Plant Disease Detection (December 2023)

Abduhu Khan, Hajra Mohsin, and Varda Sarfraz

National University of Sciences and Technology

ABSTRACT

This project proposes a comprehensive solution for plant disease detection. Machine learning algorithms combined with image processing techniques, allowed the system to utilize advanced feature extraction on raw image data. Machine learning models used included Random Forest and Convolutional Neural Networks trained on these features for disease classification. The project also contains user-friendly interfaces for practical integration into agriculture. The goal is to provide a scalable, efficient tool for disease detection, contributing to sustainable agriculture and global food security.

1. INTRODUCTION

**Problem Statement:**

Develop a system that utilizes artificial intelligence for the detection of plant diseases to assist farmers in implementing timely and effective disease management strategies.

**Objectives:**

• To identify plant diseases by performing image processing on the dataset.

• To apply various machine learning models and techniques for incorporating feature extraction to achieve prediction results of the diseases.

• To compare the performance of multiple machine learning models on the same dataset to achieve the required goal.

1. LITERATURE REVIEW

The article mentions that plant disease detection has a pivotal role in agricultural management as it relies heavily on agricultural organizations and local clinics. To address this need, the literature mentioned that the adopted method was a deep learning approach that focused on Convolutional Neural Networks (CNNs) with an emphasis on the AlexNet and GoogleNet architectures applied to leaf images. The dataset, mentioned in the article, consists of 54,306 images categorized into 38 classes, each standardized to 256 x 256 pixels. Experimental configurations were highlighted, emphasizing the importance of deep learning architectures, dataset characteristics, and the distribution of train-test sets. Performance evaluations of the various deep learning models showcased the overall accuracies ranging from 85.53% to 99.34%. A noteworthy aspect of this literature discussed the potential implementation of this approach on smartphones which would have the capability to provide a scalable and accessible solution for plant disease detection. Importantly, the methodology discussed in the article is complementary rather than a substitute for traditional laboratory tests, which suggested that by integrating real-world data from smartphones, improved accuracy can be achieved with additional contextual information.

DATASET OVERVIEW

The dataset used in this project is called the “New Plant Disease Dataset” which was acquired from Kaggle that contained approximately 87,000 images. This extensive dataset is comprised of 38 distinct classes, including healthy and diseased leaves across various plant species. This dataset also provides a rich resource for developing and testing machine-learning models for plant disease classification.

METHODOLOGY

**Image Loading and Preprocessing:**

* Dataset Loading: A dataset having images of both diseased and healthy plant leaves was acquired under different conditions varying in lightning and orientation of leaves.
* Image Resizing: The dimensions of all images were standardized, mainly to 64x64 pixels to ensure consistency in further image processing steps to be carried out.
* Depth Conversion: Image data was converted to 8-bit unsigned integer depth to create a uniform pixel representation.

**Color Space Transformation:**

* RGB to HSV Conversion: Preprocessed images were then transformed from the RGB version to the HSV version. HSV representation separated the information of colors into hue, saturation, and value components to facilitate better feature extraction.

**Image Segmentation:**

* HSV Thresholding: The leaves and spots were differentiated based on masks that were created after the application of HSV thresholds. This initial segmentation aided in isolating disease-affected areas.
* Morphological Operations: The process of opening was carried out to eliminate noise from the images using morphological operations and enhance the accuracy of disease region isolation.
* Contour Identification: Contours were detected in the segmented regions, serving as a basis for extracting shape-based features. Small contours were filtered out based on a specified minimum contour area.

**Feature Extraction from Segmented Regions:**

* Statistical Features: To have an overview of the overall intensities that span the image, the statistical features were also considered such as mean and standard deviation of pixel intensities.
* Texture Features (LBP): Local Binary Pattern (LBP) was applied to capture texture information, aiding in discerning patterns indicative of disease.
* Shape Features (Contour-based): To study the shape-based characteristics of the leaves, contour-based segmentation was done.
* Color Histogram Features: Color histograms were made to get an idea of the color distribution based on which differences in healthy and diseased leaves can be identified.

**Model Input Creation:**

* Feature Vector Construction: Features extracted from both the segmented leaf and spot regions were combined to create a comprehensive feature vector for each image. This vector was concatenated with the histogram-based features obtained in earlier steps.

**Dataset Splitting and Standardization:**

* Data Division: The dataset was already divided into the training and testing portions.
* Feature Standardization: All the compiled features are then standardized using a standard scaler so that they contribute equally to the model.

**Model Training:**

* Random Forest Classifier: The Random Forest Classifier with a specified number of trees (i.e., 2000) was utilized for training. This ensemble learning technique works wonders in handling complex datasets and doing appropriate hyperparameter tuning thus avoiding overfitting or it reduces it to a greater extent.

**Model Evaluation:**

* Performance Assessment: The model's performance on the standardized testing dataset was evaluated using the accuracy metric.
* Accuracy Calculation: Model accuracy was quantified as a main performance metric, providing an overall performance review of the model's effectiveness in the classification of healthy and diseased plant images.

AI MODEL (Random forests)

**Introduction:**

* Ensemble Learning: Random Forests were used to create multiple decision trees, which allowed for robust predictions by leveraging the collective strength of these trees.

**Why Random Forest?**

* Versatility: This model showcased its effectiveness by being able to handle multiclass classification.
* Overfitting: Random Forests (known for their robustness) were also able to combat overfitting, making it suitable for our selected dataset with varied characteristics.

**Application:**

* Feature Importance: The model was able to provide valuable insights into image features which assisted in the interpretation of the classification process.
* Diversity: By incorporating multiple trees, training each on different subsets of data, reduced the risk of overfitting which helped to enhance model generalization.

**Training:**

* Number of Trees: 2000 trees were employed in the ensemble to help improve the model's predictive performance.
* Standardization: Images extracted from the dataset underwent standardization to ensure consistent dimensions, boosting the convergence of the training process.

**Evaluation:**

* Accuracy Metric: After training and testing, the model's performance was evaluated using the accuracy metric.
* Comparative Analysis: Results were compared with the other model used which was Convolutional Neural Networks (CNN), to gauge relative performance.

**Advantages:**

* High Accuracy: The Random Forest model was able to achieve 80% accuracy, thus living up to its reputation of having the ability to generate accurate predictions.
* Interpretability: By trial and error, through this model, we were able to interpret which individual features were most important in the decision-making process.

**Future Considerations:**

* Parameter Tuning: Further improvements may include investigating enhancements through hyperparameter tuning and fine-tuning the model for optimal performance.
* Ensemble Methods: Alternative ensemble methods could be explored as avenues for improving the model's robustness and predictive capabilities.

1. AI MODEL (CNN)

**Introduction**

* The CNN model in our project is designed for image classification tasks using deep learning techniques. This section provides an overview of the architecture, components, and functionalities of CNN.

**Model Architecture**

* **ImageClassificationBase:** Defines the base class for our model containing training, validation, and logging methods for the training process.

**Convolution Blocks**

* **ConvBlock**: Function defined to create sequential convolutional layers followed by batch normalization and ReLU activation. It supports optional max-pooling.

**CNN Architecture**

* conv1: Initial convolutional block.
* conv2: Followed by a pooling layer.
* res1: First residual block consisting of two ConvBlocks.
* conv3 and conv4: Additional convolutional blocks with pooling.
* res2: Second residual block with two ConvBlocks.

**Classifier**

* Consists of a max-pooling layer, flattening, dropout, and a linear layer for classification.

**Model Functionalities**

* **Forward Propagation:** The forward function defines the forward pass through the network. The input data is processed through convolutional layers, residual connections, and the classifier to generate predictions.
* **Training Steps**: Uses cross-entropy as a loss function.
* **Validation Steps**: Calculates loss and accuracy during validation.
* **Epoch Logging**: Prints details including learning rate, training loss, validation loss, and accuracy epoch-wise.

**Model Initialization**:

* Weights are initialized using Kaiming initialization for Conv2d and Linear layers. Biases are initialized to zero.

**Weight Decay**

* Weight decay of **1e-5** is applied to all Conv2d and Linear layers to prevent overfitting and encourage regularization.

1. COMPARISON

Discussing the overall performance of the two models, taking accuracy as a performance metric, the random forest model gave an accuracy of 78.6% where whereas the Convolutional Neural network gave an accuracy of 98.07%.

The convolutional model performed well for the testing dataset and predicted all the images from the test dataset correctly, however, it could not perform well on images from other sources. It would either predict the leaf or the disease correctly. In 2 out of 13 cases, the image was perfectly classified.

In the random forest classifier, the results on test images gave the above-discussed accuracy and performed reasonably on the outsourced images compared to the accuracy of the model, but no better than the Convolutional Neural Network

1. CONCLUSION

In conclusion, the utilization of Random Forest Classifier has proven to be a successful strategy for plant disease detection. The model is adaptable and handles diverse datasets well, it is resilient to overfitting contributing to its effectiveness in this domain. This method of learning through an ensemble approach not only ensures accurate predictions but also provides valuable insights into different image features and their significance. As we look ahead, to further enhance our project, fine-tuning parameters and exploring alternative ensemble methods remain promising avenues in the field of plant pathology using digital image processing.

1. CODE LINKS

Random Forest and feature extraction code:

<https://colab.research.google.com/drive/1c6wYfmZH92MQtO2uxJ84SA91dbee1gP0?usp=sharing>

CNN code:

<https://colab.research.google.com/drive/1ZDmLJw7pfvSWZWvzd00_5XJopPBKJm8v?usp=sharing>