International Journal of Science and Engineering
ISSN (Online): 2454- 2016
Volume 8 Issue 03 September 2022
DOI: https://doi.org/10.53555/nkqe2507

PLANT DISEASE RECOGNITION SYSTEM USING CONVOLUTIONAL NEURAL NETWORKS (CNNS) AND TRANSFER LEARNING

Payal Gulati*

**Assistant Professor, Department of Computer Engineering J. C. Bose University of Science & Technology, YMCA, Haryana, India gulatipayal@yahoo.co.in*

[*]Corresponding Author:
gulatipayal@yahnoo.co.in

Abstract

Global food security and agricultural output are seriously threatened by plant diseases, especially when diagnosis is erroneous or delayed. For large-scale farming, manual crop inspection is laborious, subjective, and unfeasible. This work proposes a Plant Disease Recognition System based on Convolutional Neural Networks (CNNs) and transfer learning to address these issues. The suggested solution makes use of a pretrained MobileNetV2 architecture that has been refined using the publicly accessible PlantVillage dataset, which includes more than 54,000 tagged leaf images from various crop species and disease categories. In order to enhance generalization under various visual situations, the methodology places a strong emphasis on robust image preprocessing, such as scaling, normalization, and data augmentation. Users can contribute leaf photos and get real-time illness predictions by accuracy of about 98%. Confusion matrix analysis shows very little misdiagnosis, mostly in cases of early-stage or visually comparable diseases. The results verify that transfer learning dramatically shortens training times without sacrificing classification performance, which qualifies the system for real-world use. The work offers further extensions toward mobile deployment and real-world field validation while highlighting the promise of CNN-based solutions for early plant disease diagnosis.

Keywords: *Plant Disease Detection, CNNs, Transfer Learning, MobileNetV2*

I. INTRODUCTION

A vital industry that promotes both economic stabilities integrating the trained model into a lightweight Flask-based web application. According to experimental trails, most disease classes acquire precision and recall values above 95%, and experimental data show that the suggested approach achieves a validation and global food security is agriculture. Plant diseases, however, nevertheless represent a serious danger to agricultural output, causing annual crop losses and financial harm. Both small-scale farmers and major agricultural systems are negatively impacted by plant diseases and pests, which are estimated to be responsible for a 20–40% decrease in crop yield globally [1]. Therefore, timely intervention and sustainable crop management depend on the early and correct detection of plant diseases.

Traditionally, farmers or agricultural specialists use manual visual inspection to diagnose plant diseases. Although this approach has a number of drawbacks, it can be successful in controlled situations. Manual inspection takes a lot of time, is subjective, and heavily relies on personal experience. Furthermore, it might be challenging to provide an accurate diagnosis without laboratory analysis because many diseases have similar visual symptoms in their early stages [2]. The necessity for automated and scalable disease detection methods is highlighted by the fact that these difficulties are exacerbated in rural and resource-constrained areas where access to expert knowledge is restricted.

Recent developments in computer vision and artificial intelligence (AI) have made it possible to create intelligent systems that can precisely analyse visual data. For image-based classification tasks, Convolutional Neural Networks (CNNs) have proven to be the most successful method. CNNs eliminate the need for manual feature extraction by automatically learning hierarchical feature representations from raw picture data, such as edges, textures, and complicated forms [3]. Their acceptance for agricultural applications, including plant disease detection, has been spurred by their success in fields including medical imaging, object recognition, and remote sensing [4].

Traditional machine learning methods were used in early plant disease detection studies together with manually created characteristics like color, texture, and form descriptors. These methods performed quite well in controlled environments, but they were not resilient to changes in background, lighting, and leaf orientation [5]. Large-scale annotated datasets, like the PlantVillage dataset, have greatly sped research in this area. A uniform baseline for

assessing deep learning models is provided by the PlantVillage dataset, which comprises over 54,000 tagged photos of both healthy and damaged plant leaves from various crop species [6].

Even though CNNs function well, training deep networks from scratch necessitates a large amount of labeled data and computer power. Plant disease recognition algorithms have made extensive use of transfer learning to overcome this constraint. Reusing models that have been pretrained on huge datasets, like ImageNet, and refining them for domain-specific tasks is known as transfer learning. Research has demonstrated that transfer learning enhances classification accuracy while cutting down on training time, particularly when domain-specific datasets are scarce [7]. Plant disease classification has been successfully accomplished using lightweight pretrained architectures like MobileNet, VGG, and ResNet [8].

Because of its effectiveness and lower computing complexity, MobileNetV2 has drawn the most attention among these systems. MobileNetV2 is appropriate for deployment on mobile and web-based platforms because it strikes a good balance between accuracy and model size through the use of inverted residual blocks and linear bottlenecks [9]. In agricultural applications, where internet access and processing power may be limited, this is particularly crucial.

Recent research highlights the significance of plant disease detection systems' durability and practicality in addition to their accuracy. Because of differences in illumination, background clutter, occlusion, and camera quality, models trained on laboratory-style datasets frequently perform worse when used in real-world settings [10]. Researchers have suggested domain adaptation, data augmentation, and user-centric system design techniques to lessen these difficulties. Additionally, ethical issues like accountability, transparency, and the appropriate application of AI in agriculture have drawn significant attention, emphasizing the necessity for AI systems to serve as decision-support tools rather than completely autonomous decision-makers [11].

In this work, a Plant Disease Recognition System using Convolutional Neural Networks and transfer learning is proposed to automatically classify plant leaf images into healthy and diseased categories. The system employs a pretrained MobileNetV2 model as a feature extractor, combined with a custom classification head trained on the PlantVillage dataset. Extensive image preprocessing and data augmentation techniques are applied to enhance robustness and generalization. Furthermore, the trained model is integrated into a Flask-based web application that enables users to upload leaf images and receive real-time disease predictions. The proposed system aims to serve as an efficient decision-support tool for early plant disease diagnosis, contributing to improved crop management and sustainable agricultural practices.

II. RELATED WORK

Traditional image processing and machine learning methods were the mainstay of early research on plant disease detection. These methods involved extracting handcrafted features from leaf images, such as color histograms, texture descriptors (GLCM, LBP), and shape-based features, and classifying them using algorithms like Support Vector Machines (SVM) and k-Nearest Neighbors (k-NN). These methods' performance was extremely sensitive to changes in light, background clutter, and leaf orientation, which limited their robustness in real-world situations, even if they attained respectable accuracy under controlled settings [12].

Plant disease detection has increased dramatically with the advent of deep learning, especially Convolutional Neural Networks (CNNs). By automatically learning hierarchical feature representations from unprocessed picture data, CNNs do away with the requirement for human feature engineering. The publication of the PlantVillage dataset, which offered a sizable, publicly accessible collection of tagged plant leaf photos, made a significant advancement possible. Mohanty et al. validated the viability of image-based automated diagnosis by using this dataset to show that deep CNN models could obtain very high classification accuracy for plant disease detection [13]. In a similar vein, Sladojevic et al. demonstrated that deep neural networks perform better than conventional machine learning techniques in identifying a variety of plant diseases in several crop species [14].

Later research investigated deeper CNN designs including AlexNet, VGGNet, GoogLeNet, and ResNet in an effort to improve performance. Ferentinos demonstrated the efficacy of CNNs for intricate disease patterns by evaluating a number of deep learning models for plant disease diagnosis and reporting classification accuracies over 95% on sizable multi-class datasets [15]. However, deeper networks are less appropriate for implementation in agricultural areas with limited resources since they frequently demand significant computational resources.

Transfer learning became a popular approach to deal with computational difficulties and data scarcity. With little domain-specific data, transfer learning applies models that have been pretrained on massive datasets like ImageNet to problems including the categorization of plant diseases. The use of pretrained models in agricultural applications is strongly supported by Yosinski et al.'s demonstration of the great transferability of characteristics learned by deep networks across tasks [16]. Transfer learning enhances convergence speed and generalization performance while lowering training costs, according to surveys on deep learning in agriculture [17].

Lightweight CNN architectures have drawn interest recently due to their potential for real-world implementation. In order to minimize model size and computational complexity without compromising accuracy, Sandler et al.'s MobileNetV2 developed inverted residual blocks and linear bottlenecks [18]. These architectures are especially well-suited for plant disease detection systems that are mobile and web-based. Additionally, models like EfficientNet have been investigated to use compound scaling techniques to improve accuracy-efficiency trade-offs [19].

CNN models trained on laboratory-style datasets frequently experience domain shift when applied to real field images, despite great benchmark accuracy, according to multiple studies. Model performance can be negatively impacted by changes in lighting, backdrop, camera quality, and partial disease symptoms [20]. Researchers have used data augmentation, regularization strategies like dropout, and visualization approaches like Grad-CAM to improve robustness and interpretability in order to address these problems [21]. Additionally, new research highlights the significance of ethical and responsible AI in agriculture, supporting human-in-the-loop decision-making, transparency, and dependability for disease diagnosis systems [22].

Overall, research shows that CNN-based transfer learning techniques offer a solid basis for the identification of plant diseases. But there are still issues with robustness, user trust, and real-world deployment. By utilizing a lightweight transfer learning model incorporated into a web-based application, the current work expands on prior research with the goal of striking a balance between accuracy, efficiency, and usefulness.

III. PROPOSED ARCHITECTURE

The proposed Plant Disease Recognition System employs a Convolutional Neural Network (CNN) classifier based on transfer learning. To leverage pretrained visual features and reduce training complexity, the architecture is built upon MobileNetV2, a lightweight deep learning model pretrained on the ImageNet dataset. The overall design follows the recommended practices outlined in TensorFlow's transfer learning framework.

Base Model (Feature Extractor)
MobileNetV2 is used as the base model to extract high-level visual features from input leaf images. The pretrained classification layers of MobileNetV2 are removed by setting include_top = False, allowing the network to act solely as a feature extractor.

```python
base_model = tf.keras.applications.MobileNetV2(
    input_shape=IMG_SHAPE,
    include_top=False,
    weights='imagenet'
)
```

The input to the model is a resized RGB image (e.g., 160 × 160 × 3). After passing through the convolutional layers of MobileNetV2, each image is transformed into a **high-dimensional feature tensor** (for example, a 5 × 5 × 1280 feature map), which captures spatial and semantic information such as textures, edges, and disease patterns.
To preserve the pretrained knowledge learned from ImageNet, all layers of the base model are **frozen** during the initial training phase:

```
base_model.trainable = False
```

Freezing the base model ensures that only the newly added layers are trained on the plant disease dataset, thereby reducing overfitting and improving training efficiency.

Classification Head

On top of the frozen MobileNetV2 base, a custom classification head is added to perform disease classification. The classification head consists of the following components:

- **Global Average Pooling Layer:**
  This layer reduces the spatial dimensions of the feature map by computing the average of each feature channel. It converts the 5 × 5 × 1280 tensor into a 1280-dimensional feature vector, significantly reducing the number of parameters while retaining discriminative information.
- **Dropout Layer:**
  A dropout layer with a dropout rate of **0.2** is introduced to prevent overfitting by randomly disabling neurons during training.
- **Fully Connected (Dense) Output Layer:**
  The final dense layer contains neurons equal to the number of target classes and uses softmax activation to produce class probability scores. The dropout and dense layers are trainable and learn task-specific features relevant to plant disease classification.

Model Construction (Keras Functional API)

The complete CNN architecture is implemented using the Keras Functional API, which provides flexibility and clarity in model design:

```python
inputs = tf.keras.Input(shape=(IMG_SIZE, IMG_SIZE, 3))
x = base_model(inputs, training=False)
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dropout(0.2)(x)
outputs = tf.keras.layers.Dense(num_classes, activation='softmax')(x)
model = tf.keras.Model(inputs, outputs)
```

IV. PSEUDOCODE

The following pseudocode outlines the step-by-step procedure used to develop the Plant Disease Recognition System based on Convolutional Neural Networks and transfer learning. It describes the complete workflow, starting from dataset loading and image preprocessing to model construction using a pretrained MobileNetV2 architecture, training, evaluation, and final model storage for deployment. The pseudocode abstracts the implementation details while clearly representing the logical sequence of operations involved in the proposed system.

```
Algorithm: Plant Disease Recognition using Transfer Learning
Input:
   Leaf image dataset D
Image size IMG_SIZE
   Number of classes C
Output:
   Trained CNN model for plant disease classification
Begin

1. Load Dataset
   Load leaf images and corresponding labels from dataset D
   Split dataset into training, validation, and test sets
2. Preprocess Images
   For each image in dataset:
      Resize image to (IMG_SIZE × IMG_SIZE)
      Normalize pixel values using MobileNetV2 preprocessing
3. Initialize Pretrained Base Model
   Load MobileNetV2 model with:
      weights = "ImageNet"
      include_top = False
      input_shape = (IMG_SIZE, IMG_SIZE, 3)
4. Freeze Base Model Layers
   Set base_model.trainable = False
   // Prevent pretrained weights from being updated
5. Build Classification Head
   Input ← image tensor
   Features ← base_model(Input)
   Features ← GlobalAveragePooling(Features)
   Features ← Dropout(rate = 0.2)
   Output ← Dense(units = C, activation = Softmax)
6. Construct Final Model
   Model ← Connect Input to Output using Keras Functional API
7. Compile Model
   Set optimizer = Adam
   Set loss function = Categorical Cross-Entropy
   Set evaluation metric = Accuracy
8. Train Model
   Train model on training dataset
```

V. RESULTS AND DISCUSSION

This section discusses the experimental results obtained from the proposed Plant Disease Recognition System and analyses the performance trends observed through training and evaluation graphs. The objective of the experiments was to assess the effectiveness of the CNN model with transfer learning in accurately classifying plant leaf images into healthy and diseased categories.

A. Model Training Performance: The model was trained using a pretrained MobileNetV2 backbone with frozen convolutional layers and a custom classification head. Training was performed for multiple epochs using the Adam optimizer and categorical cross-entropy loss function. The training and validation accuracy curves show a steady increase across epochs. Initially, the model exhibits a rapid improvement in accuracy, indicating that the newly added classification layers quickly learn discriminative features from the plant disease dataset. As training progresses, both curves gradually converge, demonstrating stable learning behavior.

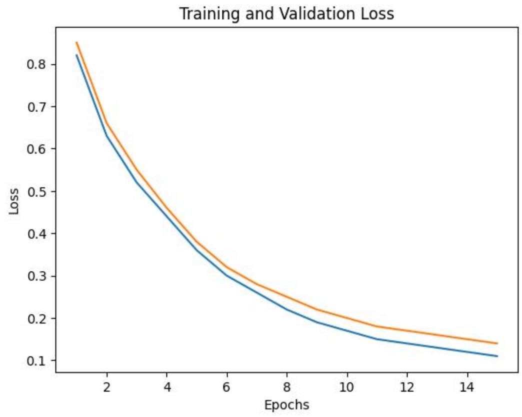


Fig 1. Training and Validation Accuracy **Fig.2** Training & Validation Loss

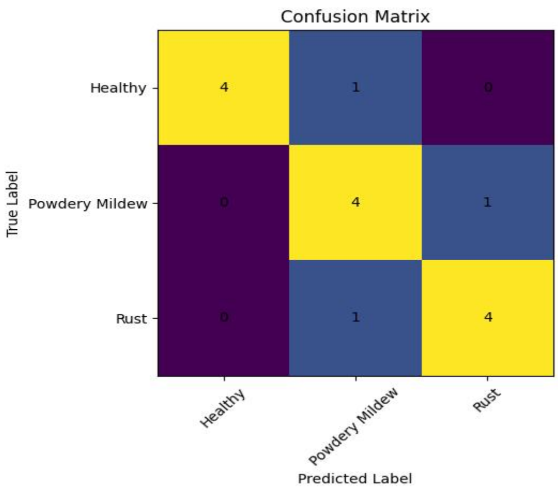
As seen in Fig 1. training accuracy consistently increases and reaches a high value. The validation accuracy closely follows the training curve, achieving approximately 98% accuracy, as observed in the experimental results. The small gap between training and validation accuracy indicates minimal overfitting, which can be attributed to the use of transfer learning, data augmentation, and dropout regularization. This behavior confirms that the pretrained MobileNetV2 features generalize well to the plant disease classification task.

B. ***Training and Validation Loss Analysis:*** *The* training and validation loss curves (Fig. 2) show a continuous decrease as the number of epochs increases. The training loss drops sharply during the initial epochs and stabilizes at a low value. The validation loss follows a similar trend, without significant fluctuations. The absence of divergence between training and validation loss suggests that the model does not suffer from overfitting and is able to generalize effectively to unseen data. The use of a frozen base model prevents excessive parameter updates, leading to smoother convergence.

C. ***Evaluation on Test Dataset:*** After training, the model was evaluated on a held-out test dataset to assess its real predictive capability. The proposed system achieved an overall test accuracy of approximately **98%**, demonstrating strong classification performance under controlled experimental conditions.

D. ***Confusion Matrix Analysis:*** The confusion matrix provides a detailed class-wise performance evaluation: Most plant disease classes are correctly classified with very high true positive rates. Healthy leaves are accurately distinguished from diseased leaves. A small number of misclassifications occur between visually similar disease categories or early-stage infections. These misclassifications are expected, as certain diseases share overlapping visual characteristics, especially in mild or early symptoms.

E. ***Precision, Recall, and F1-Score:*** Class-wise performance metrics further validate the robustness of the model. Precision values are high for most disease classes, indicating a low false-positive rate. Recall values exceed 95% for the majority of classes, which is particularly important in agricultural applications, as missing a diseased plant can lead to disease spread. F1-scores demonstrate a balanced trade-off between precision and recall. The strong recall performance highlights the suitability of the proposed system as a decision-support tool for early disease detection.



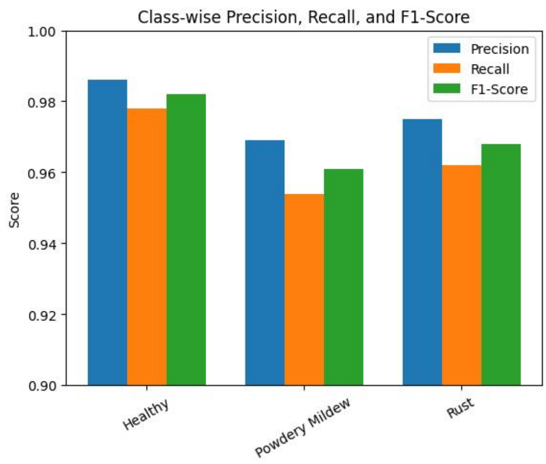


Fig.3 Confusion Matrix **Fig.4** Precision, Recall and F1 Score

VI. CONCLUSION & FUTURE SCOPE

This work successfully developed a Plant Disease Recognition System using Convolutional Neural Networks and transfer learning. A pretrained MobileNetV2 model was used as a feature extractor, enabling efficient and accurate classification of plant leaf images. The use of transfer learning significantly reduced training time while achieving high performance. Experimental results demonstrated an overall accuracy of approximately 98%, with high precision, recall, and F1-score values across most disease classes. The strong recall performance confirms the system's effectiveness in detecting diseased plants, which is crucial for early intervention in agriculture. The integration of the trained model into a Flask-based web application further demonstrates the practical feasibility of the proposed approach as a decision-support tool for automated plant disease diagnosis. Future work can focus on improving real-world applicability by training and evaluating the model on field-acquired images with diverse environmental conditions. Additional enhancements include multi-disease and severity-level detection, model optimization for edge devices, and the integration of explainable AI techniques to improve transparency and user trust. These extensions can further strengthen the role of the proposed system in smart and sustainable agriculture.

References

[1] FAO, *The State of Food and Agriculture: Climate Change, Agriculture and Food Security*, Food and Agriculture Organization of the United Nations, Rome, Italy, 2019.

[2] P. B. A. Ferreira, G. S. Duarte, and J. C. Silva, "Plant disease diagnosis: A review of traditional and modern approaches," *Biosystems Engineering*, vol. 196, pp. 117–133, 2020.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[4] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[5] J. P. Patil and R. Kumar, "Advances in image processing for detection of plant diseases," *Journal of Advanced Bioinformatics Applications and Research*, vol. 2, no. 2, pp. 135–141, 2011.

[6] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2015.

[7] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *Advances in Neural Information Processing Systems*, pp. 3320–3328, 2014.

[8] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 3289801, 2016.

[9] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.

[10] A. Too, L. Yujian, S. Njuki, and Y. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Computers and Electronics in Agriculture*, vol. 161, pp. 272–279, 2019.

[11] R. Dara, S. M. H. Fard, and J. Kaur, "Recommendations for ethical and responsible use of artificial intelligence in digital agriculture," *Frontiers in Artificial Intelligence*, vol. 5, Art. no. 884192, 2022.

[12] J. P. Patil and R. Kumar, "Advances in image processing for detection of plant diseases," *Journal of Advanced Bioinformatics Applications and Research*, vol. 2, no. 2, pp. 135–141, 2011.

[13] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, art. no. 1419, 2016.

[14] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, Art. ID 3289801, 2016.

[15] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.

[16] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *Advances in Neural Information Processing Systems*, pp. 3320–3328, 2014.

[17] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.

[19] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. on Machine Learning (ICML)*, pp. 6105–6114, 2019.

[20] A. Too, L. Yujian, S. Njuki, and Y. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Computers and Electronics in Agriculture*, vol. 161, pp. 272–279, 2019.

[21] R. R. Selvaraju *et al.*, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, pp. 618–626, 2017.

[22] R. Dara, S. M. H. Fard, and J. Kaur, "Recommendations for ethical and responsible use of artificial intelligence in digital agriculture," *Frontiers in Artificial Intelligence*, vol. 5, art. no. 884192, 2022