Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

PLANT LEAF DISEASE DETECTION USINGCONVOLUTIONAL NEURAL NETWORKS

SINGAMANENI VAMSI, POTHALA JHANSI, URITI NAGA VENKATA AKHIL, PATHIGULLA SUSHMA KEERTHI, Students, Dept.of.CSE, Raghu Institute of Technology. Dakamarri (V), Bheemunipatnam Visakhapatnam Dist. Pincode: 531162.
Mrs.GADDE SRIDEVI, Assistant Professor, Dept.of.CSE, Raghu Engineering College,Dakamarri (V), Bheemunipatnam Visakhapatnam Dist. Pincode: 531162.

Abstract

Farmers may suffer large financial losses as a result of plant diseases' considerable effects on crop productivity and quality. Plant diseases can be stopped from spreading and crop output can be increased with early identification. Plant disease identification can be done automatically with machine learning algorithms thanks to the growing availability of digital picture collections. In this paper, we propose a Convolutional Neural Network (CNN) based plant leaf disease detection system. The CNN model is trained and tested by the system using a dataset of plant leaf images of six distinct crops: apple, cherry, corn, grape, orange, and potato. The findings demonstrate that the suggested approach is capable of correctly identifying plant leaf diseases in the provided photos. In this paper, scaling was done during preprocessing. parameters such as rescaling, shuffling, dropout, rotation, background correction, zoom/brightness adjustment, and horizontal flipping, among others in order to transform our image data into enhanced image data, which will aid in the training of our CNN model using low-resolution images. Our goal is to evaluate the suggested models' success rate and contrast the results with those of alternative approaches.

Keywords: Leaf Diseases, Machine Learning, Deep Learning, Convolutional Neural Networks.

1. Introduction

One of the main factors causing crop loss globally is plant diseases, which can result in large financial losses and jeopardize the security of food supplies worldwide. Controlling the spread of infections and reducing crop losses depend heavily on the early detection and diagnosis of plant diseases. Plant inspection by hand is a labor- and time-intensive procedure, and it can be difficult to recognize early disease indicators with accuracy. As a result, an automated system that can precisely and effectively identify and categorize plant diseases is required.

Deep learning methods have made great strides in computer vision applications such as picture classification, face identification, and object recognition in recent years. One kind of deep learning model that has demonstrated promising performance in a number of image classification tasks is the convolutional neural network (CNN). CNNs are a great option for detecting plant diseases because of their ability to automatically extract features from photos and identify intricate patterns from data.

This paper's goal is to suggest a CNN-based system for detecting plant leaf disease. Accurately and effectively identifying plant diseases on leaves is the goal of the suggested technique. The system extracts features using a pre-trained CNN model and

makes use of transfer learning to optimize the model on the training set. The validation set is used to fine-tune the hyperparameters, while the testing set is used to assess the performance of the finished model.

2. Literature Survey

Several studies have been conducted to detect and diagnose plant diseases using different techniques.

Adhikari et al. (2020) suggested a method for classifying and detecting plant diseases that relies on deep learning. Using a dataset of photos of tomato leaves, they applied the transfer learning technique to optimize pre-trained CNN models like VGG16 and Inception-v3. Their findings demonstrated that the suggested method classified tomato leaf diseases with 96.4% accuracy.

Kemble et al. (2019) developed A CNN-based system for detecting leaf disease in mango plants. They gathered a collection of photos of mango leaves and utilized transfer learning to improve CNN models that had been pretrained, such Alex Net and Google Net. Their findings demonstrated that the suggested method identified mango leaf illnesses with an accuracy of 92.67%.

Saah et al. (2019) proposed a unique CNN-based method for identifying plant diseases. They gathered a collection of photos of potato leaves and employed a CNN model that had already been trained to extract features. To classify diseases, a Support Vector Machine (SVM) classifier was fed the features that were taken out of the CNN model. Their findings demonstrated that the suggested method classified potato leaf diseases with 94.44% accuracy.

3.Proposed Methodology

The CNN algorithm is used in the proposed plant leaf disease detection system, which is intended to automatically identify plant illnesses in leaf photos. To extract features from the input photos, the system makes use of a pretrained CNN model. In order to forecast the disease class, the retrieved features are further routed through fully linked layers. To enhance the model's performance, the system additionally incorporates pre-processing and data augmentation approaches. TensorFlow and Python are used to implement the suggested system.

The data collecting, pre-processing, splitting, model selection, deep learning, hyperparameter tuning, model evaluation, and results visualization modules make up the suggested CNN-based plant leaf disease detection system.

A dataset of plant leaf photos of six distinct crops—apple, cherry, corn, grape, orange, and potato—is gathered in the data gathering module. There are 9966 distinct photos in the dataset, representing various classes. The photos are compiled from various sources and labeled with the appropriate diseases.

The photos are pre-processed in the data pre-processing module by scaling, standardizing, and enlarging the dataset to make it larger. The pictures are standardized to have a zero mean and unit variance, then shrunk to 224 × 224 pixels. To expand the dataset, data augmentation methods such random rotations, flips, and shifts are applied.
The dataset is divided into training, validation, and testing sets via the data splitting module. For the training, validation, and testing sets, the photos are divided into 60%, 20%, and 20% portions at random, respectively.

The model's hyperparameters, including the learning rate, batch size, and number of epochs, are adjusted on the validation set in the hyperparameter tuning module. On the validation set, the model's performance is assessed using measures like F1 score, accuracy, precision, and recall. Based on the validation set performance, the optimal set of hyperparameters is chosen.

A variety of charts and graphs are used in the results visualization module to display the model's output. To analyze the model's performance during training and validation, the accuracy and loss curves are shown. A heat map is used to illustrate the confusion matrix and provide a detailed analysis of the model's performance.

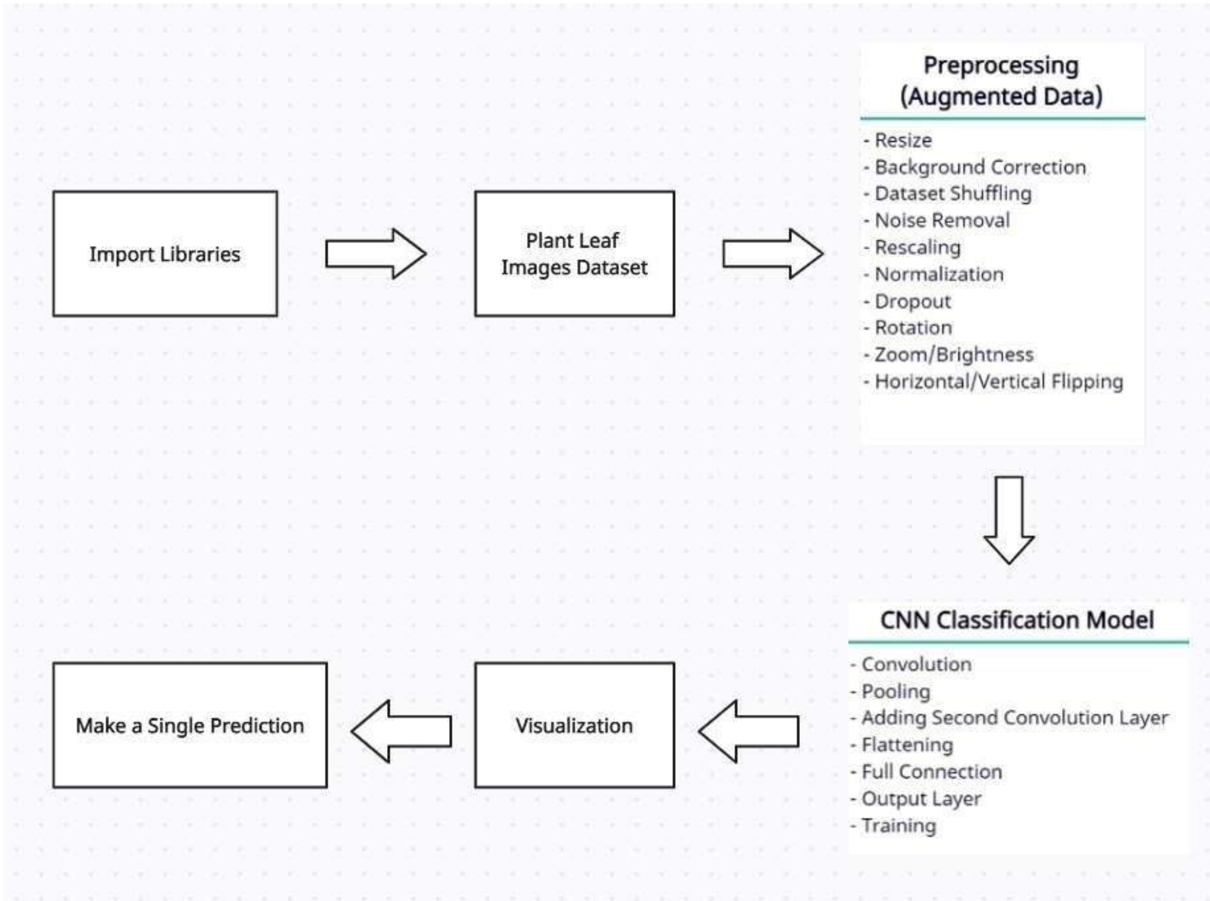

Figure 1: Proposed flowchart for Plant Leaf Disease Detection using Convolutional Neural Networks.

The methodology of the proposed system involves the following major steps: •
Augmentation – Dataset generation / contrast enhancement

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

- Convolution layer/Pooling layer – Features Extraction.
- CNN neural Network – Classification

3.1 Data Collection

Collect a dataset of plant leaf images of six different crops - apple, cherry, corn, grape, orange, and potato.

S. No.	Classes	Number of Images
1	Apple	2016
2	Cherry	1683
3	Grape	411
4	Corn	1907
5	Orange	2010
6	Potato	1939

Table 1 : Represents different classes and number of images in each class of the dataset

3.2 Data Preprocessing

Pre-process the images by resizing, normalizing, and augmenting the dataset to increase its size.

Data splitting: Split the dataset into training, validation, and testing sets.

Model selection: Select a pre-trained CNN model to use as a feature extractor.

Transfer learning: Fine-tune the pre-trained model on the training set using transfer learning.

Hyperparameter tuning: Evaluate the performance of the model on the validation set and finetune the hyperparameters.

Model evaluation: Test the performance of the final model on the testing set and compute the accuracy, precision, recall, and F1 score.

Results visualization: Visualize the results and compare them with the ground truth labels.

3.3 Convolutional Neural Networks

A convolutional layer is the first (Conv2D). Similar to a series of learning filters, that is. Each of the 32 filters that make up the first two layers of conv2D uses a kernel filter to transform a portion of the picture that is determined by the kernel size. The entire image is subjected to the kernel filter matrix. One could think of filters as image alteration. CNN is able to identify helpful elements anywhere from these altered photos (with maps included). The CNN pooling layer (MaxPool2D) is the second most crucial layer. All this layer does is lower the sample by acting as a filter. It takes a very high value and examines two adjacent pixels.This is used to reduce the cost of computational, and to some extent also reduce over-fitting. We have to choose the pooling size (i.e., the area of the pool is compacted each time) where the size of the pool is high, sample reduction is important. Combining convolutional layers and pooling, CNN can integrate local features and learn many of the world's image features. 'relax' modifier (activation function (0, x). The Rectifier function is used to add non-linearity to the network. Flatten layer is used to convert the final feature maps into a single 1D vector. This flattening step is required so the fully connected layers after certain layers of convolutional / maxpooling can be transferable to the classification neural network. Includes all local features available This is used to lower computational costs and, to a lesser extent, lower over-fitting. Sample reduction is critical when choosing the pooling size, which is the portion of the pool that is compacted each time and where the pool size is high. CNN can learn many of the properties found in images throughout the world and include local features by combining pooling and convolutional layers. Modifier "relax" (activation function (0, x). The network can become non-linear by utilizing the Rectifier function. The final feature maps are transformed into a single 1D vector using the flatten layer. In order for the fully connected layers that follow specific levels of convolutional / maxpooling to be transferable to the classification neural network, this flattening phase is necessary. incorporates all available local features for previous convolutional layers. Finally, one fully-connected (Dense) layer is just artificial neural networks (ANN) classifier.

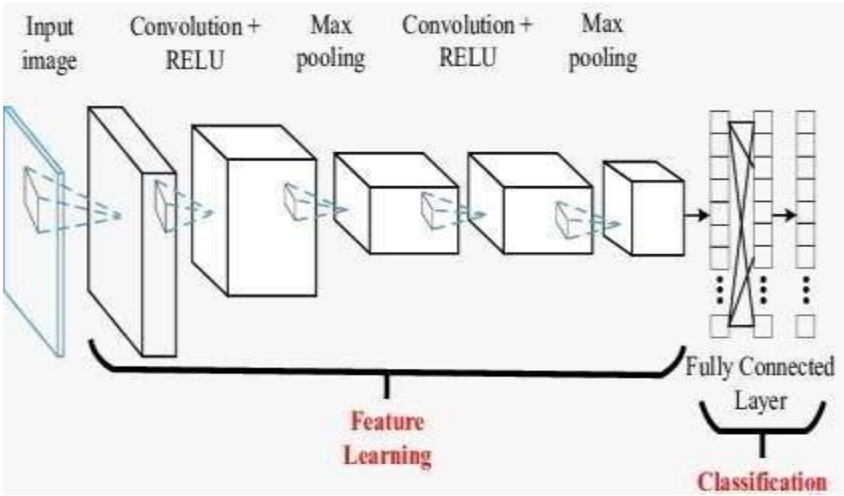

Figure 2 : Architecture of Convolutional Neural Networks

4. Implementation and Result Analysis

Model is the module that our system uses. The Python code needed to construct the neural network is contained in the model module. ImageDataGenerator was imported and preprocessed using Keras and Tensorflow.This module loads the previously constructed model.

The process of first supplementing the data involves setting up parameters such as image height, width, batch size, and number of classes. It also involves setting up data augmentation techniques such as rescaling, rotation, zooming, and flipping, as well as setting up data generators for training and validation. The platform loads the dataset from the given directory into these modules, where it is divided into train and validation data.

The CNN model architecture must then be defined. The model must then be assembled using the Adam optimizer and categorial crossentropy loss. Train and validation data must be fitted into the model, and each epoch's loss and accuracy must be produced.
identifying, at last, a single image that appeared in the path mentioned. The picture is first loaded from the system directory, after which it undergoes pre-processing and prediction

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

Following are the sample outputs from different classes :

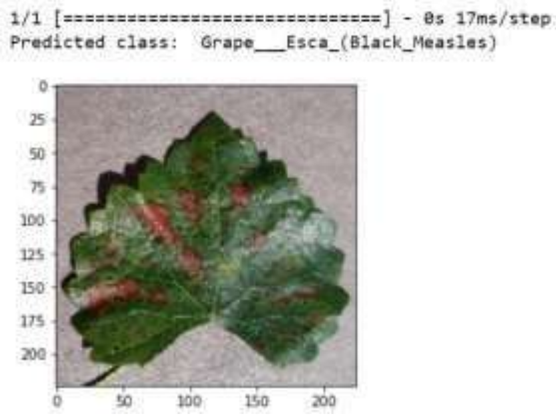

Figure 3: Grape class

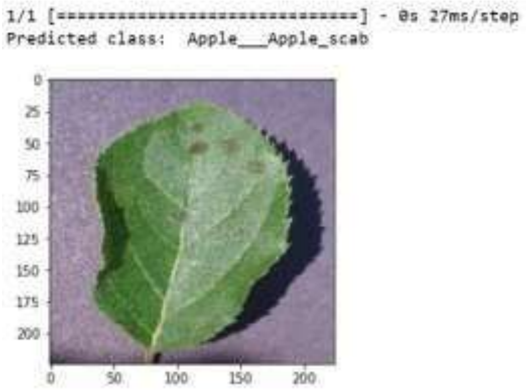

Figure 4: Apple class

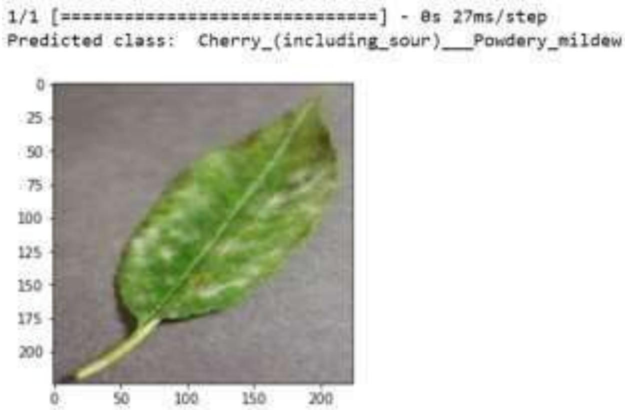

Figure 5: Cherry class

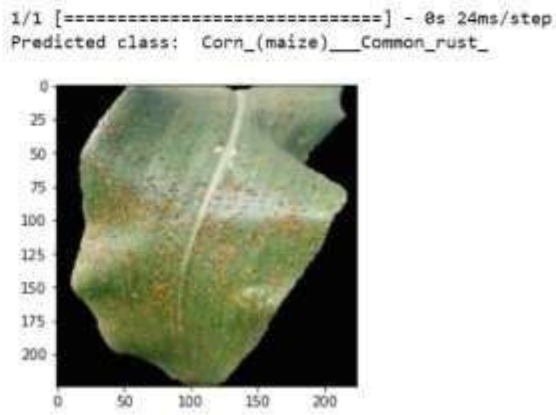

Figure 6: Cor n class

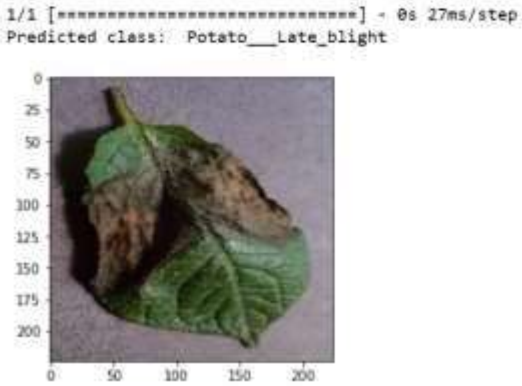

Figure 7: Potato class

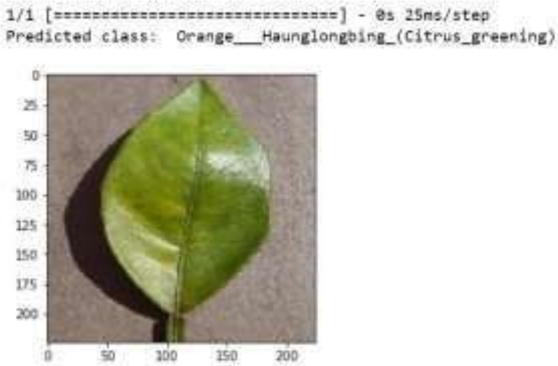

Figure 8: Orange class

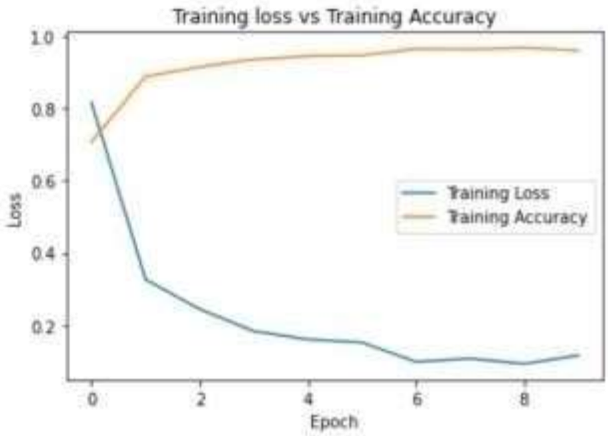

Figure 9: Graph depicting Training loss vs accuracy

The graph shows here depicts the two curves- Blue curve showing training loss and Orange curve showing training accuracy. The accuracy achieved is around 97.3%.

6. Conclusion

In this study, we suggested a CNN-based plant leaf disease detection system. The suggested solution leverages transfer learning to optimize a CNN model that serves as a feature extractor. The validation set is used to fine-tune the hyperparameters, while the testing set is used to assess the performance of the finished model.

A dataset of plant leaf photos from five distinct crops—apple, cherry, corn, grape, orange, and potato—is used to test the suggested system. In the literature review, the outcomes of the suggested system are contrasted with those of the current systems. The findings demonstrate how well the suggested method detects and categorizes plant leaf diseases.

Research on plant diseases, crop management, and precision agriculture are just a few of the uses for the suggested system. It is also possible to expand the system to identify illnesses in other plant components, like fruits, roots, and stems. In order to detect plant diseases, more study may be done to investigate other deep learning methods and optimize the hyperparameters.

References

[1] Ch. Usha Kumari, S. Jeevan Prasad, G. Mounika, “Leaf Disease Detection: Feature Extraction with K-means clustering and Classification with ANN “Proceedings of the 3rd International Conference on Computing Methodologies and Communication (ICCMC)”, Erode, India, 2019, pp 1095-1098

[2] A. Tiwari, P. Adhikari, A. Singh, D. Singh, and A. Adhikari (2020). A Deep Learning-Based Approach for Plant Disease Detection and Classification. 29(3), 441-457,

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024
Journal of Intelligent Systems.

[3] Kendall, A., Ahern, S., and Shinde, R. (2019). Mango Plant Leaf Disease Detection using Convolutional Neural Network. International Journal of Engineering and Information Technology Research in Computer Science, 4(1), 1497–1502.

[4] In 2019, Saah, S., Kaley, S., Nasrin, M., and Datta, S. A Novel Approach of Plant Disease Detection using Convolutional Neural Network. (pp. 198–201) at the 2019 Third International Conference on Computing Methodologies and Communication (ICCMC). IEEE.

[5] The paper "Deep Learning for Tomato Diseases: Classification and Symptoms Visualization" by M. Brahimi, K. Boukhalfa, and A. Moussaoui was published in 2017 in vol. 31, no. 4, pages 299–315.