IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 2, Issue 3, November 2022

Classification of Tomato Plant Diseases using Deep Learning Based Classifier from Images of Leaves

Shivraj Rahane[1], **Shubham Rahane**[2], **Bhushan Patole**[3], **Tejaswini Shelke**[4],
Dr. Sunil khatal[5], **Prof. Mundhe B. B.**[6]
Students, Department of Computer Engineering[1,2,3,4]
Professor, Department of Computer Engineering[5,6]
Sahyadri Valley College of Engineering & Technology, Rajuri, Junnar, Pune, Maharashtra, India[1,2,3,4,6]
Sharadchandra Pawar College of Engineering, Pune, Maharashtra, India[5]

Abstract: *Crop diseases cost farmers a lot of money every year in the majority of agricultural nations. Therefore, early detection of tomato plant diseases has become a major research focus. In this study, a deep convolutional neural network model is utilised to distinguish between diseased and healthy plants and to categorise the diseases that affect tomato plants. To identify ill plants and their diseases from photos of tomato plants, we employed the VGG16 deep CNN classifier. We used the Plant Village dataset, which includes healthy plants and ten different classes of tomato leaf photos. With a pre-trained VGG16 model and the transfer learning approach, this dataset exhibits a satisfying classification performance of roughly 95.5%. This model's top 2 accuracy for identifying illnesses in tomato plants is 99%. Our trained model exhibits a performance of about 100% to distinguish ill plants from healthy plants without utilising any segmentation or pre-processing of leaf pictures.*

Keywords: Transfer Learning, Tomato Disease, and Deep Convolutional Neural Network

I. INTRODUCTION

Most people in Bangladesh rely on agriculture and farming for their livelihood. Because of the local climate, tomatoes have grown to be a significant crop and part of our economy. Early diagnosis of these illnesses may enable individuals to take precautions and lessen their loss. The market demand for tomatoes and their nutritional worth make them an essential crop. Because of the taste and nutritional benefits of tomatoes, there is concern. Vitamin E, Vitamin C, and Beta Carotene are all abundant in tomatoes. These antioxidants are crucial for maintaining good health. This widely grown crop is vulnerable to insects and pests that cause various diseases in tomato plants. To manually identify tomato plant illnesses, you must be familiar with them. Because not everyone is aware of every sign of tomato plant illnesses, this identification has certain limitations. Farmers face significant challenges every year in order to produce healthy harvests. The production line is weakened by bug and pest attacks, which poses a serious threat to our economy and, in particular, to our farmers. Although farmers employ a lot of insecticides and pesticides to protect tomato plants from diseases, they frequently fail to do so because they are unaware of the conditions that cause the disease and how to prevent it. Again, overuse of insecticides and pesticides is detrimental to our existence and way of life. Crop damage can also result from misreading disease symptoms and using insecticides in excess or insufficiently. Therefore, in order to produce tomatoes at a sufficient level, it is crucial and required to recognise tomato plains infections. However, farmers are short on professionals who can keep an eye on the plants, spot diseases, and take preventive measures like applying a certain amount of insecticides, among other things. It takes a lot of time and energy to manually identify tomato illnesses while checking the plants frequently. Again, there are occasions when farmers find it challenging to consult experts in other cities and to take precautions against rare diseases. Once more, using pesticides and insecticides excessively is harmful to our survival and way of life. Misreading disease symptoms and spraying insecticides too much or too little can also cause crop harm. Therefore, it is essential and required to recognise tomato plan is infections in order to produce tomatoes at an adequate level. However, farmers lack qualified individuals who can monitor the plants, detect diseases, and take preventive actions like, for example, spraying a specific amount of insecticides. To manually detect tomato diseases when examining the plants frequently requires a lot of time and effort. Again, there are times when farmers find it difficult to speak with specialists in distant cities and to take preventative measures against

DOI: 10.48175/IJARSCT-7635

unusual diseases. Recent studies concentrate on the textural characteristics of tomato leaves with defects. Because disease attacks alter the colour, shape, and smoothness of the leaves, using the leaves to identify sick tomato plants will be more successful. To identify tomato plant illnesses, Vetal [1] et al. suggested a multi class Support Vector Machine (SVM). Each class had 80 photos from which features could be extracted and trained into a multi-class SVM classifier. These illnesses affected four different tomato plant varieties. They used a variety of pre-processing techniques to improve classification results. To effectively extract texture information from images, grayscale images were transformed to HSI format. They used kurtosis filters to smooth the images and median filters to decrease noise. On the basis of hue, the defective portions of leaf images were segmented. All of these pre-processing stages ready the dataset for features extraction and SVM training. They evaluated their classification performance against Artificial Neural Network, K Nearest Neighbor, Probabilistic Neural Network, etc., and were able to identify four different tomato plant illnesses with a classification accuracy of 93%.Two distinct methods were suggested by Budihal [2] et al. to identify fungus and spot disease in tomato plants. The performance of spotting spots in binary images was hampered by leaf veins. This is why they converted RGB photos into HSI images, which reduced noise brought on by the presence of leaf veins. To accurately identify spots from leaf damage, they used otsu threshold. To apply a threshold value, they concentrated on the I value (intensity of light amplitude) of the HSI form. Some parameters were examined to check the intensity of the fungus attack in tomato leaves in order to detect fungus assault. The position of the fungus was also discovered using the segmentation method. The significant drawback of this research was that only spot or fungal attacks could be detected using this technique. Beyond fungus and spot attacks, other leaves were identified using this method as healthy leaves regardless of whether they had any other disorders. To identify tomato plant illnesses, Prajwala [3] et al. suggested using a deep convolutional neural network as a classifier. They employed a LeNet model that has fully linked, pooling, and simple convolution layers. The training photos were downsized in the pre-processing stage for the following training process. For false positive samples and class unbalanced data, Alvaro [4] et al. presented a three unit tomato disease identification procedure. Faster RCNN was employed to discover affected areas with bounding box predictions in a refinement filter bank framework, according to their discussion. Each bounding box included a forecast of the infected area's location and a confidence score indicating the disease's class. The second unit, the CNN filter bank, was utilised to check if the samples were correctly classified as part of the classification task. The false positive samples were finally isolated by the final integration unit. This algorithm performed well in terms of classification, however the CNN Filter Bank's complexity made recognition take longer. To classify tomato plant illnesses, we have suggested using a deep CNN pre-trained model. This method does not require the use of noise reduction or noise filtering techniques, and its classification accuracy is nearly 100%.

II. METHODOLGY

2.1 Text Font Deep Convolutional Neural Network

Using deep learning, the Deep Convolutional Neural Network (DCNN) is a classifier. When given images as input, a deep CNN adjusts weights and biases based on different objects so that the model can identify those classes. This classifier stands out from others in that it involves far less image preprocessing than other classifiers. A deep CNN often has the capacity to teach and train model features.

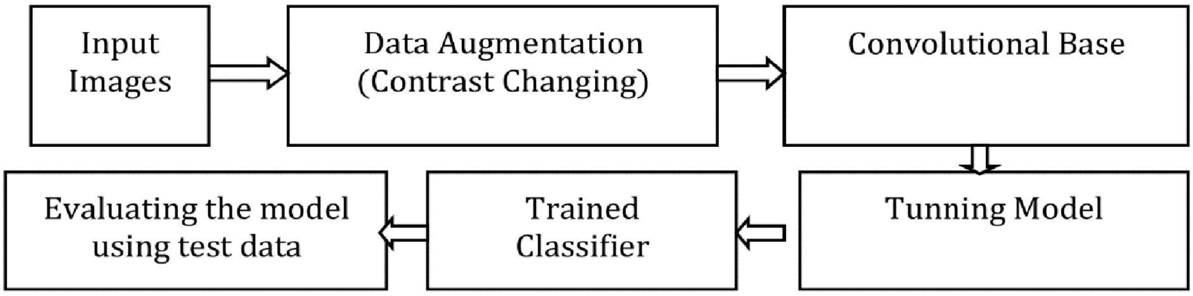
Images' spatial and temporal features are extracted using convolutional neural networks. CNN multiplies the input picture values with bias and weights using a linear calculation. The input images are run through a series of filters by a CNN, and the filtered results are then sent to an activation function. During model training, numerous convolution filters are applied at a defined number of epochs, and the combined results produce an activation map. This is referred to as a feature map, which truly distinguishes between distinct classes to be recognised. In the fully connected layer, each of these retrieved features is sent via a nonlinear activation function called Relu. The output is transformed into a one-dimensional vector that is assigned to the number of classes to be classified by a fully connected layer. A pooling layer is an additional component of a convolutional neural network. The size of the convolved features is decreased by the pooling layer, lowering computational expenses. A decrease in learnable features aids the network in extracting dominating characteristics from the feature map since convolution into deeper layers creates additional parameters to be taught..

DOI: 10.48175/IJARSCT-7635

2.2 VGG16 Architecture

VGG16 is a classifier that uses a deep learning method and includes 16 convolutional layers in its network. This deep CNN model, which Simonyan [5] et al. suggested, outperformed the AlexNet model. The improvement was carried out by switching the convolution filters' type. VGG16 uses multiple smaller convolution filters, such as 33, in place of kernel size convolution filters like 1111 or 55. The capacity to learn complicated features from training data is increased by using smaller kernel stack filters.

For network training, the fixed size 224224 pictures used by VGG16 are supplied. In order to make the input feature linear, the model first performs convolution using a number of 33 filters, followed by a convolution of 11 filters. In order to maintain the same spatial resolution as the input data, the model fixes the stride value to 1 and adds spatial padding to the intermediate output. Max pooling layers function with a 2 stride. These all map to fully connected layers as output characteristics. Our relevant data from the tomato disease recognition task are used to train a pre-trained VGG16 model because we have suggested adopting a transfer learning strategy. The output of the completely connected layers is translated to these 10 separate classes because we want to categorise 10 different classes. The model is being fine-tuned in this manner. A deep CNN model's lowest layers learn low level features that are problem-dependent and non-specific. Instead, we use our work dataset to train the deeper layers because they learn more specialised and problem-dependent features. In order for the early layers to initialise weights from the prior pre-trained knowledge, we freeze the initial layers. Overfitting the model is also decreased by freezing the first layers. Incorrect weight initialization during training causes the model to perform poorly or fails to demonstrate a generalised model on unknown data. This issue can be resolved using a pretrained model by initialising weights from the trained model. Finally, the enhanced data is ready to train the adjusted model, and the trained model will be able to identify various diseases of tomato plants.



2.3 Dataset

(a) Target Spot (b) Yellow leaf (c) Mosaic Virus (d) Bacterial Spot

(e) Early Blight (f) Leaf Mold (g) Late Blight

(h) Septoria Leaf spot (i) Spider Mites (j) Healthy Leaf

Figure 1: Different types of Tomate diseased leaf and healthy leaf

We made use of the tomato leaf photos from the Plant Village dataset's ten different classes. There are between 1500 and 3000 photos in each class. We employed the data augmentation technique to balance this unbalanced dataset. To balance this dataset, we have added to and increased the number of sample data each class as necessary. The sample data's contrast is changed to enhance the data. Target Spot, Yellow Leaf, Mosaic Virus, Bacterial Spot, Early Blight, Leaf Mold, Late Blight, Septoria Leaf Spot, Spider Mites, and Healthy Leaf are the 10 classes that will be identified through our effort.

III. RESULTS OF EXPERIMENTS

We have classified many tomato leaf diseases using a pre-trained VGG16 model. We employed hold out cross validation to separate the training, validation, and test data in order to produce a generalised model. A total of 60% of the data is used for training, 20% is utilised to assess the accuracy of the validation, and the remaining 20% is the test set. As a result of our usage of the transfer learning approach, the model is started with the weights from prior training, and as new characteristics are learned and applied to specific tasks, these weights are updated.

- Model : VGG16
- Input Image Size: 224×224
- Optimizer: Adam Learning
- rate: 0.0001
- Dropout: 0.05
- No of Classes: 10

Learning rate during training period represents the speed at which the model will reach a certain training and validation accuracy. If the learning rate is sufficiently high, the model becomes saturated early in the training process. Dropout reduces the number of training parameters. We need to correctly classify tomato illnesses on samples outside of the training set in order to develop a generalised model. Dropout layer subtracts a fraction of learnable features, enhancing the ability to efficiently learn complex features.

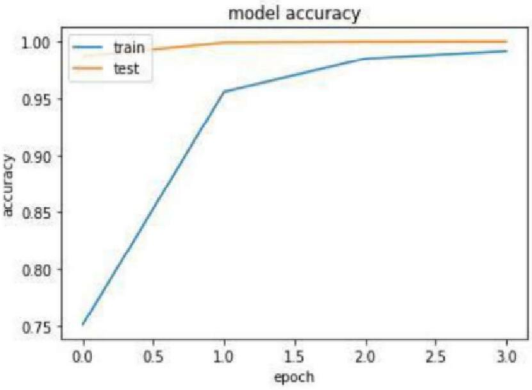


Figure 2: Training VS Validation Accuracy of VGG16

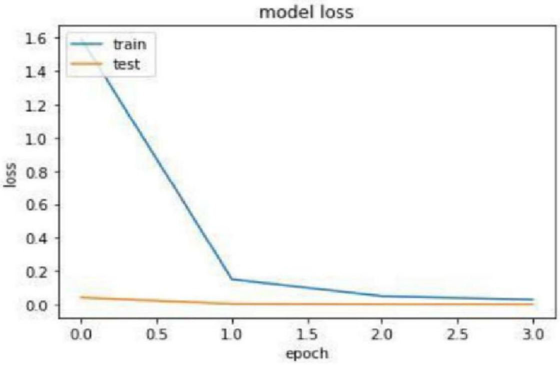


Figure 3: Training VS Validation Loss of VGG16

The dropout layer stops the model from fitting too tightly. Whether the model is generalising or not is determined by the validation accuracy and training accuracy. In Figure 2, the model's validation accuracy is greater than its training accuracy. How well the model can recognise the training samples at the current training period is determined by training accuracy. The model then categorises the validation set using the training phase as a basis. The model is warned not to become overfitted when the validation accuracy is higher than the training accuracy. Figure 3 shows that both training and validation loss reduce as the number of training epochs increases. For this model, validation loss is always lower than training loss. Figure 4 shows how the model performs when performing this categorization task. This trained classifier displays successful categorization on the test set. Only the "Septoria Leaf Spot" class displays an accuracy below 90% at around 95.5%. The majority of the test samples are in diagonal positions, which shows that the model correctly predicts the class to which they actually belong. Because septoria leaf spot illness has both brown spots like bacterial spot and yellow tints like yellow leaf, it conflicts more with other classes and reduces the accuracy of classification. The Mosaic virus can be identified with 100% accuracy.

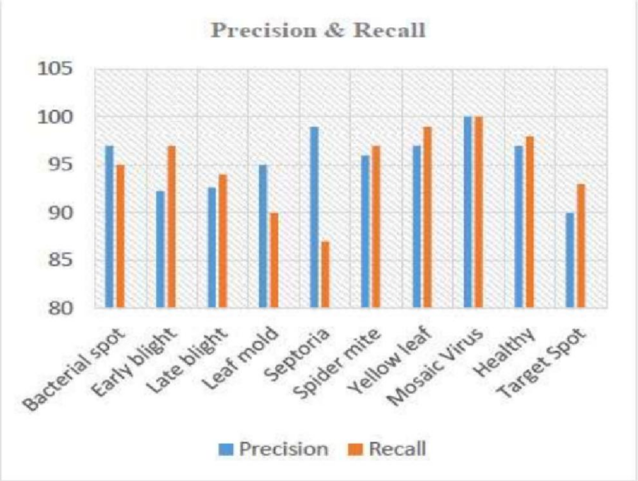
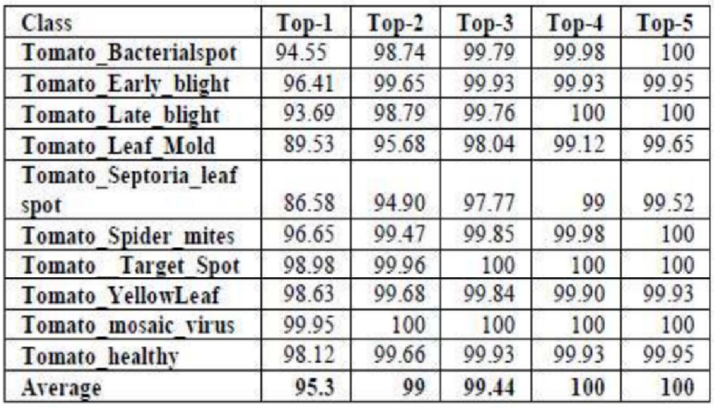


Figure 4: Precision And Recall For Tomato Disease classification.

TABLE I. TOP-1 TO TOP-5 ACCURACY



Class	Top-1	Top-2	Top-3	Top-4	Top-5
Tomato_Bacterialspot	94.55	98.74	99.79	99.98	100
Tomato_Early_blight	96.41	99.65	99.93	99.93	99.95
Tomato_Late_blight	93.69	98.79	99.76	100	100
Tomato_Leaf_Mold	89.53	95.68	98.04	99.12	99.65
Tomato_Septoria_leaf spot	86.58	94.90	97.77	99	99.52
Tomato_Spider_mites	96.65	99.47	99.85	99.98	100
Tomato__Target_Spot	98.98	99.96	100	100	100
Tomato_YellowLeaf	98.63	99.68	99.84	99.90	99.93
Tomato_mosaic_virus	99.95	100	100	100	100
Tomato_healthy	98.12	99.66	99.93	99.93	99.95
Average	**95.3**	**99**	**99.44**	**100**	**100**

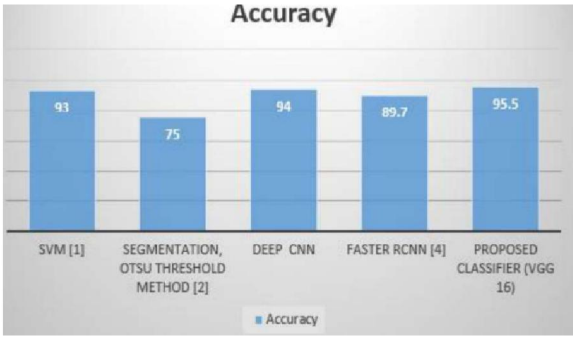


Figure 5: Comparison Among the related works with VGG16

Precision and recall for each class are evaluated using test data in Fig. 5. With an average precision value of 95.6% and an average recall value of 95%, the model produces a strong classification result. Table I lists the model's accuracy from top-1 to top-5. Top-n accuracy denotes that the confidence score for the actual class label is among the top n accuracy values in a sorted confidence score. Top-2 accuracy of our model across the plant village dataset reaches 99%, which is highly encouraging in the diagnosis of tomato illnesses. Farmers will benefit from an estimation of the appropriate illness because this recognition task will allow them to choose the best insecticide or pesticide for their crop. A farmer will be helpful if he is informed about the closest (based on the confidence score) predicted disease in the event that any disease was incorrectly predicted. In table I, Top-2 accuracy has a binary classification accuracy of approximately 100% for whether the plant is healthy or not. In Fig. 6. Our model has been contrasted with those of other authors. Performance-wise, VGG 16 outperforms all of these earlier works. Our trained algorithm can accurately and successfully discriminate between healthy and ill plants (approximately).


(a) Yellow leaf (b) Target Spot (c) Misclassified


(e) Early Blight (f) Late Blight (g) Misclassified

Figure 6: Some examples of misclassified test data.

The model incorrectly identified Fig. 7(c) as a yellow leaf for curl virus. This specimen is a leaf that has been impacted by septorial leaf spots. Another sample incorrectly identified a septoria leaf spot as a target spot-affected leaf. Both of these two symptoms are present in septoria leaf spot because the yellow leaf is predominant, with a yellow shade within the leaf, and the target spot has brown dark spots at a specific affected location. This raises the Septoria leaf spot misclassification rate. In addition, (g) contains a sample of the leaf mould disease, which some test data incorrectly anticipated to be either early blight or late blight. Reshaping or deformation of leaf shape happens in the early blight, late blight, and leaf mould classes. Additionally, this results in some misclassification. Without any preprocessing, our trained VGG16 model outperforms other classifiers significantly. Instead of having a lot of noisy data and inter-class similarities between several classes, this model accurately recognises tomato plant illnesses.

IV. CONCLUSION

Farmers in the majority of agricultural nations struggle to offer preventive measures in the event of a fungal or bacterial attack on crops or other diseases. A significant portion of the annual loss in agriculture will be reduced if they are given the necessary diagnosis system. In order to identify ill tomato plants and categorise the specific disease, we trained a deep CNN model. Because of a lack of background knowledge, visual observation to identify diseases may be inaccurate. The farmers will be assisted in solving this by a trained classifier. The categorization task becomes challenging because there are so many similarities between classes. We discovered a trained model to classify tomato plant illnesses with a good level of performance using the transfer learning technique in VGG16

ACKNOWLEDGEMENT

The Green University of Bangladesh has contributed to the cost of this study.

REFERENCES

[1]. S. Vetal, R.S. Khule, "Tomato Plant Disease Detection using Image Processing", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 6, Issue 6, June 2017.

[2]. S. Budihal, R. Sandhya, S. D Hajawagol, S. R. Navi, "Detection Of Disease In Tomato Leaf". International Journal of Advanced Computer Engineering and Communication Technology (IJACECT). ISSN (Print): 2278-5140, Volume-4, Issue-1, 2015.

[3]. P. TM, A. Pranathi, K. S. Ashritha, N. B. Chittaragi, S. G. Koolagudi, "Tomato Leaf Disease Detection Using Convolutional Neural Networks", Eleventh International Conference on Contemporary Computing (IC3), 2-4 August, 2018, Noida, India.

[4]. A. F. Fuentes, S. Yoon, J. Lee and D. S. Park, "High Performance Deep Neural Network Based Tomato Plant Diseases And Pest Diagnosis System With Refinement Filter Bank", Frontiers in Plant Science, Volume 9, Article 1162, August 2018.

[5]. K. Simonyan, A. Zisserman, "Very deep convolutional networks for large scale image recognition". International Conference on Learning Representations (ICLR), 2015.

[6]. K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[7]. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, "Rethinking the inception architecture for computer vision, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[8]. C. Szegedy, V. Vanhoucke, S. Ioffe, "Inception-v4, Inception- ResNet and the Impact of Residual Connections on Learning", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[9]. F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[10]. C. Zhang, P. Zhou, C. Li, L. Liu, "A convolutional neural network for leaves recognition using data augmentation", IEEE International Conference on Computer and Information Technology, 2015.

[11]. W. Jeon and S. Rhee, "Plant leaf recognition using a convolution neural network", International Journal of Fuzzy Logic and Intelligent Systems Vol. 17, No. 1, 2017.

[12]. J. Amara, B. Bouaziz, A. Algergawy, "A Deep Learning- based Approach for Banana Leaf Diseases Classification." In: BTW (Workshops). 2017, pp. 79-88.