Classification of Palm Trees Diseases using Convolution Neural Network

Marwan Abu-zanona[1], Said Elaiwat[2], Shayma'a Younis[3], Nisreen Innab[4], M. M. Kamruzzaman[5]

Department of Computer Sciences, College of Shari'a and Islamic Studies in Al Ahsaa,
Al Imam Mohammad IbnSaud Islamic University, Al Ahsaa, Saudi Arabia[1]
Department of Computer Science, College of Computer and Information Sciences,
Jouf University, Sakakah, Saudi Arabia[2,5]
Department Biomedical Informatics engineering,
Yarmouk University, Irbid, Jordan[3]
Department of Computer Science and Information Systems, College of Applied Sciences,
AlMaarefa University, Riyadh, Saudi Arabia[4]

***Abstract*—The palm tree is considered one of the most durable trees , and it occupies an advanced position as one of the most famous and most important trees that are planted in different regions around the world, which enter into many uses and have a number of benefits. In the recent years , date palms have been exposed to a large number of diseases. These diseases differ in their symptoms and causes, and sometimes overlap, making the diagnosing process with the naked eye difficult, even by an expert in this field. This paper proposes a CNN-model to detect and classify four common diseases threatening palms today, Bacterial leaf blight, Brown spots, Leaf smut, white scale in addition to healthy leaves. The proposed CNN structure includes four convolutional layers for feature extraction followed by a fully connected layer for classification. For performance evaluation, we investigate the performance of the proposed model and compare to other CNN- structures, VGG-16 and MobileNet, using four evaluation metrics: Accuracy, Precision, Recall and F1 Score. Our proposed model achieves 99.10% accuracy rate while VGG-16 and MobileNet achieve 99.35% and 99.56% accuracy rates, respectively. In general, the performance of our model and other models are very close with a minor advantage to MobileNet over others. In contrast, our model characterized by simplicity and shows low computational training time comparing to others.**

Keywords*—*Palm trees diseases; convolutional neural networks; mobileNet; VGG-16

I. Introduction

The date palm is considered one of the most important fruit trees in the Arab and Islamic world, as Arab countries account for 71% of its trees in the world, and 81% of the total global production, while this percentage rises to 99% (103.95 million trees) of the number of trees. The world's date palm, amounting to 105 million, when combining the Arab and Islamic worlds, according to the Food and Agriculture Organization of the United Nations. According to the latest statistics adopted in the world, including the statistic conducted by the Egyptian Embassy in Brazil in 1990 as shown in table I, it was found that the palm tree is one of the main agricultural products produced by the Arab countries and is considered a major element in supporting the macro economy in these countries, so it is very important to pay attention to the quality and quantity of production Palm trees, but unfortunately, the quality, quality and quantity of palm trees are greatly threatened with confinement due to the common palm diseases these days, where in general palm trees are threatened by 4 main types of diseases, which are, Bacterial leaf blight, Brown spots, Leaf smut, white scale, the nature and symptoms of these diseases are different in their form, in the area of their appearance and distribution on palm trees, so it is very important to reveal modern techniques that contribute greatly to discovering them before they cause tremendous pressure on the quality and quantity of palm trees produced.

Symptoms of leaf smut Small irregular brown to black spots occurred on the upper and bottom surfaces of rachis and fronds, ranging in size from 3 to 7 mm [1]. Bacterial Leaf blight symptoms were elongated brown to black patches that grew in size and spread across a considerable region, creating cankers on the midrib [1].

Brown spot disease is characterized by the appearance of non-specific dark spots, and as the infection progresses, the center of the spot turns to a pale color, but the edges remain brown to gray. The spots appear on the leaves, thorns, and the middle vein (the leaves). The size of the spots ranges from one to several centimeters, but their size and color may vary according to the fungus that causes them..Another serious risk is a lethal pest called white scale .White palm Scale is a species of armoured scale insect. This means that they produce a hard outer coating that covers the body, which protects them from pathogens. They're also well protected from topical pesticides. White Scale insects attack palm by sucking the sap through a fine, thin feeding-tubes. Infestations rarely kill plants but can impact vigour [2]. The methods usually used by farmers depend mainly on observing the affected foliage with the naked eye by experts. Unfortunately, this method is not effective because of the distance of the palm tree from the ground and at the same time due to the somewhat similarity between the symptoms of the four palm diseases mentioned previously. In addition to that, the manual examination of palm leaves is time-consuming, especially in the case of large farms. In this paper, a CNN-based model is proposed to detect four of the most frequent palm illnesses is suggested. These illnesses are, Bacterial leaf blight, brown spots, Leaf smut, white scale. The proposed model model characterized by simplicity(easy

TABLE I. Statistics Done by the Embassy of Egypt Economic and Commercial Office in Brazil,: March 18, 2019 [1]

Top 10 Largest Date Producers in the World		
Rank	**Country**	**Production (1000 Metric Tons)**
1st	Egypt	1,373,57
2nd	Saudi Arabia	1,122,82
3rd	Iran	1,016,61
4th	United Arab Emirates	900,00
5th	Algeria	690,00
6th	Iraq	619,18
7th	Pakistan	557,28
8th	Oman	268,01
9th	Tunisia	180.00
10th	Libya	165.95

to train) and the efficiency comparing to the sate of the art techniques. Our model achieves 99.10% average accuracy rate to detect Bacterial leaf blight, brown spots, Leaf smut, white scale diseases and healthy leaves. 56

The remainder of this paper is organized as follows: Section II presents a summary of related work. Section III illustrates System architectures including our proposed model in details. The next section presents the experimental results of our model and compare it with other models. Finally, the conclusion in addition to the feature work are given in Section V.

II. Related Work

This section shows the previous studies that dealt with the problem of palm tree diseases. In [1], The researchers used various classifiers to detect the three most frequent palm illnesses today: leaf spot, blight, and red palm weevil. CNN was used to distinguish between Leaf Spots and Blight Spots illnesses, and SVM was used to distinguish between the Red Palm Weevil pest and the Leaf Spots sickness. The Kaggle dataset (about 90k photos) was utilized as-is, resulting in a 65 percent accuracy for the first CNN model. The data set was split into two parts: 70% for training and 30% for testing. The accuracy ratio success rates for the CNN and SVM algorithms were 97.9% and 92.8%, respectively.

In [3], the illnesses in tomato fruit were detected using multiple classifiers, the first of which was CNN and LVQ. Leaf blight, bacterial spot, late blight, sitoria leaf spot, and yellow curved leaf disease are among the illnesses he classified. The dataset utilized in this experiment was divided into 400 photos to train the model and 100 images to assess the quality and efficiency of the model, which had a size of 512 x 512 and an accuracy of 86%. The only difficulty this study faces is the degree of symptom similarity among the illnesses that affect tomato fruit.

Jiang et al. [4] proposed a system to identify apple leaf illnesses using deep learning algorithms. To identify and categorize alternate leaf spot, brown spot, mosaic, spot Gray, rust, and eventually brown apple leaf disease, the proposed system employed a new CNN based technique called INAR-SSD based on a framework called Caffe on the GPU platform. Set of data The data utilized is made up of 26377 photos that were separated into two groups: one for training the model and the other for verifying the model. The model speed was 23.13 frames per second, and the accuracy of the findings was 78.80 percent mAP.

The proposed work by [5] investigated the ability to identify chimera and anthracnose infections in palm trees. The detection procedure began with the use of a digital camera to capture photographs of the damaged palm trees, which were then subjected to a series of image processing processes following segmentation using the k-mean method. The Gray Correlation Matrix was created by applying the characteristics extraction phase on the segmented pictures using gyro compatibles (GLCM). The accuracy of the findings proved the effectiveness of the support vector machine (SVM) in such circumstances, with Chimera having a 97 percent accuracy and anthracnose having a 95 percent accuracy.

In [6], The authors used a program called Therma CAM Researcher to analyze pictures collected with an uncooled infrared thermal camera linked to a microbolometer sensor in this work. To obtain credible maps of leaf temperature, this software was enhanced with local climatic variables and leaf emission. Each experiment had 4-5 duplicates of control trees and 8-10 replicates of infected trees in each group. The Crop Water Stress Index (CWSI) was produced, and the damaged trees were identified with up to 75% accuracy using this value. In[7], The authors of this article wanted to see if they could tell the difference between healthy palm trees and palm trees afflicted with the red palm weevil by using water stress and temperature rates. Aerial photos were taken using an uncooled infrared thermal camera and a microbolometer connected to begin the procedure. The palm canopy was separated from the soil using a set of image processing techniques, and the watershed method was used to build the palm canopy diagram using the photos. This study yielded no numerical data, however it did show that various palm plants had variable water stress levels.

III. System Architecture

A. Data Preprocessing

Since the resolution and size of acquired 2D images are usually inconsistent, using them directly as training data is not applicable. In addition, using very high resolution images is time-consuming and may lead to high computational cost. Therefor, all images are resized to 60 x 60 resolution. Furthermore, The correlation between data within the image can slow down the learning process. To address this, we apply image whitening [8] based on Zero Component Analysis (ZCA) to reduce the correlation between data and make key features such as edges and curvatures more prominent, and thus easy to detect by CNN. To achieve that the pixel values in each image is normalized to be between 0 and 1 as the following:

$$\hat{X} = X/255 \tag{1}$$

, where X and $\hat{X}$ are the image before and after normalization, receptively. After that the image is mean normalized by defining the mean value along each feature dimension (pixel position) of training images and then subtracted from the image as the following:

$$\bar{X} = \hat{X} - \mu \tag{2}$$

, where μ represents the mean vector across all the features of $\hat{X}$ while $\bar{X}$ represents the mean normalized image. Finally, the whitened image X_{ZCA} is defined based on Singular Value

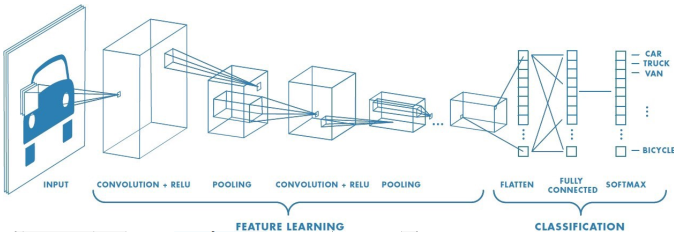

Fig. 1. An Overview of a Convolutional Neural Network (CNN) Architecture [21].

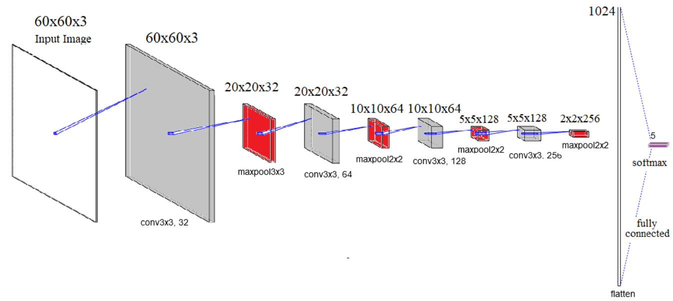

Fig. 2. The Architecture of Proposed CNN.

Decomposition (SVD) of the covariance matrix of $\bar{X}$ as the following

$$X_{ZCA} = U.diag(1/\sqrt{diag(S)+\varepsilon}\ ).U^T.\bar{X} \tag{3}$$

The function $diag(.)$ returns the diagonal matrix of the input matrix. The variable U and S represent the Eigen vector and Eigen value of the SVD of covariance matrix while the hyper-parameter ε represents whitening coefficient.

B. Convolutional Neural Networks

The Convolutional Neural Network CNN [9], [10] is a deep learning model capable to achieve both feature coding (feature extraction) and classification in a single coherent architecture. In contrast, traditional neural networks are limited for classification task and require well-defined feature (engineered features) to achieve high performance [11], [12]. In general, CNNs are widely applied in many computer vision applications such as face recognition [13], object detection [14], Natural language processing [15], medical image analysis and others [16]. Fig. 1. shows the basic architecture of the CNN which involves three different set of layers, namely, ‘Input Layer’, ‘Feature Learning Layers’ and ‘Classification Layers’. The first set is a single layer, includes the input image (usually colored image). The second set is a sequence of successive Convolution and Pooling layers which resulting in extracted features. The last set usually involves fully connected layers to achieve classification tasks. The main advantage of CNN is the ability to define local features such as horizontal edges through convolution layers by maintaining the same structure of the input image whilst using a small number of parameters. For example, regardless of the size of the input image, the number of parameters depends on the size and the number of filters (e.g., 10 filters of [3 x 3] size).

In the resents years, numerous attempts have been made to enhance CNN’s initial design (by Lecun et al. [9]) in order to achieve better performance. A popular examples of CNN architecture are AlexNet [17], VGG-16 [18], ResNet [19] and MobileNet [20]. In this work, a simple CNN architecture is firstly proposed for the problem of Palm Trees Diseases classification and then the performance of our CNN is compared to two well-known CNN structures: “VGG-16” and “MobileNet”.

1) Proposed CNN: In this section, we introduce a simple CNN architecture for efficient learning especially with limited training data size. The proposed architecture includes four convolutional layers followed by fully connected layer and softmax as output layer as shown in Fig. 2.

The input layer includes the input image with dimensions [60, 60,3] where the first two represent the width and the height of the image while the last value represents the number of channels (three channels for red, green and blue). The first convolutional layer includes 32 filters of each [3 x 3] size, followed by a maxpool of 3x3. In the next three convolutional layers, 64 filters [3 x 3] are used for the second layer, while 28 and 256 filters are used for the third and forth layers, respectively. Each of these convolutional layers is followed by a maxpool of 2x2. Finally, a flatten process is applied, resulting in a layer with 25600 features, fully connected to a dense layer with activation function “softmax”.

2) VGG-16: VGG-16 was originally proposed by Simonyan and Zisserman [18] in 2014. The basic idea behind VGG-16 is to increase the depth of the network by adding more convolutional layers while using very small [3 x 3] convolution filters in all layers. Fig. 3 shows the The architecture of VGG-16.

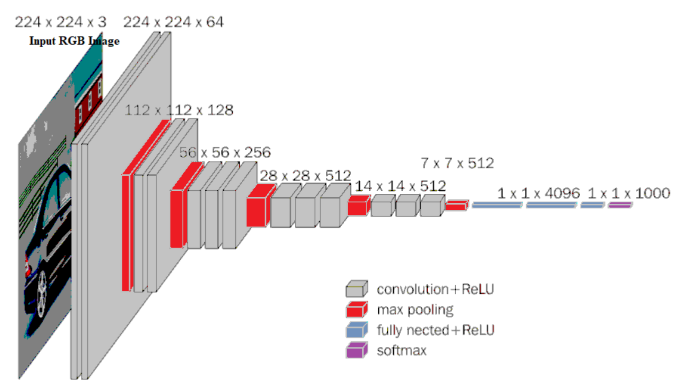

Fig. 3. The Architecture of VGG-16.

During the training phase, the only preprocessing step is to subtract the mean RGB value (calculated over the training set) from each pixel. After that, the image is passed through a set of convolutional layers with small filters ([3 x 3] size). To preserve the spatial resolution the same after the convolution, the stride and the spatial padding are both set to one pixel. Some of the convolutional layers are followed by a max-pooling layer, so a total of five max-pooling layers are applied in the whole network.

3) MobileNets: MobileNet is deep learning model, proposed by Howard et al. [20] of Google Research team. This

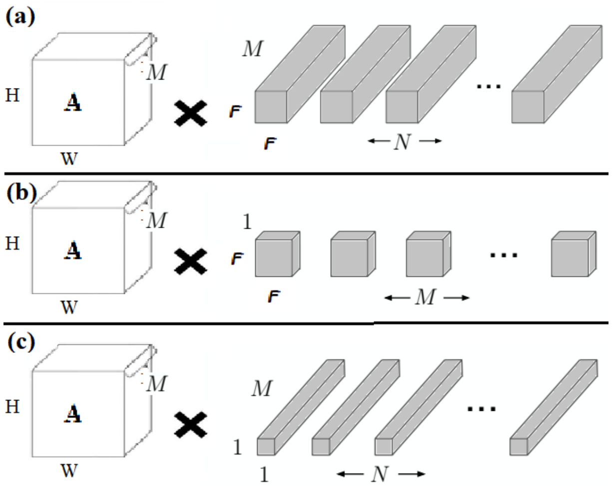

Fig. 4. (a): Standard Convolution Filters. (b): Depthwise Convolutional Filters. (c): Pointwise Convolutional Filters [20].

model was presented to effectively maximize the performance of CNN under limited resources which is ideal for mobile and embedded vision applications. MobileNets is characterized by light weight and hyper-parameters (small number of required wights comparing to other CNN models), which allows to trade-off between latency and accuracy. To achieve that, the standard convolution filters [Fig. 4(a)] replaced by two layers: depthwise convolution [Fig. 4(b)] and and pointwise convolution [Fig. 4(a)].

The computational cost C of the standard convolutional layer can be represented as:

$$C = F \cdot F \cdot M \cdot N \cdot H \cdot W \quad (4)$$

In contrast, MobileNets is capable to reduce the computational cost as:

$$C = F \cdot F \cdot M \cdot H \cdot W + M \cdot N \cdot H \cdot W \quad (5)$$

which includes the cost of the depthwise convolution plus the cost of the pointwise convolution. By comparing the ratio between Eq. 4 and 5 based on [3 x 3] depthwise separable convolutions, MobileNets can achieve 8 to 9 times less computational cost than standard convolutions [20].

IV. Experimental Results

In this section, the performance of the proposed CNN model, in addition to VGG 16 and MobileNet, are evaluated by combining two datasets, 'Date Palm' [22] and 'Leaf_Disease_3' [23].

A. Dataset

Table II gives a summary of each dataset. In our experiments, Date Palm [22] and Leaf_disease_3 [23] datasets are combined to build a new dataset (DP-disease dataset). In more details, Date Palm dataset includes two types of diseases: Brown Spots with 470 samples and White Scale with 958 samples. In addition to that, the dataset also includes 1013 samples of Healthy leaves. In contrast, Leaf_disease_3 dataset includes three different types of diseases: Bacterial Leaf Blight, Brown Spot and Leaf Smut. The dataset is partitioned into training and testing sets where 82,200 samples are used for training and 27,360 samples are used for testing. The number of samples for each disease is 27,360 in the training set and 9120 in the testing set. By combining samples from both datasets, it allows use to include more diseases (four different disease) in addition to the healthy case.As a result, the classification task in all experiments embeds five classes: 1) Bacterial Leaf Blight, 2) Leaf Smut, 3) Brown Spots, 4) White Scale and 5) Healthy leaves.

TABLE II. The Characteristic of Applied Datasets

Dataset type	Total images	Training images	Testing images
Date Palm	2631	Not partitioned	Not partitioned
Leaf_disease_3	102360	82200	20160
Combined dataset (DP-disease dataset)	4471	4023	448

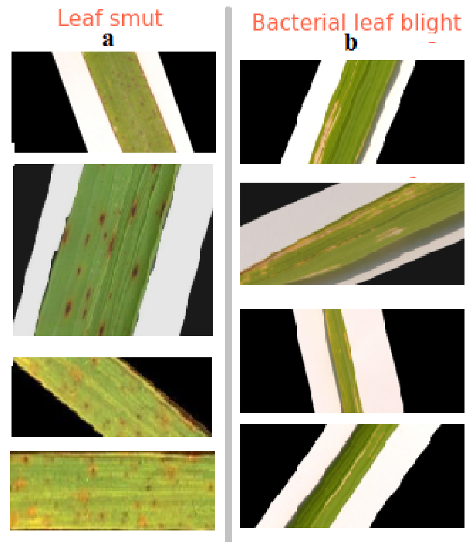

Fig. 5. Samples from: (a) Leaf Smut Disease and (b) Bacterial Leaf Blight Disease.

Fig. 5(a) shows samples from Bacterial Leaf Blight disease where 1,013 images with dimension 3081 x 897 are included in our experiments. Fig. 5(b) shows samples from Leaf Smut disease where 827 images with dimension 510 x 383 are included in our experiments.

Samples from Brown Spots disease and White Scale disease are shown in Fig. 6(a) and (b) where 470 images and 958 images are used in our experiments for Brown spots disease and White scale disease, respectively. For the case of healthy leaves, Fig. 6(c) shows sample of Healthy leaves where 1,203 images are included in our experiments.

B. Experiment Configurations

In our experiments, 4023 samples from different type of diseases were used to train each CNN model while 448 samples from different type of diseases were used to test the performance of each CNN model. The size of input images are unified in all experiments to be 60 x 60 x 3, where the first two dimensions represent the width and height while the last

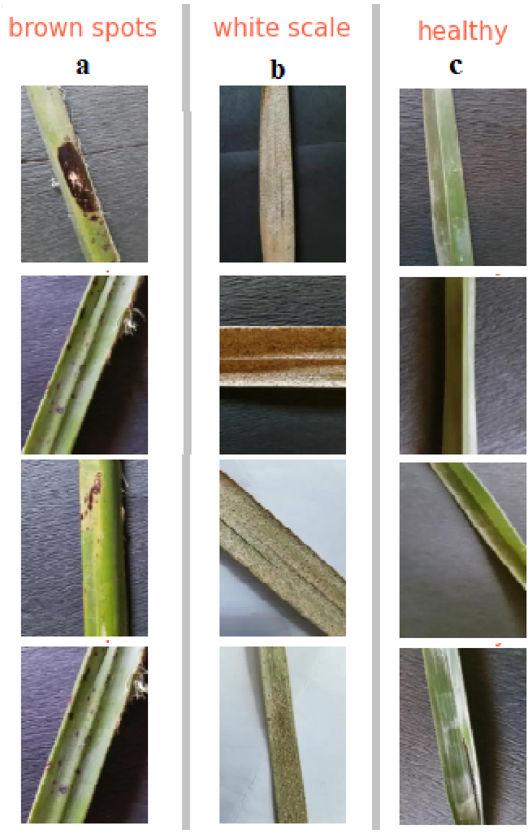

Fig. 6. a)Samples from Brown Spots Disease. b)Samples from White Scale Disease. c) Samples from Healthy Leaves.

TABLE III. THE HYPER-PARAMETERS OF PROPOSED CNN MODE

Hyper Parameters	**Proposed CNN Model**	
Convolution Layers	4	
Number of neurons for each convolution layer	First Layer	32
	Second Layer	64
	Third Layer	128
	Fourth Layer	256
Max Pooling Layers	4	
Fully Connected Layer	1	
Dropout Layer value	NaN	
Activation function	Relu	Softmax
Train Split	0.8	
Test Split	0.2	
Batch Normalization	4	
Kernal window size	(3,3)	(2,2)
Pool size	(3,3)	(2,2)
Trainable params	395,13	
epochs	25	
Batch size	10	
Optimizer	Adam	

one represent the number of color channels (R G B). During the training phase, Batch Gradient Descent were applied in all experiments. In addition to that, the number of epochs sets to 10 which compromises between sufficient training process and over-fitting. More details about the hyper-parameters of each CNN model are shown in Table III, IV and V.

C. Performance Evaluation

The performance evaluation of each model was analyzed using four known evaluation metrics: Accuracy, Precision,

TABLE IV. THE HYPER-PARAMETERS OF VGG-16 MODE

Hyper Parameters	**VGG16 Model**	
Convolution Layers	13	
Number of neurons for each convolution layer	First Layer	64
	Second Layer	64
	Third Layer	128
	Fourth Layer	128
	Fifth Layer	256
	Sixth Layer	256
	Seventh Layer	256
	Eighth Layer	512
	ninth Layer	512
	Tenth Layer	512
	eleventh Layer	512
	twelfth Layer	512
	Thirteenth Layer	512
Max Pooling Layers	5	
Fully connected Layer	3	
Dropout Layer value	NaN	
Activation function	Relu	Softmax
Train Split	0.9	
Test Split	0.1	
Kernal window size	(3,3)	
Pool size	(3,3)	
Trainable params	15,245,125	
epochs	25	
Batch size	10	
Optimizer	Adam	

TABLE V. THE HYPER-PARAMETERS OF MOBILENET MODEL

Hyper Parameters	**MobileNet Model**	
Conv2d Layer	14	
Depthwise Conv2D	13	
Number of neurons for each convolution layer	First Layer	32
	Second Layer	64
	Third Layer	128
	Forth Layer	256
	Fifth Layer	512
	Sixth Layer	1024
Global Average Pooling	1	
Fully connected Layer	2	
Dropout Layer value	1	
Activation function	Relu	Softmax
Train Split	0.9	
Test Split	0.1	
Kernal window size	(3,3)	
Pool size	(7,7)	
Trainable params	5,249,29	
epochs	25	
Batch size	10	
Optimizer	Adam	

Recall and F1 Score. The Accuracy of each CNN model at each type of disease is defined as the following:

$$Accuracy = \frac{TP + TN}{T} \tag{6}$$

, where TP and TN represent number of True Positive and True Negative classifications, respectively, while T represents the total number of samples. Table VI reports the performance of each model based on accuracy. We can notice that MobileNet achieved 99.56% accuracy rate in average which is the best performance. In contrast, our model and VGG16 achieved 99.10% and 99.35% accuracy rate, respectively. It is clear that the performance of the three models are relatively very close with slight variations less than 0.5% .

The Precision of each CNN model at each type of disease is defined as the following:

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

TABLE VI. The Performance of each Model based on Accuracy

Diseases	Our model	MobileNet model	Vgg16 model
Brown Spots	99.55%	100.00%	100.00%
white Scale	99.35%	99.75%	99.35%
Bacterial Leaf Blight	98.79%	98.98%	98.99%
Leaf Smut	99.20%	98.88%	99.5%
Healthy	98.90%	99.90%	98.90%
Average	**99.10%**	**99.56%**	**99.35%**

TABLE VII. The Performance of each Model based on Recall

Diseases	Our model	MobileNet model	Vgg16 model
Brown Spots	100.00%	100.00%	100.00%
white Scale	98.55%	98.55%	99.75%
Bacterial Leaf Blight	96.27%	96.20%	97.27%
Leaf Smut	98.73%	98.73%	98.86%
Healthy	99.15%	99.67%	99.55%
Average	**98.54%**	**98.63%**	**99.08%**

, where TP and FP represent number of True Positive and False Positive classifications, respectively. Table VII reports the performance of each model based on Recall. The highest average precision rate is 99.08% reported by MobileNet, followed by VGG16(98.63%) and our model (98.54%). It is clear that the performance of the three models are relatively very close with slight variations less than 1%.

The performance of each CNN model based Recall and $F_1 Score$ are defined as the following:

$$Recall = \frac{TP}{TP+FN}$$
$$F_1 Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

, where TP and FN represent number of True Positive and False Negative classifications, respectively. Table VIII and IX report the performance of each model based on precision and $F_1 Score$, receptively. The reported results show that the performance of the three models in both measures are very close with variations less than 1%.

From the previous reported results, the performance of VGG16 and MobileNet in addition to the proposed model very close with a slight advantage to MobileNet over others. We believe that VGG16 can achieve better performance with huge training data. However, in several applications, huge training data is not always available. In contrast, MobileNet and our model showed more capability to deal with limited resources such as training data and training time. Fig. 7 shows the average training time of each model. Note that our model achieved the lowest average training time of 20.048s/epoch while MobileNet achieved 343.85s/epoch average training time followed by VGG16 with a huge rise (2559s/epoch). Thus, the proposed model is capable compromise between the performance (high) and the complexity (simple structure with low training time).

TABLE VIII. The Performance of each Model based on Precision

Diseases	Our model	MobileNet model	Vgg16 model
Brown Spots	100.00%	100.00%	100.00%
white Scale	98.55%	99.75%	98.75%
Bacterial Leaf Blight	99.55%	99.55%	98.93%
Leaf Smut	99.33%	99.33%	99.63%
Healthy	98.73%	97.73%	98.88%
Average	**99.23%**	**99.27%**	**99.23%**

TABLE IX. The Performance of each Model based on F1 Score

Diseases	Our model	MobileNet model	Vgg16 model
Brown Spots	100.00%	100.00%	100.00%
white Scale	98.55%	98.87%	98.98%
Bacterial Leaf Blight	99.27%	99.27%	99.65%
Leaf Smut	98.94%	98.94%	98.54%
Healthy	99.15%	99.67%	99.65%
Average	**99.18%**	**99.35%**	**99.36%**

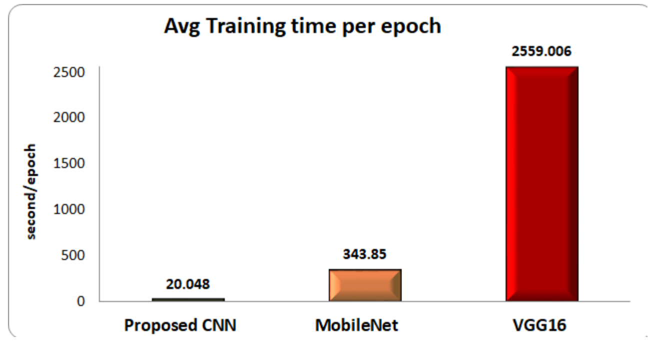

Fig. 7. Avg Training Time per Epoch for each Model.

V. Conclusion

This paper presented an efficient CNN-model for detecting and classifying palm tree frequent diseases, including Bacterial leaf blight, Brown spots, Leaf smut and white scale. The proposed model consists of three stages. Firstly, data preprocessing was applied on all images by normalizing them using Whitening Transform to reduce the correlation among data within the image. The second stage included four convolutional layers with max pooling to extract distinctive features. Finally, the classification stage was defined using fully connected layer with softmax to detect the type of palm disease. Furthermore, this work investigated the performance of two will-known CNN- models, namely MobileNet and VGG-16, for classifying palm tree diseases. To evaluate the performance of proposed CNN-model and other models, a new dataset (DP-disease dataset) was built by combining data samples from Date Palm and Leaf_disease_3 datasets. This combination allowed us to detect four deferent diseases in addition to healthy cases. A set evaluation metrics was then applied to evaluate the performance of each model including accuracy, Precision, recall and F1 score. Our experimental results showed that the performance of the three models were very close. However, our model characterized by simplicity and low computational training cost (20.048s/epoch) comparing to others (343.85s/epoch and 2559s/epoch for MobileNet and VGG-16, respectively).

Future avenues of work include integrating the proposed model with object segmentation methods such R-CNN for accurately identifying the affected parts of palm trees.

References

[1] H. Alaa, K. Waleed, M. Samir, M. Tarek, H. Sobeah, and M. A. Salam, "An intelligent approach for detecting palm trees diseases using image processing and machine learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 7, pp. 434–441, 2020.

[2] "White Scale," {https://candidegardening.com/GB/insects/ace6662f-61ca-4806-99b7-81f4b040faed}.

[3] M. Sardogan, A. Tuncer, and Y. Ozen, "Plant leaf disease detection and classification based on cnn with lvq algorithm," in *2018 3rd International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2018, pp. 382–385.

[4] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59 069–59 080, 2019.

[5] M. M. Kamal, A. N. I. Masazhar, and F. A. Rahman, "Classification of leaf disease from image processing technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, no. 1, pp. 191–200, 2018.

[6] O. Golomb, V. Alchanatis, Y. Cohen, N. Levin, Y. Cohen, and V. Soroker, "Detection of red palm weevil infected trees using thermal imaging," in *Precision agriculture'15*. Wageningen Academic Publishers, 2015, pp. 322–37.

[7] Y. Cohen, V. Alchanatis, A. Prigojin, A. Levi, and V. Soroker, "Use of aerial thermal imaging to estimate water status of palm trees," *Precision Agriculture*, vol. 13, no. 1, pp. 123–140, 2012.

[8] K. K. Pal and K. S. Sudeep, "Preprocessing for image classification by convolutional neural networks," in *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 2016, pp. 1778–1781.

[9] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[10] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Computer Science Review*, vol. 40, p. 100379, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013721000198

[11] S. Elaiwat, M. Bennamoun, F. Boussaïd, and A. A. El-Sallam, "A curvelet-based approach for textured 3d face recognition," *Pattern Recognit.*, vol. 48, pp. 1235–1246, 2015.

[12] S. Elaiwat, "Holistic word descriptor for lexicon reduction in handwritten arabic documents," *Pattern Recognition*, vol. 119, p. 108072, 2021.

[13] N. K. Mishra, M. Dutta, and S. K. Singh, "Multiscale parallel deep cnn (mpdcnn) architecture for the real low-resolution face recognition for surveillance," *Image and Vision Computing*, vol. 115, p. 104290, 2021.

[14] J. Yuan, H.-C. Xiong, Y. Xiao, W. Guan, M. Wang, R. Hong, and Z.-Y. Li, "Gated cnn: Integrating multi-scale feature layers for object detection," *Pattern Recognition*, vol. 105, p. 107131, 2020.

[15] M. Eltay, A. Zidouri, and I. Ahmad, "Exploring deep learning approaches to recognize handwritten arabic texts," *IEEE Access*, vol. 8, pp. 89 882–89 898, 2020.

[16] C. Lee, C. Tanikawa, J.-Y. Lim, and T. Yamashiro, "Deep learning based cephalometric landmark identification using landmark-dependent multi-scale patches," 2019.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, p. 84–90, may 2017.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[20] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[21] "CNN architecture," https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way\\-3bd2b1164a53.

[22] "Date Palm Dataset," {https://www.kaggle.com/hadjerhamaidi/date-palm-data}.

[23] "Leaf Diseases 3 Dataset," {https://www.kaggle.com/research27/leaf-disease-3}.