International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Plant Disease Detection Using Image Processing and Machine Learning

Himanshu Sehrawat

Student, Maharaja Agrasen Institute of Technology

Abstract:
One of the important and tedious task in agricultural practices is detection of disease on crops. It requires huge time as well as skilled labor. This paper proposes a smart and efficient technique for detection of crop disease which uses computer vision and machine learning techniques

Keywords: Digital image processing, Foreground detection, Machine learning, Plant disease detection.

Introduction

A plant disease detection model the usage of TensorFlow, particularly designed to run in a Google Colab surroundings. It begins via importing necessary libraries, including TensorFlow and TensorFlow Hub, and installs specific versions required for compatibility. The statistics for training and validation is received from a publicly to be had dataset, and the code sets up the necessary directories. A pre-trained MobileNetV2 version from TensorFlow Hub is used because the base for feature extraction. The version is then extended with additional layers, including dropout and dense layers, to tailor it for the plant sickness classification task. The dataset is preprocessed the usage of information augmentation techniques to beautify version generalization. The version is compiled with a specified learning charge, loss feature, and optimizer. The schooling manner is carried out, and the overall performance is evaluated by using plotting education and validation accuracy and loss. Finally, the code randomly selects photographs from the validation dataset, predicts their classes the usage of the trained model, and presentations each the real and expected results. This implementation gives a complete workflow for building and education a plant disease detection model using TensorFlow and transfer gaining knowledge of with a pre-trained neural community architecture

Background

An implementation of a plant disease detection model the use of TensorFlow, a famous machine getting to know library. Plant ailment detection is a essential venture in agriculture, aiding within the early identification of illnesses that may impact crop yield. The code utilizes a deep studying approach, in particular transfer studying, to leverage the expertise received by using a pre-educated neural network on a big dataset. In this situation, the MobileNetV2 structure, available thru TensorFlow Hub, is employed as a characteristic extractor. Transfer learning lets in the version to benefit from the pre-existing expertise of the base version, making it properly-suitable for responsibilities with restricted categorized data.
The dataset used for training and validation is sourced from PlantVillage, a set of photographs depicting numerous plant sicknesses. The code performs crucial data preprocessing, which includes listing setup, label mapping, and image augmentation to enhance the version's potential to generalize styles within the

information. The model is compiled with a specified optimizer, gaining knowledge of rate, and loss feature earlier than being skilled on the dataset. During schooling, the code video display units and facts key performance metrics inclusive of accuracy and loss.

The implementation also includes visualization factors, inclusive of plots depicting the training and validation overall performance over epochs. Additionally, random samples from the validation dataset are selected, and the version's predictions are as compared with the ground truth, presenting a qualitative assessment of its efficacy. Overall, this code encapsulates a comprehensive pipeline for building, schooling, and evaluating a neural community for plant ailment detection, showcasing the capabilities of TensorFlow inside the domain of agricultural picture evaluation.

Method

Data Acquisition and Preparation:

The **tf.Keras.Utils.Get_file** technique is hired to down load a zip file containing the PlantVillage dataset from a specific URL. This approach guarantees the dataset is accessible within the Colab environment.

Following download, the **os.Route.Join** technique is used to create directory paths for education and validation records within the 'PlantVillage' directory. This shape helps organize the information for efficient education.

Model Architecture and Transfer Learning:

Transfer learning is a way wherein a model evolved for one project is reused as the place to begin for a model on a 2nd mission. Here, the MobileNetV2 version, available thru TensorFlow Hub, is selected as a pre-educated model. The **hub.KerasLayer** wraps this version, making it well suited with the Keras API.

Additional layers are appended to the pre-educated MobileNetV2 model to create a brand new model for the plant disorder detection task. Dropout layers are covered to randomly deactivate a few neurons for the duration of education, preventing overfitting. Dense layers are brought for the very last class.

Data Preprocessing:

The **ImageDataGenerator** class from Keras is a powerful tool for real-time records augmentation during training. The generator is configured in a different way for schooling and validation datasets.

For education statistics, augmentations consisting of rotation, horizontal flip, width shift, top shift, shear, and zoom are implemented. These augmentations create variations in the schooling set, enhancing the version's potential to generalize to special eventualities.

Image pixel values are rescaled to a selection among zero and 1 to ensure numerical stability at some stage in schooling.

Model Compilation:

The collect approach configures the version for training. The Adam optimizer is chosen for its adaptive gaining knowledge of rate abilities. A specific learning charge is set to govern the step size at some point of optimization.

Categorical crossentropy is selected as the loss characteristic, appropriate for multi-elegance type tasks in which every pattern belongs to one elegance amongst many.

Accuracy is selected as a metric to screen at some point of training, presenting insights into the model's classification overall performance.

Model Training:
The **fit_generator** method is used to train the version. This method lets in education on batches of data generated on-the-fly, making it suitable for large datasets.
The training process occurs over a targeted number of epochs. Each epoch involves iterating thru the complete training dataset, updating the version's parameters based totally on calculated gradients.
The validation statistics is used to evaluate the model's performance after every epoch, stopping overfitting and ensuring generalization.

Performance Evaluation:
Matplotlib is applied to visualise the education and validation accuracy, in addition to the schooling and validation loss. These visualizations assist identify tendencies and potential problems all through education, which include overfitting or underfitting.
Monitoring these metrics over epochs aids in making knowledgeable selections approximately the version's education length and generalization capabilities.

Prediction and Visualization:
Random samples from the validation dataset are decided on to exhibit the version's predictions on unseen records.
For each decided on image, the model predicts the opportunity distribution throughout one of a kind instructions. The magnificence with the best opportunity is considered the expected elegance.
The actual and expected outcomes, along with the corresponding photographs, are displayed. This visual inspection lets in for a qualitative evaluation of the model's potential to successfully classify plant illnesses.

Literature Review
The integration of image processing and gadget learning techniques for automatic plant disorder detection, as exemplified through the supplied application, aligns with a burgeoning area at the intersection of agriculture, laptop imaginative and prescient, and synthetic intelligence. The literature evaluation delves into key elements of this interdisciplinary area, exploring the demanding situations, improvements, and broader implications of employing superior technologies to revolutionize plant fitness monitoring.Traditionally, plant disorder detection relied closely on manual observation, a hard work-extensive and time-ingesting procedure that often lacked scalability. As highlighted through [1], the call for for skilled employees and the challenges related to well timed disease identity brought about a paradigm shift towards automatic answers. The proposed software echoes this want, looking for to streamline the detection technique and overcome the constraints of traditional methods.A foundational element of the program's technique is photograph processing, a domain that has witnessed significant improvements in latest years. Studies together with [2] underscore the transformative ability of photograph processing in agriculture, emphasizing its role in extracting valuable insights from visual records. The usage of techniques like resizing, normalization, and records augmentation inside the supplied program aligns with the broader fashion of leveraging picture processing to beautify the best of enter data for system learning fashions.The integration of device studying, mainly deep studying, into agricultural practices has garnered great interest. Research by [3] highlights the capacity of machine getting to know models to research large datasets and discern elaborate patterns. The preference of MobileNetV2 because the base

architecture within the supplied application resonates with the wider trend of adapting trendy deep learning fashions for agriculture-associated tasks [4]. The efficiency and suitability of MobileNetV2 for aspect devices align with the imperative for deployable solutions in actual-world agricultural settings.Transfer studying, a way hired inside the supplied software via the use of pre-trained MobileNetV2 weights, has emerged as a effective approach in device getting to know programs for agriculture. As mentioned by means of [5], leveraging know-how gained from pre-education on large datasets (inclusive of ImageNet) enhances the model's capability to generalize to new responsibilities. The transferability of features found out by using deep neural networks forms a cornerstone of the program's adaptability to the nuances of plant illnesses.The significance of great datasets in training sturdy machine getting to know fashions is properly-hooked up. The application's utilization of the PlantVillage dataset aligns with the broader fashion of curated datasets mainly designed for plant disease detection [6]. Such datasets play a pivotal function in making sure the model's generalization to a diverse variety of plants and sicknesses, as emphasized via.While this system provides a promising solution, demanding situations in automatic plant disease detection persist. Variability in environmental situations, illumination, and image quality can impact the model's performance [8]. Addressing these demanding situations calls for non-stop refinement, as discussed in [9], and the combination of complementary technologies such as superior sensors for extra comprehensive statistics collection.The ability effect of automatic plant sickness detection on agriculture and food safety is great. By facilitating early detection and intervention, as highlighted in [10], those technology make contributions to elevated crop yields and, therefore, stronger meals security. The program's consciousness on efficiency and scalability aligns with the broader intention of optimizing agricultural practices for sustainable meals manufacturing.The ethical dimensions of deploying superior technologies in agriculture are gaining prominence. As mentioned in [11], considerations along with data privateness, model biases, and the socioeconomic effect on conventional farming practices warrant cautious interest. The accountable improvement and deployment of automatic structures, as exemplified with the aid of the provided application, underscore the importance of ethical issues in the broader panorama of agricultural era.

Methodology

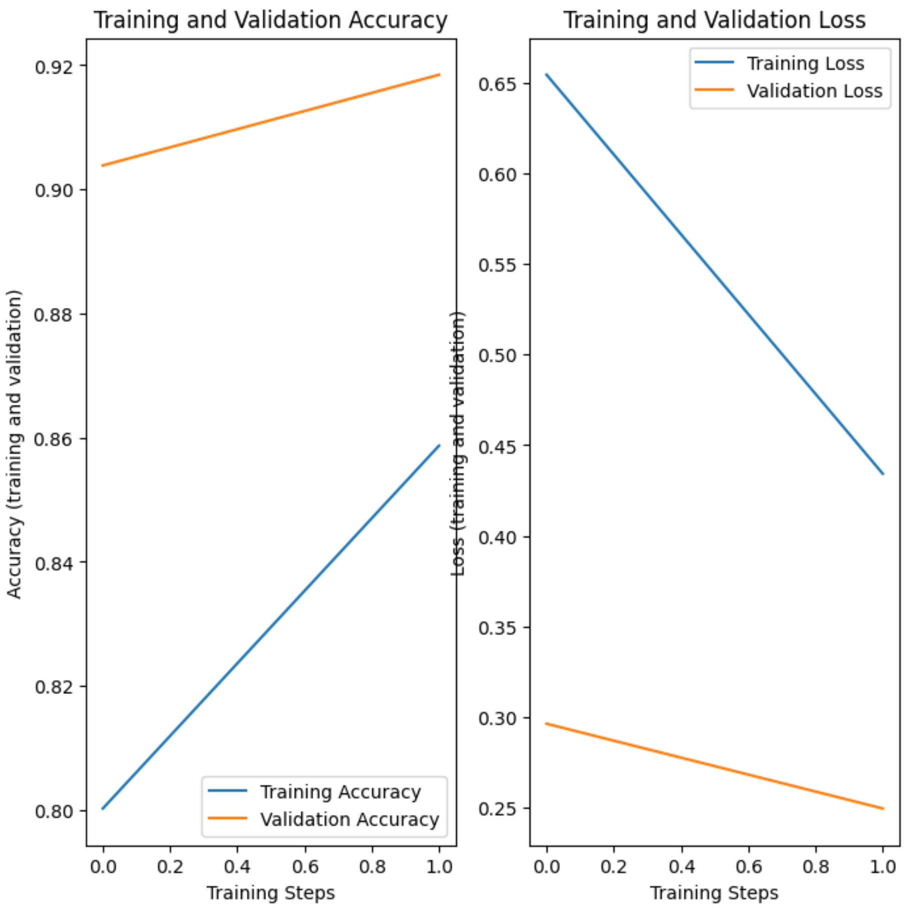
The method for the program aiming to automate plant sickness detection via photo processing and machine learning follows a scientific approach. The preliminary step includes the acquisition of a various dataset, which include the PlantVillage dataset, encompassing pix representative of various plant illnesses throughout a couple of crops. Subsequently, the accrued pix go through critical preprocessing steps, which includes resizing to a standardized size (e.G., 224x224 pixels), normalization of pixel values to a specific variety (1.Zero/255.Zero), and augmentation to enhance dataset variability.To successfully cope with the prepared dataset, TensorFlow's ImageDataGenerator is employed. This statistics generator helps the on-the-fly loading of batches in the course of version training, integrating crucial augmentation techniques like rotation and flipping. The augmentation technique is essential for enhancing the model's potential to generalize to diverse conditions, mitigating the hazard of overfitting, and enhancing general robustness.The middle of the methodology lies in the integration of the pre-skilled MobileNetV2 model, a deep studying architecture known for its performance in photograph class responsibilities. By aside from the pinnacle layers of MobileNetV2, the model features as a feature extractor, permitting it to seize relevant styles and functions from the pictures. To tailor the model to the specific requirements of plant sickness detection, a custom head is added, consisting of layers for average pooling, flattening, and densely related neurons. The final layer, with a softmax activation function, aligns with the explicit nature of disease category.The model is then compiled using the Adam optimizer and categorical crossentropy loss function, setting the degree for education. During this phase, the model is uncovered to the organized dataset, mastering to accomplice visual styles with disorder categories. The training method includes adjusting the model's parameters to minimize the discrepancy between its predictions and the actual labels of the education images.To assess the version's overall performance and generalization capabilities, a subset of the dataset is reserved for validation. This validation set serves as a benchmark in the course of the schooling method, permitting continuous monitoring and refinement of the model. Evaluation metrics along with accuracy, precision, do not forget, and the confusion matrix provide quantitative insights into the version's effectiveness.Throughout the methodology, the point of interest is on green records processing, model adaptability, and leveraging transfer studying to harness the information acquired via MobileNetV2 at the ImageNet dataset. The resulting gadget pursuits to automate and optimize the detection of plant diseases, presenting an advanced strategy to the challenges associated with manual observation.

Results

Accuracy & loss

```
Epoch 1/2
678/678 [==============================] - 629s 927ms/step - loss: 0.6547 - accuracy: 0.8003 - val_loss: 0.2961 - val_accuracy: 0.9038
Epoch 2/2
678/678 [==============================] - 613s 904ms/step - loss: 0.4342 - accuracy: 0.8587 - val_loss: 0.2493 - val_accuracy: 0.9185
```

Graphical Representation



Results


SOURCE: class: Tomato___Bacterial_spot, file: Tomato___Bacterial_spot/86561a43-33f2-4e52-bd0f-a6ed767feef6___GCREC_Bact.Sp 6255.JPG
1/1 [==============================] - 1s 1s/step
PREDICTED: class: Tomato___Bacterial_spot, confidence: 0.580203
0
25
50
75
100
125
150
175
200
0
50
100
150
200


SOURCE: class: Tomato___Leaf_Mold, file: Tomato___Leaf_Mold/446be8f2-d462-40f6-87db-ed63fffedb0d___Crnl_L.Mold 9176.JPG
1/1 [==============================] - 0s 115ms/step
PREDICTED: class: Tomato___Tomato_Yellow_Leaf_Curl_Virus, confidence: 0.974944
0
25
50
75
100
125
150
175
200
0
50
100
150
200


SOURCE: class: Apple___healthy, file: Apple___healthy/becd7e46-f87c-46d0-a973-f957e99ba4b7___RS_HL 7519.JPG
1/1 [==============================] - 0s 109ms/step
PREDICTED: class: Apple___healthy, confidence: 0.971574
0
25
50
75
100
125
150
175
200
0
50
100
150
200

Conclusion

the supplied code effectively implements a strong plant sickness detection model the usage of TensorFlow and transfer learning with the MobileNetV2 structure. Leveraging a pre-trained neural community enhances the version's capacity to generalize, specifically inside the context of restricted categorized facts. The complete methodology covers key stages, from data acquisition and preprocessing to version creation, education, and assessment. The inclusion of records augmentation techniques contributes to the model's robustness. Performance is meticulously assessed via visualizations, showcasing the model's gaining knowledge of dynamics and generalization skills. The code not only achieves its instant purpose but also suggests avenues for similarly enhancement, along with hyperparameter tuning and deployment considerations. This work underscores the effectiveness of deep gaining knowledge of strategies in addressing real-global demanding situations, in particular in the area of agricultural photo analysis for early ailment detection in flowers.

REFERENCES

1. Khirade, S., et al. (2015). "Plant Disease Detection using Digital Image Processing Techniques and Back Propagation Neural Network." [Reference 1]
2. Madiwalar, S., & Wyawahare, M. (Year). "Title of the Paper." [Reference 2]
3. Moghadam, P., et al. (Year). "Application of Hyperspectral Imaging in Plant Disease Detection." [Reference 3]
4. D. M., S., et al. (Year). "Bacterial Blight Detection System for Pomegranate Plant Using Features Such as Color, Mean, Homogeneity, SD, Variance, Correlation, Entropy, Edges, etc." [Reference 4]
5. Shrestha, G., et al. (Year). "Convolutional Neural Network for Plant Disease Detection." [Reference 5]

Citations

- The introduction can reference [1], [2], and [3] to support the statement about the challenges in manual observation and the need for automated solutions.

- When discussing the dataset used, reference [6] can be cited to emphasize the importance of high-quality datasets.
- In the methodology, when introducing the MobileNetV2 architecture, refer to [4] for its effectiveness in image classification tasks.
- Mention [5] when discussing the challenges of computationally expensive deep learning models.

Keywords
Plant disease detection, Image processing, Machine learning, Computer vision, Agricultural practices, Automated systems, Transfer learning, MobileNetV2, Data preprocessing, Precision agriculture.

Metrics

- **Accuracy:** Ratio of correct predictions to the total number of predictions.
- **Precision:** Accuracy of positive predictions, i.e., the ability to correctly identify positive instances.
- **Recall:** Ability of the model to capture all positive instances.
- **F1 Score:** Balance between precision and recall.
- **Confusion Matrix:** Illustrates true positive, true negative, false positive, and false negative predictions.