IDENTIFYING PLANTS DISEASE AND PROVIDE PLANT DISEASE SUPPLIMENTS DETECTION SYSTEM USING CNN MODEL

Anushka Srivastava, Asmita Singh, Atishay Jain, Avneesh Kumar

B. Tech Student

Department of Computer Science and Engineering

Inderprastha Engineering College, Sahibabad, Ghaziabad, India

Priya Upadhyay

Professor and Mentor

Abstract: In ancient times, Plants and crops that are afflicted by pests or diseases have an impact on agricultural productivity. Generally, farmers and professionals examine plants with their naked eyes in order to discover and identify illness. However, this procedure is time-consuming and frequently wrong.

In recent years, technological advancements have spawned a slew of new ways to complement old procedure. In picture classification challenges, deep learning approaches are particularly powerful and successful. This purpose of this study is to present a novel method to the construction of a disease recognition model using CNN, which supports plant leaf image classification utilizing convolutional networks and the Deep Learning algorithm.

Using the deep convolutional neural network that we will train, test, and validate. The model is able to discriminate between damaged and healthy leaves. Plants are the world's primary food source so infections and illnesses are a significant hazard, and the most frequent method of diagnosing plant diseases is to examine the plant body for visible signs and growth. Different research efforts intend to identify realistic techniques to plant protection and support our farmers as an alternative to the old time-consuming process.

Introduction: This method represents a technological advancement by employing a deep learning approach – CNN model, with a dataset titled "plant village dataset," which contains images in 39 different classes of different plants and background images, with each class containing nearly 60,000 images of leaves of different plants predicting whether they are healthy or not.

To increase the size of our data collection, we are employing six Augmentation approaches. Image flipping, gamma correction, noise injection, PCA color augmentation, rotation, and scaling are the six data augmentation techniques utilized in diagnosing plant diseases. It has been effectively implemented in every discipline, including end-to-end learning, image classification, data augmentation, and so on.

Neural networks translate an input, such as a picture of a damaged plant, to an output, such as a crop-disease pair. A neural network's nodes are mathematical functions that accept numerical inputs from the incoming edges and provide a numerical output as an outgoing edge.

Here, we're extracting features with CNN and utilizing them to build our model, which we'll train further. We are defining the filter size for the conv layer and the pool layer, as well as the form of each layer. Then, we employ the torch framework to train, validate, and test our model.

Deep neural CNN networks are taught by fine-tuning the network parameters so that the mapping improves with time. This is a difficult process that has been enhanced by a variety of conceptual and engineering advances.

Literature Survey: The paper presents a novel approach for detecting disease in plants by training a convolutional neural network.

The proposed approach is based on the Deep learning for image-based plant detection" by Prasanna Mohanty.[1] This is better than a simple model of random selection, a more diverse set of training data can help to increase the accuracy. Other variations of model or neural network training may also yield higher accuracy, paving the way for everyone to easily detect plant disease.

Malvika Ranjan [2] proposed an approach to detect diseases in plants using the captured image of the diseased leaf in the paper "Detection and Classification of Leaf Disease Using Artificial Neural Network."

"Detection of unhealthy regions of plant leaves and classification of plant leaf diseases using texture features," according to the paper [3] According to S. Arivazhagan, the disease identification process consists of four major steps: The colour transformation structure is used first for the input RGB image, and then the green pixels are detected and uninvolved.

Kulkarni et al. in the paper ―Applying image processing technique to detect plant diseases" [4], a methodology for early and accurately plant diseases detection, using artificial neural network (ANN) and diverse image processing techniques.

Finally, classifier is used to classify the disease based on the extracted features.Jyotsna Bankar et al. proposed the use of the inception v3 model in classifying animals of various species in their paper Convolutional Neural Network Based Inception v3 Model for Animal Classification [5].

Proposed Method: Plants are prone to a variety of disease-related illnesses and convulsions caused by environmental factors such as temperature, humidity, excess or inadequate food, light, and the most prevalent illnesses such as bacterial, viral, and fungal infections. We employ the CNN algorithm in the suggested system to identify illness in plant leaves since it achieves the highest accuracy if the data is good.

A. Dataset: This collection contains 39 distinct kinds of plant leaf and background photos. There are around 60,000 photos in the collection. To increase the size of the data-set, we applied six distinct augmentation approaches. There are total of 39 Classes to forecast using the CNN Model.

B. Performance Evaluation: We run all of our experiments across a range of train-test set splits to get a sense of how our approaches will perform on new unseen data, as well as to keep track of if any of our approaches are overfitting.

These splits are 80–20 (80% of the whole dataset used for training, and 20% for testing), 60–40 (60 percent of the whole dataset used for training, and 40% for testing), and 50–50 (50 percent of the whole dataset used for training, and 50% for testing (20 percent of the whole dataset used for training, and 80 percent for testing), shape = (channels, height , width). The overall calculation during evaluation is also called Convolutional Arithmetic.

C. Data Augmentation:

•Image Flipping - Depending on the item in the image, the image can be flipped horizontally or vertically. It also allows us to flip images in the left-right and up-down directions.

•Gamma Correction - This adjusts the image's overall brightness. Also including changes in the red-green-blue ratios (RGB).

•Noise Injection - Adding noise to a model during training can increase the network's resilience, resulting in improved generalization and faster learning. (Using NumPy in Python, add noise to a single). Create random noise using a variable called 'noise.' - Make Dataset noisier (Dataset = Dataset + noise). - Divide the Noise Dataset into three sections: - 1. 70% of the time is spent on training. 2. Validation is worth 15%. 3.15% is set aside for testing.

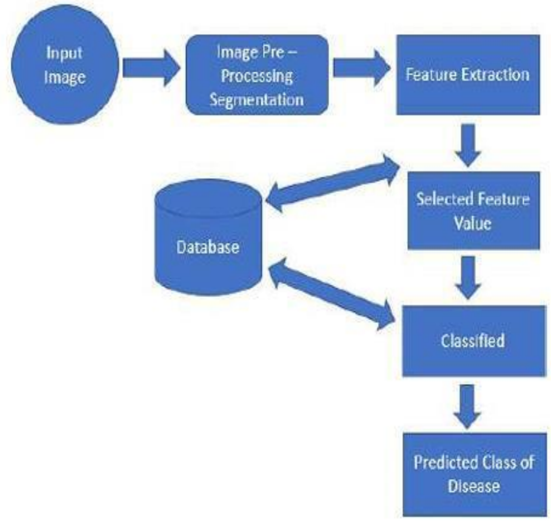
• **PCA Color Augmentation -** Atter RGB channel intensities following natural picture variances.

• **Rotation** - A source picture is rotated by a random number of degrees clockwise or anticlockwise, altering the location of an item in the frame, ranging from 0 to 360 degrees.

• **Scaling -** We choose a small picture size within a certain dimension range at random. We're utilizing PyTorch in our project, and we're using transformations for Data Augmentation, which acts as a filter for all photos.

D. Image Acquisition: This is the method of obtaining photos using a camera by visiting the location or using other accessible resources such as image databases or internet repositories. The pictures are taken in three colors: red, green, and blue (RGB), for which a color transformation structure is developed and a device independent color transformation is implemented.

E. Image Pre-processing: Firstly, we should have indices and then divide the data into train, test and validation data. Then, we create an object of Subset Random sampler which is generally used to sample our data and later we use this sampler to train and test the data loader.



SYSTEM OVERVIEW:

Feature Selection: Feature selection is a critical step in all machine learning problems. For model creation, we employ a convolutional neural network. In the model, we use ReLU as the activation function to remove non-linearly, Batch Normalization to normalize the neuron weights, but for the final layer, we must use SoftMax activation. In PyTorch, we have a cross-entropy loss that is a hybrid of SoftMax and category cross-entropy loss.

Classification Algorithm: For classification, CNN was employed. The training dataset accounts for 80% of the total, while the testing dataset accounts for the remaining 20%. Following fine-tuning, a pre-trained VGG16 model achieved roughly 87 percent on Train data, 84 percent on Validation data, and 83 percent on Test data. VGG-16 is composed of three main components:

● **Convolution layer-** Filters are applied to images in this layer to extract features. The size of the kernel and stride are the most important parameters.

● **Pooling layer-** Its purpose is to reduce the spatial size of a network in order to reduce the number of parameters and computations.

● **Fully Connected-** These are connections to previous layers that are fully connected, as in a simple neural network.

Conclusion:

This model demonstrated its value as a crucial tool for modern institutions and organizations. This conclusion reflects on the key takeaways and benefits of implementing such a system:

1.Accuracy and Efficiency: The Plant Village dataset was used to evaluate the accuracy and performance of the models. Despite the fact that this dataset contains a large number of images of various plant species and their diseases.

2.Reliability: Multispectral imaging is a new technology that has been used in a variety of fields of study. As a result, it should be combined with efficient DL architectures to detect plant diseases even before symptoms appear.

3.Visualization: A method of visualizing disease spots in plants should be implemented in order to save money as declined use of pesticide/fungicide/herbicide unnecessarily.

4.Real-Time Tracking: To understand the factors influencing plant disease detection, such as dataset classes and size, learning rate, illumination, and so on, a comprehensive study is required.

5.Adaptibility: Because the severity of plant diseases varies over time, DL models should be improved/modified to detect and classify diseases throughout their entire life cycle.

6.Potential: The DL model should be efficient for a wide range of lighting conditions, so the datasets should not only represent the real environment but also include images taken in various field scenarios.

References:

1. S. D. Khirade and A. B. Patil, "Plant Disease Detection Using Image Processing," 2015 International Conference on Computing Communication Control and Automation, 2015, pp. 768-771, doi: 10.1109/ICCUBEA.2015.153.

2. S. C. Madiwalar and M. V. Wyawahare, "Plant disease identification: A comparative study," 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), 2017, pp. 13-18, doi: 10.1109/ICDMAI.2017.8073478.

3. P. Moghadam, D. Ward, E. Goan, S. Jayawardena, P. Sikka and E. Hernandez, "Plant Disease Detection Using Hyperspectral Imaging," 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2017, pp. 1-8, doi: 10.1109/DICTA.2017.8227476.

4. S. D.M., Akhilesh, S. A. Kumar, R. M.G. and P. C., "Image based Plant Disease Detection in Pomegranate Plant for Bacterial Blight," 2019 International Conference on Communication and Signal Processing (ICCSP), 2019, pp. 0645-0649, doi: 10.1109/ICCSP.2019.8698007.

5. G. Shrestha, Deepsikha, M. Das and N. Dey, "Plant Disease Detection Using CNN," 2020 IEEE Applied Signal Processing Conference (ASPCON), 2020, pp. 109-113, doi: 10.1109/ASPCON49795.2020.9276722.

6. Mohanty SP, Hughes DP and Salathé M (2016) Using Deep Learning for Image-Based Plant Disease Detection. Front. Plant Sci. 7:1419. doi: 10.3389/fpls.2016.01419

7. R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural Features for Image Classification," in IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, no. 6, pp. 610-621, Nov. 1973, doi: 10.1109/TSMC.1973.4309314.

8.Breiman,L.Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324.