Tomato Leaf Disease Detection using Convolutional Neural Networks

Prajwala TM, Alla Pranathi, Kandiraju Sai Ashritha, Nagaratna B. Chittaragi*, Shashidhar G. Koolagudi

Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal

Email: {prajwala.tm, pakasarka, ashritha1615, nbchittaragi}@gmail.com, koolagudi@nitk.ac.in

Abstract—The tomato crop is an important staple in the Indian market with high commercial value and is produced in large quantities. Diseases are detrimental to the plant's health which in turn affects its growth. To ensure minimal losses to the cultivated crop, it is crucial to supervise its growth. There are numerous types of tomato diseases that target the crop's leaf at an alarming rate. This paper adopts a slight variation of the convolutional neural network model called LeNet to detect and identify diseases in tomato leaves. The main aim of the proposed work is to find a solution to the problem of tomato leaf disease detection using the simplest approach while making use of minimal computing resources to achieve results comparable to state of the art techniques. Neural network models employ automatic feature extraction to aid in the classification of the input image into respective disease classes. This proposed system has achieved an average accuracy of 94-95% indicating the feasibility of the neural network approach even under unfavourable conditions.

Keywords—leaf disease detection, neural network, convolution, LeNet

I. Introduction

India is a country with a majority of the population relying heavily on the agricultural sector. Tomato is the most common vegetable used across India. The three most important antioxidants namely vitamin E, vitamin C and beta-carotene are present in tomatoes. They are also rich in potassium, a very important mineral for good health. Tomato crop cultivation area in India spans around 3,50,000 hectares approximately and the production quantities roughly sum up to 53,00,000 tons, making India the third largest tomato producer in the world. The sensitivity of crops coupled with climatic conditions have made diseases common in the tomato crop during all the stages of its growth. Disease affected plants constitute 10-30% of the total crop loss. Identification of such diseases in the plant is very important in preventing any heavy losses in yield as well as the quantity of the agricultural product. Monitoring the plant diseases manually is a difficult task due to its complex nature and is a time consuming process. Therefore, there is a need to reduce the manual effort put into this task, while making accurate predictions and ensuring that the farmers' lives are hassle free.

Visually observable patterns are difficult to decipher at a single glance, leading to many farmers making inaccurate assumptions regarding the disease. As a result, prevention mechanisms taken by the farmers may be ineffective and sometimes harmful. Farmers usually come together and implement common disease prevention mechanisms, as they lack expert advice on how to deal with their crop infestation [2]. There has been circumstances where due to inadequate knowledge or misinterpretation regarding the intensity of the disease, over-dosage or under-dosage of the pesticide has resulted in crop damage. This is the underlying motivation for the proposed methodology that aims to accurately detect and classify diseases in the tomato crop.

The methodology suggested in the paper pertains to the most common diseases found in the tomato plant like, Bacterial leaf spot and Septorial leaf spot, Yellow Leaf Curl among many others. Any leaf image given as input can be classified into one of the disease classes or can be deemed healthy. The database used for evaluation is a subset of Plant Village [6], a repository that contains 54,306 images of 14 crops infested with 26 diseases. The subset includes around 18160 images of tomato leaf diseases.

Broadly, the proposed methodology consists of three major steps: data acquisition, pre-processing and classification. The images used for the implementation of the proposed methodology were acquired from a publicly available dataset called Plant Village, as mentioned earlier. In the next step, the images were re-sized to a standard size before feeding it into the classification model. The final step is the classification of the input images with the use of a slight variation of the deep learning convolutional neural network (CNN) standard model called the LeNet which consists of the convolutional, activation, pooling and fully connected layers.

The paper is organized as follows: Section II focuses on the prominent work done in regard to the concerned field. Section III elucidates the proposed methodology and the model used along with the steps taken to obtain the necessary results. Section IV pertains to the results and the analysis of the proposed methodology. Section V includes the conclusion of the paper and provides the scope for future work.

II. Literature Survey

It is important to recognize the previous research done in regard to this field to be able to correctly advance in the right direction. Plant leaf disease detection has been a major research area in which both image processing and deep learning techniques have been widely used for its accurate classification. In this paper, we discuss the most popularly incorporated techniques in literature in the relevant field. Two common tomato plant diseases look like the ones shown in

---

*Dept. of Information Science and Engg., Siddaganga Institute of Technology, Tumkur

Fig. 1 and Fig. 2 and healthy tomato leaves are shown in Fig. 3.
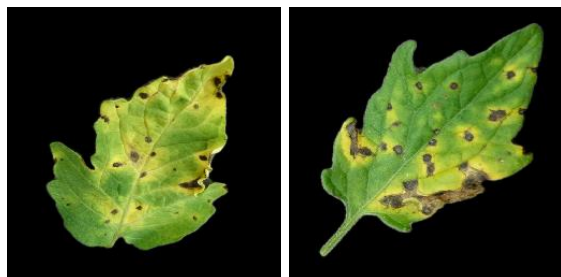

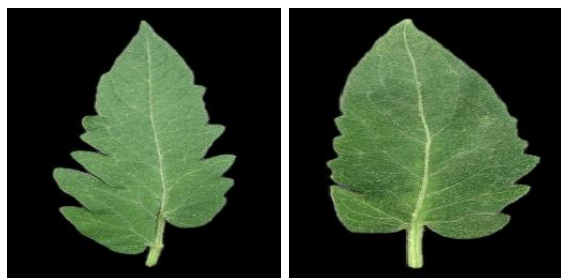Figure 1: Septoria leaf spot

Figure 2: Yellow Leaf Curl

Figure 3: Healthy

Monitoring a large field of crops is a tedious task, if done manually. It is necessary to minimize the human effort put into plant supervision. Hence this is a popular research domain attracting many researchers. Several works related to plant diseases are observed in literature.

The authors of the paper [7] have proposed an efficient method that identifies whether a tomato leaf is healthy or infected. The image given as input was first pre-processed by removing the background and the noise present was eliminated with the help of erosion technique. Gray Level Co-occurrence Matrix (GLCM) was used for texture feature extraction from the enhanced image. Support Vector Machine (SVM) classifier was trained using different kernel functions and the performance has been evaluated using N-fold cross-validation technique. The proposed system has achieved an accuracy of 99.83% using the linear kernel function with the SVM classifier. Even though the obtained accuracy is high, it is not sufficient enough to predict or differentiate between healthy or diseased leaves. Also, the type of disease was not identified.

In order to overcome the problem of the above paper, the authors in [3] have proposed various segmentation, feature extraction and classification techniques that identify and detect the type of the disease using the diseased image to conduct classification. The leaf image given as input to the system was pre-processed by smoothing it or enhancing the image by performing histogram equalization. To obtain the affected area, different segmentation techniques like K-Means clustering have been proposed. The features were then extracted from the segmented region and calculated using GLCM. After feature extraction, the diseases can be detected with the help of Artificial Neural Networks (ANN) or Back Propagation Neural Networks. The drawback of segmenting the image using K-Means clustering is that the process proposed was semi-automated as the user has to explicitly select the cluster which contains the diseased part.

The paper [8] describes a method which makes use of the Gabor wavelet transformation technique for the purpose of feature extraction which helps in the disease identification of tomato leaves. The extracted features were input to the SVM classifier for training which then determines the type of disease of the infected tomato leaf. Resizing of the images, elimination of noise and background removal have been carried out in the pre-processing stage. The paper has made use of Gabor transformation to identify the textual patterns of the affected leaf and extract appropriate features. Disease classification was carried out using SVM with different kernel functions and performance has been evaluated using cross-validation technique. An accuracy of 99.5% has been shown to have achieved according to the experimental results of the system proposed. The main disadvantage of using Gabor transformation for feature extraction is that it is computationally intensive.

In [9], the authors have used a simple approach for the classification of the diseased tomato leaves into various classes namely Tomato late blight, Septoria spot, Bacterial spot, Bacterial canker, Tomato leaf curl and Healthy. A dataset of 383 images which have been captured using a digital camera has been used for the purpose of implementation. Otsu's method for image segmentation has been applied on the dataset. Color features have been obtained using the RGB color components while shape features have been obtained using *regionprops* function and texture features have been obtained from GLCM. All the extracted features have been combined to form a feature extraction module. Supervised learning techniques have been used for classification by training the decision tree classifier. Though the accuracy is high, decision tree has its own set of disadvantages – over fitting in case of noisy data and the amount of control that the user has over the model is relatively less.

Deep convolutional neural networks have been trained in [6] for the identification of 26 diseases in 14 different crop species. The authors make use of the standard AlexNet [4] and GoogleNet [10] architectures for this purpose. A public repository which contains 54,306 images of both diseased leaves and healthy plant leaves has been used for this purpose. The dataset has been created by collecting the images of the plant leaves in a controlled environment.

The authors have conducted a performance analysis on both these architectures by carrying out the model training in two ways. It is performed from scratch in the first case and by using transfer learning in the second. Transfer learning corresponds to the process of adapting pre-trained weights obtained by training models on the ImageNet dataset. The model implementation has been carried out using the Caffe framework giving an accuracy of 99%. This portrays the feasibility of this approach. However, on testing the trained model against a set of sample test images obtained from online public data sources which are quite different from the train set, the model accuracy falls to 31.4%. This is a common problem faced in neural networks owed to the train and test sets belonging to different distributions.

The authors of [1] propose an approach where they detect and classify banana leaf diseases namely Banana sigatoka and Banana speckle. They have performed the training of deep learning models under certain challenging conditions. These conditions comprise of illumination, complex background, different images resolution, size and orientation. They effectively demonstrate the accuracy of this approach and the very less computational efforts required.

III. PROPOSED METHODOLOGY

The proposed approach includes the three important stages namely: Data Acquisition, Data pre-processing and Classification. Flow diagram is shown in Fig. 4 and current section includes the brief discussions of the same.
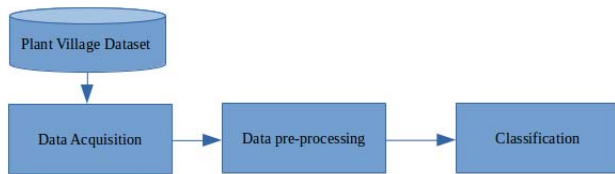

Figure 4: Proposed methodology

A. Data Acquisition

The tomato leaf disease images have been taken from the Plant Village repository [5]. Images for the diseases were downloaded using a python script. The acquired dataset consists of around 18160 images belonging to 10 different classes. The dataset includes images of all major kinds of leaf diseases that could affect the tomato crop. Each of the downloaded images belongs to the RGB color space by default and were stored in the uncompressed JPG format.

B. Data pre-processing

The acquired dataset consisted of images with minimal noise and hence noise removal was not a necessary pre-processing step. The images in the dataset were resized to 60×60 resolution in order to speed up the training process and make the model training computationally feasible.
The process of standardizing either the input or target variables tends to speed up the training process. This is done through

improvement of the numerical condition of the optimization problem. It is also made sure that the several default values involved in initialization and termination are appropriate. For our purpose, we normalize the images to get all the pixel values in the same range by using the mean and the standard deviation. In machine learning terms, it is called as the Z-score.

C. Classification

Convolutional neural networks (CNN) can be used for the creation of a computational model that works on the unstructured image inputs and converts them to corresponding classification output labels. They belong to the category of multi-layer neural networks which can be trained to learn the required features for classification purposes. They require less pre-processing in comparison to traditional approaches and perform automatic feature extraction which gives better performance. For the purpose of tomato leaf disease detection, we have experimented with several standard deep learning architectures like AlexNet [4], GoogleNet [10] and the best results could be seen with the use of a variation of the LeNet architecture [5].

LeNet is a simple CNN model that consists of convolutional, activation, pooling and fully connected layers. The architecture used for the classification of the tomato leaf diseases is a variation of the LeNet model. It consists of an additional block of convolutional, activation and pooling layers in comparison to the original LeNet architecture. The model used in this paper been shown in Fig. 5.

Each block consists of a convolutional, activation and a max pooling layer. Three such blocks followed by fully connected layers and softmax activation are used in this architecture. Convolutional and pooling layers are used for feature extraction whereas the fully connected layers are used for classification. Activation layers are used for introducing non-linearity into the network.

Convolutional layer applies convolution operation for extraction of features. With the increase in depth, the complexity of the extracted features increases. The size of the filter is fixed to 5×5 whereas number of filters is increased progressively as we move from one block to another. The number of filters is 20 in the first convolutional block while it is increased to 50 in the second and 80 in the third. This increase in the number of filters is necessary to compensate for the reduction in the size of the feature maps caused by the use of pooling layers in each of the blocks. The feature maps are also zero padded in order to preserve the size of the image after the application of the convolution operation. The max pooling layer is used for reduction in size of the feature maps, speeding up the training process, and making the model less variant to minor changes in input. The kernel size for max pooling is 2×2. ReLU activation layer is used in each of the blocks for the introduction of non-linearity. Also, Dropout regularization technique has been used with a keep probability of 0.5 to avoid overfitting the train set. Dropout regularization randomly drops neurons in the network during each iteration of training in order to reduce the variance of the model and simplify the network which aids in prevention
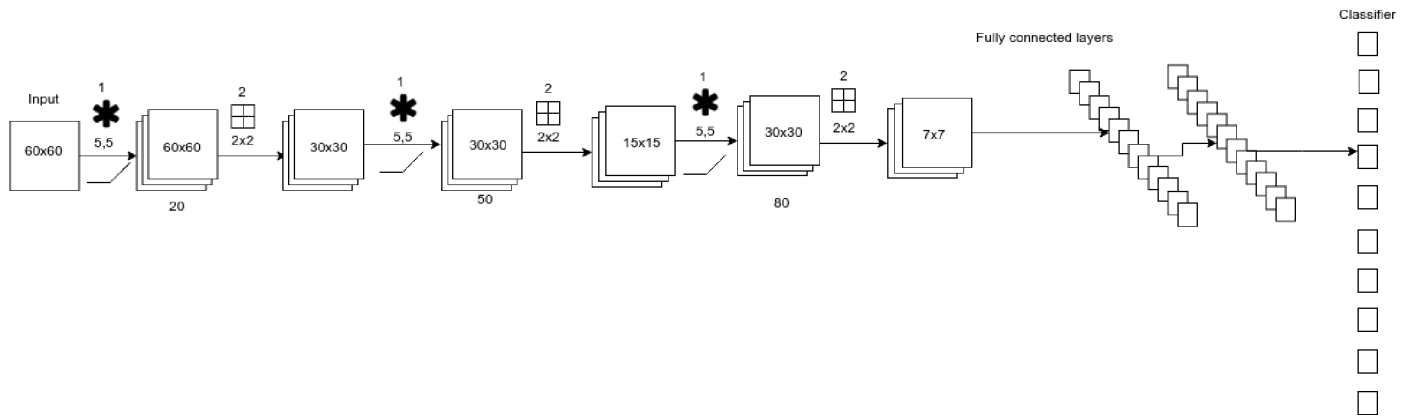
Figure 5: Model architecture

of overfitting. Finally, the classification block consists of two sets fully connected neural network layers each with 500 and 10 neurons respectively. The second dense layer is followed by a softmax activation function to compute the probability scores for the ten classes.

IV. EXPERIMENTAL SETTINGS

The implementation of the proposed methodology has been carried out on the Plant Village dataset. It consists of around 18160 images belonging to 10 different classes of tomato leaf diseases. Keras, a neural network API written in Python, has been used for the model implementation.

Out of the 18160 images, 4800 images were set aside for testing and 13360 images were used for training. In order to increase the dataset, automatic data augmentation techniques has been used by randomly rotating the images by a small amount of 20 degrees, horizontal flipping, vertical and horizontal shifting of images. The optimization was carried out using Adam optimizer with categorical cross entropy as the loss function. Batch size of 20 has been used and the model has been trained for 30 epochs. The initial learning rate has been set to 0.01 and it is reduced by a factor of 0.3 on plateau where the loss stops decreasing. Early stopping has also been used in order to monitor the validation loss and stop the training process once it increases. All the experiments were performed on Intel Core i3-4010U CPU.

V. RESULTS AND ANALYSIS

To evaluate the performance of the proposed model, a set of quantitative metrics comprising of accuracy, precision, recall and F1-score have been used. The results are reported in Table 1. They show the highest values of the quantitative metrics obtained until the corresponding epoch number.

TABLE I. RESULTS AND ANALYSIS

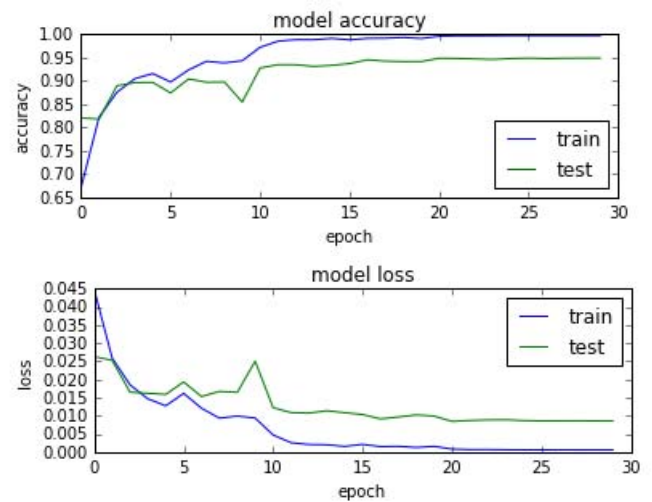| No.of epochs	Accuracy	Precision	Recall	F1-Score
10	0.9041	0.9012	0.9012	0.9012
20	0.9452	0.9449	0.9449	0.9449
30	0.9485	0.9481	0.9481	0.9481

Figure 6: Plots of accuracy and loss against epochs

A highest validation accuracy of 94.8% was obtained over 30 epochs of training, while a high 99.3% of training accuracy was reported. An average validation accuracy of 94% has been obtained. This is an effective measure of the classification made by the deep learning model. The plots of train and test accuracy and loss against the epochs in Fig. 6 provide a means of visualization and indication of the speed of model convergence. It can be seen that the model has stabilized around 20 epochs and the metrics do not show a significant improvement in the last 10 epochs. The results show that the model performs well on the dataset and can be used as a means for classification of the 10 tomato leaf diseases with minimum resource requirements.

The implementation process requires minimum hardware requirements unlike large neural networks which generally have high computational resource requirements or the use of a Graphics Processing Unit. This is due to less number of training parameters owed to the presence of fewer layers with less filter sizes and smaller train size images. Unlike other state of the art models, the model implementation can be carried out

on CPU with minimum time owing to the simplicity. Also, the variation of the LeNet model adopted is simple to understand and easy to implement. The model thus, provides a simple and effective way of solving the problem of plant disease detection with results comparative to [6], where the authors deal with plant diseases of multiple crops. With less resource constraints and minimal data, the model gives comparative results to traditional state of the art techniques.

VI. Conclusion and Future Work

Agricultural sector is still one of the most important sector over which the majority of the Indian population relies on. Detection of diseases in these crops is hence critical to the growth of the economy. Tomato is one of the staple crops which is produced in large quantities. Hence, this paper aims at detection and identification of 10 different diseases in the tomato crop. The proposed methodology uses a convolutional neural network model to classify tomato leaf diseases obtained from the Plant Village dataset. The architecture used is a simple convolutional neural network with minimum number of layers to classify the tomato leaf diseases into 10 different classes. Different learning rates and optimizers could also be used for experimenting with the proposed model as a part of the future work. It could also include experimentation with newer architectures for improving the performance of the model on the train set. Thus, the above mentioned model can be made use of as a decision tool to help and support farmers in identifying the diseases that can be found in the tomato plant. With an accuracy of 94-95% the methodology proposed can make an accurate detection of the leaf diseases with little computational effort.

References

[1] Jihen Amara, Bassem Bouaziz, Alsayed Algergawy, et al. "A Deep Learning-based Approach for Banana Leaf Diseases Classification." In: *BTW (Workshops)*. 2017, pp. 79–88.

[2] Hui-Ling Chen et al. "Support vector machine based diagnostic system for breast cancer using swarm intelligence". In: *Journal of medical systems* 36.4 (2012), pp. 2505–2519.

[3] S. D. Khirade and A. B. Patil. "Plant Disease Detection Using Image Processing". In: *2015 International Conference on Computing Communication Control and Automation*. Feb. 2015, pp. 768–771. DOI: 10.1109/ICCUBEA.2015.153.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[5] Yann LeCun et al. "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4 (1989), pp. 541–551.

[6] Sharada P Mohanty, David P Hughes, and Marcel Salathé. "Using deep learning for image-based plant disease detection". In: *Frontiers in plant science* 7 (2016), p. 1419.

[7] Usama Mokhtar et al. "SVM-based detection of tomato leaves diseases". In: *Intelligent Systems' 2014*. Springer, 2015, pp. 641–652.

[8] Usama Mokhtar et al. "Tomato leaves diseases detection approach based on support vector machines". In: *Computer Engineering Conference (ICENCO), 2015 11th International*. IEEE. 2015, pp. 246–250.

[9] H Sabrol and K Satish. "Tomato plant disease classification in digital images using classification tree". In: *Communication and Signal Processing (ICCSP), 2016 International Conference on*. IEEE. 2016, pp. 1242–1246.

[10] Christian Szegedy et al. "Going deeper with convolutions". In: Cvpr. 2015.