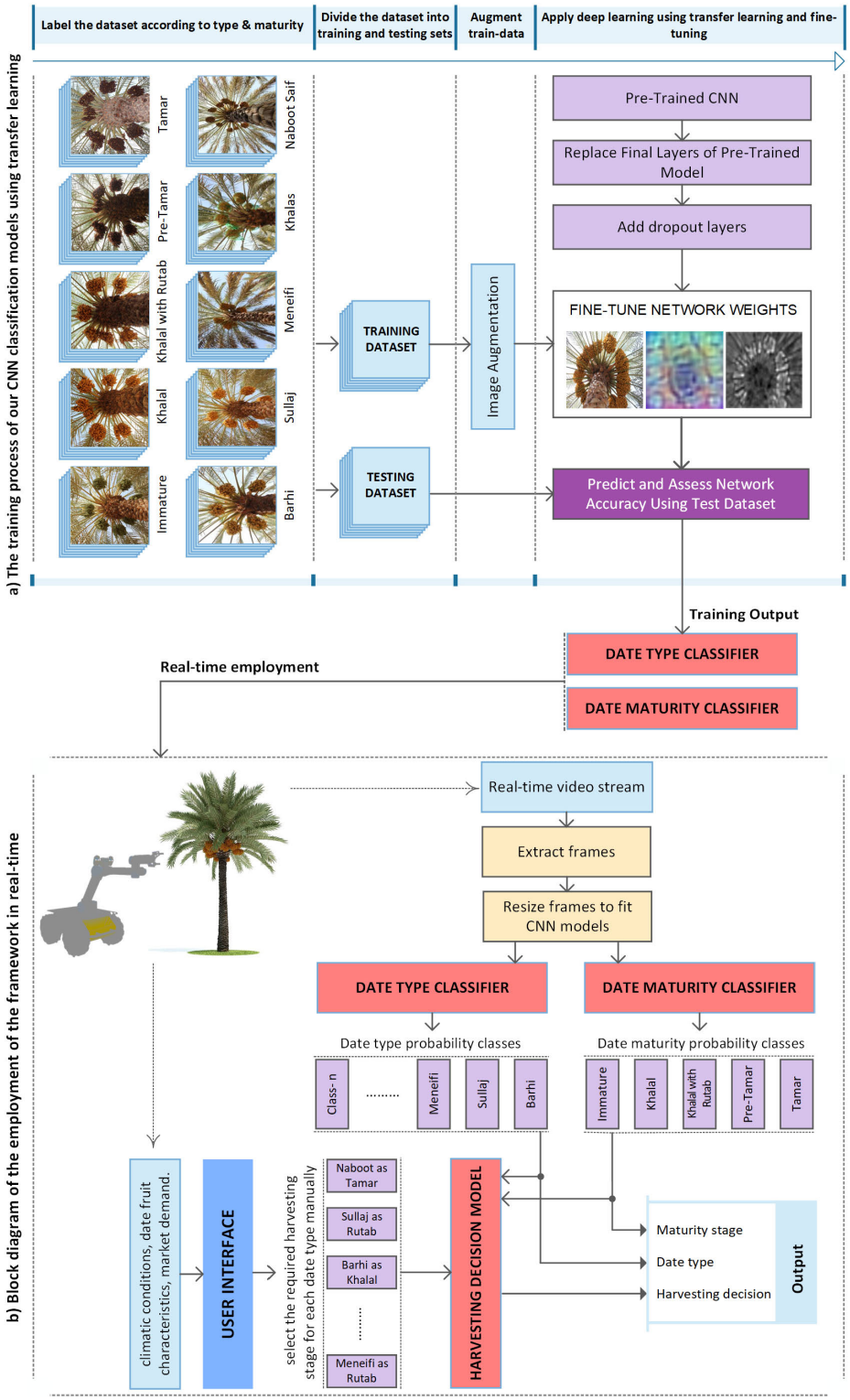


FIGURE 2. Block diagram of the proposed date fruit classification framework. (a) Training phase, (b) real-time employment.

384, and 256 kernels of dimension 3×3, respectively. Max-pooling layers use non-linear down-sampling to abstract the network. They thus retain the most important features and reduce the number of parameters that the network needs to learn, which diminishes network computation. The first two fully connected layers have 4096 neurons and the last one

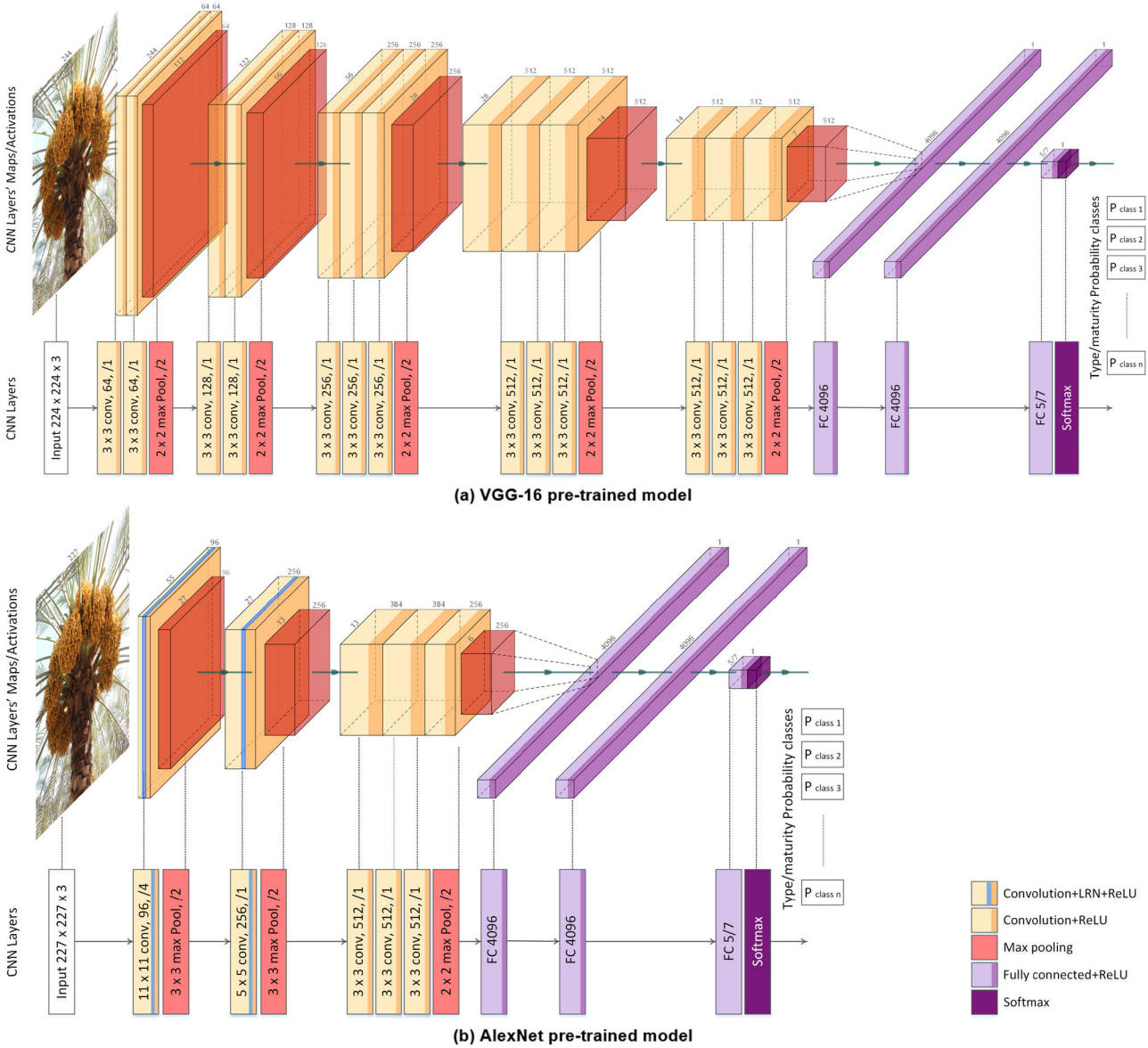


FIGURE 3. Deep learning architectures of the proposed date fruit classification framework based on the VGG-16 (a) and AlexNet (b) pre-trained models.

has 1000 neurons, which is the same as in the ImageNet classes. The neurons in fully connected layers are connected to all the neurons in the previous layer. The last layer of the model provides the classification output using the softmax classifier.

C. VGGNET ARCHITECTURE

VGG-16, which is the D-configuration in [29], also known as VGG-D, is the best network configuration among the VGGNets built by the VGG group [29]. VGG-16 (VGGNet) achieved 92.7% top-five accuracy in the ImageNet ILSVRC2014 challenge. Its architecture is popular in the literature due to its uniform structure and high accuracy in classification tasks. VGG-16 has a deeper network than AlexNet and consists of 16 trainable layers including 13 convolutional layers and three fully connected layers (FIGURE 3-**a**). The model features a homogeneous and smooth architecture that only uses filters of size 3×3 with a stride of one for convolutions, and 2×2 pooling with a stride of two in all layers. The convolutional layers are grouped into five blocks. Adjoining blocks are linked through a max-pooling layer, which performs down-sampling by half along the spatial dimensions. Max-pooling reduces the dimensions of the layers from 224×224 in the first block to 7×7 after the last one. The number of convolution filters remains fixed within one block and doubles after each max-pooling layer from 64 in the first block to 512 in the last block. As in AlexNet, the ReLU layer is tailed after each convolutional and fully connected layer. The fully connected layers in VGG-16 have the same configurations as in AlexNet.

VGG-16 does not use normalization layers due to their minimal contribution to enhancing accuracy.

D. TRANSFER LEARNING

Transfer learning makes CNNs work effectively in many visual classification tasks, even if their datasets have insufficient or limited size [35]. In transfer learning and fine-tuning techniques, the CNN network of a pre-trained model, trained on a large-scale dataset, can be used either as a feature extractor [36] or as a weight initializer rather than generating weight values randomly [37]. The work in [35] discussed the best practices for transfer learning with fine-tuning.

CNN models have millions of trainable parameters; for example, AlexNet has 60 million and VGG-16 has 138 million. Training these parameters from scratch is difficult using relatively small datasets such as the one in this work. CNN networks can easily memorize small datasets, which lead to overfitting. Hence, to prevent overfitting, transfer learning was used in this study. Moreover, the size of the dataset was increased using image augmentation performed on the training datasets randomly using different values. During the training phase, the date images in each batch were randomly subjected to the following operations: horizontal reflection, horizontal and vertical translation with a random value in the range $[-30\ 30]$ pixels, and horizontal and vertical scaling with a random rate in the range [0.9 1.1].

We used transfer learning with fine-tuning as demonstrated in [38]. In the fine-tuning technique, the weights of the CNN models were trained starting from the transferred values for all the layers excluding the last fully connected layer. Because this layer was trained to classify 1000 categories (ImageNet classes), we replaced it with a new one to classify the new tasks classes, as shown in FIGURE 3. The weights of the new layer were initialized with random values and its learning rates were raised compared with the rest of the CNN model. In fine-tuning, learning rates are reduced for original layers and boosted for new layers. This changes the CNN model by only a small amount, as the weights were optimized using a large dataset and only need to be modified slightly. However, the new layers change their weights quickly and therefore learn much faster. We also used dropout layers with a 0.5 probability after the first two fully connected layers. The dropout layer generalizes the network and prevents overfitting [39]. At the end of the last fully connected layer, a softmax layer was used to produce the classification outputs. The images in the training and testing sets were resized to $227 \times 227 \times 3$ and $224 \times 224 \times 3$ before feeding them into the AlexNet and VGG-16 networks, respectively. Finally, the models were trained using the training datasets.

We experimented with different hyper-parameter settings to make the CNN models generalize well. For both CNN models (AlexNet and VGG-16), the learning rates of the original and new layers were adjusted to 0.0001 and 0.002, respectively. The weights of the last fully connected layer were initialized using a Gaussian distribution with zero mean and 0.01 standard deviation. The weights of the remaining layers were initialized using the pre-trained models. A stochastic gradient descent optimizer with a momentum of 0.9 was used to train the CNN models. A weight decay (regularization) of 0.0001 was added to the cross-entropy loss function to help reduce overfitting.

TABLE 1. The number of images taken in the six imaging sessions for all the date types in the dataset.

Session no.	Session date	Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	Total per session
1	29-6-2016	219	190	152	124	188	**873**
2	19-7-2016	193	161	183	245	262	**1044**
3	2-8-2016	330	256	298	156	351	**1391**
4	10-8-2016 13-8-2016	719 -	425 -	445 -	401 -	530 453	**2973**
5	3-9-2016	283	263	217	286	232	**1281**
6	21-9-2016	67	90	-	212	141	**510**
Total per date type		**1811**	**1385**	**1295**	**1424**	**2157**	**8072**

III. DATASET

To build a robust vision system, a rich image dataset was created with 8072 images of more than 350 date bunches that belong to 29 date palms in an orchard environment. It included five date types: Naboot Saif, Khalas, Barhi, Meneifi, and Sullaj. Images of the five date types were captured at six imaging sessions, as illustrated in TABLE 1. These sessions covered all date fruit maturity stages (immature, Khalal, Rutab, and Tamar) as well as intermediate phases (i.e. transition between the maturity stages). Therefore, the dataset had a large degree of variation that represents the challenges in a natural environment and date fruit orchards. These variations in images included different angles and scales, different daylight conditions (e.g. poorly illuminated images), and date bunches covered by bags. Sample images are shown in FIGURE 4.

Date fruit classification using this dataset is challenging. In the case of type classification, some date types can easily be distinguished based on their visual appearance, whereas it is difficult, for even a specialist, to distinguish some types, as shown in FIGURE 6. In addition, there are large differences between date bunches of the same type, as they vary in terms of maturity level, illumination, bagging state, scale, and angle (FIGURE 4, **5**). Maturity classification of date fruits is also considered to be challenging because dates grow in large clusters (bunches) and can be harvested in different maturity stages that overlap because the dates in one bunch do not mature at the same time. This fact makes them hard to classify or even label by an expert, as shown in FIGURE 7. Furthermore, many external effects make classification based on maturity difficult. This includes images of date bunches covered by bags and images with poor illumination. All these variations were included in the dataset to help build a

FIGURE 4. Sample images of the dataset showing large variations in scales, angles, and illumination with some date bunches covered by bags.

FIGURE 5. Maturity levels of the Sullaj date at the six imaging sessions starting from the immature stage to the tamar stage.

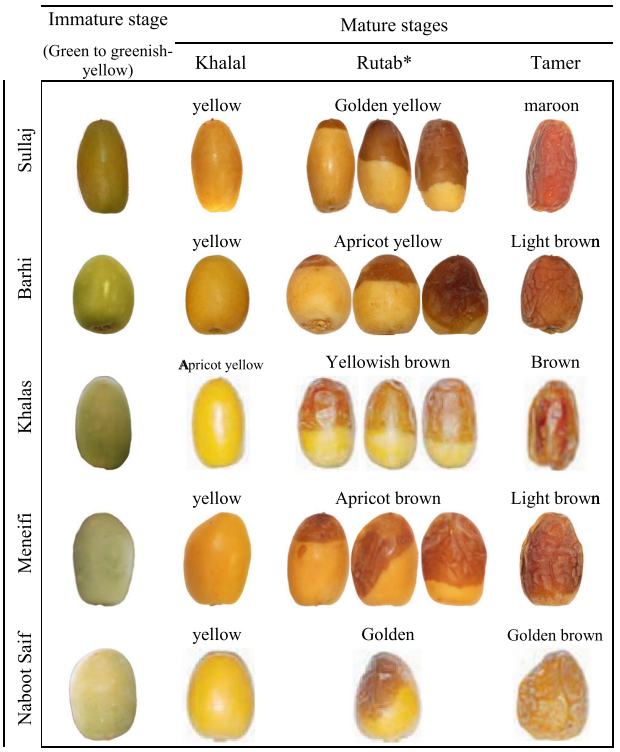
FIGURE 6. Sample images of three date types at the Khalal stage that are difficult to distinguish.

FIGURE 7. Sample images of date bunches that have individual dates at different maturity levels, which makes them hard to label or classify.

robust vision system that provides accurate results in natural environments.

A. DATASET LABELING

Each image in the dataset had two labels: one for type and one for maturity phase. The dataset was labeled into five type classes and seven maturity classes (with three inter-



** The Rutab stage starts with the appearance of the color at the date's tip and then spreads gradually to the whole date.*

FIGURE 8. Sample images of individual dates in the four maturity stages of the five date types. The description of dates colors at each stage is according to [40].

mediate classes). **TABLE 2** shows the distribution of the dataset images between these classes. Type-based labeling was performed during dataset collection by an expert marking the selected date trees in the orchard. For maturity-based labeling, images were labeled according to the decision to harvest date bunches and maturity index of individual dates. The maturity index typically indicates when an individual fruit is ready to harvest. However, for date fruits, several aspects should be taken into consideration when making the harvesting decision. First, dates grow as bunches and the individual dates in one bunch do not mature uniformly. Second, dates are harvested by different methods: either by cutting the whole bunch (when most of the dates are ripe) or by selecting and picking individual ripe dates. Third, dates are harvested in different maturity stages (Khalal, Rutab, and Tamar). Based on these factors, we categorized the date bunches into seven classes (phases): Immature-1, Immature-2, pre-Khalal, Khalal, Khalal-with-Rutab, pre-Tamar, and Tamar. These maturity classes are based on the color and texture of dates, as shown in FIGURE 8, as well as the harvesting decisions and methods described by experts and farmers, as demonstrated in TABLE 3. Hence, in this paper, maturity stages refer to the maturity status of date fruits (immature, Khalal, Rutab, and Tamar), whereas maturity phases refer to the seven maturity classes.

Since individual dates in a bunch do not mature uniformly, this lead to overlaps between the seven maturity classes,

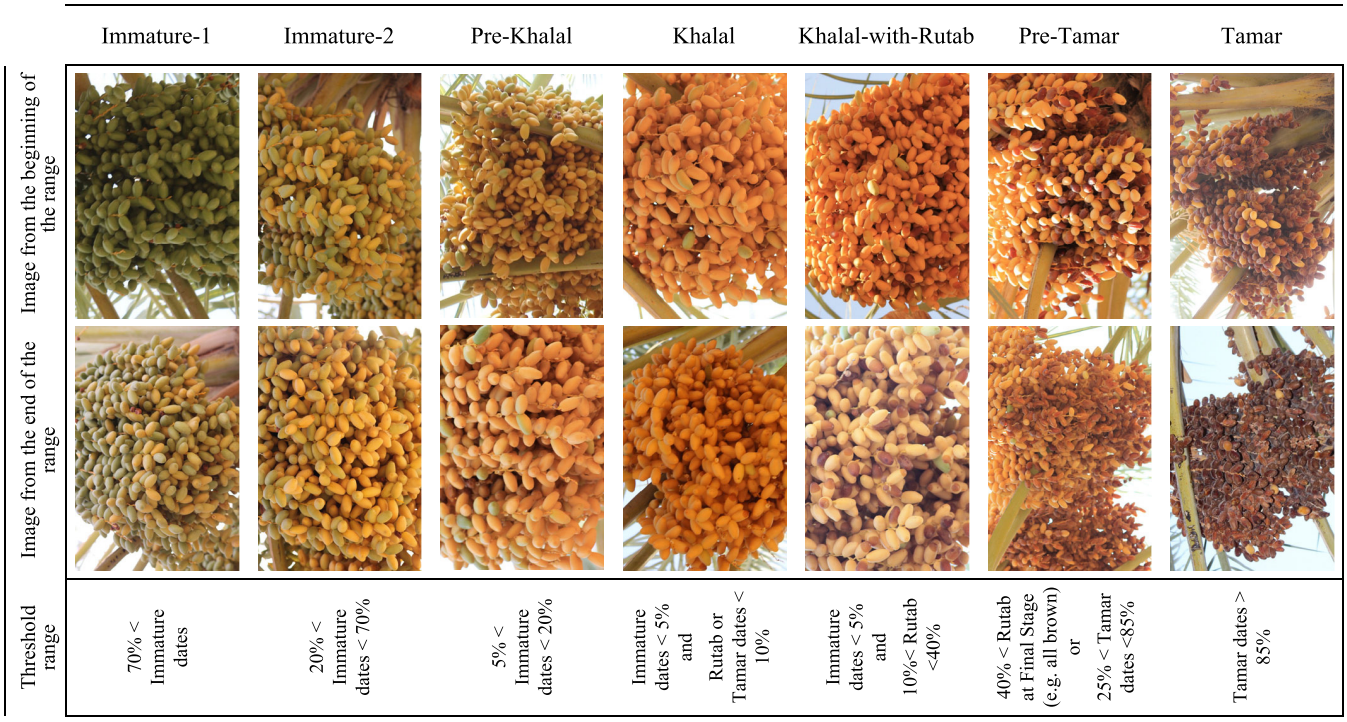


FIGURE 9. Sample images of Sullaj dates labeled into the seven maturity classes based on the threshold values described in TABLE 3.

TABLE 2. Distribution of the dataset images between the seven maturity classes and five type classes.

Maturity class		Type class: Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	
	1 **Immature-1**	412	232	332	332	261	**1569**
	2 **Immature-2**	261	113	135	45	184	**738**
	intermediate-class-1	*300*	*24*	*176*	*190*	*22*	*712*
	3 **Pre-Khalal**	280	156	92	16	121	**665**
	4 **Khalal**	187	90	83	-	299	**659**
	intermediate-class-2	*49*	-	*223*	*146*	*210*	*628*
	5 **Khalal-with-Rutab**	266	416	247	197	534	**1660**
	intermediate-class-3	*1*	-	-	*5*	*62*	*68*
	6 **Pre-Tamar**	42	263	-	281	109	**695**
	7 **Tamar**	6	90	-	212	353	**661**
	no class	7	1	7	-	2	17
		1811	**1385**	**1295**	**1424**	**2157**	**8072**

Immature dates = **1%**
Khalal dates = **59%**
Rutab dates = **39%**
Tamar dates = **1%**

Since immature dates < 5%, and Rutab dates <40%, this image is labeled in the **Khalal-with-Rutab** class

Immature dates = **3%**
Khalal dates = **95%**
Rutab dates = **2%**
Tamar dates = **0%**

Since immature dates < 5%, and Rutab dates <10%, this image is labeled in the **Khalal** class

FIGURE 10. Demonstration of the labeling process into the maturity classes. The process depended on the visual estimation of the number of individual dates in date bunches that belonged to the four maturity stages.

making it difficult to categorize all the images in the dataset into these seven classes. In some cases, when date images have a high degree of confusion (e.g. an equal chance of being classified into two classes), we labeled them as an intermediate class between their two confused adjacent classes. Therefore, there are three intermediate classes between the seven main classes, as shown in TABLE 2. FIGURE 9 shows sample images of date bunches labeled based on the seven maturity classes. Intermediate classes were not used to train the system; however, during the testing phase, the images of these classes were classified by the system into one of the confused adjacent classes.

The process of labeling date images between the maturity classes was performed by an expert. We set an approximate thresholding for each class, as shown in FIGURE 9 and described in TABLE 3, to make this process more accurate. The threshold values depended on the number of individual dates in bunches that belong to the four maturity stages, as shown in **FIGURE 10**. The threshold ranges were approximations based on information gathered from experts and farmers, with no distinct boundaries between the adjacent classes.

B. DIVIDING THE DATASET INTO TRAINING AND TESTING SETS

Since the date dataset has several types in different maturity stages and the number of images per type and maturity stage differ, dividing the dataset randomly into training and testing

TABLE 3. Description of the proposed maturity classes (phases).

Maturity phase/class	Harvesting decision	Description by experts	Approximate threshold according to experts	More information about the maturity phase
Immature-1	**No harvest action**	Most dates are in the immature stage	70% < Immature dates	
Immature-2	**No harvest action,** but the farmers may pick individual Khalal dates	A large amount of immature dates	20% < Immature dates < 70%	Sometimes farmers start to collect individual mature Khalal dates in this phase (selective harvesting)
Pre-Khalal	**(Transition phase)** Usually no harvest action, but the farmer may harvest date bunches as Khalal	Most dates are in the Khalal stage, but some are in the immature stage	5% < Immature dates < 20%	Usually no harvest action in this phase, but sometimes farmers harvest the whole palm even if some bunches have immature dates, which are then removed after harvesting
Khalal	Date bunches are ready to be **harvested as Khalal**	Most dates are in the Khalal stage. A small amount of immature or Rutab dates is acceptable	Immature dates < 5% and Rutab or Tamar dates < 10%	Usually Khalal dates are harvested by cutting the whole bunch (bunch-based harvesting)
Khalal-with-Rutab	Dates are ready to be **harvested as Rutab** and date bunches can be harvested as Khalal	A sufficient amount of dates have reached the Rutab stage. A small amount of immature or Tamar dates is acceptable. A large amount of Khalal dates is acceptable	Immature dates < 5% and 10%< Rutab < 40%	Usually Rutab dates are harvested individually (selective harvesting) by picking the ripe dates over several days. Alternatively, the whole bunch can be harvested and Rutab are separated from Khalal and each sold alone
Pre-Tamar	**(Transition phase)** Usually no harvest action, but farmers may pick individual Rutab or Tamar dates	A large amount of dates have reached the final Rutab stage (e.g. all brown) or Tamar stage	40% < Rutab at Final Stage (e.g. all brown) or 25% < Tamar dates < 85%	When date bunches reach this phase, usually they are left until the Tamar phase, i.e. when the majority of dates in the bunch are in the Tamar stage. However, sometimes farmers collect Rutab or Tamar dates individually (selective harvesting)
Tamar	Date bunches are ready to be **harvested as Tamar**	Most dates are in the Tamar stage. A small amount of Khalal or Rutab dates is acceptable	Tamar dates > 85%	In this phase, most dates are in the Tamar stage and they are usually harvested by cutting the whole bunch (bunch-based harvesting)

sets affects classification accuracy due to bias. Therefore, we divided the dataset into training and testing sets based on rules to ensure the robustness of the classification models and sufficient distributed data to validate the system correctly:

1. The number of training images for all classes should be equal. This gives the classification model an equal chance of learning each class fairly.
2. At least 30% of the images should be included in the testing set for each date type as well as for each maturity stage or imaging session. This will help validate the models by testing them using enough images distributed over all date types, maturity stages, and imaging sessions.
3. The selected training images (after image equalization; rule 1) should be distributed equally between all imaging sessions, for the type classification model, and between all date types, for the maturity classification model. This will help build robust classification models independent of the maturity stage or date type.
4. In cases of conflict, priority is given to the rules in sequence from rule 1 to rule 3.

TABLE 4 shows the number of training and testing images after applying these rules. The training dataset for date type classification consisted of 4530 images (906 per class). As Meneifi had the lowest number of images, we took 30% for testing (389 images) and 70% for training (906 images). This number of training images was fixed for all the other types, and the rest of the images were used for testing, as shown in TABLE 4-**a**. Hence, the testing set amounted to 44% of the overall dataset (3542 images: Barhi 905, Khalas 479, Meneifi 389, Naboot Saif 518, and Sullaj 1251).

The same distribution rules applied to date maturity classification, which had 3227 training images, 461 per class (40% of overall dataset) distributed among all date types, as shown in TABLE 4-**b**. The remaining images were used as the testing dataset.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

We performed many experiments for the three classification tasks. In all the experiments, the CNN models were trained by one GPU, Nvidia GeForce GTX 1060 6 GB, with an Intel Xeon E5-2600 CPU and 28 GB RAM, using Matlab2018b. The batch sizes were set to 128 and 32 training images for the AlexNet and VGG-16 models, respectively. The batch sizes were set depending on the hardware capabilities (GPU) and depth of CNN models used (number of trainable parameters). The improvement in the CNN models was tracked during the training phase by testing them on the testing datasets every 50 iterations. Using the testing datasets, the proposed models achieved final accuracies of 99.01%, 97.25%, and 98.59% for date type classification, date maturity classification, and harvesting decision, respectively.

The performance of the classification models was evaluated using speed and accuracy measures. The speed of the model was measured by the classification rate, number of classified frames (images) per second (fps), and average classification time (i.e. the average time spent by the

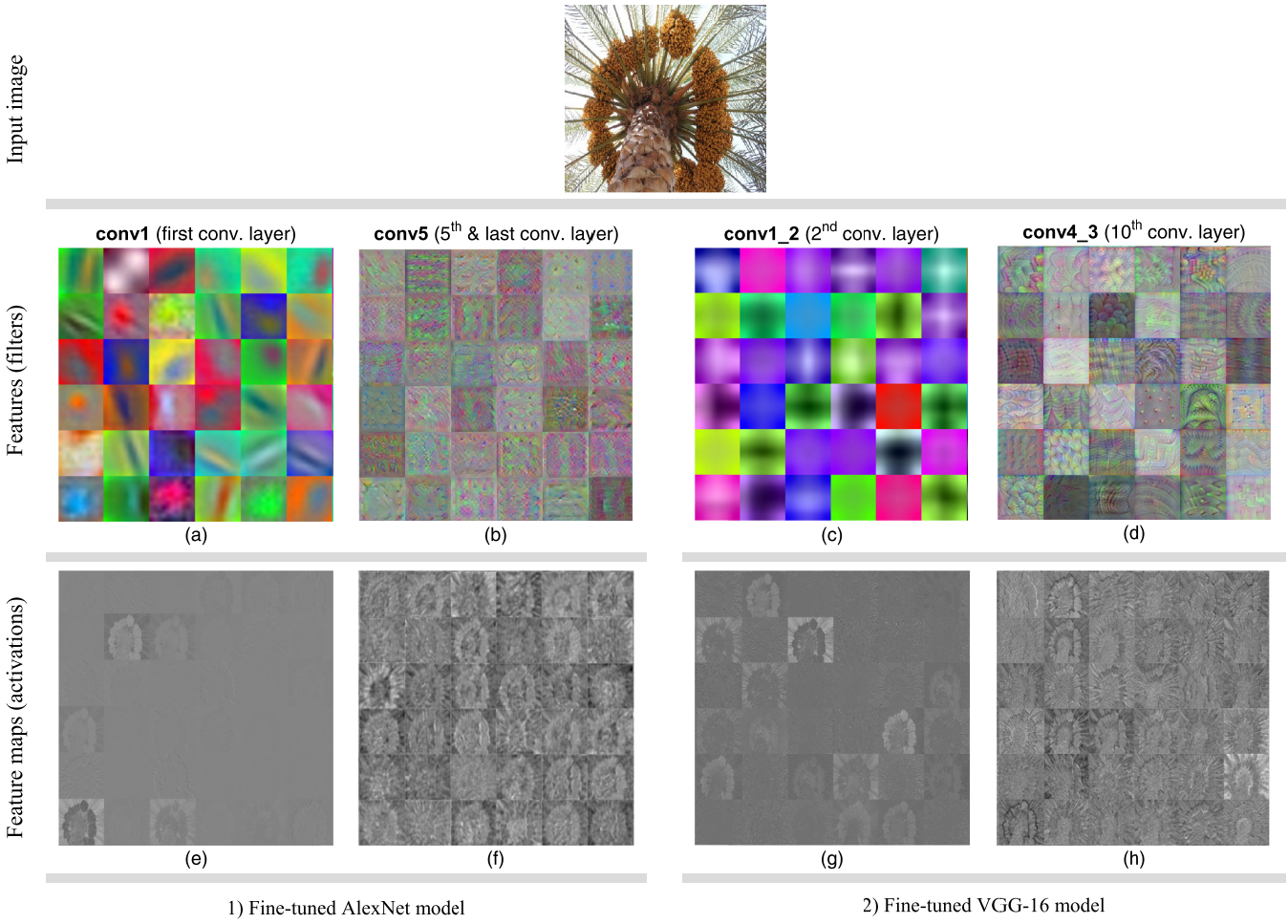


FIGURE 11. Visualization of the type classification models based on AlexNet (1) and VGG-16 (2). The figure shows the first 36 features (filters) of the first and last convolutional layers at the top as well as the activations of these features at the bottom.

model classifying one frame (seconds/frame)). The accuracy of the model was evaluated using the positive predictive value (PPV) or precision, true positive rate (TPR) or recall, f score, and accuracy (ACC). These measures were calculated by Equations (1), (2), (4), and (4), respectively.

For class x, if TP_X is the true positive (i.e. the number of images correctly predicted as belonging to class x), the PPV is the number of true positives divided by the total number of images predicted as belonging to class x. The TPR is defined as the number of true positives divided by the actual number of images in class x. The f-score is used to combine PPV and TPR into one measure using the harmonic mean. The overall accuracy in Eq. (4) was calculated using balanced accuracy, which normalizes the true positive for each class by the number of images in the class and divides their sum by the number of classes. Balanced accuracy ensures that all classes contribute equally to the calculation of overall accuracy even if the number of samples in the classes is unequal. In our experiments, we demonstrated the confusion matrices of the classification models using the same format as in TABLE 5.

$$\mathrm{PPV_X} = \frac{TP_X}{\mathrm{TotalPredicted_X}} \tag{1}$$

$$\mathrm{TPR_X} = \frac{TP_X}{\mathrm{TotalActual_X}} \tag{2}$$

$$f\,\mathrm{score_X} = \frac{1}{{}^{\alpha}/_{TPR_X} + {}^{\alpha}/_{PPV_X}} \tag{3}$$

$$\begin{aligned} &where : \propto = 0.5\ gives\ equal\ weight\ to\ TPR\ and\ PPV \\ &\mathrm{ACC} = \frac{\sum_{i=1}^{n} TP_i/I_i}{n} \quad where : n = no.\ of\ classes, \\ &\qquad\qquad I_i = no.\ of\ images\ in\ classe\ i \end{aligned} \tag{4}$$

FIGURE 11 presents some of the visualization results for the fine-tuned AlexNet and VGG-16 CNN models. The first row in the figure (a to d) visualizes the first 36 features (filters) of the first and last convolutional layers for AlexNet and VGG-16, respectively. The second row visualizes their 36 activations. The filters in the first convolutional layers (a and c) mostly contain colors and edges, which indicates that they are color filters and edge detectors. In the last layers (b and d), the filters represent more detailed and compacted features. We can also see that the filters in all the layers are well formed, smooth, and have no noisy patterns, indicating a well-converged network. Noisy patterns usually indicate

TABLE 4. Dataset images in the training and testing sets. (a) The number of training and testing images for the date type classification task showing their distribution among the six imaging sessions. (b) The number of training and testing images for the date maturity classification task showing their distribution among the five date types.

(a) Date type classes

			Date type class (Training images : Testing images)					
			Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	
Imaging session	1	29-6-2016	153 : 66	133 : 57	106 : 46	87 : 37	132 : 56	
	2	19-7-2016	135 : 58	113 : 48	128 : 55	172 : 73	171: 91	
	3	2-8-2016	190 : 140	179 : 77	209 : 89	109 :47	171:180	
	4	10-8-2016	191 : 528	234 : 191	311: 134	195 : 206	171:812	
	5	3-9-2016	190 : 93	184 : 79	152 : 65	195 : 91	162 : 70	
	6	21-9-2016	47 : 20	63 : 27	-	148 : 64	99 : 42	
			906 : 905	906 : 479	906 : 389	906:518	906:1251	**Total 4530 : 3542**

(b) Date maturity classes

			Date type class (Training images : Testing images)					**Total 3227 : 3420**
			Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	
Date maturity class	1	Immature-1	93 : 319	92 : 140	92 : 240	92 : 240	92 : 169	461 : 1108
	2	Immature-2	128 : 133	79 : 34	95 : 40	32 : 13	127 : 57	461 : 277
	3	Pre-Khalal	192 : 88	109 : 47	64 : 28	11 : 5	85 : 36	461 : 204
	4	Khalal	131 : 56	63 : 27	58 : 25	-	209 : 90	461 : 198
	5	Khalal-with-Rutab	92 : 174	92 : 324	92 : 155	92 : 105	93 : 441	461 : 1199
	6	Pre-Tamar	29 : 13	178 : 85	-	178 : 103	76 : 33	461 : 234
	7	Tamar	4 : 2	63 : 27	-	148 : 64	246 : 107	461 : 200

insufficient training time or very low regularization, which may lead to overfitting. The corresponding activations (feature maps of these 36 features) are shown at the bottom (e, f, g, and h). In these feature maps, the light (white) pixels in some location indicate strong positive activations at that position, whereas the dark (black) pixels indicate strong negative activations. Mostly gray feature maps indicate weak activation on the input image. Since we employ ReLUs, the initial activations appear relatively dense; however, as the training progresses, they look more localized and sparser.

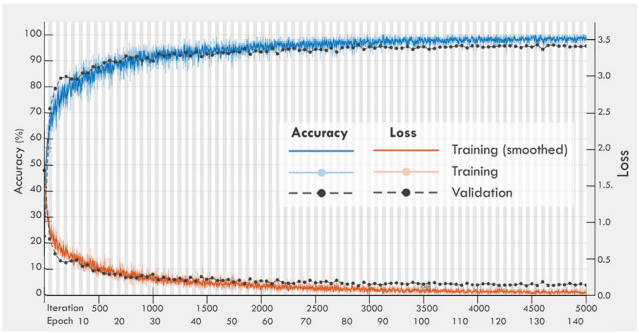
A. DATE TYPE CLASSIFICATION MODEL

In both the pre-trained CNN networks, the last fully connected layer was set to five neurons, equal to the number of date type classes. The AlexNet and VGG-16 CNN networks were trained using 4530 training images (TABLE 4-a) for 7000 iterations. The VGG-16 model needs more iterations than AlexNet to pass all the training examples in the dataset because it has more learnable parameters. In 7000 iterations, VGG-16 and AlexNet adjusted their weights using all the training images during 50 and 200 epochs, respectively.

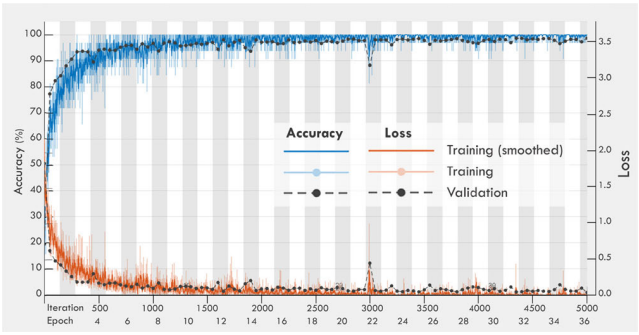
After fine-tuning, the CNN models were tested on the testing dataset containing 3542 images (**TABLE 4-a**) and they achieved accuracies of 96.51% and 99.01% for the AlexNet and VGG-16 models, respectively. The VGG-16 model achieved better accuracy than AlexNet due to its deeper network. The multiple stacked smaller filters in VGG-16 are better than the larger ones in AlexNet because a higher number of non-linear layers increases the network depth. The deep architecture of VGG-16 enables it to learn more complex features and therefore it is more accurate. The training and validation accuracies and losses during the training phase are shown in **FIGURE** 12. The losses of the pre-trained models decrease slowly. After around 4000 iterations, both the pre-trained models reach almost their highest accuracies and stabilize in further iterations with little improvement. The performance of the fine-tuned CNN models on the testing dataset is illustrated by the confusion matrices in **TABLE 6** and summarized in **TABLE 7**.

B. DATE MATURITY CLASSIFICATION MODEL

The date dataset was labeled into seven maturity classes according to the maturity state of the date fruit bunches, as illustrated in the dataset labeling subsection. These maturity classes represent the four maturity stages of date fruits (immature, Khalal, Rutab, and Tamar). The objective of the maturity classification model was to classify the images of date fruit bunches into the four maturity stages using the proposed seven classes. We conducted



(a) AlexNet



(b) VGG-16

FIGURE 12. The behavior of the training and validation accuracies and losses of the date type classification model during the training phase.

TABLE 5. The confusion matrix for the multiclass classification problem with classes $C_1, C_2, \ldots, C_n$. TP, TPR, and PPV refer to the true positive, true positive rate, and positive predictive value, respectively.

Actual class \ Predicted class	C_1	C_2	....	....	C_n	
C_1	TP_1					TPR_1
C_2		TP_2				TPR_2
....			....			....
....				....		....
C_n					TP_n	TPR_n
	PPV_1	PPV_2	....	....	PPV_n	ACC

many experiments to determine the best selection of maturity classes and their training examples to train the CNN networks. Here, we present three experiments for training the CNN models using three selections of the maturity classes.

1 - In the first experiment, the CNN models were trained on the seven maturity classes using 3227 training images (**TABLE 4 -b**) for 5000 iterations. The CNN models achieved accuracies of 90.1% and 92.3% on the testing dataset containing 3420 images for the AlexNet and VGG-16 models, respectively. **FIGURE 14** shows that both the pre-trained models reached almost their highest accuracies after around

TABLE 6. The confusion matrices (as illustrated in TABLE 5) of the type classification model on the testing dataset.

(a) Predicted class based on AlexNet

Actual class	Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	
Barhi	881	2	6	11	5	97.3
Khalas	2	473	2	1	1	98.7
Meneifi	14	1	367	3	4	94.3
Naboot Saif	9	1	4	500	4	96.5
Sullaj	19	2	24	10	1196	95.6
	95.2	98.7	91.1	95.2	98.8	**96.5**

(b) Predicted class based on VGG-16

Actual class	Barhi	Khalas	Meneifi	Naboot Saif	Sullaj	
Barhi	896	1	4	2	2	99.0
Khalas	0	478	1	0	0	99.8
Meneifi	2	5	381	1	0	97.9
Naboot Saif	5	0	0	513	0	99.0
Sullaj	3	0	6	0	1242	99.3
	98.9	98.8	97.2	99.4	99.8	**99.0**

TABLE 7. Results of the proposed type classification models on the testing dataset.

Method	Accuracy (%)	PPV (%)	TPR (%)	$fscore$ (%)	Classification rate (fps)	
					GPU*	CPU*
Transfer learning based on VGG-16	**99.01**	**98.82**	**99.01**	**98.92**	48.33	1.35
Transfer learning based on AlexNet	96.51	95.83	96.51	96.17	**83.96**	**17.65**

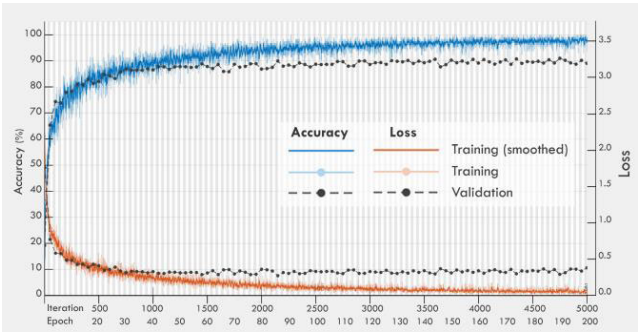
** GPU: Nvidia GeForce GTX 1060 6 GB, CPU: Intel Xeon E5-2600 with 28 GB RAM.*

3000 iterations and did not show significant improvement in further iterations.

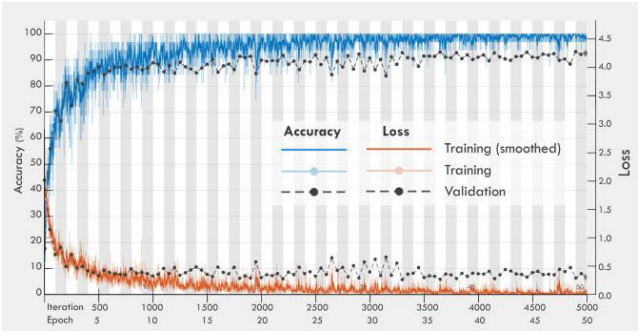
The achieved accuracies are relatively low because adjacent classes significantly overlap, which is clearly noticeable in FIGURE 9. The confusion matrices in TABLE 8 show that in both CNN models, most of the misclassified images were predicted between the prior or subsequent classes of the actual class. For example, as shown in TABLE 8**-b**, 13 of 14 Khalal sample images were predicted either as pre-Khalal (prior class) or Khalal-with-Rutab (subsequent class). For the images misclassified as Tamar (last columns in TABLE 8), most had poor illumination, as shown in **FIGURE 13**. This can be justified by the fact that dark images lost most of their color features and thus were closer to the features of the Tamar class that contain dates with dark colors (e.g. dark brown).

Due to the small inter-class variation in the adjacent classes, it is difficult for the CNN models to learn distinct features for these classes. Therefore, to increase the accuracy of the maturity classification model, we need to train the CNN

FIGURE 13. Samples of misclassified images using the maturity classification CNN models on the seven maturity classes.



(a) AlexNet



(b) VGG-16

FIGURE 14. The behavior of the training and validation accuracies and losses of the date maturity classification model based on the seven maturity classes.

TABLE 8. The confusion matrices (as illustrated in TABLE 5) of the maturity classification model on the testing dataset based on seven maturity classes.

(a) Predicted class based on AlexNet

Actual class	Immature-1	Immature-2	Pre-Khalal	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature-1	1063	34	0	1	1	0	9	95.9
Immature-2	5	266	6	0	0	0	0	96.0
Pre-Khalal	0	7	165	12	17	0	3	80.9
Khalal	0	3	12	163	15	4	1	82.3
Khalal-with-Rutab	0	5	41	31	956	149	17	79.7
Pre-Tamar	0	0	0	0	1	229	4	97.9
Tamar	1	0	0	0	0	3	196	98.0
	99.4	84.4	73.7	78.7	96.6	59.5	85.2	**90.11**

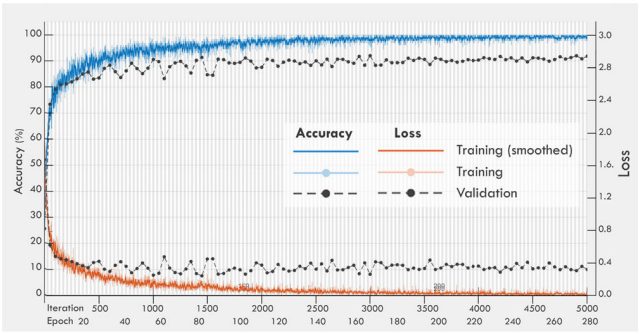
(b) Predicted class based on VGG-16

Actual class	Immature-1	Immature-2	Pre-Khalal	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature-1	1086	13	0	2	1	0	6	98.0
Immature-2	11	251	13	1	1	0	0	90.6
Pre-Khalal	0	1	165	27	9	0	2	80.9
Khalal	0	0	8	184	5	0	1	92.9
Khalal-with-Rutab	0	0	18	46	1056	74	5	88.1
Pre-Tamar	0	0	0	1	0	226	7	96.6
Tamar	0	0	0	0	0	2	198	99.0
	99.0	94.7	80.9	70.5	98.5	74.8	90.4	**92.3**

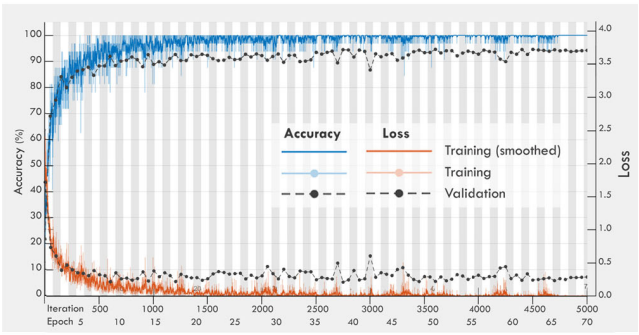
models with fewer maturity classes that have more distinct features and directly represent the required maturity stages of date fruits. This will help the classification model learn the clear and strong features related to date maturity stages. Then, in the test process, it will automatically classify other confused images to the appropriate nearest classes. This was done in the next experiment.

2 - In the second experiment, we excluded the immature-2 class and merged the pre-Khalal and Khalal classes. For the harvesting objective, date bunches in the pre-Khalal and Khalal phases are considered to be in the Khalal stage. We only separated these two classes in the dataset labeling section to allow the farmer to give date bunches in the Khalal phase priority to be harvested before those in the pre-Khalal phase.

The CNN models were thus trained to classify five maturity classes using 2305 training images for 5000 iterations. We achieved accuracies of 93.36% and 95.8% on the testing dataset containing 3604 images for the AlexNet and VGG-16 models, respectively. The behavior of the training and validation accuracies and losses during fine-tuning are shown in **FIGURE 15**. The confusion matrices in **TABLE 9** show that most of the misclassified images (270 of 292 and 176 of 192 in **TABLE 9** a and b, respectively) belong to the Khalal or Khalal-with-Rutab classes due to the high similarity between these and their adjacent classes. The Khalal and Khalal-with-Rutab classes (phases) represent the harvesting indicator of date bunches in the Khalal and Rutab stages, respectively. Hence, they cannot be merged or excluded. Instead, it is possible to increase the number of training examples of these two classes compared with the other classes, which will increase the ability of the CNN to learn their fine features and therefore improve the overall accuracy of the maturity classification model. This was done in the final experiment.

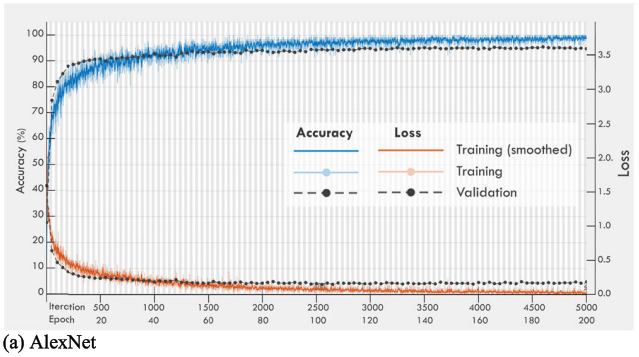


(a) AlexNet

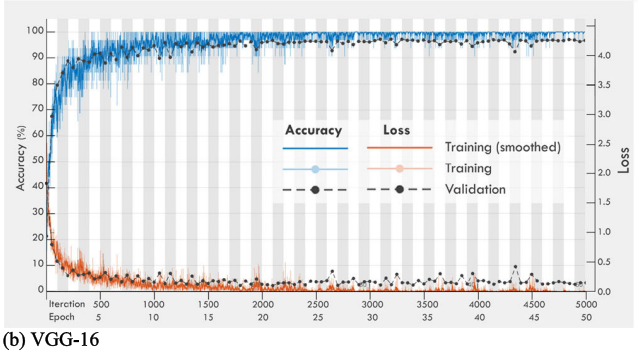


(b) VGG-16

FIGURE 15. The behavior of the training and validation accuracies and losses of the date maturity classification model based on the five classes.



(a) AlexNet



(b) VGG-16

FIGURE 16. The behavior of the training and validation accuracies and losses of the date maturity classification model based on the five maturity classes with doubling the number of training examples of the Khalal and Khalal-with-Rutab classes.

3 - The third experiment used the same classes as in the second experiment but with twice the number of training examples for the Khalal and Khalal-with-Rutab classes and

TABLE 9. The confusion matrices (as illustrated in TABLE 5) of the maturity classification model on the testing dataset based on the five maturity classes.

(a) Predicted class based on AlexNet

Actual class	Immature	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature	1098	2	3	0	5	99.1
Khalal	9	736	111	6	1	85.3
Khalal-with-Rutab	0	66	1056	62	15	88.1
Pre-Tamar	0	0	2	229	3	97.9
Tamar	1	0	0	6	193	96.5
	99.1	91.5	90.1	75.6	88.9	**93.4**

(b) Predicted class based on VGG-16

Actual class	Immature	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature	1101	2	0	1	4	99.4
Khalal	4	812	47	0	0	94.1
Khalal-with-Rutab	0	76	1074	46	3	89.6
Pre-Tamar	0	0	1	229	4	97.9
Tamar	0	0	0	4	196	98.0
	99.6	91.2	95.7	81.8	94.7	**95.8**

TABLE 10. The confusion matrices (as illustrated in TABLE 5) of the maturity classification model on the testing dataset based on the five maturity classes with doubling the number of training examples of the Khalal and Khalal-with-Rutab classes.

(a) Predicted class based on AlexNet

Actual class	Immature	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature	1095	4	3	0	6	98.8
Khalal	1	360	40	0	1	89.6
Khalal-with-Rutab	0	34	685	19	0	92.8
Pre-Tamar	1	0	1	224	8	95.7
Tamar	0	0	0	4	196	98.0
	99.8	90.5	94.0	90.7	92.9	**95.0**

(b) Predicted class based on VGG-16

Actual class	Immature	Khalal	Khalal-with-Rutab	Pre-Tamar	Tamar	
Immature	1101	5	0	0	2	99.4
Khalal	0	390	12	0	0	97.0
Khalal-with-Rutab	0	43	690	5	0	93.5
Pre-Tamar	0	0	1	229	4	97.9
Tamar	0	0	0	3	197	98.5
	100.0	89.0	98.2	96.6	97.0	**97.3**

fewer testing images. The CNN models were trained for 5000 iterations on 3277 training images (922 for Khalal and Khalal-with-Rutab and 461 for the other classes). The test accuracies rose to 94.98% and 97.25% for the AlexNet and VGG-16 models, respectively. The CNN models reached nearly their highest test accuracies after around 3000 iterations and stabilized in the subsequent iterations, as shown in **FIGURE** 16. The confusion matrices in **TABLE 10** show that

FIGURE 17. Samples of misclassified images using the maturity classification CNN models on the five maturity classes.

TABLE 11. Results of the proposed maturity classification model on the testing dataset using the different settings of the maturity classes.

Experiment number	Setting of maturity classes	Pre-trained model	Accuracy (%)	PPV (%)	TPR (%)	*fscore* (%)	Classification rate (fps)	
							GPU*	CPU*
1	Seven maturity classes with an equal number of training samples	VGG-16	92.3	86.98	92.3	89.56	57.43	1.36
		AlexNet	90.1	82.51	90.11	86.14	83.75	15.66
2	Five maturity classes with an equal number of training samples	VGG-16	95.78	92.61	95.78	94.17	48.93	1.31
		AlexNet	93.36	89.05	93.36	91.16	93.17	16.47
3	Five maturity classes with an unequal number of training samples	VGG-16	**97.25**	**96.17**	**97.25**	**96.71**	48.17	1.34
		AlexNet	94.98	93.56	94.98	94.27	93.68	15.35

* *GPU: Nvidia GeForce GTX 1060 6 GB, CPU: Intel Xeon E5-2600 with 28 GB RAM.*

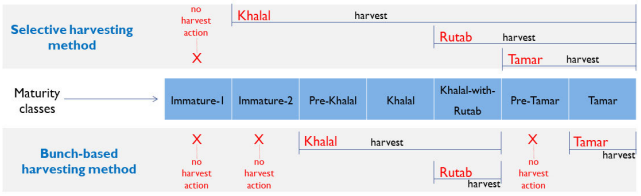


FIGURE 18. The harvesting decision of dates depending on the harvesting methods during the development period.

increasing the training samples of the Khalal and Khalal-with-Rutab classes enhanced the accuracy of the CNN models. However, these two classes still challenge the maturity classification model due to their low inter-class variation and wide intra-class variation. **FIGURE** 17 shows samples of some misclassified images. **TABLE 11** presents the results of the three experiments.

C. HARVESTING DECISION SYSTEM

Dates are harvested and sold as ripe fruit in three stages: Khalal, Rutab, and Tamar. The choice to harvest in one or other stage depends on many factors such as climatic conditions, date fruit type, and market demand [2]. The proposed system determines the decision to harvest date bunches in two steps. First, the user enters the required harvesting stage for each date type, according to the climatic conditions, market demand, and so on. Next, the system automatically recognizes the types of date bunches in the orchard and

Date type	Maturity stage: Khalal	Rutab	Tamar
Barhi	√	√	√
Meneifi		√	√
Sullaj		√	√
Khalas	√	√	√
Naboot Saif		√	√

FIGURE 19. Possible harvesting stages of the five date types in the dataset.

defines their maturity stages, then making the harvesting decision.

Date fruits can be harvested either by picking individual mature dates (selective harvesting) or by cutting the whole bunch when the most dates are mature (bunch-based harvesting). The latter method is used in large orchards for commercial production. Therefore, in this experiment, we investigate the harvesting decision associated with bunch-based harvesting. In bunch-based harvesting, date bunches labeled Pre-Khalal, Khalal, and Khalal-with-Rutab are harvested in the Khalal stage, whereas only date bunches labeled Khalal-with-Rutab and Tamar are harvested as Rutab or Tamar, respectively, as shown in **FIGURE 18**.

According to the date fruit characteristics, Barhi and Khalas are consumed in all maturity stages, whereas Meneifi, Naboot Saif, and Sullaj are consumed in Rutab or Tamar [40], as shown in **FIGURE 19**. In this experiment, we test the harvesting decision model for the five date types in the most frequent harvesting stage selected for each type. We assume that the required harvesting stage of Barhi is Khalal, of Meneifi and Sullaj is Rutab, and of Khalas and Naboot Saif is Tamar.

The fine-tuned VGG-16 models of date type and maturity classification in the previous experiments were used to determine the harvesting decision in this experiment. The harvesting decision system was tested using a dataset containing 2682 images of the five date types. Each date type in the dataset was categorized into two classes (harvest and not harvest), as shown in **FIGURE 20**. The performance of the harvesting decision model on the testing dataset is illustrated by the confusion matrix in **TABLE 12**. This matrix consists of four large cells that report the number of true

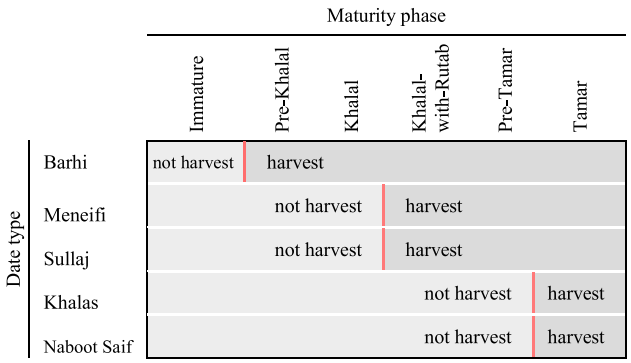


FIGURE 20. Labeling each date type in the testing dataset according to the harvesting decision. Decisions are based on the assumption that the required harvesting stage of Barhi is Khalal, Meneifi and Sullaj is Rutab, and Khalas and Naboot Saif is Tamar.

TABLE 12. The confusion matrix of the harvesting decision model on the testing dataset.

Required harvesting decision		Barhi as Khalal	Meneifi as Rutab	Sullaj as Rutab	Khalas as Tamar	Naboot Saif as Tamar	Barhi before Khalal	Meneifi before Rutab	Sullaj before Rutab	Khalas before Tamar	Naboot Saif before Tamar	
		Predicted harvesting decision by our framework										
		Harvest					Not harvest					
Barhi as Khalal*	Harvest	239	1	0	0	0	0	0	0	0	0	99.6
Meneifi as Rutab*	Harvest	0	73	0	0	0	0	1	0	0	0	98.6
Sullaj as Rutab*	Harvest	1	1	430	0	0	0	0	29	0	0	93.3
Khalas as Tamar*	Harvest	0	0	0	27	0	0	0	0	0	0	100
Naboot Saif as Tamar*	Harvest	0	0	0	0	63	0	0	0	0	1	98.4
Barhi before Khalal	Not harvest	0	0	0	0	0	318	1	0	0	0	99.7
Meneifi before Rutab	Not harvest	0	3	0	0	0	1	287	0	1	0	98.3
Sullaj before Rutab	Not harvest	0	0	2	0	0	0	0	293	0	0	99.3
Khalas before Tamar	Not harvest	0	0	0	1	0	0	1	0	501	0	99.6
Naboot Saif before Tamar	Not harvest	0	0	0	0	2	2	0	0	0	403	99.0
		99.6	93.6	99.5	96.4	96.9	99.1	99.0	91.0	99.8	99.8	**98.6**

* The required harvesting stage for each date type, which is entered manually depending on many factors such as climatic conditions, date fruit characteristics, and market demand.

positives, false positives, false negatives, and true negatives of the decision to harvest date bunch images. Each large cell consists of five small cells related to the five date types. The rows represent the required harvesting decision for each date type and the columns represent the predicted harvesting decision. The overall framework performance is shown in **TABLE 13**. This TABLE reports the speed and accuracy measures of the date type and maturity classification models based on the VGG-16 network and the performance of the harvesting decision system based on the entered decision factors and two classification results.

D. COMPARISON WITH OTHER CLASSIFICATION METHODS

We compared the performance of the proposed classification method with that of the methods used in other studies

TABLE 13. Performance of the proposed classification models (speed and accuracy measures).

Method	Accuracy (%)	PPV (%)	TPR (%)	$fscore$ (%)	Average classification time (msec/image)	
					GPU*	CPU*
Date type classification model	99.01	98.82	99.01	98.92	20.6	740.2
Date maturity classification model	97.25	96.17	97.25	96.71	20.7	745.9
Harvesting decision model	98.59	97.46	98.59	98.02	35.9	1298

* *GPU: Nvidia GeForce GTX 1060 6 GB, CPU: Intel Xeon E5-2600 with 28 GB RAM.*

of date fruit type and maturity classification, as shown in TABLE 14. The author in [10] used support vector machine to classify date types using LBP and WLD local texture descriptors combined with size and shape features. He classified date images of four date types and achieved an accuracy of 98.1%. In [11], the authors used 15 features including color, shape, size, and texture descriptors to classify dates according to their types using neural networks. The study reported an accuracy of 98.6%; however, only 140 images were used to classify and test seven date classes. The authors in [12] classified date images based on their types using a dataset containing 5000 images of 10 date types. They used an RGB color histogram, a gray-level co-occurrence matrix, and four shape features including area and eccentricity, combined with a Gaussian mixture model. They reported an accuracy of 97.5% with a classification time of 0.029 seconds. All these studies used single date images with a uniform background. Hence, previously used methods are unsuitable for real-life applications in natural environments where the variety in the data is enormous (e.g. in date orchards).

Few studies of maturity classification have been performed, and they all have the same limitations discussed above. One such study [16] used a color distribution analysis and back projection to classify one date type into four maturity classes. The authors achieved a classification accuracy of 97.5% on single date images. In another study [17], a taxonomy classification method with RGB color and texture features including contrast, entropy, and homogeneity was used to classify a date type into four maturity classes. The authors reported an overall accuracy of 88.33% with a 0.34-second classification time.

In the most recent approach [14], the researchers proposed a date type classification based on a pre-trained CNN model and reported an accuracy of 99.2%. They used images of single and multiple dates for four date types. However, the studied date types had wide visual inter-class variation and their images were all taken after the production stage. Hence, the classification task was much simpler than the classification in this work.

All previous approaches have used single or multiple date images taken after the harvesting or production stages.

TABLE 14. Performance comparison between the methods.

(a) Date fruit type classification

Method	Dataset type	No. of testing images	Accuracy (%)	Average classification time (msec/image)	
				GPU	CPU
Proposed with VGG-16	Date fruit bunches in an orchard environment	3542	99.01	20.6	740.2
Proposed with AlexNet			96.51	11.9	56.6
[10]	Single dates with uniform background	10-fold validation on 800	98.1	-	
[11]		10-fold validation on 140	98.6	-	
[12]		1000	97.5	29	
[14]	Single and multiple dates after the production	1000	99.2	-	

(b) Date fruit maturity classification

Proposed with VGG-16	Date fruit bunches in an orchard environment	2682	97.25	20.7	745.9
Proposed with AlexNet			94.98	10.7	65.1
[16]	Single dates with a uniform background	353	97.51	-	
[17]		100	88.33	340	

TABLE 15. Performance comparison between the methods using the same dataset adopted in this work.

Method	Accuracy (%)			Average classification time of harvesting decision system (msec/image)	
	Date type classification	Date maturity classification[(1)]	Harvesting decision	GPU[(2)]	CPU[(2)]
Proposed with VGG-16	99.01	97.25	98.59	35.9	1298
Proposed with AlexNet	96.51	94.98	95.51	22.3	115.8
[10]	81.22	80.38	80.45	125	129.2
[11]	58.82	63.03	62.01	37.6	37.9
[14]	95.92	93.40	95.28	17.5	103

[(1)] *Using the same training and testing datasets described in the maturity classification model subsection - the third experiment.*

[(2)] ***GPU**: Nvidia GeForce GTX 1060 6 GB, **CPU**: Intel Xeon E5-2600 with 28 GB RAM.*

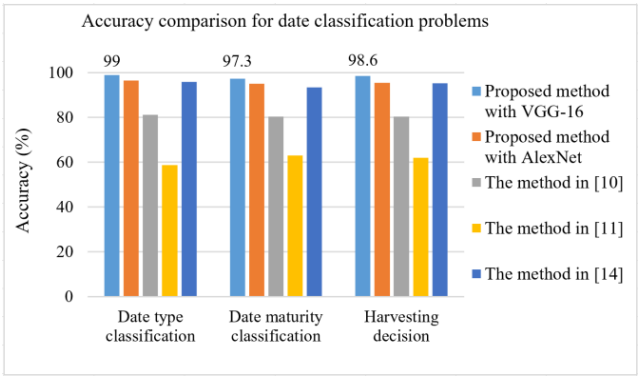


FIGURE 21. Comparison between the accuracy obtained by the proposed approach for date classification problems and the approaches in [10], [11], and [14].

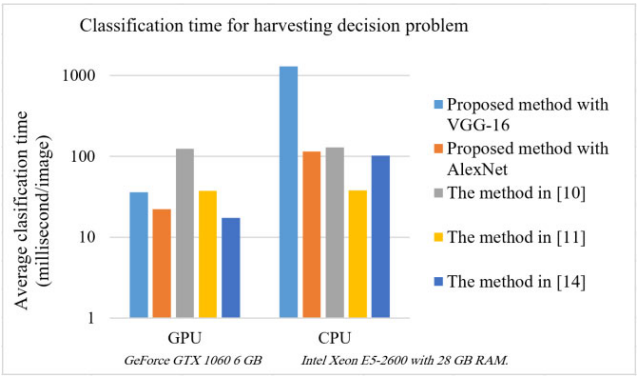


FIGURE 22. Average classification time of the date harvesting decision system based on the proposed method and the methods in [10], [11], and [14], using a machine with a GPU and CPU.

By contrast, this work deals with date bunch images in different pre-maturity and maturity stages with a large degree of variation in an orchard environment, which makes them challenging to classify.

TABLE 15 compares the performance of the proposed approach with that of other recent approaches using the same dataset that we created. The methods described in [10], [11], and [14] were carried out and evaluated using the same

environment and machine used in the previous experiments. To conduct the methods of [10] and [11], color and texture features were used and size and shape features were ignored because they cannot identify the maturity stage of dates or their types as bunches. A k-nearest neighbor classifier was chosen to implement [11] and LBP was chosen as the texture descriptor in [10]. **FIGURE 21** shows the average accuracy obtained by the proposed approach in the three classification problems and by the approaches in [10], [11], and [14]. **FIGURE 22** presents the average classification time of each method using a machine with and without a GPU. The results show that the proposed approach achieved excellent classification accuracies, which outperformed those of the other approaches and had a low classification time using a GPU.

V. CONCLUSION

A real-time machine vision framework for date fruit harvesting robots in an orchard environment was proposed based on deep learning. The framework consisted of three models used to classify date fruit bunches according to their type, maturity, and harvesting decision. Transfer learning with fine-tuning was used in the classification tasks. Two pre-trained CNN models were investigated, namely AlexNet and VGG-16. To build a robust machine vision system, we used a rich image dataset of five date types for all maturity stages. The dataset was designed with a large degree of variation that represents the challenges in natural environments and date fruit orchards. The proposed approach achieved excellent classification accuracies on this challenging dataset with a high classification rate. The results showed that a pre-trained CNN could achieve robust date fruit classification without the pre-processing of images to remove background noise or enhance illumination. The best accuracies were obtained by the fine-tuned VGG-16 model, which achieved 99.01%, 97.25%, and 98.59% accuracies with classification times of 20.6, 20.7, and 35.9 msec for the date fruit type, maturity, and harvesting decision classification models, respectively. As for future work, we will improve the dataset by including testing images captured from different date orchards. We will also investigate more recent CNN models to minimize the usage of memory and lower computational complexity. One more area to investigate is the confusion in the maturity detection of date fruit, including labeling rules, and the interference among maturity stages.

REFERENCES

[1] Food and Agriculture Organization Corporate Statistical. (2016). *FAOSTAT*. Accessed: Oct. 18, 2018. [Online]. Available: http://www.fao.org/faostat/en/#data/QC

[2] Z. Abdelouahhab and E. Arias-Jimenez, *Date Palm Cultivation*, 1st ed. Rome, Italy: FAO, 2002.

[3] A. Al-Janobi and A. Abdulwahed, "Evaluation of field test of harvesting system for picking dates fruits based on robotic arm," in *Proc. Int. Conf. Robot. Associated High-Technol. Equip. Agricult.*, Sep. 2012, pp. 183–188.

[4] K. Kapach, E. Barnea, R. Mairon, Y. Edan, and O. Ben-Shahar"Computer vision for fruit harvesting robots—State of the art and challenges ahead," *Int. J. Comput. Vis. Robot.*, vol. 3, no. 1, pp. 4–34, Apr. 2012.

[5] P. Rajendra, N. Kondo, K. Ninomiya, J. Kamata, M. Kurita, T. Shiigi, S. Hayashi, H. Yoshida, and Y. Kohno, "Machine vision algorithm for robots to harvest strawberries in tabletop culture greenhouses," *Eng. Agricult., Environ. Food*, vol. 2, no. 5, pp. 24–30, Dec. 2008.

[6] S. S. Mehta and T. F. Burks, "Vision-based control of robotic manipulator for citrus harvesting," *Comput. Electron. Agricult.*, vol. 102, pp. 146–158, Mar. 2014.

[7] D. M. Bulanon, T. Kataoka, H. Okamoto, and S. Hata, "Development of a real-time machine vision system for the apple harvesting robot," in *Proc. Annu. Conf.*, Aug. 2004, pp. 595–598.

[8] A. I. Hobani, A. Thottam, and K. Ahmed, "Development of a neural network classifier for date fruit varieties using some physical attributes," in *Proc. Res. Bull. Agric. Res. Center*, 2003, pp. 5–18.

[9] M. Fadel, "Date fruits classification using probabilistic neural networks," *Agric. Eng. Int. CIGR J.*, vol. 9, p. 1989, Dec. 2007.

[10] G. Muhammad, "Date fruits classification using texture descriptors and shape-size features," *Eng. Appl. Artif. Intell.*, vol. 37, pp. 361–367, Jan. 2015.

[11] A. Haidar, H. Dong, and N. Mavridis, "Image-based date fruit classification," in *Proc. 7th Int. Congr. Ultra Modern Telecommun. Control Syst.*, Oct. 2012, pp. 357–363.

[12] A. Oussama and M. L. Kherfi, "A new method for automatic date fruit classification," *Int. J. Comput. Vis. Robot.*, vol. 7, no. 6, pp. 692–711, 2017.

[13] A. Kamal-Eldin and S. Ghnimi, "Classification of date fruit (Phoenix dactylifera, L.) based on chemometric analysis with multivariate approach," *J. Food Meas. Characterization*, vol. 12, no. 2, pp. 1020–1027, Jun. 2018.

[14] M. S. Hossain, G. Muhammad, and S. U. Amin, "Improving consumer satisfaction in smart cities using edge computing and caching: A case study of date fruits classification," *Future Gener. Comput. Syst.*, vol. 88, pp. 333–341, Nov. 2018.

[15] Z. Schmilovitch, A. Hoffman, H. Egozi, R. Ben-Zvi, Z. Bernstein, and V. Alchanatis, "Maturity determination of fresh dates by near infrared spectrometry," *J. Sci. Food Agricult.*, vol. 79, no. 1, pp. 86–90, Jan. 1999.

[16] D. Zhang, D.-J. Lee, B. J. Tippetts, and K. D. Lillywhite, "Date maturity and quality evaluation using color distribution analysis and back projection," *J. Food Eng.*, vol. 131, pp. 161–169, Jun. 2014.

[17] R. Pourdarbani, H. R. Ghassemzadeh, H. Seyedarabi, F. Z. Nahandi, and M. M. Vahed, "Study on an automatic sorting system for date fruits," *J. Saudi Soc. Agricult. Sci.*, vol. 14, no. 1, pp. 83–90, 2015.

[18] A. Halimi, A. Roukhe, B. Abdenabi, and N. El Barbri, "Sorting dates fruit bunches based on their maturity using camera sensor system," *J. Theor. Appl. Inf. Technol.*, vol. 56, no. 3, pp. 324–337, 2013.

[19] K. M. Ismail and K. A. Al-Gaadi, "Development of an electronic sensor for Date sorting based on moisture content," *King Saud Univ.-Agricult. Res. Center*, vol. 26, no. 4, pp. 1923–1932, 2009.

[20] A. Manickavasagan, N. K. Al-Mezeini, and H. N. Al-Shekaili, "RGB color imaging technique for grading of dates," *Scientia Horticulturae*, vol. 175, pp. 87–94, Aug. 2014.

[21] A. AI-Janobi, "Application of co-occurrence matrix method in grading date fruits," in *Proc. ASAE Annu. Int. Meeting*, Orlando, FL, USA, 1998, pp. 3024–3098.

[22] A. A. Al-Janobi, "Date inspection by color machine vision," *J. King Saud Univ.*, vol. 12, no. 1, pp. 69–79, 2000.

[23] D. Wulfsohn, Y. Sarig, and R. V. Algazi, "Defect sorting of dry dates by image analysis," *Can. Agricult. Eng.*, vol. 35, no. 2, pp. 133–139, Apr. 1993.

[24] Y. A. Ohali, "Computer vision based date fruit grading system: Design and implementation," *J. King Saud Univ., Comput. Inf. Sci.*, vol. 23, pp. 29–36, Jan. 2011.

[25] A. Nasiri, A. Taheri-Garavand, and Y.-D. Zhang, "Image-based deep learning automated sorting of date fruit," *Postharvest Biol. Technol.*, vol. 153, pp. 133–141, Jul. 2019.

[26] N. Alavi, "Quality determination of Mozafati dates using Mamdani fuzzy inference system," *J. Saudi Soc. Agricult. Sci.*, vol. 12, no. 2, pp. 137–142, Jun. 2013.

[27] H. Altaheri, M. Alsulaiman, M. Faisal, and G. Muhammed, "Date fruit dataset for automated harvesting and visual yield estimation," *IEEE DataPort*, 2019. doi: 10.21227/x46j-sk98.
Please note that this reference is for a database that has already been published and uploaded in the IEEE DataPort repository.

[28] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–9.
[30] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
[31] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
[32] L. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[33] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, A. Senior, P. Tucker, K. Yang, Q. V. Le, and A. Y. Ng, "Large scale distributed deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1223–1231.
[34] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2014, pp. 647–655.
[35] B. Chu, V. Madhavan, O. Beijbom, J. Hoffman, and T. Darrell, "Best practices for fine-tuning visual classifiers to new domains," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 435–442.
[36] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Sep. 2014, pp. 806–813.
[37] A. Salvador, M. Zeppelzauer, D. Manchón-Vizuete, A. Calafell, and X. Giro-i-Nieto, "Cultural event recognition with visual ConvNets and temporal models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 36–44.
[38] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.
[39] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[40] *The Famous Date Varieties in the Kingdom of Saudi Arabia*, Ministry Agricult., Riyadh, Saudi Arabia, 2011.

HAMDI ALTAHERI (M'19) received the master's degree in computer engineering from King Saud University, in 2019, where he is currently pursuing the Ph.D. degree with the Department of Computer Engineering, College of Computer and Information Sciences. His research interests include computer vision, machine learning, deep learning, and artificial intelligence.

MANSOUR ALSULAIMAN received the Ph.D. degree from Iowa State University, USA, in 1987. Since 1988, he has been with the Computer Engineering Department, King Saud University, Riyadh, Saudi Arabia, where he is currently a Professor with the Department of Computer Engineering. He is also the Director of the Center of Smart Robotics Research, King Saud University. His research interests include automatic speech/speaker recognition, automatic voice pathology assessment systems, computer-aided pronunciation training systems, and robotics. He was an Editor-in-Chief of the *Journal of King Saud University—Computer and Information Sciences* Section.

GHULAM MUHAMMAD (M'10–SM'19) received the Ph.D. degree from the Department of Electronic and Information Engineering, Toyohashi University of Technology, Japan. He is a Professor with the Computer Engineering Department, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh, Saudi Arabia. He has authored or coauthored more than 170 publications and has been refereed in IEEE/ACM/Springer/Elsevier journals, conference papers, and book chapters. He holds a US patent.