Detection Disease Tomato Using Cnn

Youssef Laatiri, Mahjoub Mohamed and Karim Kalti

December 27, 2023

Detection Disease Tomato Using Cnn

ABSTRACT. Tomato production has been rising over recent years. Apart from nutritional values, tomato production also plays a significant role in terms of creating job opportunities for a lot of people. Even though the most suitable season for producing tomato is winter, tomato cultivation has been proved profitable throughout the year. However, the production of tomatoes gets hindered due to various diseases of tomato leaves. The goal of our research is to create a model that can predict these diseases earlier with the intent that suitable actions can be taken to mitigate the situation. This work presents a deep CNN model with over 22500 images divided into 3 parts: Training, validation, and testing. The dataset was prepared using necessary normalization, augmentation, and label encoding techniques. Total 10 classes (9 diseases and 1 healthy class) were taken into consideration. 75 instances were separated from each class for testing purposes. 80% of the remaining data was used for training and 20% was used for validation. The proposed Deep CNN is trained with various dropout values and a suitable dropout value is identified to regularize the model.

The model was eventually able to achieve a high accuracy of 98.77% for the considered dataset. The outcomes of the work shows the effectiveness along with feasibility of our suggested model .

Keywords: *Detection, Classification, Convolutional Neural Networks, dataset, image processing, accuracy, loss, Overfitting, Dropout()*

I. INTRODUCTION

The majority of tomato infections typically start in the leaves before spreading throughout the entité plant. While efforts have been made to create a favorable environment for the classification and detection of tomato leaf diseases, our garden occasionally still falls victim to these plant diseases.

The conventional method of proficiently analyzing tomato leaf diseases is both cost-prohibitive and prone to subjectivity, which poses a significant problem. With the rapid advancement of computer technology, deep learning and machine learning, a long with computer vision, have now become widely employed in diagnosing agricultural diseases. Traditional computer vision methods for crop diseases involve segmenting RGB images based on characteristics such as color, texture, and shape. However, because many diseases share similar characteristics, accurately classifying and detecting them in complex environments can be challenging. Deep neural networks are employed as à prime example of end-to-end intelligence in the realm of deep learning. They serve various purposes, and the proposed work is one such application. In a neural network, nodes receive mathematical input from incoming edges and produce numerical results as outgoing edges.

Deep neural networks effectively map the input layer to the output layer, which in this case involves taking an image of an unhealthy plant as input and producing a diagnosis of potential diseases as output. Creating a deep neural network involves stacking multiple layers of nodes. The challenge lies in ensuring that the network's architecture, the tasks performed by individual nodes, and the weights of the edges accurately transforms input data into the desired output.

Deep neural networks are improved by adjusting their parameters during training to enhance their ability to map inputs to outputs. Convolution Neural Networks (CNNs) represent a sophisticated form of deep learning architecture that replaces labor-intensive feature extraction and image preprocessing methods with an end-to-end approach, significantly speeding up the identification process compared to previous research methods. Utilizing CNNs for predicting tomato leaf diseases can increase diagnostic accuracy while reducing labor costs. For the analysis of tomato leaf diseases, only a small portion of the leaf's image contains the diseased area. To detect diseases in complex environments, this approach incorporates a thoughtful element into the CNN network model. The disease feature channel becomes the primary focus of feature extraction, and irrelevant feature channel data is eliminated.

This research proposes an enhanced CNN network model designed for the precise detection of multiple tomato leaf diseases based on their unique properties. Deep learning techniques enable fast and accurate diagnoses, facilitating the timely application of immunity methods and prevention efforts to manage these diseases more effectively. However, it's worth noting that the ability to detect multiple diseases is currently limited, and the process remains time-consuming, as people used to rely on personal experience for tomato disease identification

II. LITERATURE REVIEW

Machine learning algorithms are applied in various fields, but feature engineering remains the main problem. With the emergence of deep neural network, the promising results are available for plant pathology without laborious feature engineering. Deep neural networks significantly increase the image classification accuracy. This section provides a various deep learning technique used by researchers in plant disease identification. This section analyzes the latest research for the application of learning deep in the field of plant disease detection and more precisely the work done using the Plant Village dataset (**HUGHES, David; SALATHÉ, Marcel et al. 2015)** [1] the Results are summarized in Table 1.1.the authors of (**KAWASAKI, Yusuke; UGA, Hiroyuki; KAGIWADA, Satoshi; IYATOMI, Hitoshi.**(2015)) [2] proposed a plant viral disease detection system. The dataset used consisted of 800 images of cucumber leaves representing two different diseases and also healthy leaves. The authors used their own CNN model. The proposed system achieved classification accuracy. Maximum of 94.9% under a 4-fold cross-validation strategy.

The same authors conducted another research (**FUJITA, Erika; KAWASAKI, Yusuke; UGA, Hiroyuki ; KAGIWADA, Satoshi ; IYATOMI, Hitoshi. (2015**))[3] to detect seven types of cucumber viral diseases. They used a dataset of 7250 images including viral diseases and healthy leaves. The classifiers achieved an average accuracy of 82.3% with a four-fold cross-validation strategy. The authors of (**AMARA, Jihen; BOUAZIZ, Bassem; ALGERGAWY, Alsayed.2017**) [4] used the LeNet architecture to detect two types of diseases among healthy diseases in images of banana leaves collected in the part of the Plant Village project. the dataset contained 3700 images and the model achieved a maximum accuracy of 99.72%.

In a study conducted by **(Prajwala Tm et al.,(2018))** [5], the utilization of a CNN model named LeNet was explored for the identification of various diseases affecting tomato leaves. The research outlines the implementation of this model, which incorporates an automatic feature extraction method to streamline the classification of diverse diseases. Notably, the model achieved an impressive average accuracy ranging from 94% to 95%. The optimization process involved the use of the Adam optimizer, with categorical Cross-entropy serving as the selected loss function. The training process employed a batch size of 20 and 30 epochs for the complete model.

(**Another study by Liu Y Y et al.(2018))** [6] from Gansu Agricultural University employed the regional Faster-RCNN model to train collected images, subsequently classifying and identifying the images by classifier. The accuracy of diseased leaf recognition

ranged from 60.56% to a maximum of 75.52%. (**Ramcharan et al.**2017) [7] from the Pennsylvania State

University in the United States applied a learning migration training deep convolutional neural network to Identify three diseases of cassava and two diseases of pests. (**Similarly, Durmus¸ et al. (2017)**)[8] utilized AlexNet and SqueezeNet to classify tomato plant images into 10 classes 9 diseases and 1 healthy. On the PlantVillage dataset, the AlexNet model achieved an accuracy of 95.65%, while the SqueezeNet model achieved 94.3 %.(**Karthik et al. (2020)**) [9] employed two deep learning architectures on the PlantVillage dataset to detect three diseases in tomato plants: early blight, late blight, and leaf mold. The attention-based residual CNN architecture presented the highest accuracy of 98%.

(**Agarwal et al. (2020)**) [10] developed a CNN model with 3 convolutional, 3 max-pooling, and 2 fully connected layers to detect 10 classes (9 diseases and 1 healthy) in tomato plants, achieving an overall accuracy of 91.2% on the PlantVillage dataset.

(**Elhassouny and Smarandache. (2019)**) [11] created a smart mobile application for classifying 9 diseases of tomato plants using MobileNet (**Howard et al., (2017**)[12]. The application, utilizing 7,176 tomato leaf images from the PlantVillage dataset, achieved an accuracy of 90.3%. (**Widiyanto et al. (2019)**)[13] developed a CNN model to identify 4 diseases in tomato plants, namely, late blight, Septoria leaf spot, mosaic virus, and yellow leaf curl virus, along with healthy leaves. The model, trained on 1,000 images per class from the PlantVillage dataset, achieved an overall classification accuracy of 96.6%. Table 1 summarizes the various deep learning approaches used in the literature for the detection and classification of tomato plant diseases.In This paper**,(Romana Rahman Ema, Sk. Shalauddin Kabir, Md. Nasim Adnan, Syed Md. Galib**,(2023)) [14] authors are proposed strategies for leaf disease, disease severity stage, and pest detection using five notable Convolutional Neural Network models

(CNN) such as VGG16, Resnet50, AlexNet, EfficientNetB2, and EfficientNetB3 . The models are used for the classification of diseases and pests, with EfficientNetB3 achieving the highest accuracy for disease and pest detection in both training (99.85%) and validation (97.85%) . For severity stage identification, AlexNet achieves the best accuracy in training (69.02%) and validation (72.49%).The authers (**N. Balakrishna, Gurram Sunitha ,(2022**)) [15] are proposed a deep learning approach using CNN to detect tomato leaf diseases, achieving an accuracy of 92%. the authors used deep learning techniques, specifically with deep detector: Faster R-CNN with deep feature extractor: ResNet50, is used to detect and classify tomato disease in plants. the proposed system successfully detects early blight, leaf curl, septoria leaf spot, and bacterial spot of tomato disease even in complex plant surrounding areas

III. CNN MODELS FOR TOMATO LEAF DISEASE DETECTION

1. Convoltion neural Network (Cnn)

The architectures of Convolutional Neural Networks have not stopped evolving since 1989 The optimization of criteria and parameters, regularization, classification and structural reformulation present continuous evolutions. Moreover, a good number of these improvements are focused on the restructuring of the processing units and the design of new blocks.

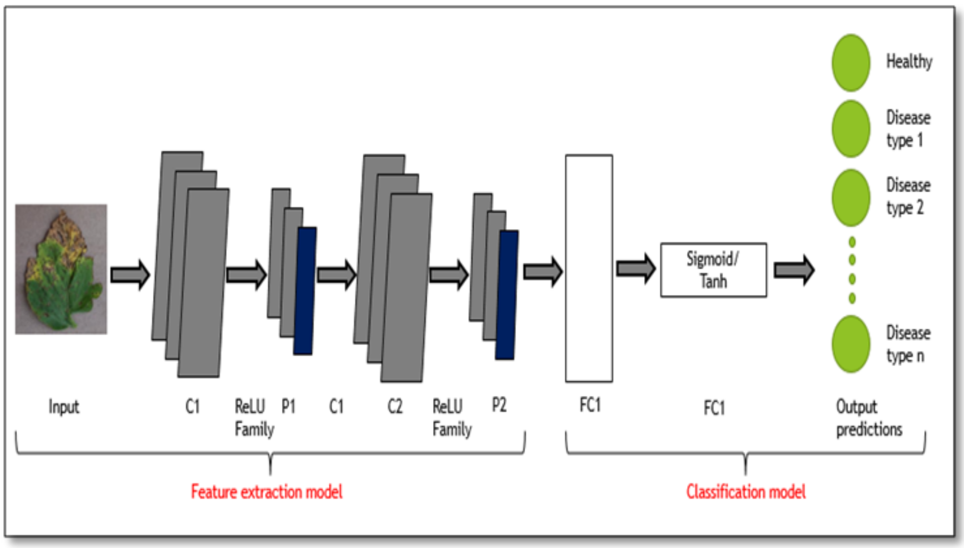


Figure 1. *Example of CNN architecture used in plant disease detection*

Convolutional neural networks (CNNs) are a subset of deep learning algorithms that have been designed to handle pixilated data. These networks are widely used in image recognition and analysis. It receives an image as input, applies a set of biases and weights that it has learned to each image, and then uses this information to tell them apart. One potential benefit of adopting CNN is that it requires far less pre-processing than previous algorithms meaning the neural network learns on its own instead of relying on filters that were manually constructed for traditional methods (**hikari, S., Unit, D., Shrestha, B., & Baiju, B. (2018**))[16]. To extract characteristics from high-dimensional data, convolutional neural networks (CNNs) are a type of artificial neural network. In this Analysis, a max-pooling layer is added to a simple CNN model consisting of three convolutional blocks. In addition, a dropout layer, a dense layer, and a flat layer were added as a conclusion as shown Fig1. The function that flattens the pooled feature maps into a single vector before sending them to a dense layer comes in between the pooling and dense layers .A CNN-based DL model was built to distinguish between healthy and TSW-afflicted images. The CNN-based model can binary and multi-classify the image collection. It has two convolutional (C) layers, two max-pooling (M) layers, one flattening (F) layer, and one fully connected (D) layer. Train the two convolutional layers with an input image to extract features using convolution [. The max-pooling layer receives the output feature vector next. This layer pools feature vectors from convolutional layers and finds the maximum value from each feature map batch(**Kingma, D.P. and Ba, J., 2015**) [17].

2. Google Net

The GoogleNet can save time in part by reducing the size of the input image while keeping the relevant spatial details intact. Several filters were applied to the publicly available Plant Village dataset to highlight the disease hotspots using GoogleNet CNN architectures. For the purpose of measuring performance and contrasting the two well-known CNN designs, we used the P, R, F1, and OAI measures across three different situations (color, grayscale, and segmented). Results showed that GoogleNet was superior to Alex Net (**Hong, H., Lin, J. and Huang, F., (2020)**)[19].

3. ResNet 50

ResNet-50 is a 50-layer deep convolutional neural network. The network can be loaded in its pre-trained state, which has been exposed to over a million images in the ImageNet database [16]. The ResNet-50 model is the basis for this 97% accurate framework. Advantages include a trained model that can improve its results by augmenting them with additional data.
Cons It might be pricey to maintain a high-configuration hardware environment for training purposes (**Jiang, D., Li, F., Yang, Y. and Yu, S., (2020**))[20].

4. DenseNet-121

Though all the models did well, the DenseNet-121 model had the highest accuracy while also being the smallest in size. DenseNet-121's results were similarly achieved by ResNet-101 and VGG16. However, ResNet-101 was much bigger, making it inappropriate for mobile devices with limited storage space. Additionally, this research can be expanded to identify and Diagnose diseases, and a lightweight model can be implemented for use on mobile devices. A better dataset can lead to better results(**Gehlot, M. and Saini, M.L.,(202**0)) **[21].**

5. InceptionV3 model

In this work, we employ Neural Computing Stick (NCS) to expedite computation and simplify detection because of their mobility, speed, and accuracy. To detect Septoria leaf spot disease in tomatoes, researchers at Intel NCS used the InceptionV3 model to create a deep learning system (**Muchtar, K., Chairuman, C., Fitria, M., Kardawi, M.Y., C.Y., (2021)** [22].

IV .Dataset Description

Taken from well-known PlantVillage (**Tairu Oluwafemi Emmanuel. (2018. [24]**)dataset. For 10 different classes, 10000 images (in total) were used where 10000 images had been used for training, 2000 images had been used for validation, Each of the images was in RGB format with 224 X 224 pixels resolution. A few samples from the dataset from different classes are illustrated in Figure 2.

Figure 2. Examples of tomato 10 classes – (1) Bacterial spot, (2) Early blight, (3) Late blight, (4) Leaf mold, (5) Septoria leaf spot, (6) Spider mites two spotted, (7) Target spot, (8) Yellow leaf curl virus, (9) Mosaic virus and (10) Tomato healthy.

V. METHODOLOGY

A deep convolutional neural network has been used for predicting tomato leaf disease in this experiment. Our experiment includes total of 10classes - 1 healthy and 9 other disease classes. Figure. 2 indicates how the whole procedure works to correctly detect tomato leaf diseases.

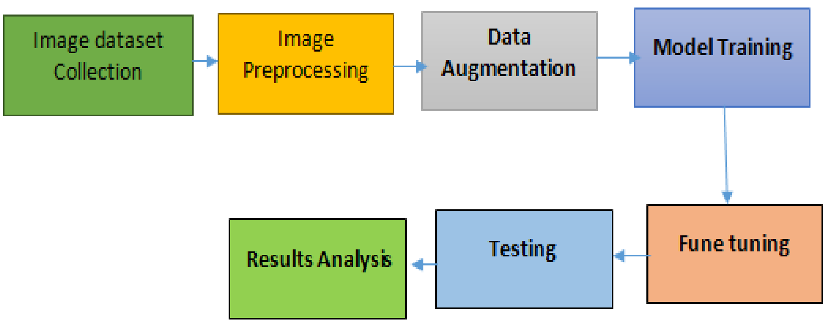


Figure2. Steps of Detections and classification process For leaf diseases

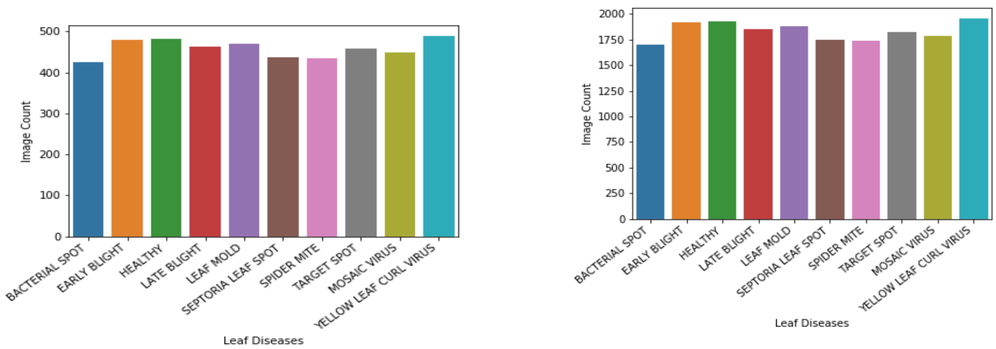


Figure.3. Leaf Diseases train and test

A. Dataset Preparation

A.1. Image preprocessing

Image preprocessing enhances the quality of the image data needed for image classification. Geometric transformations of images, such as image rotation, scaling, and translation, are used in preprocessing approaches. In this step, we decreased the resolution of all of the images to 64*64 pixels during the preprocessing stages, the original images are 224*224 pixels. It must ensure that all images are of the same size and resolution.

A.2. Augmentation Process

CNN requires a large amount of training data to achieve improved results (Shorten & Khoshgoftaar, 2019). In order to improve the model's performance, image augmentation is frequently required to create the best deep CNN model with insufficient training data. Image augmentation increases the amount of images in the data set and reduces over fitting by adding a few distorted photos to the training data. When the network learns the data rather than the overall pattern of the dataset, this is known as over fitting. Image augmentation artificially creates training images using a range of processing methods or a combination of processing methods such as image flipping, rotation, blur, relighting, and random cropping(**Chen et al., 2020**). In our study we do the following for images augmentation: scaling the Images, shearing, zooming and horizontal flipping.

Tab. 2. Fine-Tuning parameters and values used through training models

Paramétrer	Value
Batch size	128
Metrics	Accuracy,Loss,Precision
Epoch	**75**
Validation steps	none1
Optimizer	Adm
Activation Function	Softmax
lr	0.0010
Loss Function	Categorical Crossentropy

A.3. Model Architecture

A model has been built with CNN algorithm to require minimal preprocessing of images Mostly it is used for analyzing visual imagery.Convolutional Neural Network grained its popularity when CNN algorithm came as a success in image processing and deep learning

System model of this research takes raw images as input and extract features images augmentation of CNN algorithme has been used.Our model has been done upon full color or RGB(Red,Green,and Blue).At the end model feedbacked as follows:

Table2: Convolutional Network Model

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
dropout (Dropout)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 64)	18496
max_pooling2d_1 (MaxPooling2	(None, 14, 14, 64)	0
dropout_1 (Dropout)	(None, 14, 14, 64)	0
conv2d_2 (Conv2D)	(None, 12, 12, 128)	73856
max_pooling2d_2 (MaxPooling2	(None, 6, 6, 128)	0
dropout_2 (Dropout)	(None, 6, 6, 128)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 64)	294976
dense_1 (Dense)	(None, 128)	8320
dense_2 (Dense)	(None, 64)	8256
dense_3 (Dense)	(None, 10)	650

Total params: 405,450
Trainable params: 405,450
Non-trainable params: 0

Output of the model can model can be sp up into parts.Where, in the first part of full-color system model ,3 Convolutions layers with Relu activation function are to be found. And evry convolutional is followed by MaxPooling layer/where ,first convolutional layer contains input shape (62,62,3),filtrer size 32,kernelsize(3,3),padding"same",strides'(1*1).Second Convolutionnal layer contains activation Relu ,filtrer size 64 ,kernel size (3*3),padding "same" strides(1*1) followed by MaxPool pool size 2*2 ,strides(1*1). Convolutionnal number three layer contains activation Relu ,filtrer size 128 ,kernel size (3*3),padding "same" strides(1*1) followed by MaxPool pool size 2*2 ,strides(1*1).Later Batch Normalization is used for activating the previous layers at each batch model.

$$ReLu(X) = Max\ (0, X) \qquad \textbf{(1)}$$

A Dense layer with 512 units has been used by Flatten layer where ReLu, activation contained 50% of Dropout. The final output layer has used 5 units with activation SoftMax "2" and Sigmoid "(30)".

$$\sigma(z) = \frac{e^{z_i}}{\sum_{i}^{km} e^{m}} for\ i = 1 \qquad \textbf{(2)}$$

$$\varphi(z)=1/(1+e^{\wedge}(-z)) \qquad \textbf{(3)}$$

Adam algorithme has been used for reducing errors and value loss. It is a learning rate method for computing individual learning rate for different parameters where, we took Adam value as (0.0010).

A.4. Optimizer and Learning Rate

Optimization algorithme declares a sufficient change which can be ensued for the output of the work of image processing ,computer vision and deep neural network

$$\mathbf{Vt= (1-\beta2)} \quad \sum_{i=1}^{t} \beta .\mathbf{g}^2{}_i \qquad (4)$$

$$\mathbf{Li}{=}- \quad \sum_j ti, jlog(pi, \mathrm{j}) \qquad (5)$$

to perform classification and prediction task a neural network use ,a very recent study showed that classification error and mean square error perform worse than that crossentroppy function(**Janocha ,Ka.,Czarnecki,W.M. (2017)**)[15]..Cross-entroppy error do not get smaller enough ,so does the weight change.The Proposed method of Our research have used categorical cross entropy as reduced function

To make the optimizer converge faster and closer to the global minimum of the loss function an automatic Learning Rate reduction method has been used .Learning rate walks through the minimum loss using it like a step.It of global minima if higher learning rate used .After successful run of evry epoch model dynamically decreases the learning rate advantage of the fast computation time with a high learning Rate running .It also manually cheeks the accuracy and decreases learning rate after some successful epochs to reach the global minima.

VI . RESULTS AND ANALYSES

A. Accuracy Graph

75 instances from each class separated for testing and they were divided the rest of the available data in a way that 80% of the data from each class go into training and 20% is used for validation the summary shows that the model achieved 98.77% accuracy for the considered dataset .Figure 4 and 5 visualizes the summary of training and validation

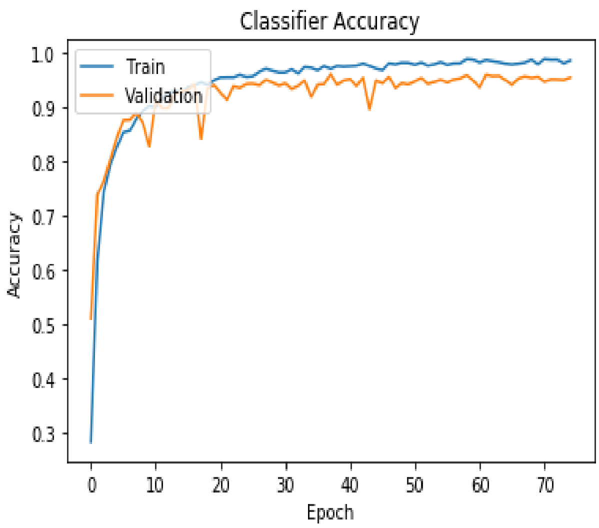
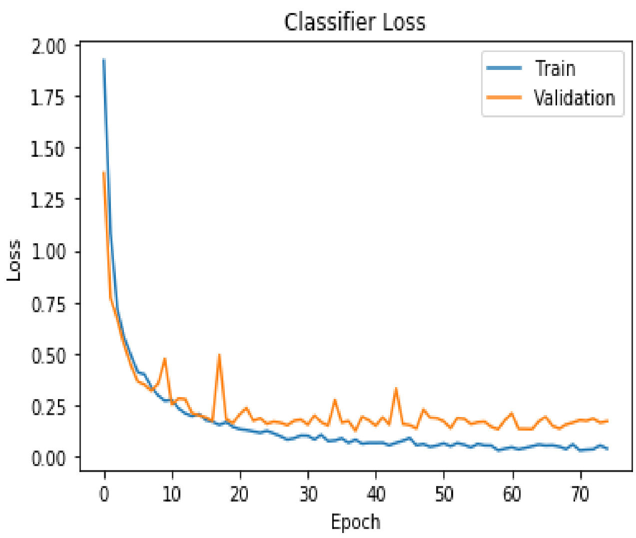


Figure.4. Accuracy Graph



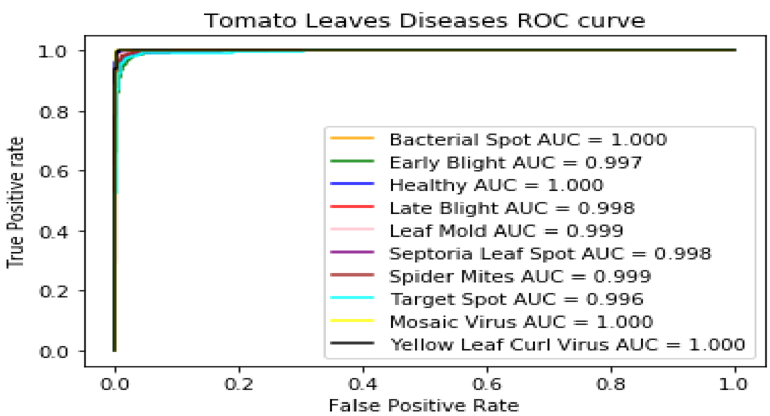
Figur.5. Loss graph

B.Confusion Matrix

	Bacterial Spot	Early Blight	Healthy	Late Blight	Leaf Mold	Septoria Leaf Spot	Spider Mite	Target Spot	Mosaic Virus	Yellow Leaf Curl Virus
Bacterial Spot	420	4	0	0	0	0	0	0	0	1
Early Blight	8	419	0	25	1	17	2	7	1	0
Healthy	2	0	476	0	0	0	0	3	0	0
Late Blight	3	12	1	436	1	8	1	0	0	1
Leaf Mold	0	0	0	3	449	14	4	0	0	0
Septoria Leaf Spot	0	2	0	2	5	423	1	1	2	0
Spider Mite	0	0	2	0	1	2	417	11	0	2
Target Spot	0	2	13	0	0	7	13	414	8	0
Mosaic Virus	0	0	0	0	0	1	0	0	447	0
Yellow Leaf Curl Virus	4	2	0	3	0	1	6	0	0	474

Figure.6 . Confusion Matrix

Confusion matrix provides the quality our proposed odel.It show how good this model is for the processed test dataset .Our confusion matrix of this proposed model has given 420 true images for **Bacterial Spot** where total predicted images were 425 and only 5 images has founded false. **Early Blight** has 419 true images where predicted images 480 and 61 images are False. **Healthy** has given 476 images true where total predicted images were 481 and only 5 are false .**Light Blight** has given 436 images true where predict 463 images , only 27 images are founded false . **Leaf Mold** has given 449 images true where predict 470 images, only 21 images are founded false. **Septoria leaf Spot** has 423 true images 436 where predicted images, only 13 images are False. **Spider mit** has given 417 images true where predicted images 434, only 17 images are false. **Target Spot** has given 414 images true where predict 457 images, only 43 images are founded false. **Mosaic Virus** has 447 true images where predicted images 448, only 1 images are False. **Yellow leaf Curl Virus** 474 true images where predicted images 490, only 16 images are False. Diagonal values of this confusion matrix are large comparing to other table values. The diagonal values of (10*10) confusion are working the best because this step has given maximum largest data amongs all table values and differentiate largest folder values amongst every row and Column with different color. Color goes deep to deeper with the increasing diagonal values the confusion matrix is show in figure 7.



Tab

3. For Data for training and Testing with Differents Models

S.NO	Model	Accuracy	TOTALIMAGES
1	InceptionV3 model	95.85%	3362
2	LeNet	97%	55000
3	Google Net	98%	10735
4	CNN	94.33%	11804
Proposed Model	***CNN***	***98.77%***	***22500***

As shown by the curve accuracy in Figure. 8 the model achieves a remarkable performance using the cnn, in addition, the curve of the loss function follows the same performance and shows that the use of this networks optimize considerably the convergence of the model. Table 3 shows the accuracies obtained on the test set from different classifiers trained . The results obtained shows that the performance of the model using cnn are better than those with InceptionV3, Lenet, GoogleNet,CNN

VII.CONCLUSION

This reaserch raises an innonative system that can detect the affected “tomato “ leaves to increase productivitry of tomato in the agricultural industry worldwide .With this proposed method we can now proudly show that the best model has been established for detecting tomatoes leaf diseases .Image processing and CNN (convolution neural network) algorithm have gained expertise acceptance and encouraged us to try a newelly invented technology .This research thoroughly investigate and analyzed affected tomato leaves and left no possible way to detect the diseases more accratly then other system .Our research has feed local farmers along with the farmers of the whole world from the length and hard working manual disease detection procedure .We now aim to develop an android based software that can detect plant diseases from affected images and propose probable solution and list of to do things while diseases attack yields as the future enhancement of this research .

VIII. REFERENCES

[1]. Hughes, David ; Salathé, Marcel et al. *An open access repository of images on plant health to enable the development of mobile disease diagnostics*. *arXiv preprint arXiv :1511.08060*. 2015.

[2].Kawasaki, Yusuke; UGA, Hiroyuki; Kagiwada, Satoshi; IYATOMI, Hitoshi. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In: *International symposium on visual computing*. 2015, p. 638-645.

[3]. FUJITA, Erika ; Kawasaki, Yusuke; UGA, Hiroyuki ; KAGIWADA, Satoshi ; IYATOMI, Hitoshi. Basic investigation on a robust and practical plant diagnostic system. In: *2016 15th IEEE international conference on machine learning and applications (ICMLA)*. 2016, p. 989-992.

[4].AMARA ,Jihen; BOUAZIZ, Bassem; ALGERGAWY, Alsayed. A deep learning based approach for banana leaf diseases classification. *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband*. 2017.

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

[6]. LIU Y Y, FENGQ, YANG S. *Detecting grape diseaes based on convolutional neural network [J]. Journal of Northeast Agricultural University, 2018, 49(3): 73 – 83.*

[7].Ramcharan A, Baranowski K, Mccloskey P, et al. *Deep learning for image-based cassava disease detection [J]. Frontiers in Plant Science*, 2017, 8 : 1852.

[8]. Durmus¸, H., Günes¸, E.O., Kırcı, M., 2017. *Disease detection on the leaves of the tomato plants by using deep learning. In: 2017 6th International Conference on Agro- Geoinformatics. IEEE*, pp. 1–5.

[9].Karthik, R., Hariharan, M., Anand, S., Mathikshara, P., Johnson, A., Menaka, R., *Attention embedded residual CNN for disease detection in tomato leaves*. 2020. *Appl. Soft Comput. 86, 105933*

[10]. Agarwal, M., Singh, A., Arjaria, S., Sinha, A., Gupta, S., 2020 . Toled: *Tomato leaf disease detection using convolution neural network. Procedia Comput. Sci.* 167, 293–301

[11] .Elhassouny, A., Smarandache, F. *Smart mobile application to recognize tomato leaf diseases using convolutional neural networks. In: 2019* International Conference of Computer Science and Renewable Energies (ICCSRE), IEEE, pp. 1–4.

[12]. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. *Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint rXiv: 1704.04861.*

[13] Widiyanto et al. , *Implementation of Convolutional Neural Network Method for Classification of Diseases in Tomato Leaves;* 2019 Fourth International Conference on Informatics and Computing (ICIC)

[14] . Romana Rahman Ema, Sk. Shalauddin Kabir, Md. Nasim Adnan, Syed Md. Galib, *Tomato Disease Using Deep Learning* , 01 Jul 2023-International Journal of Computing-pp 191-201

[15]. N. Balakrishna, Gurram Sunitha, Avula Karthik, K. Reddy Madhavi, *Tomato Leaf Disease Detection Using Deep Learning: A CNN Approach. 08 Dec 2022-Vol. 01, pp 1-6*

[16] hikari, S., Unit, D., Shrestha, B., & Baiju, B. (2018). *Tomato Plant Diseases Detection System*. I(September 2018), 81–86. DOI:10.13140/RG.2.2.22135.68009

[17]. Kingma, D.P. and Ba, J., 2015. Adam: *A Method for Stochastic Optimization*. ICLR. 2015. arXiv preprint arXiv:1412.6980, 9.DOI: arXiv:1412.6980

[18] Tairu Oluwafemi Emmanuel.PlantVillage Dataset, version http ;// www.kaggle.com (2018,October)

[19] Hong, H., Lin, J. and Huang, F., 2020, June. *Tomato disease detection and classification by deep learning. In 2020 International Conference on Big Data, Artificial Intelligence and Internet* of Things Engineering (ICBAIE) (pp. 25-29

[20] Jiang, D., Li, F., Yang, Y. and Yu, S., 2020, August. A tomato leaf diseases classification method based on deep learning. In 2020 chinese control and decision conference (CCDC) (pp. 1446- 1450). IEEE. DOI: 10.1109/CCDC49329.2020.9164457

[21] Gehlot, M. and Saini, M.L., 2020, December. *Analysis of Different CNN Architectures for Tomato Leaf Disease Classification.* In 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE).

[22] Muchtar, K., Chairuman, C., Fitria, M., Kardawi, M.Y., Febriana, A., Zarima, N. and Lin, C.Y., 2021, October. *Embedded-based Tomato Septoria Leaf Detection with Intel Movidius Neural Compute Stick.* In 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE) (pp. 907-908).

[23] . Janocha ,Ka.,Czarnecki,W.M.:On *Loss Function for Deep Neural Network in Classification* (2017).arXiv:1702.05659.

[24] Tairu Oluwafemi Emmanuel.PlantVillage Dataset, Version ,https://www.kaggle.com(/2018, October).