DETECTION OF DISEASES IN PLANT LEAVES USING GABOR FILTER FEATURE EXTRACTION AND CNN

Advaith Narayan[1], Noel John[2], Prapul Chandra[3], Vijet Naik[4], Dr. Sumithra Devi K A[5]

[1] *BE Student, Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Karnataka, India*
[2] *BE Student, Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Karnataka, India*
[3] *BE Student, Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Karnataka, India*
[4] *BE Student, Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Karnataka, India*
[5] *Dean & HOD, Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Karnataka, India*

ABSTRACT

Now-a-days crops have been a vital in food consumption so in order to maintain the crops that are healthy and can be consumed, we have to detect the crop leaves that are not affected through disease, and we have used The tomato crop is a significant staple in the Indian market with high business esteem and is delivered in enormous amounts. Sicknesses are inconvenient to the plant's wellbeing which thusly influences its development. To guarantee negligible misfortunes to the developed yield, it is urgent to administer its development. There are various kinds of tomato illnesses that focus on the yield's leaf at a disturbing rate. This paper receives a slight variety of the convolutional neural organization model called Le net to distinguish and recognize infections in tomato leaves. The primary point of the proposed work is to discover an answer for the issue of tomato leaf infection location utilizing the easiest methodology while utilizing min-imal processing assets to accomplish results equivalent to best in class strategies. Neural organization models utilize programmed include extraction to help in the order of the info picture into individual sickness classes. This proposed framework has accomplished a normal precision of 94-95% showing the practicality of the neural organization approach much under ominous conditions.

Keyword: *Leaf disease detection, Neural network, Convolution, LeNet*

---

1. INTRODUCTION

India is a country with a larger part of the populace depending intensely on the horticultural area. Tomato is the most well-known vegetable utilized across India. The three most significant cell reinforcements to be specific nutrient E, nutrient C and beta-carotene are available in tomatoes. They are likewise plentiful in potassium, a vital mineral for great wellbeing. Tomato crop development region in India ranges around 3,50,000 hectares roughly and the creation amounts generally summarize to 53,00,000 tons, making India the third biggest tomato maker on the planet. The affectability of harvests combined with climatic conditions have made illnesses normal in the tomato crop during every one of the phases of its development. Infection influenced plants comprise 10-30% of the complete yield misfortune. ID of such infections in the plant is vital in forestalling any hefty misfortunes in yield just as the amount of the farming item. Checking the plant sicknesses physically is a troublesome assignment because of its perplexing nature and is a tedious interaction. Consequently, there is a need to decrease the manual exertion put into this assignment, while making exact expectations and guaranteeing that the ranchers' lives are sans bother. Outwardly detectable examples are hard to unravel at a solitary look, prompting numerous ranchers making incorrect presumptions with respect to the sickness. Accordingly, avoidance components taken by the ranchers might be inadequate and once in a while hurtful. Ranchers generally meet up and carry out normal illness anticipation instruments, as they need master exhortation on the most proficient method to manage their harvest pervasion [2].

The technique recommended in the paper relates to the most widely recognized sicknesses found in the tomato plant like, Bacterial leaf spot and Septoria leaf spot, Yellow Leaf Twist among numerous others. Any leaf picture given as information can be ordered into one of the infection classes or can be considered sound. The information base utilized for assessment is a subset of Plant Town [6], a vault that contains 54,306 pictures of 14 harvests swarmed with 26 infections. The subset incorporates around 18160 pictures of tomato leaf sicknesses.

2. LITERATURE SURVEY

Perceive the past research done concerning this field to have the option to accurately progress the correct way. Plant leaf infection recognition has been a significant exploration region wherein both picture handling and profound learning procedures have been generally utilized for its precise order. In this paper, we talk about the most prominently fused procedures in writing in the pertinent field. Two normal tomato plant diseases look like the ones shown in Fig. 1 and Fig. 2 and healthy tomato leaves are shown in Fig. 3.

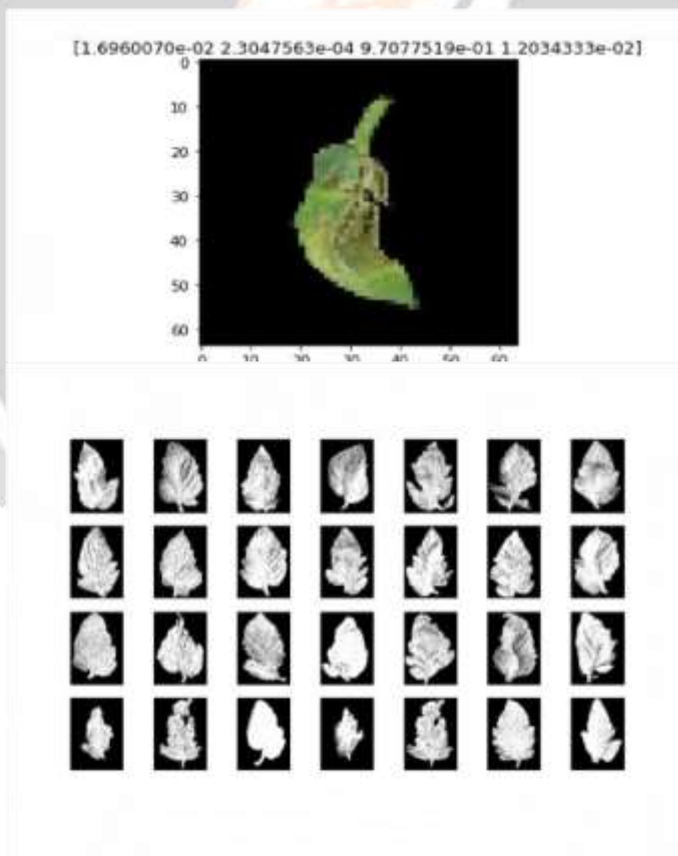


Fig 1. Septoria Leaf Spot

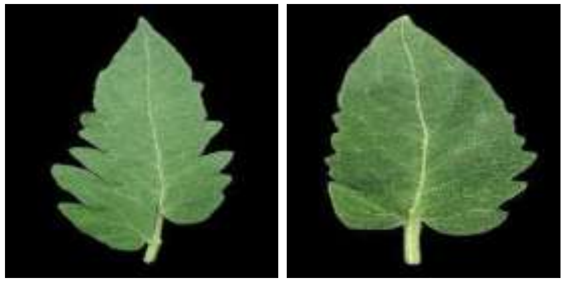


Fig 3: Healthy

Observing a huge field of yields is a drawn-out task, whenever done physically. It is important to limit the human exertion put into plant management. Consequently, this is a well-known exploration area drawing in numerous analysts. A few works identified with plant sicknesses are seen in writing.

The creators of the paper [7] have proposed a productive strategy that recognizes whether a tomato leaf is solid or contaminated. The picture given as info was first preprepared by eliminating the foundation and the commotion present was killed with the assistance of disintegration method. Dim Level Co-event Network (GLCM) was utilized for surface element extraction from the improved picture. Backing 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.

To beat the issue of the above paper, the creators in [3] have proposed different division, include extraction and arrangement methods that recognize and distinguish the kind of the sickness utilizing the unhealthy picture to lead order. The leaf picture given as contribution to the framework was pre-prepared by smoothing it or improving the picture by performing histogram adjustment. To acquire the influenced region, distinctive division procedures like KMeans bunching have been proposed. The highlights were then extricated from the fragmented district and determined utilizing GLCM. After highlight extraction, the sicknesses can be distinguished with the assistance of Counterfeit Neural Organizations (ANN) or Back Proliferation Neural Organizations. The downside of fragmenting the picture utilizing K-Means grouping is that the interaction proposed was semi robotized as the client needs to expressly choose the bunch which contains the unhealthy part

Deep convolutional neural organizations have been prepared in [6] for the distinguishing proof of 26 infections in 14 distinctive yield species. The creators utilize the standard Alex Net [4] and Google Net [10] designs for this reason. A public vault which contains 54,306 pictures of both infected leaves and sound plant leaves has been utilized for this reason. The dataset has been made by gathering the pictures of the plant leaves in a controlled climate. The creators have led a presentation examination on both these designs via doing the model preparing two ly. It is performed without any preparation in the principal case and by utilizing move learning in the second. Move learning compares to the way toward adjusting pre-prepared loads got via preparing models on the ImageNet dataset. The model execution has been completed utilizing the Caffe system giving an exactness of 99%. This depicts the plausibility of this methodology. Be that as it may, on testing the prepared model against a bunch of test pictures acquired from online public information sources which are very not quite the same as the train set, the model exactness tumbles to 31.4%. This is a typical issue looked in neural organizations owed to the train and test

3. PROPOSED METHEDOLOGY

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.

Fig.4: Proposed Methodology

3.1 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.

3.2. 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 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 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.
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 Alex Net [4], Google Net [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 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.

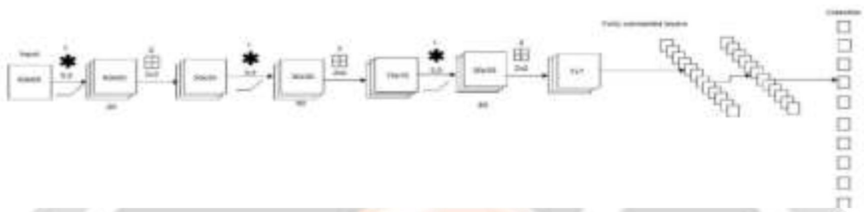
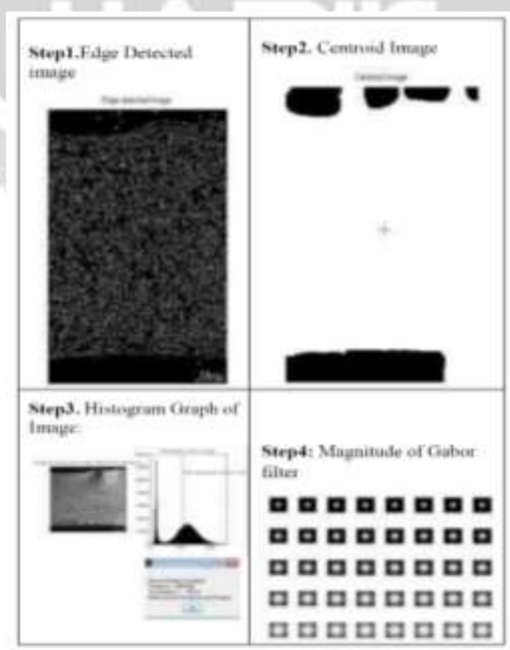


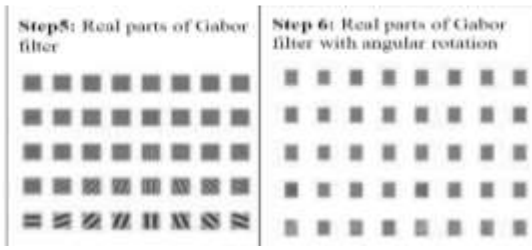
Fig.5: Model Architecture

3.3. Feature Extraction

In feature extraction, we using Gabor filter to extract filter texture features of input image. With the help of Gabor filters, we are considering GLCM features for our feature extraction of leaf. Area, Perimeter, Contrast, Energy, Homogeneity, major length, minor length, mean of gray level, these are common features we are extracted [5].

Gabor Filter: Texture features that are based on the local power spectrum obtained by a bank of Gabor filters are compared. Here we are interesting to extract texture features like Area, Perimeter, Contrast, Energy, Homogeneity, major length, minor length, mean of gray level. The filter is characterized by a preferred orientation and a preferred spatial frequency. When a small-area patch has a wide variation of features of discrete gray tone, the dominant property of that area is texture.





4. 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. To increase the dataset, automatic data augmentation techniques have 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.

5. 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 2. They show the highest values of the quantitative metrics obtained until the corresponding epoch number 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

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 a smaller 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.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 64, 64, 32)        2432
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 32, 32, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 32, 32, 64)        51264
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 16, 16, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 12, 12, 128)       204928
_________________________________________________________________
dropout (Dropout)            (None, 12, 12, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 18432)             0
_________________________________________________________________
dense (Dense)                (None, 1024)              18875392
_________________________________________________________________
dropout_1 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_1 (Dense)              (None, 4)                 4100
=================================================================
Total params: 19,138,116
Trainable params: 19,138,116
Non-trainable params: 0
```

RESULTS:

```
Epoch 1/10
42/42 [==============================] - 96s 1s/step - loss: 120.6908 - accuracy: 0.3747 - val_loss: 1.0834 - val_accuracy: 0.5
480
Epoch 2/10
42/42 [==============================] - 54s 1s/step - loss: 1.0670 - accuracy: 0.5382 - val_loss: 1.0730 - val_accuracy: 0.552
4
Epoch 3/10
42/42 [==============================] - 54s 1s/step - loss: 1.0601 - accuracy: 0.5381 - val_loss: 0.9983 - val_accuracy: 0.563
3
Epoch 4/10
42/42 [==============================] - 56s 1s/step - loss: 0.9993 - accuracy: 0.5789 - val_loss: 0.9913 - val_accuracy: 0.598
3
Epoch 5/10
42/42 [==============================] - 56s 1s/step - loss: 0.9566 - accuracy: 0.5971 - val_loss: 0.9314 - val_accuracy: 0.624
5
Epoch 6/10
42/42 [==============================] - 56s 1s/step - loss: 0.9039 - accuracy: 0.6210 - val_loss: 0.9248 - val_accuracy: 0.635
4
Epoch 7/10
42/42 [==============================] - 51s 1s/step - loss: 0.8847 - accuracy: 0.6356 - val_loss: 0.8819 - val_accuracy: 0.628
8
Epoch 8/10
42/42 [==============================] - 51s 1s/step - loss: 0.7943 - accuracy: 0.6665 - val_loss: 0.8963 - val_accuracy: 0.650
7
```

6. CONCLUSION

Agricultural sector is still one of the most important sectors 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.

7. 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 diag- nostic 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 net- works". In: Advances in neural information processing sys- tems. 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 Salathe´. "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 Confer- ence on. IEEE. 2016, pp. 1242– 1246.

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