Plant disease classification using CNN models

Jeevan EG
School Of Computer Science and Engineering,B.Tech(H)
Rastreeya Vidyalaya University
Bengaluru,India
jeevaneg.btech22@rvu.edu.in

Manoj Y N
School Of Computer Science and Engineering,B.Tech(H)
Rastreeya Vidyalaya University
Bengaluru,India
manojyn.btech22@rvu.edu.in

Nandeesha Kantli
School Of Computer Science and Engineering,B.Tech(H)
Rastreeya Vidyalaya University
Bengaluru,India
nandeeshak.btech22@rvu.edu.in

Pranav P Kulkarni
School Of Computer Science and Engineering,B.Tech(H)
Rastreeya Vidyalaya University
Bengaluru,India
pranavpk.btech22@rvu.edu.in

***Abstract*—Detecting plant diseases accurately is crucial for ensuring food security, as crop production can suffer greatly from various diseases. Traditionally, farmers or experts visually inspect leaves to identify diseases, but this process is subjective and time-consuming. To address this, a project utilizes deep learning, specifically convolutional neural networks (CNNs), known for their effectiveness in image-based tasks. By training a CNN model on a dataset of plant images, diseases can be classified swiftly and accurately. Recently, deep learning, particularly convolutional neural networks (CNNs), has become widely used for plant disease classification, overcoming many drawbacks of traditional methods. This review explores the latest CNN models used for classifying plant leaf diseases, outlining the underlying principles of deep learning in this context. It also discusses common challenges in CNN-based classification and their corresponding solutions, as well as potential future directions for advancing plant disease classification techniques.**

***Index Terms*—Plant disease detection, Deep learning (DL), Convolutional neural networks (CNN), State-of-the-art technology**

I. Introduction

A. *Problem Statement*

With the increasing demand for sustainable agricultural practices, early detection and management of plant diseases are crucial for ensuring food security and maximizing crop yield. Manual inspection of crops for disease symptoms is time-consuming and often unreliable. Leveraging the power of machine learning, specifically Convolutional Neural Networks (CNNs), can automate this process and enable accurate and efficient classification of diseased and non-diseased plant leaves.

B. *Plan*

The project will begin with the acquisition of a diverse dataset containing images of both diseased and non-diseased plant leaves, ensuring a balanced distribution of classes. These images will undergo preprocessing steps such as resizing to a uniform resolution and augmentation techniques to increase dataset variability. Subsequently, CNN architectures including LeNet, ResNet, and VGG will be selected for leaf disease classification, and their respective configurations will be designed. Using TensorFlow, the chosen architectures will be implemented, and models will be trained on the preprocessed dataset with appropriate parameters and optimization algorithms. Following training, the models will be evaluated using a separate validation dataset to assess metrics like accuracy, precision, recall, and F1-score, facilitating a comparative analysis of their effectiveness. Fine-tuning and optimization strategies, including hyperparameter tuning and regularization techniques, will be explored to further enhance classification performance. Throughout the project, detailed documentation of implementation, training, and evaluation procedures will be maintained, culminating in a comprehensive report elucidating the findings, insights, and recommendations for future research and practical applications in agriculture. Ultimately, the project aims to present its findings to stakeholders, including agricultural experts, researchers, and farmers, emphasizing the significance of automated leaf disease classification in bolstering agricultural productivity and global food security.

II. Deep Learning

Deep Learning (DL) is a neural network-based algorithm that automatically selects features from data without extensive feature engineering, enhancing accuracy and generalization in tasks like image recognition and target detection. The main DL networks include multilayer perceptrons, Convolutional Neural Networks (CNNs), and recurrent neural networks (RNNs), with CNNs predominantly used for plant leaf disease classification. CNNs typically consist of convolutional, pooling, and fully connected layers. The convolutional layer extracts features by local correlation within the image, where a kernel is moved across the image, multiplying pixel values and

Identify applicable funding agency here. If none, delete this.

summing the results. Pooling layers, such as maximum or average pooling, select features and provide translation, rotation, and scaling invariance. Convolutional and pooling layers often alternate in applications. Fully connected layers integrate multidimensional features into one-dimensional features for classification or detection tasks.

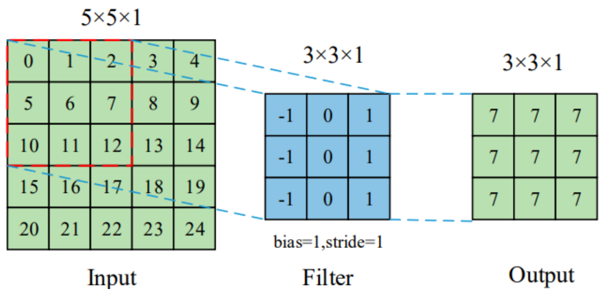

Fig. 1. The process of convolution operation.

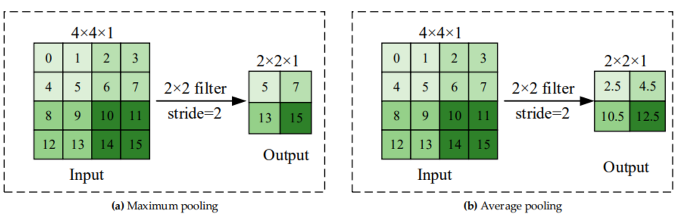

Fig. 2. The process of pooling operation.

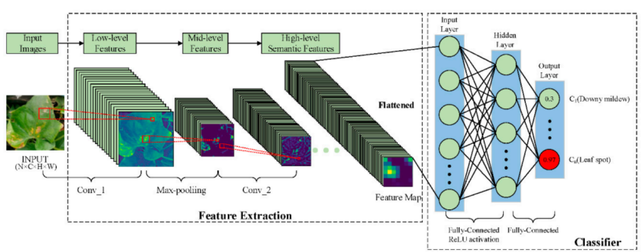

Fig. 3. Convolutional neural networks for potato leaf disease classification.

A. *Explanation of the process*

In the depicted scenario illustrated in Figure 4, a batch of images is fed into the feature extraction network to derive features, which are subsequently flattened and inputted into the classifier for disease classification. This procedure can be delineated into three main phases. Initially, in Step 1, data preparation and preprocessing are conducted. Following this, in Step 2, the model is constructed, trained, and assessed for its performance. Lastly, in Step 3, inference and deployment of the model take place, completing the process flow outlined in the figure.[1]

III. Related Work

Research into the classification of plant diseases has advanced using a spectrum of methodologies, from traditional image processing to contemporary deep learning techniques. Initially, approaches centered on tasks like segmenting images, extracting features, and employing algorithms such as Support Vector Machines (SVM) or k-Nearest Neighbors (k-NN), which relied on manually engineered features like texture and color descriptors. However, with the rise of deep learning, particularly Convolutional Neural Networks (CNNs), there's been a paradigm shift, enabling comprehensive learning directly from raw image data. Transfer learning, wherein pretrained models are adapted to plant disease datasets, has emerged as particularly promising, particularly in scenarios where data is scarce.

A. *Previous inferences*

1) Many studies, as outlined earlier, have utilized the **PlantVillage dataset** to assess the effectiveness and efficiency of various deep learning models and architectures. Despite containing numerous images of various plant species afflicted with diseases, this dataset typically features a straightforward and unadorned background. However, in real-world applications, it's essential to account for the complexities of the natural environment.

2) The utilization of **hyperspectral/multispectral** imaging represents a burgeoning technology that has found application across numerous research domains. Consequently, integrating this technology with optimized deep learning architectures holds promise for detecting plant diseases at an early stage, potentially even before visible symptoms manifest.

3) While many deep learning frameworks proposed in the literature demonstrate effective detection performance on their respective datasets, they often exhibit poor robustness when applied to other datasets. This lack of robustness highlights the need for improved deep learning models capable of adapting to diverse disease datasets while maintaining consistent performance across different data sources.[2]

B. *Modern overview*

Plant diseases present a significant challenge to global agriculture and food security. It's crucial to detect and diagnose these diseases early to mitigate crop damage. Recently, machine learning techniques, especially Convolutional Neural Networks (CNNs), have become instrumental in automating the detection and classification of plant diseases. By analyzing RGB images of diseased plant leaves, CNN models can accurately differentiate between healthy and infected plants. This paper provides an overview of a CNN-based model for classifying plant diseases using RGB images. It covers various stages of model development, including data collection, preprocessing, model architecture design, training, validation, hyperparameter tuning, evaluation, and inference. The aim is to showcase CNNs' effectiveness in tackling plant disease detection challenges and to offer a roadmap for future research in this area.[3]

IV. Methodologies

The proposed method emphasizes several crucial stages in the development of a plant disease detection model. It involves introducing a new dataset and augmentation methods, examining various classification and object detection algorithms, and putting forth a novel approach for plant disease detection.

a)Deep learning models, particularly Convolutional Neural Networks (CNNs), have been extensively utilized for detecting and classifying plant diseases across various crops like maize, tomato, banana, cucumber, and cassava. Different CNN architectures such as AlexNet, GoogLeNet, ResNet, and VGG have been employed with varying success rates. Evaluation metrics like accuracy, sensitivity, specificity, and F1-score have been commonly used, with ResNet-50 often performing well when combined with Support Vector Machines (SVM). However, previous studies have overlooked the potential of visualization techniques to interpret disease symptoms in plants, leaving room for future research in enhancing the interpretability of deep learning models in plant pathology.

- Research has introduced visualization methods to effectively identify and visualize plant diseases. For example, one study introduced a saliency map to visualize disease symptoms and achieved high accuracy using the CaffeNet CNN architecture. Filters were used to indicate disease spots, and other studies employed architectures like AlexNet and GoogLeNet on the PlantVillage dataset. Performance metrics such as precision, recall, F1 score, and overall accuracy were assessed, with GoogLeNet generally outperforming AlexNet. Visualization techniques like activation in the initial layers clearly depicted disease spots.
- Furthermore, modifications to existing models like LeNet for olive plant disease detection have been explored, incorporating segmentation and edge maps to identify diseases. DL models with detectors such as Faster-RCNN, RFCN, and SSD have been used to mark diseases in plants, with ResNet-50 showing promising results. Heat maps, ROC curves, and feature maps have been utilized to evaluate model performance, with techniques like occlusion aiding in disease region recognition.
- Various crops have been targeted, with specific models and techniques tailored to each. For instance, in banana leaf disease detection, ResNet-50, Inception-V2, and MobileNet-V1 were employed with detectors like Faster-RCNN and SSD. Similar approaches have been applied to detect diseases in soybean, tomato, wheat, radish, and cucumber plants, with segmentation, feature extraction, and visualization aiding in disease identification.
- Overall, deep learning models combined with visualization techniques offer promising avenues for accurately identifying and visualizing plant diseases, thereby aiding in effective disease management in agriculture.[4]

b)Convolutional Neural Networks (CNNs) are specialized neural networks specifically designed for tasks in computer vision, such as pattern recognition and classification. What distinguishes CNNs is their ability to autonomously learn and extract features from images during the training process, eliminating the manual feature extraction required in traditional methods. CNNs consist of different types of layers, with the convolutional layer being central. This layer extracts features from input images by applying small arrays of numbers, called kernels or filters, across the input. Each kernel generates a feature map, capturing different aspects of the input image. Multiple convolutional layers may be utilized based on the input image's size and complexity.

- After the convolutional layer, the pooling layer is employed to decrease the dimensionality of the feature maps. Pooling conducts downsampling operations, consolidating information in the feature maps while reducing computational complexity. Various pooling methods, like max-pooling, min-pooling, and average-pooling, help distill the most pertinent features.
- The output feature maps from the convolutional or pooling layers are then converted into one-dimensional vectors, with each input connected to every output through weighted connections. This layer is known as the dense layer or fully connected layer. CNNs can contain one or more fully connected layers, with the final layer having the same number of outputs as the classification task's classes, enabling the mapping of extracted features to class labels. Leveraging the hierarchical structure of their layers, CNNs automatically learn and extract meaningful features from images, making them highly effective for a variety of computer vision tasks.[5]

c)To ensure the reliability and effectiveness of our methods and evaluate their performance on new, unseen data, we conduct experiments using various train-test set splits, including 80-20, 60-40, 50-50, 40-60, and 20-80. These splits represent the proportions of the dataset allocated for training and testing, allowing us to assess how well our models generalize across different dataset proportions and detect potential overfitting issues. With the PlantVillage dataset, which may contain multiple images of the same leaf from different angles, we maintain data integrity and prevent data leakage by ensuring that images of the same leaf are either in the training or testing set, but not both. This guarantees that our models are evaluated on genuinely unseen data during testing. Throughout the experiments, we compute performance metrics such as mean precision, mean recall, mean F1 score, and overall accuracy, typically at the end of each training epoch. Precision measures the ratio of true positive predictions to all positive predictions, while recall measures the ratio of true positive predictions to all actual positives. The F1 score, a balanced evaluation of precision and recall, provides an overall assessment of model performance, and overall accuracy indicates the proportion of correctly classified instances among all instances. By averaging the F1 score across all experimental setups, we obtain a comprehensive measure of our models' performance, enabling us to compare their effectiveness across different train-test splits.[5]

d)In our study, we introduced additional modifications to our experimental approach. Firstly, we reduced the image resolution to 128×128 pixels to examine how this change affects our models' performance. This adjustment was aimed at potentially reducing computational complexity while still maintaining effective feature extraction capabilities.

Moreover, we chose to solely use RGB images and disregarded other versions of the dataset, such as segmented and color variations. This decision was made to simplify our experimental approach and ensure consistency in our analysis. By focusing on RGB images, which are the standard color model in digital imaging, we aimed to avoid potential biases introduced by variations in image representation.

Overall, these adjustments to our experimental setup allowed us to conduct a more focused and streamlined analysis. By investigating the impact of downscaled images and exclusively utilizing RGB representations, we aimed to gain deeper insights into our models' performance and their ability to accurately classify plant leaf images.[6]

V. Experiments

A. Data Collection

The dataset utilized in this study is derived from an extensive collection of plant images sourced from https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset?resource=download .Specifically, it comprises approximately 87,000 RGB images depicting both healthy and diseased crop leaves, categorized into 38 distinct classes. Offline augmentation techniques were employed to recreate this dataset from its original form, which is accessible on a corresponding GitHub repository. The dataset is organized into training and validation sets, with a split ratio of 80% for training and 20% for validation, maintaining the original directory structure. Furthermore, a separate directory containing 33 test images was subsequently created for prediction purposes. This comprehensive dataset serves as a valuable resource for research endeavors in the field of agricultural disease detection and classification.[7]

B. Architecture

*1) **LeNet-5**:* The LeNet-5 architecture, a convolutional neural network (CNN) based on gradient-based learning, was initially successful in hand-written digital character recognition. Illustrated in, the typical structure of LeNet-5 comprises an input layer representing hand-written digital images of digits 0 to 9, each with dimensions of 32×32, and an output layer with 10 nodes corresponding to the digits 0 to 9. Beyond the input and output layers, LeNet-5 generally consists of six layers, including three convolutional layers, two pooling layers, and one fully connected layer. The convolutional layers employ 5×5 convolutional kernels, while the pooling layers utilize 2×2 kernels. The fully connected layer further reduces the neuron count from 120 to 84, aiding in parameter training efficiency.[8]

*2) **AlexNet**:* AlexNet comprises 5 convolutional layers, 3 max-pooling layers, 2 Normalized layers, 2 fully connected layers, and 1 SoftMax layer. Each convolutional layer is composed of a convolution filter and employs the Rectified Linear Unit (ReLU) activation function. The max-pooling function is applied in the pooling layers, and the input size is fixed due to the presence of fully connected layers, typically noted as 224x224x3, but due to padding, it effectively becomes 227x227x3. Notably, AlexNet boasts over 60 million parameters.[9]

*3) **GoogLeNet/Inception**:* GoogLeNet, a variant of the Inception Network, is a deep convolutional neural network consisting of 22 layers. Developed by Google researchers, this architecture was presented in the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14). It successfully addressed computer vision tasks like image classification and object detection. For details on its performance, refer to the conclusion section of this article.[10]

*4) **VGG-16**:* In the ImageNet dataset, which comprises over 14 million training images across 1000 object classes, the VGG16 model demonstrates a test accuracy of 92.7 percentage. Noteworthy for its performance, VGG16 is among the top models from the ILSVRC-2014 competition. It enhances upon AlexNet by employing sequences of smaller 3×3 filters instead of large filters. In contrast to AlexNet, where the kernel sizes are 11 for the first convolutional layer and 5 for the second layer, VGG16 adopts this strategy. The researchers dedicated several weeks to training the VGG model, leveraging NVIDIA Titan Black GPUs.[11]

*5) **ResNet-50**:* ResNet50 stands out as a formidable image classification model capable of achieving cutting-edge results when trained on extensive datasets. Its notable innovation lies in the incorporation of residual connections, which enable the network to learn residual functions mapping input to output. These connections facilitate the training of deeper architectures without encountering the issue of vanishing gradients.

The architecture of ResNet50 comprises four main components: convolutional layers, identity blocks, convolutional blocks, and fully connected layers. Convolutional layers extract features from input images, while identity and convolutional blocks process and transform these features. Finally, fully connected layers are employed for the ultimate classification task.[12]

C. Sequential Model

Our machine learning project employs a Convolutional Neural Network (CNN) architecture implemented using TensorFlow for the classification of plant images. The model is constructed as a sequential stack of layers, starting with convolutional layers initialized for images of size 256x256 pixels. ReLU activation functions are applied after convolution operations, followed by pooling layers with a stride of 2 for downsampling. Dropout regularization with a rate of 0.25 is utilized to prevent overfitting. The Adam optimizer is employed along with the cross-entropy loss function for model optimization during training. The dataset consists of labeled images representing various plant species. Here, we further introduced 4 more neural layers for efficiency in the output. Through training, the model learns to extract meaningful features from the images and classify them into the corresponding plant species. After training and evaluation on a validation set, the model is deployed for practical applications such as automated plant species identification in agricultural or ecological contexts. Overall, this project showcases the effectiveness of CNNs in handling image classification tasks, particularly in the domain of plant species classification.

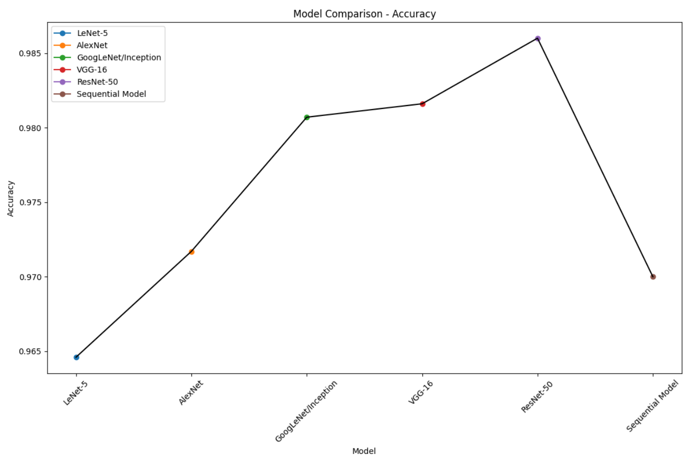

Fig. 4. Model Accuracy

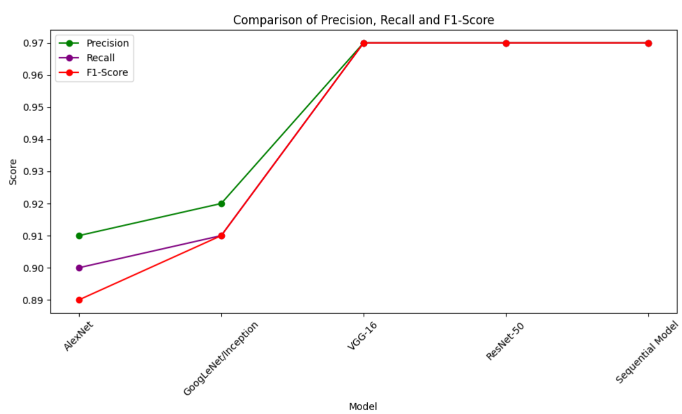

Fig. 5. Comparision of Precision, Recall and F1-Score.

VI. Conclusion

This study harnesses the capabilities of deep learning to develop an automatic plant disease detection system. The system utilizes a straightforward classification mechanism, leveraging the feature extraction capabilities of Convolutional Neural Networks (CNNs). For prediction, the model utilizes fully connected layers. The research draws from a publicly accessible collection of 87,000 images, including images from both experimental conditions and actual environments. The system demonstrates strong testing accuracy on the publicly accessible dataset and performs well on additional images. This suggests that CNNs are highly suitable for automatic detection and diagnosis of plant diseases. Furthermore, the system has potential applications in integrated into mini-drones for live disease detection in cultivated areas. Despite being trained on the Plant Village dataset with only 38 classes, the system is capable of detecting whether a plant is diseased or not, as symptoms often exhibit similarities across plant species. To further enhance accuracy and expand the system's capabilities, additional images from actual environments can be incorporated into the dataset to improve performance on real-condition images of leaves and classify a broader range of plant and disease types.

References

[1] Jinzhu Lu,Lijuan Tan,ORCID and Huanyu Jiang 3ORCID "Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification".Vol 11(8), 707, 27 July 2021, https://doi.org/10.3390/agriculture11080707.

[2] Muhammad Hammad Saleem ,Johan Potgieter and Khalid Mahmood Arif , Plant Disease Detection and Classification by Deep Learning, Department of Mechanical and Electrical Engineering, School of Food and Advanced Technology, Massey University, Auckland 0632, New Zealand, 31 October 2019, pp.468, https://doi.org/10.3390/plants8110468.

[3] Mohanty, S. P., Hughes, D. P., Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in plant science, 7, 215232.

[4] Chohan, Murk, et al. "Plant disease detection using deep learning." International Journal of Recent Technology and Engineering 9.1 (2020): 909-914.

[5] Li, Lili, Shujuan Zhang, and Bin Wang. "Plant disease detection and classification by deep learning—a review." IEEE Access 9 (2021): 56683-56698.

[6] Ferentinos, Konstantinos P. "Deep learning models for plant disease detection and diagnosis." Computers and electronics in agriculture 145 (2018): 311-318.

[7] Dataset collected from https://www.kaggle.com/datasets/vipoooool/new-plant- diseases-dataset?resource=downloa

[8] Guangfen Wei,Gang Li ,Jie Zhao and Aixiang He , School of Information Electronic Engineering, Shandong Technology and Business University, Yantai 264005, China,8 January 2019, 19(1), 217; https://doi.org/10.3390/s19010217

[9] Information collected from: https://medium.com/@siddheshb008/alexnet-architecture-explained-b6240c528bd5

[10] Information collected from: https://towardsdatascience.com/deep-learning-googlenet-explained-de8861c82765

[11] Information collected from: https://datagen.tech/guides/computer-vision/vgg16/

[12] Information collected from : https://datagen.tech/guides/computer-vision/resnet-50/