Available online at : http://www.advancedscientificjournal.com
http://www.krishmapublication.com
IJMASRI, Vol. 3, issue 1, pp.922- 926, Dec. -2023
https://doi.org/10.53633/ijmasri

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY ADVANCED SCIENTIFIC RESEARCH AND INNOVATION (IJMASRI)

ISSN: 2582-9130 **IBI IMPACT FACTOR 1.5** **DOI: 10.53633/IJMASRI**

RESEARCH ARTICLE

IMAGE-BASED PLANT DISEASE DETECTION USING DEEP LEARNING

Mandeep Singh

Department of Information Technology, Maharaja Agrasen Institute of Technology, Rohini, Delhi
Email: hapysingh1313@gmail.com

Abstract

In this study, we are forecasting diseases connected to the potato plant using machine learning algorithms and image processing techniques. Because of the illnesses, a significant portion of the potato plants is lost, which harms the farmer's yield. We are putting out a technique that will be able to identify plant illness by examining symptoms associated with the plant's leaf to reduce the number of farmers lost to the disease of the potato plant. It will be able to identify plant-related disease and offer some recommendations for disease prevention and treatment. This image will be classified using our model, and the outcomes will be displayed alongside the Convolution Neural Networks-labelled photos.

Keywords: Plant Disease, CNN, Prediction, Deep Learning, Image-Based, TensorFlow

Introduction

Because plants are exposed to the outside environment and are very susceptible to illnesses, the prevention and control of plant disease have long been hotly debated topics. The prompt and precise identification of disease is typically crucial for controlling plant disease since effective protective measures are frequently put in place following a proper diagnosis. We are employing image processing and machine learning techniques, which will aid us in the process, to identify the sickness of the potato plant. By examining the image of the potato plant leaves, we can identify diseases that affect the potato plant. Based on our model, the image of the plant leaves will be categorised. The image of the plant leaves will be classified based on our model. Multiple pre-processing will be done on the image and then it will be fed to the prediction model which will tell us about the disease of the potato plant.

Plant diseases, poor nutrition, natural catastrophes, and other factors are major contributors to crop loss. Food shortages, price increases, and even farmer suicide are caused by crop failure. 17% of farmer suicide cases are the result of crop failure.

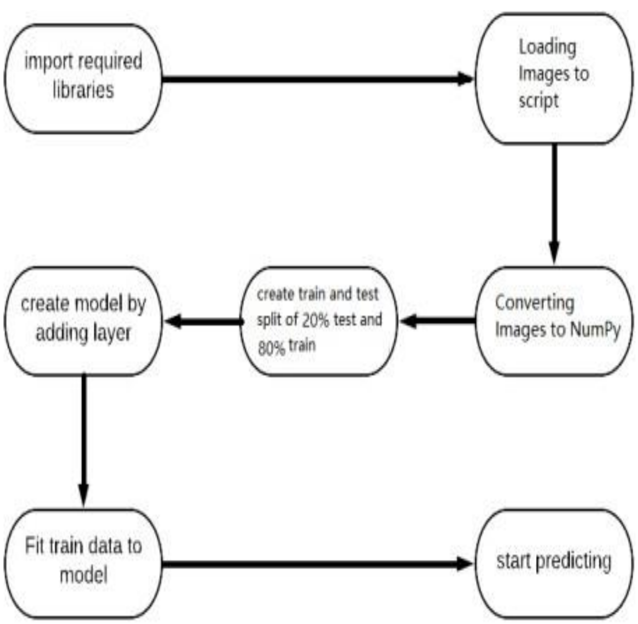
Diseases cause a significant portion of the potato and cotton plants to die, which has a negative impact on the farmer's yield. To prevent farmers from losing their livelihoods due to diseases of the cotton and potato plants, we are putting forth a method that can identify plant diseases by examining symptoms in the plant's leaves.

Proposed Methodology:

The algorithms for object detection and picture classification actively utilise convolution neural networks. In this effort to identify potato plant illness, Convolution Neural Networks (CNN) are used for model training and classification.

With the aim of identifying the potato plant disease by taking photographs with our special model, we trained our machine learning model on the images of the potato plant leaves using the deep convolution neural network architecture. There are 10,000 photos of potato plant leaves that are categorised into three illnesses that affect potato plants that are included in the data collection. We have attained the highest accuracy of 99.35% using our model.

Based on the three classes in 993 out of the 1000



photos, the model properly categorises potato plant disease and determines if it is healthy. Due to the low CPU requirements of our model and the speed at which the classification is performed, it may be easily employed in mobile devices, web applications, and other platforms.

If the image is tainted, that is, if the background is not uniform, then our algorithm can quickly identify the disease connected to potato plants. Additionally, our model can quickly identify the features in the image of potato plant leaves and divide it into three different groups. Our model will categorise the image and produce a list of potential classes along with their numerical probabilities.

Dataset Collection:

We run all of our experiments across a wide range of train-test set splits, including 80-20 (80% of the whole dataset used for training, and 20% for testing), 60-40 (60% of the whole dataset used for training, and 40% for testing), 50-50 (50% of the whole dataset used for training, and 50% for testing), and 40-60 (40% of the whole dataset use for training, and 60% for testing), to get a sense of how our approaches will perform on new unseen data, as well as to keep track

It should be noted that the Plant Village dataset frequently contains multiple images of the same leaf (taken from various orientations), and we have the mappings for such cases for 41,112 images out of the 54,306 images. We also make sure that during all test-train splits, all the images of the same leaf go either in the training set or the testing set. Additionally, we compute the mean precision, mean recall, mean F1 score, and overall accuracy across the whole training period for each trial at regular intervals (at the end of every epoch). For the purpose of comparing the outcomes of all of the various experimental settings, we utilise the final mean F1 score.

Pre-processing:

Images are resized into dimensions of 256 x 256 and loaded into the NumPy array of images. We take just the images of single leaves attached to the potato plants.

Performed model optimization and prediction, both on the downscaled images. We obtained three different versions of the Original Dataset by performing image operations and transformations.

Images are resized into dimensions of 256 x 256

The versions are defined below one by one:

In color: In this version, various transformations are applied only to the dimensional factors like angle, height and width, orientation. Keeping the colors untouched and original.

Gray-scaled: In this version, the dimensions, straighten property(angle), and orientation is kept original. Only the colors are downscaled to gray. 20

Segmented: In this version the leaves are segmented from every image of the Original Dataset i.e. the segmented data only contains the leaves only, not even the background. Segmentation can decrease the Dataset biasing and also reduces size. Thus, it also improves processing speed.

Model Training:

We trained our model with 50 epochs on Google's Data Science Community Kaggle.

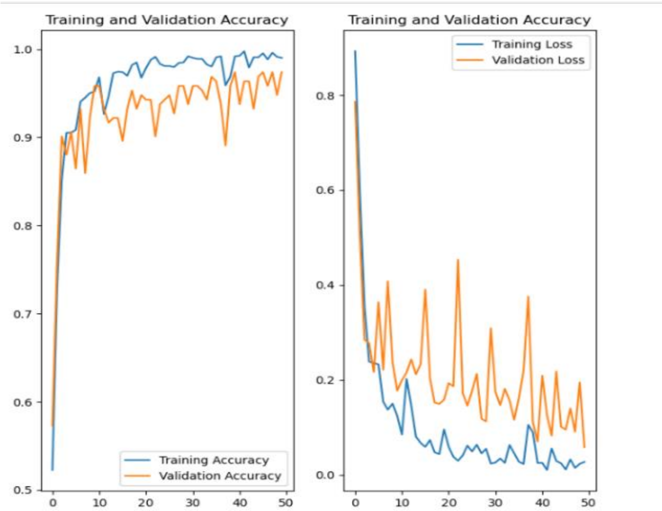
```
In [67]: model.summary()

Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential (Sequential)     (32, 256, 256, 3)         0

 sequential_1 (Sequential)   (None, 256, 256, 3)       0

 conv2d_12 (Conv2D)          (None, 254, 254, 32)      896

 max_pooling2d_12 (MaxPoolin  (None, 127, 127, 32)     0
 g2D)

 conv2d_13 (Conv2D)          (None, 125, 125, 64)      18496

 max_pooling2d_13 (MaxPoolin  (None, 62, 62, 64)       0
 g2D)

 conv2d_14 (Conv2D)          (None, 60, 60, 64)        36928

 max_pooling2d_14 (MaxPoolin  (None, 30, 30, 64)       0
 g2D)

 conv2d_15 (Conv2D)          (None, 28, 28, 64)        36928

 max_pooling2d_15 (MaxPoolin  (None, 14, 14, 64)       0
 g2D)

 conv2d_16 (Conv2D)          (None, 12, 12, 64)        36928

 max_pooling2d_16 (MaxPoolin  (None, 6, 6, 64)         0
 g2D)

 conv2d_17 (Conv2D)          (None, 4, 4, 64)          36928

 max_pooling2d_17 (MaxPoolin  (None, 2, 2, 64)         0
 g2D)

 flatten_1 (Flatten)         (None, 256)               0

 dense (Dense)               (None, 64)                16448

 dense_1 (Dense)             (None, 3)                 195
```

Which provides us the resources for training the model in the best time by the use of GPUs. As the learning graph becomes constant after a certain number an epoch hence, we go with 100 (not too less and not too much). The Batch size used for training is 64. It took around 20 minutes to complete the learning process. The evaluation of model is provided in the next chapter of this report.

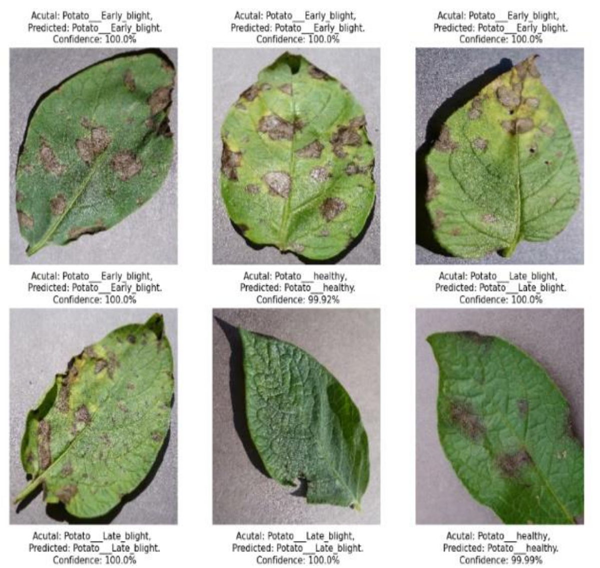
Layers are stacked up in the following manner: A convolution layer with 32 filters, a filter size of 3 and an activation function relu.

- An average pooling layer with a window size of 2.
- A convolution layer with 32 filters, a filter size of 3 and an activation function relu.
- An average pooling layer with a window size of 2.
- A convolution layer with 64 filters, a filter size of 3 and an activation function relu.
- An average pooling layer with a window size of 2.
- A Flatten layer with default parameters.
- A dense layer with 64 units and an activation function relu.

Outputs:



Training and Validation Accuracy Graph



Output results for test batch consisting 10% of the original data with confidence percentage

Conclusion:

We start off by pointing out that, using test and train data, we were able to obtain accuracy of 99.25 on a dataset with 9 class labels. However, the real-world prediction's accuracy can range from 70 to 100 percent (rounded off), depending on a number of factors, including the size, colour saturation, angle, and segmentation of the leaf.

The system described here is not intended to replace any existing systems for the detection of plant diseases; rather, it will enhance the functionality of current systems and assist in their modernization.

It is sometimes very difficult to detect the sickness of the plant solely on the basis of their visual inputs, making laboratory tests ultimately more trustworthy than detection based solely on photographs of plant leaves. Technology advancements have led to a considerable improvement in the image quality taken by mobile devices, which will allow our system to forecast events with great precision and certainty. An increase in image resolution will enable the capture of more information about plant leaves, improving prediction and increasing accuracy.

Future Scope:

- In future, following work can be done in this field:
- The accuracy and stability of the model can be increased.
- Addition of more Neural Networks and Deep Learning Modules.
- Making easy access and communication among the users.
- Creating a more attractive interface.
- Creating an Integrated Platform where all the different modules are attached as a single module.
- The model can be used for transfer learning.

Reference

1. Prakash, R., Saraswathy, G.P and G Ramalakshmi. (2017). Detection of leaf diseases and classification using digital image processing IEEE International Conference on Innovations in Information, Embedded and Communication Systems (2017), pp. 1-4.
2. Pooja, V., Das, R and Kanchana, V. (2017). Identification of plant leaf diseases using image

processing techniques IEEE International Conference on Technological Innovations in ICT for Agriculture and Rural Development (2017), pp. 130-133.

3. Saleem, M.H., Potgieter. Arif K Mahmood. (2019). Plant Disease Detection and Classification by Deep Learning Plants (Basel), 8 (11) (2019), pp. 1-22.
4. Using Deep Learning for Image-Based Plant Disease Detection methods article Front. Plant Sci., 22 September 2016 Sec. Technical Advances in Plant Science.
5. Using Deep Learning for Image-Based Plant Disease Detection https://www.frontiersin.org/articles/10.3389/fpls.2016.01419/full.
6. Plant Village Dataset Dataset of diseased plant leaf images and corresponding labels This dataset was gotten from spMohanty's GitHub Repo Kaggle.

*****