Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 218 (2023) 1825–1833

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

TomConv: An Improved CNN Model for Diagnosis of Diseases in Tomato Plant Leaves

Preeti Baser[a], Jatinderkumar R. Saini[b*], Ketan Kotecha[c]

[a]*National Forensic Sciences University, Gandhinagar-382 007, India.*
[b]*Symbiosis Institute of Computer Studies and Research, Symbiosis International (Deemed University), Pune-411 016, India.*
[c]*Symbiosis Centre for Applied Artificial Intelligence, Symbiosis International (Deemed University), Pune-412 115, India.*

Abstract

Crop disease in the plant is a significant issue in the agriculture sector, and it is currently very difficult to detect these illnesses in crop leaves. The foundation of the global economy is agriculture. India ranks second in the production of tomatoes worldwide. The tomato crop is affected by various diseases which lead to a reduction in product quality and quantity. The advancement in computer vision and deep learning opens up the door for predicting diseases that appear in the crops. The aim of this paper is classification among 10 different categories of tomato plant leaves using the proposed novel TomConv model which deploys an improved Convolutional Neural Network (CNN). For this purpose, the publicly available dataset called PlantVillage comprising of more than 16000 images of tomato leaves, both diseased and healthy was used for the experimentation purpose. The proposed model is the simplest model among all the available state-of-the-art models. The tomato leaf images were preprocessed for reducing the size in 150×150 dimension. The model constitutes four layered CNN followed by a max pooling layer. The model splits the corpus into training and validation datasets in 80:20 ratio, is trained under 105 epochs for tomato leaf images, and achieved an accuracy of 98.19%. The proposed model is compared with existing models under different parameters such as no. of classes, no. of layers, and accuracy. The results are promising as they outperform all the available state-of-the-art models.

Keywords: Computer Vision; Convolutional Neural Network (CNN); Dense Layer; Fully Connected Network (FCN); Max Pooling; Rectified Linear Unit (ReLU); Softmax; Tomato

* Corresponding author. Tel.: +91-968-768-9708.
E-mail address: saini_expert@yahoo.com

10.1016/j.procs.2023.01.160

1. Introduction

The most extensively grown crop in India is the tomato. Tomato are the most preferred crop by indian farmers because of its huge demand in domestic consumption as well package food industry. It is the world's third largest vegetable crop after potato and onion. More than seven billion people over the world live on it as food. Due to technological advancement in the agriculture domain, farmers are now able to produce sufficient food to satisfy increasing demand. Plant disease plays a major concern for the smallholder farmers whose livelihood completely depends on healthy crops. Therefore, plant disease detection in the early stage is very necessary in order to minimize the damage, reduce production costs and raise the global income at large. In the past farmers used to meet experts and based on their suggestions farmers were applying proper pesticides to cure the plant disease. However, this method requires a lot of time, so farmers must rely on reliable professionals. Therefore, a quick, precise and less expensive technique to identify disease in plant leaves is required. This will improve overall yield of tomato crop and it gives more income to farmers. Hence, farmers are more encouraged to use scientific tools in agriculture activities.

The latest trends in image processing and machine learning technologies are helpful to farmers to identify weed detection appearing on plant leaves on an early basis [1], for phenotyping of plant disease symptoms [2]. Nowadays, modern technological advancement in the field of artificial intelligence helps farmers to diagnose disease on plant in order to reduce pesticides. Recently, deep learning is widely used for various plant images which are publicly available on the PlantVillage dataset [3].

Tomato is a nutrient-dense plant that is widely cultivated as a vegetable. Around 160 million tonnes of tomatoes are consumed each year worldwide. According to data, small farmers provide more than 80% of total agricultural output, but over 50% of their harvests are lost due to disease and pests. It is vital to do research on field crop disease diagnostics since parasitic insects and diseases are highly affecting tomato growth. Tomato production is threatened by different types of diseases that appear over tomato leaves. However, early diagnosis of these types of diseases can help the farmers to take preventive actions and save their crops. Therefore, there is a definite requirement for system to diagnose the diseases in the tomato plant leaves. Additionally, machine learning (ML) approaches have already been effectively uses to diagnose various diseases in a variety of other plant types, including cotton and rice. Advancement in computer vision, particularly Convolutional Neural Network (CNN) has shown reliable findings in the areas of image classification. In this paper, we take a first step towards classifying tomato leaf disease. Fig. 1 shows sample images of tomato plant leaves. However, the existing study shows model that use transfer learning of existing models which are highly dense and complex convolutional neural network model. Even these models consume memory resources and execution time. Therefore, motivation of our study is to propose lightweight, simplest and accurate model. The model is developed and trained from the scratch for the tomato leaf images.

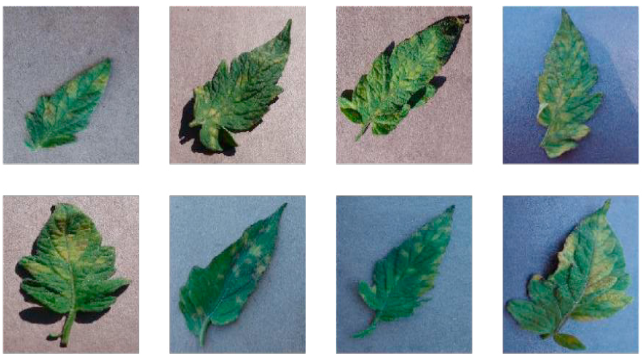

Fig. 1 Sample leaf images of Tomato Plant

The paper is organized as follows. Section 2 talks about the literature review of the available techniques. The detailed methodology comprising of dataset, materials and methods is discussed in Section 3. Section 4 then discusses the findings. Finally, the paper is concluded with a discussion on future directions for further enhancement of the proposed algorithm.

2. Literature Review

Atole and Park [4] used a pretrained model called AlexNet to classify disease in 600 rice plants leaf images. Using stochastic gradient descent with a batch size of 20 and an initial learning rate of 0.0001, they were able to obtain accuracy of 91.23%. Dhakal and Shakya [5] used a deep learning model to identify plant disease. They extracted features, segmentation and classified patterns of captured leaves into four classes. This model measured accuracy of 98.59% under 20 epochs. Rangarajan et al. [6] used images of tomato leaves from PlantVillage dataset as input and used pretrained models namely AlexNet and VGG16 architecture. This model was trained for hyperparameters such as bias rate, weight and batch size. VGG16 architecture outperformed under all criteria compared to the AlexNet. VGG16 architecture achieved accuracy of 96.19% while AlexNet achieved 95.81%.

Durmus et al. [7] used pretrained models such as AlexNet and SqueezeNet to classify 9 different diseases and healthy tomato plant leaves. AlexNet architecture gave accuracy of 95.65% whereas SqueezeNet architecture noted 94.3% accuracy. Brahimi et al. [8] classified tomato leaves into 8 different diseases using the AlexNet and GoogleNet model. The accuracy achieved by these architectures was compared and found to be higher than Random Forest (RF) and Support Vector Machines (SVM). Too et al. [9] used deep learning models such as VGG, Inception V4, DenseNet and ResNet in order to classify disease in plant leaves. Finally, the DenseNet model attained a 99.75% accuracy rate. Foysal et.al. [10] demonstrated feature extraction such as color, shape and texture of plant image. These segmented tomato leaves are fed into the classification algorithm. Finally, they identified six types of diseases and yielded overall accuracy of 97.3%.

To recognize five different tomato diseases, Sabrol and Satish[11] designed a 15-layered deep convolutional neural network. Tm et al. [12] created the LeNet architecture to identify diseases in tomato plants. This architecture achieved average accuracy of 94-95% under unfavourable conditions. Further, they proposed a model to classify diseases of PlantVillage dataset consisting of more than 54,000 diseased and non-diseased leaf images. Based on species and diseases, these images were further partitioned into 38 classes. The present research work distinctly focuses just on the tomato plant leaves to diagnose the diseases.

The CNN model was constructed for classifying disease of 500 natural images of rice plant leaves [13]. This model was trained under 10-fold cross validation with accuracy level 95.48% which was higher than the conventional model. Balakrishna and Rao [14] proposed two methods KNN and PNN for identification of healthy and unhealthy 600 tomato leaves. The experimental analysis revealed that PNN classifier outperformed KNN. Hong et al. [15] used five deep neural network models namely Resnet50, Xception, MobileNet, ShuffleNet, Densenet121 and Xception. The best recognition accuracy of DensNet_Xception was found to be 97.10%. The authors [16] used pretrained model DensNet161 and VGG16 with transfer learning. The accuracy of DensNet161 was highest with a value of 95.65%. Shijie et. al. [17] used VGG16 for feature extraction and SVM for classification of the diseases for tomato plant leaves. They reported an accuracy of 88% for the test dataset.

Nandhini and Ashokkumar [18] reported an accuracy of 99% for tomato plant leave disease diagnosis. However, it is notable that they used only a toy dataset of around 6200 leaves and experimented with only 4 classes. For their research, they also combined CNN with the Improved Crossover based Monarch Butterfly Optimization(ICRMBO) algorithm. The database used by the researchers was the famous PlantVillage dataset. Zhou et al. [19] reported an accuracy of 95% after using 200 epochs with more than 13000 images sourced from the Artificial Intelligence (AI) Challenger 2018 dataset. Zhao et al. [20] reported accuracy of 96.81% on a toy dataset of 4585. They experimented with 10 categories. Abbas et al. [21] reported accuracy of 97.11% for 10-category classification of tomato plant disease prediction using synthetic images. In addition to the diagnosis of tomato plant leaf diseases, there have also been research works reported on tomato plant root disease [22], tomato plant seedlings [23], tomato-pest identification [24], and tomato yield using Artificial Neural Networks (ANN) [25].

3. Methodology

In this section, the detailed discussion on dataset, materials and methods has been presented. The present research work used 16,012 images of tomato leaves for the study from the PlantVillage publicly available dataset. This dataset contains 9 types of leaf diseases of the tomato plant. It also contains a 10 category of healthy leaves for the comparison and empirical purpose. That means we have trained our model for 10 classes such as Class 1 : "Bacterial spot", Class 2 : "Early blight", Class 3 : "Late blight", Class 4 : "Leaf Mold", Class 5 : "Septoria Leaf Spot", Class 6 : :Two Spot Spider Mite:, Class 7 : "Target Spot", Class 8 : "Yellow-Leaf Curl Virus", Class 9 : Mosaic Virus", Class 10 : "Healthy".

The entire methodology is separated into a number of stages like collecting the images for classification process, pre-processing images, designing the model, and finally training and validating the model for tomato disease identification using the improved CNN model. Tomato leaves are pre-processed and resized into 150×150 for speedy processing. We divided the dataset in 80:20 ratio across the training and validation sets. The training set contains 12,830 images while 3,182 images constitute the validation set. CNN model takes these images to extract features that classify among 10 classes of tomato leaves disease.

The model for image classification is shown in the Figure 2. Image pre-processing and CNN-based classification are two main components of this model. The constructed model is trained for a public dataset of tomato images. These images pass through different CNN layers in order to train the model. Afterward, this trained model is checked for accuracy if the model gives desired result, then it is deployed for final production.

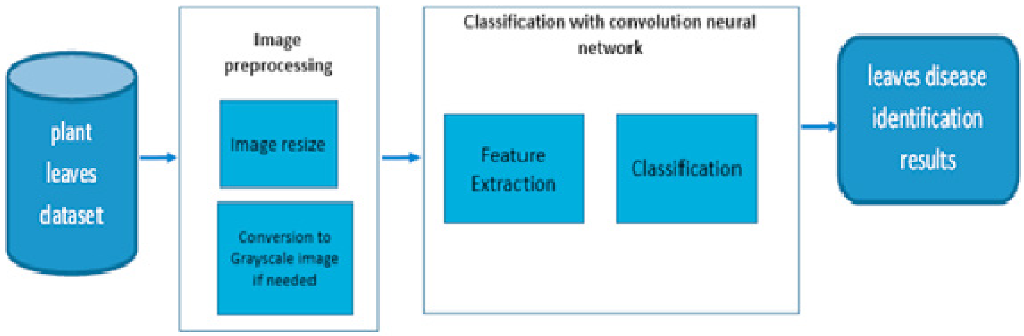

Fig. 2 CNN-based Image Classification Model

The CNN architecture has four convolutional layers as shown in Table 1. Every convolutional layer is followed by a ReLU function. Features are computed at the four convolutional layers followed by max pooling. The proposed CNN model as shown in Table 1 starts with input feature map with 150×150 dimension and 3 for colors R, G and B. The first convolutional block has 32 filters of size 3×3 with ReLU function. The first convolutional neural network outputs 32 images of 148×148 size. A MaxPooling layer with 32 filters of size 2×2 is added after the first convolutional neural network, scaling down the 32 images from size 148×148 to size 74×74. The second convolutional layer receives the output from the preceding layer. The second convolutional block contains 64 ReLU functioned 3×3 sized filters. The second convolutional neural network outputs 64 images of 72×72 size. A MaxPooling layer with 64 filters of size 2×2 is added after the second convolutional neural network, scaling down the 64 images from size 72×72 to size 36×36. The output generated by second layer is fed into third convolutional layer. The third convolutional block has 128 filters of size 3×3 with ReLU function. The third convolutional neural network outputs 128 images of 34×34 size. A MaxPooling layer with 128 filters of size 2×2 is added after the third convolutional neural network, scaling down the 128 images from size 34×34 to size 17×17. The output generated by third layer is fed into forth convolutional layer. The fourth convolutional block has 256 filters of size 3×3 with ReLU function. The fourth convolutional neural network outputs 256 images of 15×15 size. A MaxPooling layer with 256 filters of size

2×2 is added after the third convolutional neural network, scaling down the 256 images from size 15×15 to size 7×7.

Every time when an image pass through a convolutional layer, the output is normally passed through an activation function. ReLU, is also known as a rectified linear unit, is widely used in convolutional neural network. According to eq. (1), the ReLU function returns the highest value between 0 and the supplied value. If the input value is negative, the output will be zero, and if it is positive, the output will be the same as the input value.

$$f(x) = \max(0, x)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

In the later stage flatten layer and two dense layers are applied. This dense layer makes the total number of network trainable parameters 3,602,506 in the coloured images. This fully connected layer calculates the class probability. Last fully connected layer (FC) mapping the 2570-array to a new array of length 10. Each has the ten different class probabilities of the input image. These probabilities are classified by the Softmax function as given in eq. (2). Dropout layer contains 60% dropout.

$$\text{softmax}(z_i) = \frac{\exp(z_i)}{\sum_j exp(z_j)}\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

Table 1. Convolutional Neural Network (CNN) Architecture

Layer	Output Shape	No. of parameters
Input Layer	(150, 150, 3)	0
Conv-1	(148,148,32)	896
MaxPool-1	(74, 74, 32)	0
Conv-2	(72,72,64)	18,496
MaxPool-2	(36, 36, 64)	0
Conv-3	(34, 34, 128)	73,856
MaxPool-3	(17, 17, 128)	0
Conv-4	(15, 15, 256)	2,95168
MaxPool-4	(7, 7, 256)	0
Flatten	12544	0
Dropout	12544	0
Dense-1	256	3211520
Dense-2	10	2570

In order to measure the performance of CNN model, cross-entropy loss function is used. When the convolutional neural network is initialized with random values then the loss function gives high value of loss function. The main objective of training CNN network is to reduce the loss function as low as possible. When the CNN gives a low value of loss function the model gives the higher accuracy.

$$Loss = -\sum_{c=1}^{M}(y_c - \log \hat{y}_c)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3)$$

Where M is the number of classes and $\hat{y}$ is the model's prediction for that class.

4. Results and Discussion

The proposed model was trained on Windows 10 operating system, “Intel® Core™ i5-8250U CPU @ 1.60GHz 1.80 GHz processor with 8 GB RAM” including NVIDIA GetForce MX130 GPU using Keras , TensorFlow and other libraries under Python environment.

TomConv, the suggested model, is compared with current state-of-the-art models. Table 2 presents this comparison. It is noteworthy that we have focused only on the diagnosis of diseases for the Tomato plant leaves and hence in Table 2 we have not compared the proposed model with other similar plant leave disease diagnosis research works (like for rice plant leaves), in order to prevent dilution of the comparison. Further, in order to have a fair comparison of the research works, we have considered only those research works in Table 2.

Table 2. Comparative Analysis of CNN model on tomato plant leaves disease

Authors	Size of dataset	No. of Classes	No. of Layers	CNN Model	Transfer Learning	Parameters (Millions)	Accuracy
Rangarajan et al. [6]	13,262	7	23	VGG16	Yes	138M	96.19%
Durmus et al. [7]	18,160	10	8	AlexNet	Yes	60M	95.65%
Tm et al. [12]	18,160	10	7	LeNet	No	60K	94-95%
Hong et. al. [15]	13,112	10	121	DensNet121_Xception	Yes	29.2M	97.10%
Shijie et al. [17]	7040	11	16	VGG16	Yes	138M	88.00%
Zhou et al. [19]	13,185	9	6	RRDN	No	NA	95.00%
Zhao et al. [20]	4585	10	NA	ResNet50+SeNet	Yes	27.5M	96.81%
Abbas et al. [21]	Synthetic Images	10	NA	DenseNet121	Yes	29M	97.11%
Proposed Work	**16,012**	**10**	**7**	**TomConv**	**No**	**3M**	**98.19%**

NA: Not Available

We have experimented with at least 5 classes and the dataset size is at least 4500. TomConv performs well and it is also the simplest form of the model. The term itself has been coined from the individual words Tomato (Tom) and Convolutional (Con) Neural Network.

The proposed model was trained under the parameters given in Table 3. Table 3 shows that the maximum epoch size is 105, the batch size is 32, the learning rate is 0.0001, the optimizer is the "Adam" optimizer, and the loss function is " categorical cross-entropy". The model was finally been tested using the NVIDIA GPU execution environment.

Table 3 Experimental parameters used for training the model

Parameter Name	Parameter Values
Batch Size	32
Optimizer	Adam
Learning rate	0.0001
Loss	Categorical Cross entropy
Epochs	105
Execution environment	NVIDIA GPU

To analyse the robustness of the proposed approach, The PlantVillage collection, which includes thousands of publicly accessible photos of healthy and diseased leaves from various plants was used in a series of experiments. The model has been trained for 105 iterations, and the accuracy and loss of the training and validation phases produced by the model are noted. This demonstrates the way that previous hidden layers transfer activation to the next layer neuron and how neurons are learned by the network. With a total of 105 training epochs, the Adam optimizer is used to train the model architecture. Table 4 displays experimental findings.

Table 4. Last 15 results of loss and accuracy recorded from105 epochs

Epochs	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
91	0.2005	0.9795	0.2222	0.9728
92	0.1797	0.9836	0.2052	0.9833
93	0.1901	0.9806	0.2580	0.9615
94	0.1848	0.9808	0.1849	0.9794
95	0.1873	0.9799	0.2335	0.9673
96	0.2087	0.9739	0.2785	0.9632
97	0.1899	0.9817	0.2141	0.9746
98	0.1760	0.9840	0.2442	0.9705
99	0.1720	0.9844	0.2259	0.9702
100	0.1976	0.9750	0.2774	0.9566
101	0.1903	0.9792	0.2989	0.9469
102	0.1751	0.9822	0.1873	0.9775
103	0.1578	0.9868	0.1910	0.9713
104	0.1750	0.9807	0.1819	0.9797
105	**0.1743**	**0.9819**	**0.2005**	**0.9757**

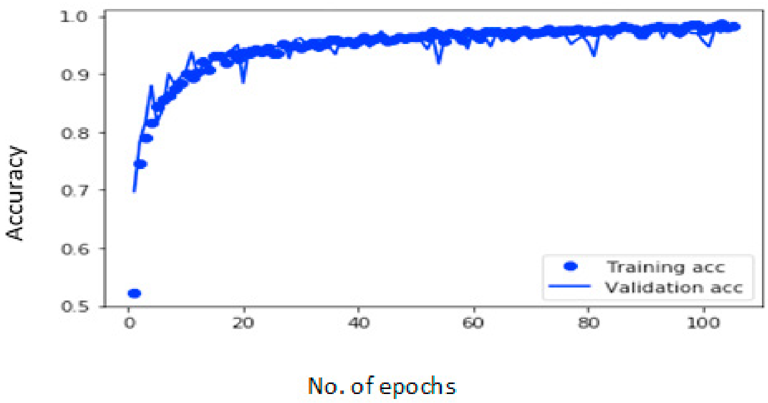

Fig. 3 Training and Validation Accuracy

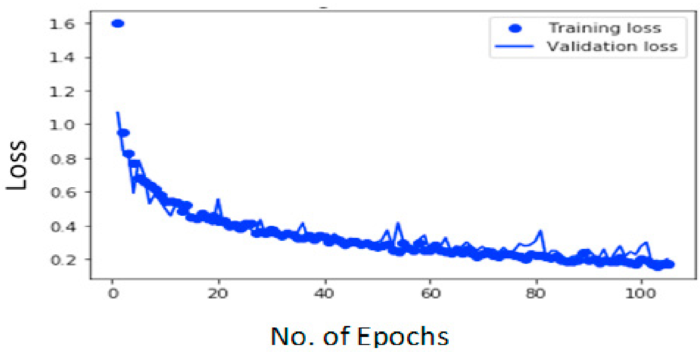

Fig. 4 Training and Validation loss

It is observed that validation accuracy is 97.57% which is acceptable compared to actual training set accuracy 98.19%. It is also clear that loss is reduced as shown in graph given in Fig. 4. The accuracy result of validation and training dataset is given in Fig 3.

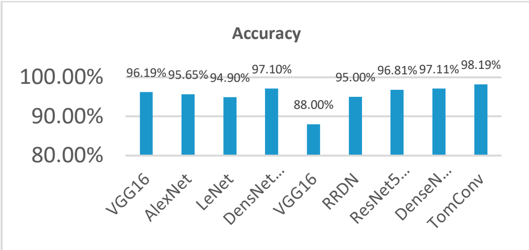

Fig. 5 Comparison with the state-of-art models with proposed model

The comparison with the state-of-art models with the proposed TomConv model is shown in the Figure 5. It is clear from the above figure that TomConv gives higher accuracy as compared to other models.

5. Conclusion, Limitations and Future Directions

The proposed TomConv model helps to identify diseases found on leaves and achieved an accuracy of 98.19% for training data of tomato plant leaves. In our research, we used pictures of healthy and diseased tomato plant leaves to construct a convolutional neural network model for disease identification. We utilized 16,012 images from PlantVillage dataset to train and test the mode. Ten distinct disease are included in this dataset. We were able to attain the best accuracy rate of 98.19% after splitting the dataset in 80:20 ratio. When compared to the current state-of-the-art models, the proposed model produces better outcomes. Moreover, the TomConv architecture is an optimized convolutional neural network that limits the parameter number and operations as much as possible. The proposed model is unique and presents an excellent contribution to the scientific community in multiple terms. Firstly, it uses a standard and publicly available dataset rather than devising one's own which many researchers do but is subject to bias. Also, the dataset is large compared to the datasets used by many other researchers. Additionally, unlike various other researchers, we did not make use of synthetic images. Secondly, it deals with as many as 10 classes. Finally, the reported accuracy of the proposed model on a large publicly available dataset is better than all the results reported by the researchers till date. The proposed model was trained for the disease diagnosis of the tomato plant leaves captured in a controlled environment. In the future, we will improve this model for the tomato plant leaves images captured even in an uncontrolled environment. Further, we will extend our work to detect disease from other parts of tomato plant like root, stem and flower.

References

[1] Vibhute A., and Bodhe S.K. (2012), “Applications of Image Processing in Agriculture: A Survey”, *International Journal of Computer Applications*, 52(2):34-40. doi: 10.5120/8176-1495

[2] Mutka A.M., and Bart R.S. (2015), “Image-Based Phenotyping of Plant Disease Symptoms”, *Frontiers in Plant Science*, 5(734):1-8. doi: 10.3389/fpls.2014.00734

[3] Mohanty S.P., Hughes D.P., and Salathe M. (2016), “Using Deep Learning for Image-Based Plant Disease Detection”, *Frontiers in Plant Science*, 7(1419):1-10. doi: 10.3389/fpls.2016.01419

[4] Atole R.R., and Park D. (2018), “A Multiclass Deep Convolutional Neural Network Classifier for Detection of Common Rice Plant Anomalies”, *International Journal of Advanced Computer Science and Applications*, 9(1):67-70. doi: 10.14569/IJACSA.2018.090109

[5] Dhakal A., and Shakya S. (2018), “Image-Based Plant Disease Detection with Deep Learning”, *International Journal of Computer Trends and Technology*, 61(1):26-29. doi: 10.14445/22312803/IJCTT-V61P105

[6] Rangarajan A.K., Purushothaman R., and Ramesh A. (2018), “Tomato crop disease classification using pre-trained deep learning algorithm”, proc. of International Conference on Robotics and Smart Manufacturing (RoSMa-2018), Procedia Computer Science,133:1040–1047. doi: 10.1016/j.procs.2018.07.070

[7] Durmuş H., Güneş E.O., and Kırcı M. (2017), "Disease detection on the leaves of the tomato plants by using deep learning", proc. of the 6th International Conference on Agro-Geoinformatics, Fairfax, pp. 1-5. doi: 10.1109/Agro-Geoinformatics.8047016

[8] Brahimi M., Boukhalfa K., and Moussaoui A. (2017), "Deep learning for tomato diseases: classification and symptoms visualization", Applied Artificial Intelligence, 31:299-315.

[9] Too E.C., Njuki S., Yingchun L., and Yujian L. (2018), "A comparative study of fine-tuning deep learning models for plant disease identification", Computers and Electronics in Agriculture", 161:272-279. doi: 10.1016/j.compag.2018.03.032

[10] Foysal M.F.A., Islam M.S., Abujar S., and Hossain S.A. (2020), "A novel approach for tomato diseases classification based on deep convolutional neural networks", proc. of the International Joint Conference on Computational Intelligence, pp. 583-591. doi: 10.1007/978-981-13-7564-4_49

[11] Sabrol H., and Satish K. (2016), "Tomato plant disease classification in digital images using classification tree", proc. of the International Conference on Communication and Signal Processing (ICCSP-2016), pp. 1242-1246. doi: 10.1109/ICCSP.2016.7754351

[12] Tm P., Pranathi A., SaiAshritha K., Chittaragi N.B., and Koolagudi S.G. (2018), "Tomato Leaf Disease Detection Using Convolutional Neural Networks", proc. of the Eleventh International Conference on Contemporary Computing (IC3), pp. 1-5. doi: 10.1109/IC3.2018.8530532

[13] Yang Lu, Yi S., Zeng N., Liu Y., and Zhang Y. (2017), "Identification of rice diseases using deep convolutional neural networks", Neurocomputing, 267(6):378-384. doi: 10.1016/j.neucom.2017.06.023

[14] Balakrishna K., and Rao M. (2019), "Tomato Plant Leaves Disease Classification Using KNN and PNN", International Journal of Computer Vision and Image Processing, 9(1):51-63. doi:10.4018/IJCVIP.2019010104

[15] Hong H., Lin J., and Huang F. (2020), "Tomato Disease Detection and Classification by Deep Learning", proc. of the International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE-2020), pp. 25-29. doi: 10.1109/ICBAIE49996.2020.00012.

[16] Ouhami M., Es-Saady Y., Hajji M.E., Hafiane A., Canals R., and Yassa M.E. (2020), "Deep Transfer Learning Models for Tomato Disease Detection", proc. of the International Conference on Image and Signal Processing (ICISP-2020), 12119. doi: 10.1007/978-3-030-51935-3_7

[17] Shijie J., Peiyi J., Siping H., and Haibo L. (2017), "Automatic detection of tomato disease and pests based on leaf images", Chinese Automation Congress, pp. 3507-3510. doi: 10.1109/CAC.2017.8243388

[18] Nandhini S., and Ashokkumar K. (2021), "Improved crossover based monarch butterfly optimization for tomato leaf disease classification using convolutional neural network", Multimedia Tools and Applications, 80:18583–18610. doi: 10.1007/s11042-021-10599-4

[19] Zhou C., Zhou S., Xing J., and Song J. (2021), "Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network", IEEE Access, 9:28822-28831. doi: 10.1109/ACCESS.2021.3058947

[20] Zhao S., Peng Y., Liu J., and Wu S. (2021), "Tomato Leaf Disease Diagnosis Based on Improved Convolution Neural Network by Attention Module", Agriculture, 11(651):1-15. doi: 10.3390/agriculture

[21] Abbas A., Jain S., Gour M., and Vankudothu S. (2021), "Tomato plant disease detection using transfer learning with C-GAN synthetic images", Computers and Electronics in Agriculture, 187. doi: 10.1016/j.compag.2021.106279

[22] Olowe O.M., Nicola L., Asemoloye M.D., Akanmu A.O., and Babalola O.O. (2022), "Trichoderma: Potential bio-resource for the management of tomato root rot diseases in Africa", Microbiological Research, 257. doi: 10.1016/j.micres.2022.126978

[23] Shalaby T.A., Taha N.A., Taher D.I., Metwaly M.M., El-Beltagi H.S., Rezk A.A., El-Ganainy S.M., et al. (2022), "Paclobutrazol Improves the Quality of Tomato Seedlings to Be Resistant to Alternaria solani Blight Disease: Biochemical and Histological Perspectives", Plants, 11(3), 425. doi: 10.3390/plants11030425

[24] Huang M., Chuang T., and Liao Y. (2022), "Application of transfer learning and image augmentation technology for tomato pest identification", Sustainable Computing: Informatics and Systems, 33. doi: 10.1016/j.suscom.2021.100646

[25] Belouz K., Nourani A., Zereg S., and Bencheikh A. (2022), "Prediction of greenhouse tomato yield using artificial neural networks combined with sensitivity analysis", Scientia Horticulturae, 293. doi: 10.1016/j.scienta.2021.110666