J-KOMA **Journal of Computer Science and Applications**

Universitas Negeri Jakarta
Phone: (021) 4890856
East Jakarta City
Jakarta, 13220

Open Access

E-ISSN: *2620 - 4872*
Vol*.09,* ***No****.01*
Doi*: 10.21009/JKOMA.091.08*

Recivied*: 02June 2026*
Accepted*: 23 July 2026*
Published*: 31 July 2026*

Keywords:
Plant disease;
CNN;
Dense Convolutional Network;
Deep learning;
Data augmentation.

Correspondence Email:
wasanahmed@uodiyala.edu.iq

Early Plant Disease Classification System Based on Deep Neural Network Techniques

Wasan Ahmed Ali
Department of Computer Science, College of Science. University of Diyala, Iraq.

Abstract
Plant diseases can significantly impact the economy, affecting both global and local scales. These illnesses can substantially negatively impact crop yields and quality, resulting in significant losses in crop production. Deep learning algorithms are commonly recognized as potent remedies in this regard. CNNs are a part of deep learning methods and the most widely utilized image recognition classifier, which it has illustrated exceptional proficiency in both classification and image processing. This paper presents a CNN-based approach for plant disease identification. This work suggests a CNN model with data augmentation and performs plant disease classification with data augmentation and transfer learning with DenseNet121. Sample photos are analyzed and studied through simulation regarding the contaminated region's area. An image processing technique is used to do this. Three cases of healthy (Potato, leaves, Bell Paper, and Tomato), and twelve cases of diseased plant leaves, including Potato Early Blight, bell paper bacterial spot, tomato mosaic virus, tomato target spot, tomato spider mites, tomato early blight, tomato bacterial spot, tomato late blight, Tomato Septoria Leaf spot, Tomato Yellow Leaf Curl Virus, Potato Late Blight, and Tomato Leaf Mold, were fed into the model. A validation accuracy is achieved as 96.85% with DenseNet121, and 98% with the proposed CNN model.

INTRODUCTION

Deep learning(DL) models are employed to address various agricultural issues [1]. Leaf disease is one of the issues that significantly impacts agriculture. According to estimates, plant diseases cause 35% of agricultural produce to be squandered [2]. This wasteful produce impacts the nation's economy and raises food costs in an undesirable way. According to [3], plant diseases can cause production losses of 13% to 22% per year, translating into billions of US dollars in lost revenue.
In the past, institutions like agricultural extension groups or local plant clinics have assisted in the diagnosis of diseases. More recently, by taking advantage of the growing global Internet penetration, such initiatives have also been aided by the availability of online information for disease diagnosis. Recently, mobile phone-based tools have been widely available, to take advantage of global mobile phone technology which has been adopted at levels not seen in human history [4].
Technology has helped bring about significant advancements in the agriculture field. The contaminated area of the plant has been identified using image processing and object identification techniques. Apart from their ease of use and precision, these methods are efficient [5], [6]. Therefore, improvements in internet and computer technology can help solve the issue of automatically identifying plant diseases. These advancements are crucial to scientific study because they enable the automatic classification and detection of plant disease signs through creative and clever methods [7].
DLis a subfield of machine learning that relies on a collection of algorithms [8]. Several cutting-edge deep-learning architectures have been employed to determine and detect plant diseases. In this work, developed a CNN model for identify plant disease experimented on Plant Village dataset. The dataset was first

processed, and then data augmentation was performed, followed by transfer learning with DenseNet121. The model is compared with the CNN proposed, and the model with transfer learning outperforms the proposed CNN.
The paper is arranged as follows: Section II demonstrates the existing work for classification of plant diseases. Section III illustrates the methodology used. Section IV illustrates the results. Finally, Section V summarizes the conclusion.

RELATED WORK

CNNs are a part of deep learning methods recently gaining popularity [9]. CNN is the most widely utilized image recognition classifier, and it has illustrated exceptional proficiency in both classification and image processing [1]. In Ref [10], the author presented a framework design experiment, five status design were included grouping, channel, convolution kernel, residua block number, and residual layer connection design. The Multi-scale ResNet network structure was finally established, encompassing the use of two channels, a two-layer connection mode, two residua blocks, group convolution, and a big convolution kernel. The model experimented on plant village dataset and attained an accuracy of 95.95%.
Agarwal et al. [11] proposed a CNN framework to determine nine kinds of tomato leaf disease. The model consists of two fully connected layers, three maximum pooling layers, and three convolutional layers with different filters. It achieves an accuracy of 91.2%. The model was compared with MobileNet, Inception, and VGG16 and outperformed them in test accuracy.
Y. Wang et al. [12] investigated five apple leaf infections, i.e., apple scab, alternaria leaf spot, gray spot, mosaic and cedar rust. Firstly, the training set data was expanded through the data augmentation operation (including random brightness enhancement, rotating, random chromaticity enhancement, sharpening, and random contrast enhancement) to enhance the robustness of the framework. Then the augmented training set images were trained through the improved Faster R-CNN to make the detection model more reliable. For the improved Faster R-CNN model, the attention separation mechanism ResNcst (split-attention networks) was adopted as the backbone to make the model focusing on the more important information to enhance the ability of feature extraction, a feature pyramid network (FPN) was added to fuse multi-scale features, which effectively used the shallow and deep features of the network. The framework attained an accuracy of 98% with five classes.
Li et al. [13] developed a framework called PARNet to identify five tomato leaf diseases. The model is built by combining the residual structure with the attention mechanism. Then, they employed the development of a WEB application. The PARNet model attained an accuracy of 96.84%.
T. N. Pham et al [14] developed a framework for early plant disease detection on mango leaves utilizing ANN. Firstly, an image is processed using a contrast enhancement method, and then a feature is selected utilizing a wrapper-based feature selection algorithm. The resulting features were used as input for ANN and achieved an accuracy of 89.41%.
A. Khattak [15] developed a CNN model that integrated multiple layers to extract complementary discriminative features. The model detected five citrus diseases and attained an accuracy of 94.55%.

METHODOLOGY

A. Dataset

In this work, a publicly available dataset called Plant Village [16] is used, which consists of 54,306 images with 38 different diseased/healthy leaves belonging to 14 plant species. The image size was updated to 224 224 3, and the pixel value was divided by 255 to normalize images to be suitable for the initial values of the models. To avoid overfitting, the data was partitioned into 10% validation, 70% training, and 20% testing [17].

B. Data Labeling

Data augmentation is one of several regularization strategies designed to improve model performance. Regularization techniques add information to the underlying machine learning model to reflect the problem's general features better. The most often used and simplest strategy is to include a penalty term in the cost function to limit values of model parameters, which enables the trained model to respond more consistently and smoothly to different variances of input data (during test time). Batch normalization normalizes the outputs of multiple layers, eliminating the impacts of internal covariance shifts in inputs to deep levels [18]. Data augmentation taxonomy illustrated in Figure1.

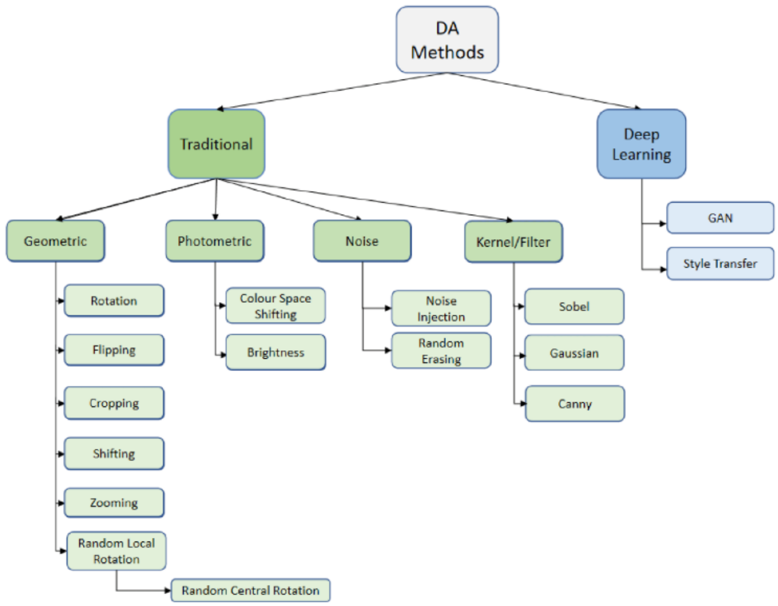

Figure 1. Sentiment Distribution

C. Transfer Learning

Transfer Learning [20], [21] is another intriguing method for preventing overfitting. It involves training a network on a large dataset, such as ImageNet [22] and then applying the starting weights in a new classification issue. Typically, only the weights of convolutional layers are duplicated rather than the whole network with fully connected layers. That is extremely useful since many image datasets contain low-level spatial properties that can be better learned with big data. Understanding the link among transmitted data domains is an ongoing research topic [23].

D. DENSENET121

The DenseNet introduces a considerable improvement in DL methods. The technique utilizes a convolutional procedure that permits more in-depth image analysis and an improved calculating procedure for image recognition. It includes pre-trained models trained on large databases such as C and Imagenet, establishing itself as a model of transfer learning. It connects all convolutional layers to the next layer and passes the feature map output from the layer before it to the next layer, the procedure known as a DENSE BLOCK [29]. Figure 2 demonstrates the process of transfer learning for DenseNet121 to detect plant disease.

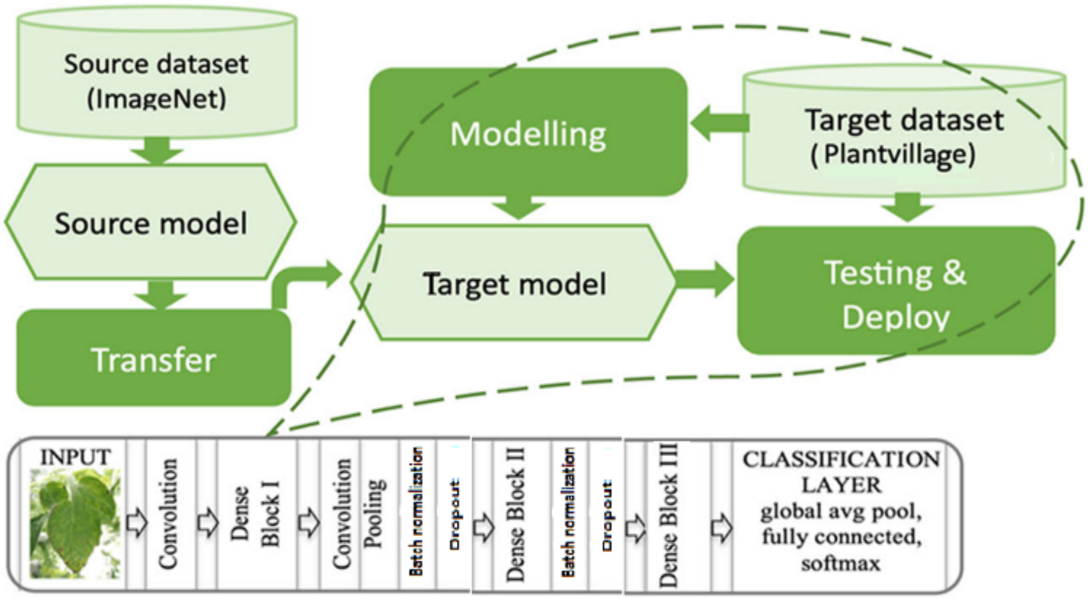

Figure 2. Details of plant disease classification with DenseNet121

E. Evaluation Metrics

Accuracy is utilized to extract the model's performance. Validation accuracy is the degree to which the trained model closely follows the learned data. In contrast, the false positive rates are the total value of every column. In contrast, FN(false negative) is each row's amount. TN (true negative) is the sum of the other diagonal numbers, whereas the diagonal numbers provide an accurate positive rate (TP). However, accuracy is computed utilizing the following equation [30], [31].

Accuracy=(tn+fp)/(tp+tn+fp+fn)

RESULT AND DISCUSSION

A. CNN Implementation

CNNs have revolutionized image processing and are now a cornerstone of many applications, from facial recognition to medical imaging. The proposed CNN model comprises of four convolutional layers with max-pooling followed by flattening, then a dense layer with 32 kernels. The summary of the suggested CNN is demonstrated in Figure 3.

As the era progressed, the data showed a steady drop in training loss and a uniform growth in training accuracy. In contrast, after the 17th epoch, relatively minimal variations showed in validation accuracy and loss. Notably, the 48th epoch showed a training accuracy of 97.36%, with a training loss of 0.0771. Furthermore, the validation accuracy reached its peak at the 47th epoch, at 98%, with a validation loss of 0.0653. Figure 4 illustrates the loss and validation and training accuracy.

Universitas Negeri Jakarta
Phone: (021) 4890856
East Jakarta City
Jakarta, 13220

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                       Output Shape          Param #
=================================================================
sequential (Sequential)            (32, 128, 128, 3)     0
sequential_1 (Sequential)          (32, 128, 128, 3)     0
conv2d (Conv2D)                    (32, 128, 128, 32)    896
max_pooling2d (MaxPooling2D)       (32, 64, 64, 32)      0
conv2d_1 (Conv2D)                  (32, 64, 64, 64)      18496
max_pooling2d_1(MaxPooling2D)      (32, 32, 32, 64)      0
conv2d_2 (Conv2D)                  (32, 32, 32, 64)      36928
max_pooling2d_2(MaxPooling2D)      (32, 16, 16, 64)      0
conv2d_3 (Conv2D)                  (32, 16, 16, 64)      36928
max_pooling2d_3 (MaxPooling2D)     (32, 8, 8, 64)        0
conv2d_4 (Conv2D)                  (32, 8, 8, 64)        36928
max_pooling2d_4 (MaxPooling2D)     (32, 4, 4, 64)        0
conv2d_5 (Conv2D)                  (32, 4, 4, 64)        36928
max_pooling2d_5 (MaxPooling2D)     (32, 2, 2, 64)        0
flatten (Flatten)                  (32, 256)             0
dense (Dense)                      (32, 64)              16448
dense_1 (Dense)                    (32, 15)              975
=================================================================
Total params: 184527 (720.81 KB)
Trainable params: 184527 (720.81 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 3. Summary of suggested CNN model

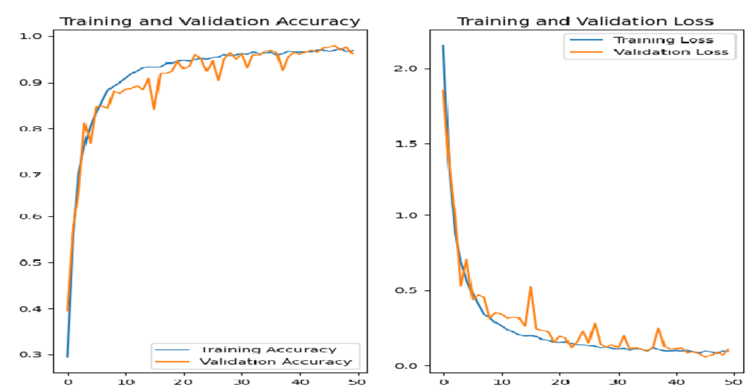

Figure 4. The CNN loss and validation and training accuracy.

B. DenseNet121 Implemntation

Data a Several parameters are required to configure the DenseNet121 layer, including input_shape with 224x224x3 and weights utilizing ImageNet. After processing DenseNet121 layer, results went through the batch normalization layer, global average pooling layer, and the dropout feature. The following phase involves a fully connected Dense layer with a 1024-kernel, resulting in a final layer using softmax activation with Dense. A summary of the DenseNet121 is demonstrated in figure 5. As the era progressed, the data showed a steady drop in training loss and a uniform growth in training accuracy. In contrast, after the 10th epoch, relatively minimal variations showed in validation accuracy and loss. Notably, the 27th epoch showed a training accuracy of 98.23%, with a training loss of 0.0568. Furthermore, the validation accuracy reached its peak at the 25th epoch, at 96.85%, with a validation loss of 0.1086. When compared with DenseNet121 the suggested CNN attained an accuracy of 98%. In comparison with related works results on plant village dataset, where ResNet (11), CNN (13), PARNet (14), ANN with wrapper feature selection (16), and CNN (17) attained accuracies of 95.95%, 98%, 96.84%, and 89.41% and 94.55%, respectively, our proposed method produces an enhanced result of 98%. Table 1 demonstrates the comparison of the proposed approach with related works.

Universitas Negeri Jakarta
Phone: (021) 4890856
East Jakarta City
Jakarta, 13220

```
Model: "model"

_________________________________________________________________
Layer (type)                   Output Shape                Param #
=================================================================
input_2 (Input Layer)          [(None, 224, 224, 3)]       0
conv2d (Conv2D)                (None, 224, 224, 3)         84
densenet121 (Functional)       (None, None, None, 1024)    7037504
global_average_pooling2d       (None, 1024)                0
(GlobalAveragePooling2D)
batch_normalization            (None, 1024)                4096
(Batch Normalization)
dropout (Dropout)              (None, 1024)                0
dense (Dense)                  (None, 256)                 262400
batch_normalization_1          (None, 256)                 1024
(Batch Normalization)
dropout_1 (Dropout)            (None, 256)                 0
dense_1 (Dense)                (None, 128)                 32896
batch_normalization_2          (None, 128)                 512
(Batch Normalization)
dropout_2 (Dropout)            (None, 128)                 0
root (Dense)                   (None, 15)                  11352
=================================================================
Total params: 7349868 (28.04 MB)
Trainable params: 7263404 (27.71 MB)
Non-trainable params: 86464 (337.75 KB)
_________________________________________________________________
```

Figure 5. Summary of DenseNet121

Table 1. A comparison of the proposed approach with previous works

Reference	Approach	Classes	Accuracy
[10]	Resnet	Tomato, potato, cucumber, pepper	95.95%
[11]	CNN	Apple	98
[13]	PARnet	Tomato	96.84%
[14]	ANN with wrapper feature selection	Mango	89.41%
[15]	CNN	Citrus	94.55%
Proposed approach	CNN	Tomato, potato, pepper	98%
	DenseNet121	Tomato, potato, pepper	96.84%

CONCLUSION

Several cutting-edge deep-learning architectures have been employed to specify and determine plant diseases. In this work, suggested a CNN framework to detect plant disease experimented on Plant Village dataset. Three cases of healthy (Potato, leaves, Tomato, and Bell Paper), and twelve cases of diseased plant leaves, including bell paper bacterial spot, Potato Late Blight, tomato (spider mites, target spot, mosaic virus, early blight, bacterial spot), late blight, Tomato Septoria Leaf spot, Tomato Yellow Leaf Curl Virus, Potato Early Blight, and Tomato Leaf Mold, were fed into the framework. Dataset first processed and then performed data augmentation then transfer learning with DenseNet121. The model is compared with CNN proposed and the framework with transfer learning is outperformed the proposed CNN. In context to the results, the CNN framework outperformed the DenseNet121 model in terms of accuracy. The developed CNN framework attained an accuracy of 98% whereas DenseNet121 attained 96.85%. For future the proposed CNN model will developed to detect more plant disease not just diseases of potato, tomato, and pepper.

Universitas Negeri Jakarta
Phone: (021) 4890856
East Jakarta City
Jakarta, 13220

REFERENCE

[1] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” Comput Electron Agric, vol. 147, pp. 70–90, 2018, doi: https://doi.org/10.1016/j.compag.2018.02.016.

[2] E.-C. OERKE, “Crop losses to pests,” J Agric Sci, vol. 144, no. 1, pp. 31–43, 2006, doi: DOI: 10.1017/S0021859605005708.

[3] D. C. He, M. H. He, D. M. Amalin, W. Liu, D. G. Alvindia, and J. Zhan, “Biological control of plant diseases: An evolutionary and eco-economic consideration,” Oct. 01, 2021, MDPI. doi: 10.3390/pathogens10101311.

[4] B. Sanou, “ICT,” Facts and Figures, 2018.

[5] U. Shruthi, V. Nagaveni, and B. K. Raghavendra, “A Review on Machine Learning Classification Techniques for Plant Disease Detection,” in 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 281–284. doi: 10.1109/ICACCS.2019.8728415.

[6] H. and S. A. K. and M. S. C. Panigrahi Kshyanaprava Panda and Das, “Maize Leaf Disease Detection and Classification Using Machine Learning Algorithms,” in Progress in Computing, Analytics and Networking, P. K. and R. S. S. and L. K.-C. Das Himansu and Pattnaik, Ed., Singapore: Springer Singapore, 2020, pp. 659–669.

[7] M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, “Image-Based Plant Disease Identification by Deep Learning Meta-Architectures,” Plants, vol. 9, no. 11, 2020, doi: 10.3390/plants9111451.

[8] B. B. Benuwa, Y. Z. Zhan, B. Ghansah, D. K. Wornyo, and F. Banaseka Kataka, “A Review of Deep Machine Learning,” International Journal of Engineering Research in Africa, vol. 24, pp. 124–136, Feb. 2016, doi: 10.4028/www.scientific.net/JERA.24.124.

[9] J. G. A. Barbedo, “Factors influencing the use of deep learning for plant disease recognition,” Biosyst Eng, vol. 172, pp. 84–91, Aug. 2018, doi: 10.1016/j.biosystemseng.2018.05.013.

[10] C. Wang, J. Zhou, H. Wu, G. Teng, C. Zhao, and J. Li, “Identification of vegetable leaf diseases based on improved Multi-scale ResNet,” Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering, vol. 36, no. 20, pp. 209–217, Oct. 2020, doi: 10.11975/j.issn.1002-6819.2020.20.025.

[11] Y. Wang, J. Wu, P. Lan, F. Li, C. Ge, and F. Sun, “Apple disease identification using improved Faster R-CNN,” Journal of Forestry Engineering, vol. 7, pp. 153–159, Feb. 2022, doi: 10.13360/j.issn.2096-1359.202104028.

[12] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, “ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network,” Procedia Comput Sci, vol. 167, pp. 293–301, 2020, doi: https://doi.org/10.1016/j.procs.2020.03.225.

[13] X. Z. Li, Y. Xu, Z. H. Wu, Z. Gao, and L. Liu, “Recognition system of tomato leaf disease based on attentional neural network,” Jiangsu J. Agricult. Sci., vol. 36, no. 3, pp. 561–568, 2020.

[14] T. N. Pham, L. Van Tran, and S. V. T. Dao, “Early Disease Classification of Mango Leaves Using Feed-Forward Neural Network and Hybrid Metaheuristic Feature Selection,” IEEE Access, vol. 8, pp. 189960–189973, 2020, doi: 10.1109/ACCESS.2020.3031914.

[15] A. Khattak et al., “Automatic Detection of Citrus Fruit and Leaves Diseases Using Deep Neural Network Model,” IEEE Access, vol. 9, pp. 112942–112954, 2021, doi: 10.1109/ACCESS.2021.3096895.

[16] D. Hughes, M. Salathé, and others, “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” arXiv preprint arXiv:1511.08060, 2015.

[17] K. C. Kamal, Z. Yin, M. Wu, and Z. Wu, “Depthwise separable convolution architectures for plant disease classification,” Comput Electron Agric, vol. 165, p. 104948, 2019.

[18] A. Mumuni and F. Mumuni, “Data augmentation: A comprehensive survey of modern approaches,” Array, vol. 16, p. 100258, 2022, doi: https://doi.org/10.1016/j.array.2022.100258.

[19] K. Alomar, H. I. Aysel, and X. Cai, “Data Augmentation in Classification and Segmentation: A Survey and New Strategies,” J Imaging, vol. 9, no. 2, Feb. 2023, doi: 10.3390/jimaging9020046.

[20] L. Shao, F. Zhu, and X. Li, “Transfer learning for visual categorization: A survey,” IEEE Trans Neural Netw Learn Syst, vol. 26, no. 5, pp. 1019–1034, 2014.

[21] K. Weiss, T. M. Khoshgoftaar, and D. Wang, “A survey of transfer learning,” J Big Data, vol. 3, no. 1, p. 9, 2016, doi: 10.1186/s40537-016-0043-6.

[22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[23] A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese, “Taskonomy: Disentangling task transfer learning,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3712–3722.

[24] S. Karen, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv: 1409.1556, 2014.

[25] G. Andrew, Z. Menglong, and others, “Efficient convolutional neural networks for mobile vision applications,” Mobilenets, vol. 10, p. 151, 2017.

[26] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[27] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1251–1258.

[28] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[29] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, “Using deep transfer learning for image-based plant disease identification,” Comput Electron Agric, vol. 173, p. 105393, 2020, doi: https://doi.org/10.1016/j.compag.2020.105393.

[30] N. A. Saeed and Z. T. M. Al-Ta’i, “Heart Disease Prediction System Using Optimization Techniques,” in New Trends in Information and Communications Technology Applications, A. M. Al-Bakry, S. O. Al-Mamory, M. A. Sahib, H. S. Hasan, G. S. Oreku, T. M. Nayl, and J. A. Al-Dhaibani, Eds., Cham: Springer International Publishing, 2020, pp. 167–177. doi: 10.1007/978-3-030-55340-1_12.

[31] N. A. Saeed and Z. T. M. Al-Ta’i, “Feature Selection using Hybrid Dragonfly Algorithm in a Heart Disease Predication System,” International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no. 6, pp. 2862–2867, 2019, doi: 10.35940/ijeat.F8786.088619.