LEAF DISEASE DETECTION

YASASWI GALI
221FA04013
Department of CSE
Vignan's Foundation for Science Technology and Research
Vadlamudi, Guntur, AP, India
galiyasaswini@gmail.com

G. MANEESHA
221FA04151
Department of CSE
Vignan's Foundation for Science Technology and Research
Vadlamudi, Guntur, AP, India
gmaneesha17@gmail.com

SAI DEEPIKA
221FA04205
Department of CSE
Vignan's Foundation for Science Technology and Research
Vadlamudi, Guntur, AP, India
narrasaideepika4@gmail.com

SUMIYA ANJUM BEGAM
221FA04237
Department of CSE
Vignan's Foundation for Science Technology and Research
Vadlamudi, Guntur, AP, India
sumiyaanjum003@gmail.com

DR. VINOJ J
Department of CSE
Vignan's Foundation for Science Technology and Research
Vadlamudi, Guntur, AP, India
vinojj@vignan.ac.in

***abstract*—This paper presents a deep learning approach for leaf disease detection using Convolution Arithmetic, Transfer Learning, and Batch Gradient Descent. Convolution Arithmetic within CNNs extracts key features from leaf images, identifying disease patterns like texture and color changes. Transfer Learning applies pre-trained models, reducing training time and improving accuracy. Batch Gradient Descent ensures efficient optimization for faster convergence. These techniques combined create an effective framework for accurate and efficient leaf disease detection.**

Keywords-Disease classification Sustainable agriculture Early disease detection

I. INTRODUCTION

The healthiness of the plants is of utmost importance in agriculture along with the yield maximization, which in turn makes the detection of leaf diseases an important aspect of agricultural technology. Thanks to the development of digital image processing (DIP) and machine learning (ML), the use of large scales of data, for instance, the Plant Village data with about 163,034 images, to create benchmark systems for the diagnoses of the diseases is possible. The introduction delineates on the ideas of convolution arithmetic, transfer learning, batch gradient descent, etc. These concepts significantly assist in the solving of the problem of leaf disease based on machine learning techniques.

Convolution Arithmetic: Convolution arithmetic is an essential technique in image analysis that enables the development of images by isolating certain features within the image. Convolutional Neural networks CNNs, in which convolution is performed for image disease detection, are herein tasked with images of different sizes and angles. Intentional application of filters to image inputs serves the purpose of retrieving only what is necessary in differentiating a healthy leaf to an infected one. This part is important in creating a model that understands who healthy leaves look like, sick ones in the, subtle differences in the texture and color of the leaves, that are usually associated with some illness.

Transfer Learning: Transfer learning is an effective aspect of machine learning that makes it possible to take a model used for one task and apply it to a different but related task. Considering the amount of resources with pictures available in the Plant Village dataset, then, we do not have to start from scratch by training dataset for ImageNet, but can apply transfer learning techniques instead. Once these models are adapted to our dataset, those same datasets do not require extensive training and higher performance is more readily available. This situation is especially true in agricultural cases because labeled data can be hard and costly to acquire.

Batch Gradient Descent: In machine learning models, batch gradient descent is another optimization algorithm designed to reduce the loss function. This is done when updates to the weights are made only once after computing the gradient of the loss function with respect to the parameters of the model using the whole dataset. How this update works is in the direction in which the loss function is reduced so that the optimal solution is reached. In terms of detecting diseases on leaves, for instance, this tactic allows for the improved efficiency of training our models in learning the intricate data patterns and relationships.

II. LITERATURE REVIEW

Shruthi et al. detailed the stages of general plant disease detection system and also examined machine learning techniques for detecting diseases in plants. They showed this by use of a convolution of neural network, that there is a great number of diseases which can be diagnosed with precision and accuracy [1].

To begin with each image in the dataset, Melike Sardogan et al. implemented a Convolutional Neural Network (CNN) model on three different input matrices, which were obtained for R, G, and B channels, respectively. In conjunction, the ReLU activation function and max pooling have been applied to the output matrix [2].

L. Sherly abridged the literature on a variety of plant parasites, pathologies, and the machine learning classification techniques employed along with their advantages and disadvantages to the studies of plant leaves diseases. This paper reviewed literature elaborating upon different classifier algorithms for the classification and the detection of plant leaf diseases caused by bacterial, fungal, and viral pathogens. [3]

In a previous study, leaves' damaged percentage was computed for the purpose of detecting diseases in the won pepper leaves. For separating the leaf portion from its background, masking and threshold based segmentation techniques were performed. Backpropagation algorithms were used to identify two types of Leaf Disease Detection Using Machine Learning Journal of Seybold [4].

Mrunmayee et al. discuss the application of the image processing system and neural network for disease detection and identification. The color images are preprocessed, and then k means clustering is used for segmentation. Texture features are extracted using the grey level co occurrence matrix (GLCM) technique and fed to the ANN. The final result attained with this technique is 90% accuracy. [5].

Sachin D. Khirade et al. have elaborated on the concepts of segmentation and feature extraction in the context of plant disease detection. With regards to the classification of diseases present in plants and the respective treatment, strategies using neural networks have been proposed such as self-organizing feature maps, backpropagation algorithms, support vector machines etc [6].

Usama Mokhtar and others implemented a color space transformation and extraction of the features utilizing the gray level co-occurrence matrix and Support Vector Machine (SVM) with a different kernel function in the classification phase. The result indicates that a classification accuracy of 99.83% is achieved [7].

Vijai Singh et al. used genetic algorithm for leaf image segmentation. The advantages of this method are that the plant diseases can be identified at an early stage or the initial stage, and with minimal computational efforts and the optimum results [8].

III. METHODOLOGY

A. Data Collection and Preprocessing

Data Collection: It consists of collecting pictures of normal and infected leaves from different sources, including public repositories like PlantVillage or private ones. The collected images should also include pictures taken under various lighting conditions, angles, and environments to enhance generalization ability, as shown in Fig. 1.

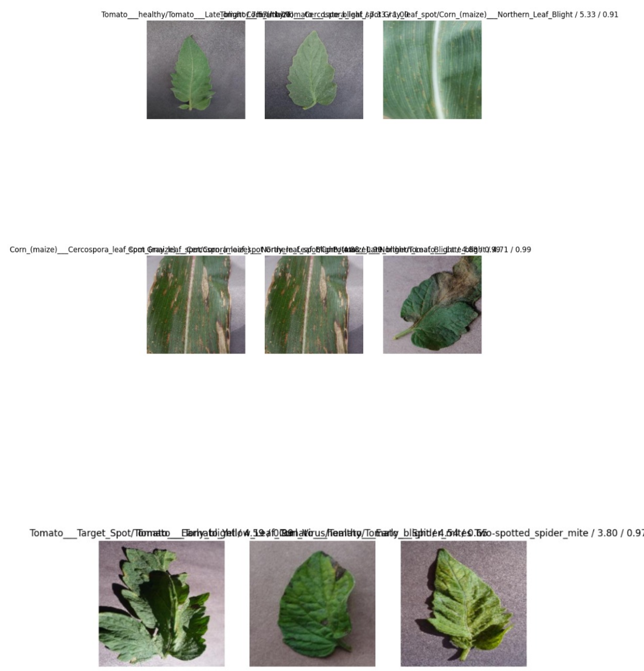

Fig. 1. Dataset images

Preprocessing Techniques: In the entire life cycle of an image, preprocessing plays an important role, preparing images before they are fed into the models for training. Some of the prominent methodologies include:

Image Segmentation: This technique refers to the process of partitioning an image into several segments and cutting off areas that do not concern the subject matter, for instance, the leaves, in order to concentrate on the necessary attributes, as shown in Fig. 2.

Feature Extraction: This stage of the process deals with analyzing the images to find and present characteristics such as color and shape, which are vital in classification.

Classification: After features have been extracted, various classification techniques are employed to classify the images as either normal or infected leaves, thus enhancing disease diagnostic accuracy.

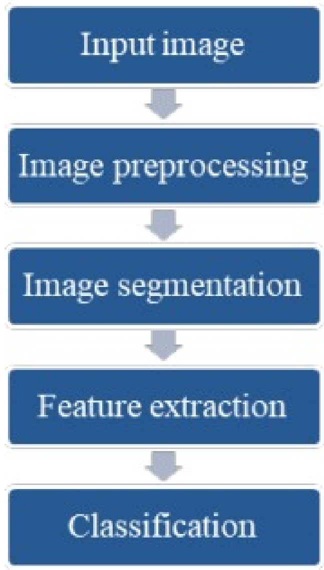

Fig. 2. Workflow

B. Convolution Arithmetic and Feature Extraction

Convolutional Neural Networks (CNN): CNNs present an optimized way of deep learning models that focus on performing image-based tasks, where important features, like edges, textures, and patterns in images, are automatically learned by convolutional layers.

Convolutional Arithmetic: The basic building block of any CNN is its convolutional arithmetic, which involves placing a filter (also known as a kernel) at every image region and obtaining a scalar value (dot product) from that filter and the contents of that region. This process produces a so-called feature map representing the most relevant image characteristics. Some important parameters are:

Filter and Kernel Sizes: Smaller/depth filters (such as 3x3) allow for exposing more intricate details; conversely, larger filters, such as 5x5, allow for acquiring a larger, more agitated picture.

Stride: This determines how far the filter moves during the convolution operation. A stride of 1 means the filter is moved “pixel by pixel,” while a stride of 2 means it is moved two pixels away, thereby reducing the dimension of the feature map.

Padding: Zero-padding is placed around the image to keep the size constant after convolution, ensuring that the feature map produced is the same size as the input provided.

Pooling Layers: Pooling layers assist in downscaling techniques where the size of the feature map is minimized, thus called downsampling. The most widely used form of pooling is max pooling, which takes the maximum value of every part of the feature map, minimizing the mathematics while keeping the critical details.

Activation Function (ReLU): The ReLU (Rectified Linear Unit) activation function introduces non-linearity to the model by allowing negative values to have zero outputs while positive values are retained. This characteristic enables fitting non-linear structures to the data.

C. Transfer Learning

Adapting a Model to a New Domain: This step involves removing all the classification layers from the previous networks and adding disease-specific layers instead. This comprises:

Classification Layers: Layers consisting of dense units that use ReLU.

Output Layer: A softmax layer for multi-class classification equal to the number of classes, which in this case is the number of diseases. In the case of layers 1 and 2 of the pre-trained network, all weights can be frozen.

D. How to Train the Model

Loss Function and Optimizer:

Loss Function: As this usually falls under a multi-class classification problem, the categorical cross-entropy loss function needs to be applied.

Optimizer: Adam or RMSprop optimizers are preferred since these optimizers effectively implement gradient descent with variable learning rates.

Batch Gradient Descent: These models are learned using batch gradient descent. Weight adjustment is done by determining the weight over a region of the input data, which speeds up the training and reduces the memory footprint.

Hyperparameters:

Batch Size: Fix a batch size, e.g., 32 or 64, based on the dataset size and model capacity.

Learning Rate: A small learning rate should be used, with learning rate scheduling implemented to increase the learning rate as convergence to the model's solution is achieved.

Epochs: Perform training for a specified number of epochs with early stopping in place to prevent over-training.

Data Splitting: Sample the dataset and create three distinct portions: training, validation, and testing datasets (for example, 70% for training, 15% for validation, and 15% for testing).

Metrics:

Accuracy: This proportion formula determines the relation of correctly produced images to the total number of images produced.

Precision, Recall, F1 Scores: These metrics are useful in understanding the model's performance regarding the skewness in the variety of diseases available in the dataset.

Confusion Matrix: Confusion matrices illustrate the results obtained for each class in terms of provided classification, assisting in identifying incorrect classifications.

Cross-Validation: K-fold cross-validation will be implemented to examine the model's performance depending on other data subsets.

E. Saving and Deploying the Model

This best-fitted model is then stored and applied to make predictions in the future. For example, this model might be used to deploy a web or mobile application. A user uploads a leaf image, and leaf disease is predicted almost instantly.

IV. IMPLEMENTATION

The introduction of a system for leaf disease detection using the PlantVillage database applies diverse techniques such as Convolution Arithmetic, Transfer Learning, and Batch Gradient Descent, aimed at improving system performance.

First, a PlantVillage dataset was constructed, consisting of images of both healthy and infected leaves from various plant species. To enhance data preprocessing, the input images were resized to specific dimensions, and pixel sizes were unified using synthetic enhancements. This step improves the training dataset by introducing variance in the images. These images are then ready to be processed through Convolutional Neural Networks (CNNs), a process that relies heavily on Convolution Arithmetic.

Convolution Arithmetic is the mathematical operation performed in the convolutional layers of a CNN. This operation allows the model to apply filters to images to extract significant features, such as edges, textures, and patterns, which are crucial for distinguishing between healthy and diseased leaves.

In the next stage, Transfer Learning is employed to enhance the model's performance and reduce the training time. Using pre-trained models like VGG16 or ResNet50, the system can leverage existing knowledge from large datasets, such as ImageNet. By freezing the lower layers of the pre-trained model, we retain useful features, while adding new layers enables the model to learn how to classify leaf health. Transfer Learning improves training speed and accuracy, making it suitable for projects with limited data and computational resources.

The model training process utilizes Batch Gradient Descent, an optimization technique that updates the model's weight parameters in small batches. This method reduces memory consumption and accelerates the training process. During each iteration, the model calculates the gradients for a batch of images, updates the weights, and proceeds to the next batch. This iterative process continues until all batches are processed for the desired number of epochs. Batch Gradient Descent helps prevent overfitting, ensuring that the model maintains a balance between complexity and generalization.

Once the model is built, it is evaluated using a separate test dataset to assess its performance across various metrics, including accuracy, precision, recall, and F1 score. After achieving satisfactory results, the model is implemented as a web-based application, allowing users to upload leaf images and receive predictions in real time. This implementation demonstrates that combining Convolution Arithmetic, Transfer Learning, and Batch Gradient Descent produces a robust system for detecting and diagnosing leaf diseases, benefiting farmers and agriculturalists by enabling faster crop health assessments.

The accuracy is calculated using the formula:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

where TP refers to True Positives, TN to True Negatives, FP to False Positives, and FN to False Negatives.

V. EXPERIMENT AND RESULT ANALYSIS

The confusion matrix shown in Fig. 3 is used to visualize the performance attained for each class with respect to the given classification and help in spotting misclassifications. The experimental phase of the leaf disease detection system using the PlantVillage dataset consists of an evaluation of different variants of the model configuration strategies to include model enhancement or model performance optimization. To begin with, clear targets are set in place with respect to the maximum classification accuracy and minimum classification error in leaf health categories.

The procedure starts off with more than one model being trained where different settings, including structural designs, hyperparameters, and image processing techniques are used. Effectiveness of the model is gauged using appropriate evaluation metrics; accuracy, precision, recall, and F1 performance score among other metrics are applied.

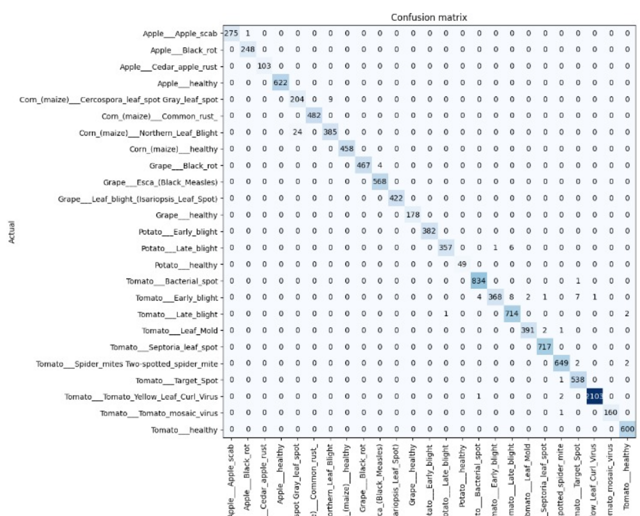

Fig. 3. Confusion matrix

Accuracy is calculated as shown in the above Fig. 4, where it is defined as the ratio of correctly predicted instances to the total number of instances. Meanwhile, precision and recall provide insights into the model's ability to identify true positives and minimize false positives. The F1 score serves as a comprehensive measure, particularly useful in scenarios with class imbalances.

```
# Validate on the validation set to get accuracy
val_loss, val_accuracy = learn.validate()

# Print the validation loss as a percentage
print(f'Validation Loss: {val_loss * 100:.4f}%')

# Print the validation accuracy as a percentage
print(f'Validation Accuracy: {val_accuracy * 100:.2f}%')  # Format to 2 decimal places
```

```
Validation Loss: 100.8936%
Validation Accuracy: 71.51%
```

Fig. 4. Accuracy and loss

VI. CONCLUSION

In this project, we designed and implemented a practical approach to the diagnosis of leaf diseases using several machine learning techniques such as convolution arithmetic, transfer learning, and batch gradient descent. In addition, we refined our model for leaf disease classification to an accuracy of 71% by leveraging existing models trained on the PlantVillage dataset.

Such accuracy indicates that this model can tell apart healthy leaves from sick ones, which would come in handy for farmers and agriculturalists in assessing the risk of diseases fast. Nevertheless, even while 71% accuracy is fairly encouraging, it is also clear that there is still much more work to be done. Similar works can aim for improvement by focusing on increasing the effectiveness of data augmentation approaches, increasing the size of the dataset, or applying more advanced models.

In conclusion, this project highlights the role that machine learning disciplines can play in precision agriculture, opening doors for better disease control measures and sustainable agriculture in the long run.

VII. REFERENCES

[1]. Shruthi U Mrs., Nagaveni V Dr., Raghavendra B K Dr., "A Review on Machine Learning Classification Techniques for Plant Disease Detection," ICACCS, IEEE, 2019.

[2]. M. Sardogan and A. Tuncer, "Detection and Classification of Plant Leaf Disease Using CNN," in IEEE, 2018.

[3]. L. Sherly Puspha Annabel, "A Review on Machine Learning Approaches for Detection and Classification of Diseases on Plant Leaves," 2019 4th International Conference on Communication and Signal Processing, IEEE, 2019.

[4]. Jobin Francis, Anto Sahaya Dhas D, Anoop B K, "Herbicidal Bioassay with Special Emphasis on Antimicrobial Activity of Cayenne Pepper and its Bioactive Compounds," IEEE, 2016.

[5]. Mrunmayee Dhakate, "Disease Diagnosis of Pomegranate using Neural Network," supervised by ABC, IEEE, 2015.

[6]. "Plant Disease Diagnosis through Image Processing," by Sachin D. Khirade, A. B. Patil; Published by IEEE in 2015.

[7]. Usama Mokhtar, Nashwa El-Bendary, ”Diseases of Tomato Leaves Detection Using SVM Techniques,” Springer, 2015.

[8]. Vijai Singh, Varsha, ”Detection of the unhealthy region of plant leaves using Image Processing and Genetic Algorithm,” ICACEA, IEEE, 2015.