Blight Diagnosis

G.Ashwini Kumari[1], Suri Dheeraj Sai[2], Alex Abrahim[3], Bojja Ritwik[4]
[1]*Assistant Professor, CSE, CMR Technical Campus, Hyderabad, India*
[2]*Student, CSE, CMR Technical Campus, Hyderabad, India*
[3]*Student, CSE, CMR Technical Campus, Hyderabad, India*
[4]*Student, CSE, CMR Technical Campus, Hyderabad, India*

***Abstract*: The early detection of diseases is important as it has social, ecological, and economic impacts on agriculture. Automatic methods for classification of crop diseases also help with taking action after detecting the symptoms of crop diseases as it plays a vital role in terms of rural employment and GDP of the country. The objective of this work is to develop a model which can predict those diseases so that the farmers can take appropriate action. This work can be done using a Convolution Neural Networks model using a dataset with 10 fold cross-validation technique to identify common crop diseases. With this model, we overcome the problem of crop diseases. The early detection of diseases is important as it has social, ecological, and economic impacts on agriculture. Automatic methods for classification of crop diseases also help with taking action after detecting the symptoms of crop diseases as it plays a vital role in terms of rural employment and GDP of the country. The objective of this work is to develop a model which can predict those diseases so that the farmers can take appropriate action. This work can be done using a Convolution Neural Networks model using a dataset with 10 fold cross-validation technique to identify common crop diseases. With this model, we overcome the problem of crop diseases.**

***Keywords*: RNN ML Algorithm, IOT software, Real Time GPS, TensorFlowLite.**

1. Introduction

India is a developed country and about 70% of the populace relies upon agribusiness. Farmers have an enormous scope of variety for choosing different reasonable yields and tracking down the appropriate pesticides for plants. Infection on plants prompts a critical decrease in both the quality and amount of farming items. The investigations of plant sickness allude to the investigations of outwardly perceptible examples on the plants. Checking of wellbeing and infection on plants assumes a significant part in the fruitful development of yields in the ranch. In the early days, the observing and investigation of plant illnesses were done physically by the aptitude individual around there. This requires a colossal measure of work and furthermore requires unreasonable handling time. The picture preparing procedures can be utilized in the plant sickness discovery. In a large portion of the cases, sickness manifestations are seen on the leaves, stem, and natural products. The plant leaf for the location of infection is viewed as which shows the sickness manifestations. This paper gives the prologue to the picture preparing strategy utilized for plant sickness recognition.

2. Proposed System

In the proposed framework from the outset the pictures are procured from the farmer. The pictures are gotten from the farmer by means of the Android Application assistance of the farmer. At that point picture will be resized in suitable arrangement then it will be transferred on worker on which a calculation is carried out utilizing Convolutional Neural Network Each Convolutional Neural Network engineering is isolated into two sections initially is include extraction and second is grouping and has four primary segments.

1. Convolutional activity.

2. Max-pooling (Down examining)

3. ReLu (Non Linearity)

4. Grouping (completely associated layer)

When picture is reached to worker it is prepared with calculation here we separate the element of picture with convolutional activity by convolving the channel over picture which delivers the element guides like edges, surface, spots, openings, shading. These highlights maps are down tested so it very well may be passed to completely associated layer for example classifier after each layer we apply ReLu for example non linearity so tackle complex issue like order .After this the machine learning code is converted in tflite file which is kept in the asserts folder in android studio and a text file is created . The Image which is captured by the user is taken in the form of bitmap and it is compared with the images in tflite file and the result is computed accordingly .The proposed model was evaluated based on Plant VillageDataset, which achieved around 98% accuracy.

This ML model consists of three major parts : Building and creating a machine learning model using TensorFlow with Keras Deploying the model to an Android application using TFLite.

Documenting and open-sourcing the development process.

The references are mentioned at the end of the paper.

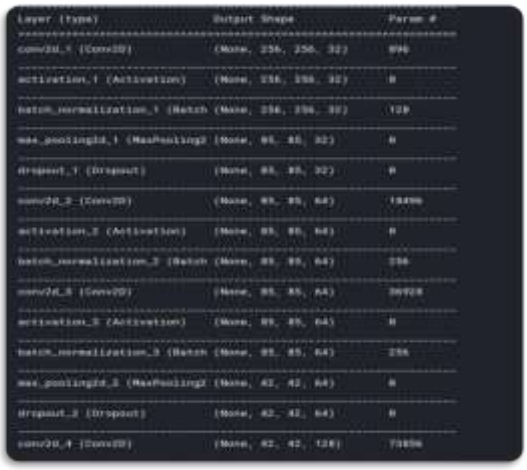
3. Methodology

A.. System Architecture

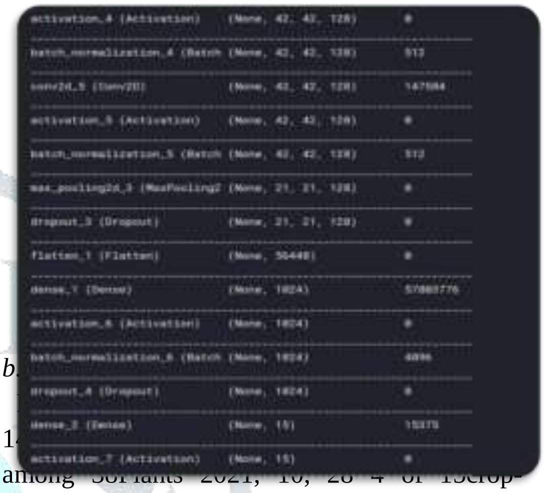
This part portrays the means associated with making and conveying the classifier. Arrangement by CNN is partitioned into three stages that tackle separate undertakings. They are convolutional layers, pooling layers, and enactment capacities, ordinarily Rectified Linear Units (ReLUs). The number of layers utilized, their plan, and the presentation of other handling units differ starting with one design then onto the next, deciding their particularity.

Fig. 1: Cnn architecture

Below are the layers used in CNN along with activation function and the total params are 58,102,671 the trainable params are 58,099,791 And non-trainable params are 2,880.



Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 256, 256, 32)	896
activation_1 (Activation)	(None, 256, 256, 32)	0
batch_normalization_1 (Batch	(None, 256, 256, 32)	128
max_pooling2d_1 (MaxPooling2	(None, 85, 85, 32)	0
dropout_1 (Dropout)	(None, 85, 85, 32)	0
conv2d_2 (Conv2D)	(None, 85, 85, 64)	18496
activation_2 (Activation)	(None, 85, 85, 64)	0
batch_normalization_2 (Batch	(None, 85, 85, 64)	256
conv2d_3 (Conv2D)	(None, 85, 85, 64)	36928
activation_3 (Activation)	(None, 85, 85, 64)	0
batch_normalization_3 (Batch	(None, 85, 85, 64)	256
max_pooling2d_2 (MaxPooling2	(None, 42, 42, 64)	0
dropout_2 (Dropout)	(None, 42, 42, 64)	0
conv2d_4 (Conv2D)	(None, 42, 42, 128)	73856



activation_4 (Activation)	(None, 42, 42, 128)	0
batch_normalization_4 (Batch	(None, 42, 42, 128)	512
conv2d_5 (Conv2D)	(None, 42, 42, 128)	147584
activation_5 (Activation)	(None, 42, 42, 128)	0
batch_normalization_5 (Batch	(None, 42, 42, 128)	512
max_pooling2d_3 (MaxPooling2	(None, 21, 21, 128)	0
dropout_3 (Dropout)	(None, 21, 21, 128)	0
flatten_1 (Flatten)	(None, 56448)	0
dense_1 (Dense)	(None, 1024)	57803776
activation_6 (Activation)	(None, 1024)	0
batch_normalization_6 (Batch	(None, 1024)	4096
dropout_4 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 15)	15375
activation_7 (Activation)	(None, 15)	0

b.

14

among 38Plants 2021, 10, 28 4 of 15crop-infection sets. This dataset contains clear pictures of plant leaves and each picture contains just one leaf. It likewise accompanies preset preparing/testing subsets which we continue in this investigation. We utilize this dataset in two distinct manners, first to arrange sickness crop pair and second to group infection paying little heed to the influenced crop. In the first configuration,32 classes (C7 to C38) are utilized as the source area Ds and 6 classes (C1 to C6) as the objective space Dt-like [38]. For this situation, the source area has 43,444 examples while the objective/novel space has 10,861 examples. In the subsequent arrangement, we revamp the dataset as indicated by the basic sickness name as in, bringing about 20 illness classes and one solid class. For infections that influence more than one plant, all examples are consolidated in one class under the name of this illness, and for the sound class, around 5000 examples were gathered from all accessible plants. The three infections with the least pictures were chosen from the adjusted dataset as target (space classes and the rest as the source area classes. The quantity of tests in the source area Ds is 44,081 and in the objective space, Dt is 1278.

Class	Crop	Disease	Samples	
			Train	Test
C23	Potato	Late blight	800	200
C24	Raspberry	Healthy	297	74
C25	Soybean	Healthy	4072	1018
C26	Squash	Powdery mildew	1468	367
C27	Strawberry	Healthy	364	92
C28	Strawberry	Leaf scorch	887	222
C29	Tomato	Bacterial spot	1702	425
C30	Tomato	Early blight	800	200
C31	Tomato	Healthy	1273	318
C32	Tomato	Late blight	1527	382
C33	Tomato	Leaf mold	761	191
C34	Tomato	Septoria leaf spot	1417	354
C35	Tomato	Spider mites Two-spotted spider mite	1341	335
C36	Tomato	Target spot	1123	281
C37	Tomato	Tomato mosaic virus	299	74
C38	Tomato	Tomato Yellow Leaf Curl Virus	4286	1071

Class	Crop	Disease	Samples	
			Train	Test
C23	Potato	Late blight	800	200
C24	Raspberry	Healthy	297	74
C25	Soybean	Healthy	4072	1018
C26	Squash	Powdery mildew	1468	367
C27	Strawberry	Healthy	364	92
C28	Strawberry	Leaf scorch	887	222
C29	Tomato	Bacterial spot	1702	425
C30	Tomato	Early blight	800	200
C31	Tomato	Healthy	1273	318
C32	Tomato	Late blight	1527	382
C33	Tomato	Leaf mold	761	191
C34	Tomato	Septoria leaf spot	1417	354
C35	Tomato	Spider mites Two-spotted spider mite	1341	335
C36	Tomato	Target spot	1123	281
C37	Tomato	Tomato mosaic virus	299	74
C38	Tomato	Tomato Yellow Leaf Curl Virus	4286	1071

Fig 3 : Dataset diseases

C.MACHINE LEARNING

1) Stage one:-

Stage one means to explore the impact that picture size has on model execution. Altogether, five pictures estimated are tried going from 150 x 150 to 255 x 255. As a default of move learning, all layers with the besides of the last two layers are frozen. These contain new loads and are explicit to the plant illness characterization task. Freezing permits these layers to be illness independently prepared, without back-propagating the inclinations. In precisely thusly, the 1cycle approach is utilized to prepare the last layers. With this total, the leftover layers are delivered. To help the calibrating interaction, a plot showing learning rate versus misfortune is produced and investigated. From this, appropriate learning is chosen, and the model is run. With results recorded, the model is re-made to the extra four picture sizes.

2) Phase two:-

Utilizing the most appropriate picture size, the Cnn model is advanced. To additionally improve the model's exhibition, extra expansion settings are added. Tasks incorporate brilliance changes (0.4,0.7) and twists (0.5). Then, the last two layers are disengaged and prepared at the default learning rate. With this total, tweaking is performed, running numerous preliminaries to test a progression of learning rates and a number of epochs.

The convolutional layer is the fundamental structure square of the convolutional neural organization. The layer's boundaries are included a bunch of learnable pieces which have a little responsive field yet reach out through the full profundity of the information volume.

Each convolutional layer has M maps of equivalent size, Mx and My , and a piece of size kx, and ky is moved over the specific district of the info picture. The skipping factors Sx and Sy characterize the number of pixels the channel/portion avoids in x- and y- heading between ensuing convolutions. The size of the yield guide could be characterized as

$$M_x^n = \frac{M_x^{n-1} - K_x^n}{S_x^n + 1} + 1,$$

$$M_y^n = \frac{M_y^{n-1} - K_y^n}{S_y^n + 1} + 1,$$

where n demonstrates the layer. Each map in layer Ln is associated with most maps Mn-1 in layer. Corrected Linear Units (ReLU) are utilized as an alternative for soaking nonlinearities. This enactment work adaptively learns the boundaries of rectifiers and improves precision at the insignificant extra computational expense. It is characterized as

$$F(z_i) = \max(0, z_i),$$

where zi addresses the contribution of the nonlinear initiation function f on the ith channel. Profound CNN with ReLUs trains a few times quicker. This technique is applied to the yield of each convolutional and completely associated layer. Regardless of the yield, the info standardization isn't needed; it is applied after ReLU nonlinearity after the first and second convolutional layer since it decreases top-1 and top-5 mistake rates. In CNN, neurons inside a secret layer are divided into "include maps." The neurons inside a component map share a similar weight and predisposition. The neurons inside the element map look for a similar component. These neurons are extraordinary since they are associated with various neurons in the lower layer. So for the primary secret layer, neurons inside a component guide will be associated with various districts of the information picture. The secret layer is sectioned into highlight maps where every neuron in an element map searches for a similar component yet at various places of the information picture. Essentially, the component map is the consequence of applying convolution across a picture. Each layer's highlights are shown in an alternate square, where perception addresses the most grounded initiation for the gave include map, beginning from the first convolutional layer, where highlights go from singular pixels to straight lines, to the fifth convolutional layer where learned highlights like shapes and certain pieces of leaves are shown.

Another significant layer of CNNs is the pooling layer, which is a type of nonlinear downsampling. Pooling activity gives the type of interpretation invariance it works freely on each profundity cut of the info and resizes it spatially. Covering pooling is advantageously applied to decrease overfitting. Additionally for decreasing overfitting, a dropout layer is utilized in the initial two completely associated layers. Yet, the weakness of dropout is that it builds preparing time 2-3 times contrasting with a standard neural organization of the specific design. Bayesian enhancement tests likewise demonstrated that ReLUs and dropout have collaboration impacts, which implies that it is invaluable when they are utilized together.

The development of CNNs allude to their capacity to learn rich mid-level picture portrayals instead of hand-planned low-level highlights utilized in other picture grouping strategies

Figure 4 shows the separated yield pictures after each convolutional and pooling layer of the profound organization. Yield pictures are marked with the name of the relating layer at the base right corner of each picture.

Fig. 4. Output Layers

D Performed tests:-

The basic methodology in estimating the execution of artificial neural organizations is parting information into the preparation set and the test set and afterward preparing a neural network on the training set and utilizing the test set for prediction. Accordingly, since the first results for the testing set and our model anticipated results are known, the exactness of our expectations can be determined. Various tests were performed with .50000 images from the database.

For the precision test, 10-fold cross-validation predictive was utilized to evaluate a prescient model. The cross-validation technique was repeated after each thousand preparing cycles. In generally assessed consequence of the test is graphically addressed as top-1, to test if the top class (the one having the most noteworthy likelihood) is equivalent to the objective marker. The top 5 error rate is there to test if the target label is one of the top 5 predictions, the ones with 5 of the greatest probabilities.

E Fine-tuning

Fine-tuning assists with expanding the exactness of expectation by making little adjustments to improve or upgrade the result. The most appropriate model for plant disease detection will be accomplished through the cycle of exploratory change of the boundaries. The green line in the diagram in Figure 5 shows the organization's prosperity on the validation test set, through preparing iterations. After each 10 thousand preparing iterations, the depiction of the model was gotten. The blue line in the diagram addresses the misfortune during the preparation stage. Through Training iterations, the loss was quickly decreased.

Fig 5 : Traning and validation accuracy

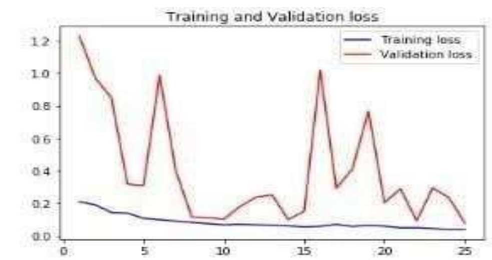


Fig 6 : Training and validation accuracy

4. Results and Discussions

The cnn architecture pictures the hidden layer output for each layer and its generated intermediate outputs are yields are summarized. In our trained model, a portion of the intermediate outputs in the shallow layers (Conv1, Conv5) feature the yellow and earthy colored injuries that are evident inside the picture . Be that as it may, in the more profound layer , attributable to the convolution and pooling layers, the picture size is too little to even think about interpretting whether such removed highlights have been held. Additionally, the worldwide normal pooling layer changes pictures over to a component vector that disposes of the spatial data, making it profoundly hard to see how the highlights are taken care of in continuing layers. It is hard to recognize whether the extricated includes decidedly add to the grouping of the info picture to the right infection class or are utilized for motivation to deny different potential outcomes Hence, understanding what the CNN has realized by just investigating the halfway yield is lacking.

Fig. 7. Main Page

After opening the application, The above page is displayed. There we can see two sections at the bottom which are named as Classify and History.

We can see two buttons named as Camera and Gallery. The users can select their preferred option.

Fig 8: Disease Selection

Whenever user clicks Gallery button it prompts to select among various options like Photos, Gallery and File Manager. The user can select any option and can choose the image of the diseased plant from his storage. The user can also select Camera option and it redirects him to his phone camera where he can manually capture the picture of the crop and it automatically processes the image.

Fig. 9. Disease Identification

Whenever user selects the diseased image the application processes it and identifies the plant ddisease and gives the output as shown in Fig 9.

Fig. 10. History

The application has a feature which saves the user past activity so that he can view it whenever needed. So that it can save his time.

5. Conclusion

Recent studies on crop diseases show how they harm the plants. For detecting the crop diseases, people are using the conventional method, human vision-based approach which is time consuming .In this paper, we have proposed a custom CNN-based model that can classify common crop diseases which are commonly found in leaves. Our new system is cost effective , simple and error free. Detecting the diseases from the crops helps in protecting the leaves surrounding it . Practical results might help in identifying other crop diseases.

6. References

[1] Neural Networks: H.A. Rowley , Dept. of Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, USA.

[2] Deep Residual Learning for tomato plant leaf disease identification – Habibollah agh atabay.Department of computer, Gonbad Kavous university , Gonbad kavous Iran.

[3] Identifying two of tomatoes leaf viruses using support vector machine, in: Information Systems Design and Intelligent Applications : U. Mokhtar, M.A. Ali, A.E. Hassanien, H. Hefny.

[4] Large-scale machine learning on heterogeneous distributed systems : M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin.

[5] Deep Residual Learning for Image Recognition : Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun

[6] Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks : Alec Radford, Luke Metz, Soumith Chintala.

[7] Learning Activation Functions to Improve Deep Neural Networks : Forest Agostinelli, Matthew Hoffman, Peter Sadowski, Pierre Baldi

[8] Applications of Convolutional Neural Networks : Ashwin Bhandare, Maithili Bhide, Pranav Gokhale, Rohan Chandavarkar .

[9] Gradient basel learning applied to document recognition: Yann LeCunn, Leon Bottou, Yoshua Bengio, Patrick Haffner.

[10]"PlantVillage", Plantvillage.psu.edu, 2020.

[11]D. Klauser, "Challenges in monitoring and managing plant diseases in developing countries".

[12]A.Muimba-Kankolongo, Food crop production by smallholder farmers in Southern Africa. Elsevier, 2018, pp. 23-27.

[13]J. Boulent, S. Foucher, J. Théau and P. St-Charles, "Convolutional Neural Networks for the Automatic Identification of Plant Diseases", Frontiers in Plant Science, vol. 10, 2019.

[14]"iNaturalist", iNaturalist, 2020.

[15]I. Ltd., "Success-stories - PlantSnap: Training the world's largest plant recognition classifier. | Imagga Technologies Ltd.", Imagga.com, 2020. .

[16]"Smartphone users worldwide 2020 | Statista", Statista, 2020.