Research on AI Applications for Plant Leaf Disease Detection and Classification

Hoang Trung Nguyen, Tran Ly Minh Hoang, Le Duc Khang, Nguyen Nhat Minh
(nguyenhtde180300@fpt.edu.vn, FPT University, Da Nang)

Abstract

Timely and precise identification of foliar diseases is crucial for improving crop output and ensuring food security. This study examines an extensive array of machine learning and deep learning techniques for the automated categorisation of plant leaf diseases. Conventional models, including Logistic Regression, Support Vector Machines (SVM), Random Forest, K-nearest Neighbours, and LightGBM, were evaluated, achieving a peak accuracy of 87% through traditional computer vision approach. On the other hand, by utilizing deep features extraction with classical classifiers, these aforementioned models reached an aggregate accuracy of 91% .

Conversely, deep learning models such as DenseNet, EfficientNet, ResNet, Vision Transformer (ViT), and Swin Transformer were employed in an end-to-end methodology. The highest-performing design achieved 95% accuracy, significantly surpassing traditional methods. These findings highlight the effectiveness of modern convolutional and transformer-based vision models in addressing visual variability, occlusion, and the presence of several conditions in field-like settings. The research introduces a scalable and efficient AI framework for practical use in agriculture, especially in resource-constrained farming regions.

Keywords: Plant disease detection, deep learning, precision agriculture, computer vision, multi-modal systems, disease classification

I. INTRODUCTION

The rising global demand for sustainable agriculture requires innovative approaches to the early and accurate identification of foliar diseases, which drastically affect crop productivity and food security. This study systematically examines the applications of artificial intelligence for the automated identification and classification of foliar diseases in crops.The research addresses key challenges in plant disease recognition, including diversity in symptom manifestation, environmental interference, and the requirement for scalable solutions suitable for eclectic agricultural circumstances.

We develop an integrated deep learning framework that leverages computer vision to enhance diagnostic precision and stability across multiple plant species. The proposed approach utilized an architecture that analyses visual data for thorough illness evaluation. Computer vision components employ sophisticated convolutional neural networks and transformer-based models to extract distinctive features from leaf images. The system is engineered for computational efficiency in mind, offering practical application in various agricultural settings.

Through extensive experimentation, the framework demonstrates robust performance in handling real-world complexities such as occlusions, fluctuating illumination conditions, and the presence of numerous concurrent diseases . The research presents three principal contributions: (1) an innovative method for vision-based plant disease diagnostics, (2) enhanced deep learning frameworks for agricultural use, and (3) effective implementation methodologies for field application.

These advancements facilitate intelligent crop surveillance, allowing for early disease identification and precision agriculture. The results connect AI research with real agricultural requirements, especially in resource-constrained areas, providing scalable solutions for global food security.

II. RELATED WORK

Recent advancements in the classification of plant leaf diseases have been predominantly propelled by deep learning (DL) and machine learning (ML) techniques, which have exhibited considerable enhancement in accuracy and efficiency relative to conventional approaches.

Zhang et al. (2024) introduced ERCP-Net, an innovative network architecture that integrates a channel extension residual structure structure with an adaptive channel attention mechanism[1]. This method augments feature extraction and elevates the classification accuracy of plant leaf diseases by concentrating on the most

informative channels within the feature maps. Their model exhibited robust performance across several disease datasets, illustrating the efficacy of attention mechanisms in plant disease diagnosis.

Calma et al. (2023) utilized MobileNetV3, a lightweight convolutional neural network designed for mobile and embedded devices, to identify cassava illnesses[2]. Their strategy enhanced the model's capacity to generalize across various illness signs and environmental variables by supplementing stem and leaf imagery. This study emphasizes the increasing trend of utilizing effective deep learning models for real-time illness diagnosis in resource-limited settings..

Additional research has investigated hybrid models that integrate DL feature extractors, such as EfficientNet and DenseNet with conventional ML classifiers such as Random Forest and AdaBoost, attaining classification accuracies surpassing 95%. Attention-based architectures and multi-head self-attention processes have been utilized to augment the model's emphasis on critical disease characteristics, resulting in enhanced detection rates even under adverse settings such as class imbalance and noisy backgrounds.

Thorough evaluations of the domain highlight the significance of transfer learning, data augmentation, and lightweight architectures to enable the effective implementation of disease classification systems in agriculture. The amalgamation of these methodologies has facilitated the creation of mobile applications and web-based tools that aid farmers in early disease detection, hence enhancing crop management and yield preservation.

III. METHODOLOGY

Our approach for automated plant leaf disease identification consists of a detailed, multi-phase workflow, as depicted in Figure 1. Initially, leaf images undergo critical preprocessing procedures, including segmentation to identify the leaf region and comprehensive data augmentation to improve dataset variety and rectify class imbalance. Subsequently, feature extraction is conducted through two distinct methodologies: handcrafted features (Shape, Color, GLCM, LBP, and basic descriptors) are utilized to capture conventional visual attributes, while deep features are derived from pre-trained convolutional and transformer-based models (VGG, DenseNet, ResNet, EfficientNet) to leverage their robust learned representations. The various elements are subsequently integrated and undergo feature selection to enhance representation. Finally, the produced feature sets are fed into a suite of classic machine learning classifiers (KNN, RF, SVM, LR, LightGBM) for training and rigorous performance evaluation, allowing for a complete comparative examination of their usefulness in differentiating distinct leaf diseases.
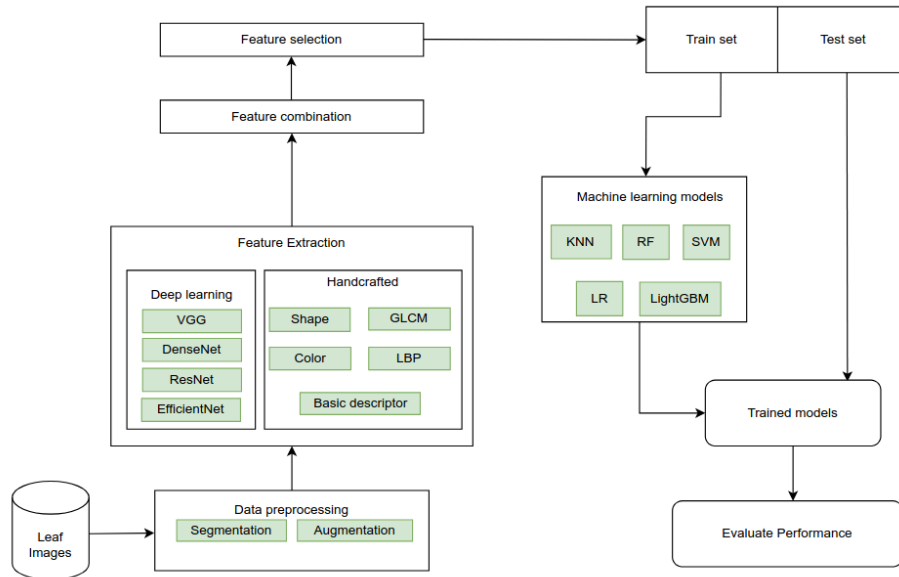

FIGURE 1: THE OVERALL PIPELINE OF OUR PROPOSED METHOD

A. Dataset

This premium picture dataset, tailored for plant disease diagnosis, consists of 16,584 images predominantly featuring plant leaves. The photos are methodically divided into 18 separate illness classifications, each equally sized at 256x256 pixels and kept in JPG format.

To ensure uniformity and reduce variability within each illness category, all pictures were obtained at a standardized distance. The dataset is systematically partitioned into three subsets for comprehensive model construction and assessment: 70% allocated for training, 15% for validation, and 15% for testing. The fundamental goal of this collection is to underpin research and development in automated plant disease categorization and detection using photography. The precise distribution of images across all 18 separate illness classes within our dataset is given in Figure 1, showing the varied number of samples available for each category and indicating potential class imbalances.

To address the observed data imbalance, where certain illness classes are severely underrepresented, we employed a two-pronged method within our deep learning pipeline. This entailed the use of rigorous Data Augmentation to synthetically enhance the diversity and volume of data, particularly for minority classes, thus augmenting the model's exposure to diverse visual patterns. Simultaneously, we employed Weighted Loss Functions in model training, which dynamically impose greater penalties on prediction failures from minority classes, hence enhancing the model's ability to learn their differentiating characteristics more effectively. These strategies collectively aim to boost the model's generalization capabilities and improve performance across all disease categories, as explained in the following subsections.
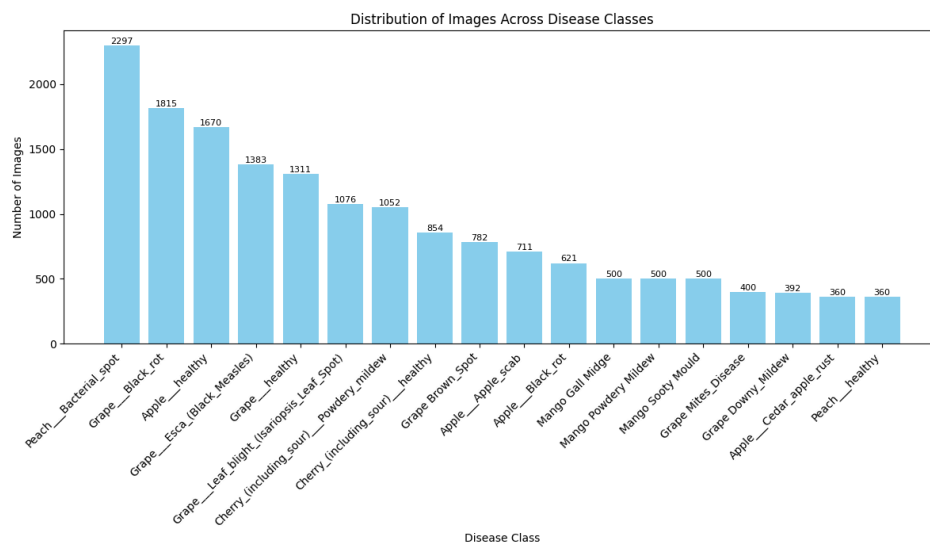

FIGURE 2: CLASS DISTRIBUTION IN THE DATASET.

B. Data Preprocessing

The data pretreatment pipeline was established to ready plant leaf pictures for effective feature extraction and categorization through machine learning techniques. The pipeline comprised three primary stages: (1) data sampling to incorporate images featuring both intricate and simple backgrounds, (2) image segmentation utilizing the Segment Anything Model (SAM) with manually constructed polygon masks in YOLO format, succeeded by the training of YOLOv11seg for the automated segmentation of the residual dataset. We additionally employ deep feature extraction with convolutional neural network (CNN) backbone models, such as ResNet, DenseNet, and EfficientNet. These measures guaranteed superior input data for subsequent machine learning classifiers.

1. Data sampling

To ensure robustness in the classification model, the dataset was curated to include a comprehensive array of plant leaf photos, featuring both intricate (e.g., overlapping leaves or varying illumination) and uncomplicated backdrops. This sampling approach sought to encompass all variations of real-world circumstances. The sampled photos underwent preprocessing using standard approaches, which included scaling to a consistent resolution (e.g., 224×224 pixels) and normalizing pixel values to the range [0, 1], as is customary in plant disease diagnosis.

2. Image segmentation

A hybrid approach was developed to provide the maximum quality in picture segmentation. A group of photos was initially generated and segmented utilizing both the Segment Anything Model (SAM) and a manually crafted polygon mask, subsequently formatted in YOLO to guarantee accurate delineation of leaf boundaries, especially in intricate scenarios including overlapping leaves or varying lighting conditions. The annotations were

utilized to train the YOLOv11 segmentation model, a cutting-edge segmentation framework, which was then applied to the remaining photos. This hybrid methodology, including SAM's automation, manual refining, and YOLOv11seg's scalability, guaranteed accurate leaf segmentation throughout the dataset. Following segmentation, data augmentation methods, including random rotations, flips, and brightness modifications, were employed to improve model resilience. Figure 3 presents an example of segmented leaves.

FIGURE 3: AN EXAMPLE OF SOME SEGMENTED LEAF

3. Data Augmentation

To rectify data imbalance among the dataset and mitigate model variance to avoid overfitting, we used a variety of image augmentation strategies, such as rotation and vertical and horizontal flipping. We employed the Albumentations library for most augmentation techniques owing to its efficacy and adaptability. In specific instances, we employed custom-designed augmentation techniques to guarantee that the modified outputs adhered to particular criteria or maintained task-relevant attributes. This amalgamation of standard and customized augmentations was formulated to improve the model's generalization capacity while preserving data integrity.

C.   Traditional Computer Vision Approach

Machine learning and conventional computer vision methods have been extensively utilized for the diagnosis of plant leaf diseases owing to their interpretability, efficiency, and adaptability[5][6]. This part builds upon previous research to investigate the application of conventional machine learning algorithms, focusing on feature extraction techniques and assessing the performance of the models.

We will utilize the segmented images obtained using the aforementioned segmentation techniques for feature extraction and machine learning algorithms, concentrating exclusively on leaves rather than the background.

a. Feature Extraction Methods

Efficient detection depends on the extraction of significant visual elements from leaf images. This research examines the subsequent computer vision techniques:

- Color Features: Fundamental color descriptors (RGB, HSV, Lab, YCbCr, and XYZ) color spaces and associated statistical summaries (mean, standard deviation, skewness, kurtosis) to encapsulate critical visual attributes for the identification of disease symptoms.
    - RGB: The predominant additive color model, depicting colors as amalgamations of red, green, and blue light intensities.
    - HSV: More comprehensible for the human eye, distinguishing color type (Hue) from its intensity (Value) and purity (Saturation).
    - Lab: A device-agnostic color space engineered for perceptual uniformity, wherein numerical variations align with human-perceived color distinctions.
    - YCbCr: Distinguishes luminance (Y) from chrominance (Cb for blue difference, Cr for red difference). Predominantly utilized in video and digital photography.
    - XYZ: A device-agnostic color space that functions as a standard reference for all other color spaces, including the complete spectrum of human color perception.

    Statistical Summaries: For each channel within these color spaces, we compute:

    - Mean: The average color intensity, signifying the predominant hue.
    - Standard Deviation: Assesses the dispersion or variance of color intensity, indicating color homogeneity.
    - Skewness: Indicates the asymmetry of the color distribution, reflecting a preference for lighter or darker tones.
    - Kurtosis : Assesses the "tailedness" of a distribution, indicating whether color intensities are concentrated or dispersed.
- Texture Features: Gray-Level Co-occurrence Matrix (GLCM)[7] and Local Binary Patterns (LBP) methodologies to measure texture patterns characteristic of damaged leaf tissues.
    - GLCM measures the frequency of various pairs of gray-level values (pixel intensities) that appear in a defined spatial relationship (distance and angle) inside a picture. For affected leaves, this indicates it can capture:
        - Changes in uniformity: Healthy leaf regions may exhibit a generally consistent texture, whereas diseased parts may display blotches or uneven patterns. The Homogeneity or Energy characteristics of GLCM would indicate this.
        - Presence of spots/lesions: If a disease induces dark spots on a lighter background, the GLCM would have a high co-occurrence of low-intensity pixels close to high-intensity pixels. The contrast and dissimilarity in these places would be significant.
        - Roughness/Smoothness: Specific illnesses may cause the leaf surface to exhibit increased roughness or smoothness. GLCM can encapsulate this through features such as Entropy or Angular Second Moment (ASM).
    - LBP characterizes the local texture around each pixel by contrasting its intensity with that of its neighbors. For affected foliage, this corresponds to:
        - Roughness/Smoothness: Specific illnesses may cause the leaf surface to exhibit increased roughness or smoothness. The Gray Level Co-occurrence Matrix (GLCM) can encapsulate this through elements such as Entropy or Angular Second Moment (ASM).
        - Roughness/Smoothness: Certain illnesses may cause the leaf surface to exhibit increased roughness or smoothness. GLCM can encapsulate this through features such as Entropy or Angular Second Moment (ASM).
- Shape Features: Zernike Moments[9][10] to examine form attributes and identify morphological alterations in leaves. Zernike Moments are orthogonal and rotation-invariant shape descriptors utilized to examine and quantify morphological alterations in plant leaves, hence facilitating the detection and diagnosis of plant diseases irrespective of leaf orientation or position within an image.
- Morphological Features: Morphological features quantify the geometric aspects of a segmented item (e.g., a leaf, a disease spot), facilitating an understanding of its shape, size, and overall structure. Alterations in these characteristics frequently act as vital indications of illness.

- Area & Perimeter: Quantifying the dimensions and border length of the impacted region.
- Aspect Ratio & Roundness: Characterizing the object's elongation or circularity.
- Solidity & Convexity: Indicating the degree to which a shape is "filled" or "indented".
- Symmetry: Identifying anomalies or abnormalities in morphology.
- Equivalent Diameter: Standardizing the size of an object to the diameter of a comparable circle.

b. Machine Learning Classifiers

Employing the retrieved features through the aforementioned methods, we apply conventional machine learning classifiers to differentiate various leaf diseases:

- Logistic Regression (LR): Employs the logistic function to transform a linear amalgamation of input features into probabilities ranging from 0 to 1. It delineates the contribution of each trait to the probability of illness manifestation.
- K-Nearest Neighbors (KNN): A basic classifier that allocates labels according to the nearest training instances in the feature space.[11]
- Support Vector Machines (SVM): identify an ideal hyperplane that segregates data points of distinct classes, increasing the margin to improve generalization. It can manage intricate and high-dimensional characteristics.[12]
- Random Forest (RF): An ensemble method that generates several decision trees by bootstrap sampling and randomly chosen feature subsets at each node. The ultimate prediction consolidates these distinct tree outputs, enhancing resilience and mitigating overfitting.[13]
- LightGBM: A gradient-boosting framework that adeptly manages extensive datasets and intricate, high-dimensional feature spaces through the sequential construction of decision trees, enhancing classification efficacy and computing efficiency.[14]

D. Deep Learning-based

Deep learning methodologies have gained prominence in the diagnosis of plant leaf diseases, with the benefit of autonomously developing intricate feature representations from unprocessed data. This section examines two deep learning methodologies: the extraction of deep features from pretrained models for use with conventional machine learning classifiers, and the end-to-end training of deep neural networks for direct disease categorization. A range of deep architectures is utilized, encompassing convolutional neural networks such as DenseNet201, ResNet50/ResNet50V2, ResNet101, and EfficientNetB0, alongside transformer-based models include Swin Transformer and Vision Transformer (ViT). This study comparatively assesses these methodologies to determine their efficacy and trade-offs in predicting plant leaf diseases.

1. Feature Extraction Using Pre-trained Deep Models

This method involves extracting deep features from the intermediate layers of pretrained convolutional and transformer models that have been trained on extensive datasets like ImageNet. These attributes function as input for several conventional machine learning classifiers, such as Logistic Regression, Support Vector Machine (SVM), Random Forest, K-Nearest Neighbors (KNN), and Light Gradient Boosting Machine (LightGBM). This hybrid technique seeks to harness the robust representational capabilities of deep models while capitalizing on the efficiency, interpretability, and variety of traditional machine learning algorithms. This subsection outlines the feature extraction pipeline, the selection of feature layers, and the comparative efficacy of the utilized classifiers.

We utilized a range of pretrained convolutional and transformer-based models as feature extractors to derive significant deep representations from leaf images. Features were specifically retrieved from the global average pooling (GAP) layer of convolutional neural networks, including VGG19, ResNet50, DenseNet201, and EfficientNet.

All models underwent pretraining on the ImageNet dataset, facilitating their effective generalization to various visual tasks beyond their initial training purposes. Input images were scaled to conform to the dimensional requirements of each design, often 224×224 pixels for convolutional neural networks (CNNs).

The dimensionality of the resultant feature vectors fluctuate based on the mode, however each functioned as a concise and semantically dense representation of the image content.

Principal Component Analysis (PCA) was employed on the extracted feature vectors to minimize redundancy and computational expenses. The quantity of primary components was established according to the

Kaiser criterion, preserving only those with eigenvalues over 1, so ensuring that each chosen component represents more variation than an average original variable.

The reduced-dimensional feature vectors were subsequently utilized as input for different conventional machine learning classifiers for disease prediction, as outlined in the following part.

2. End-to-End Deep Learning for Disease Classification

2.1 Introduction to End-to-End Deep Learning Models

Traditional methods for detecting diseases often depend on features that are manually created, such as color, texture, and shape, followed by classifiers like SVM or KNN. Unfortunately, these methods often struggle to generalize across different symptoms and environments. On the other hand, end-to-end deep learning models can learn hierarchical, task-specific representations directly from raw images. By fine-tuning architectures that have been pre-trained on large datasets (like ImageNet), these models can adapt effectively, even with limited domain-specific data. A great example is the ICS-ResNet model, which is based on ResNet50 and includes attention modules. It achieved over 97% accuracy in detecting maize diseases, highlighting the effectiveness of combining residual learning with feature attention.

The primary objective of this comprehensive deep learning approach is to exceed the constraints of conventional, manually crafted feature pipelines, offering a more advanced and flexible framework for accurate and efficient classification of plant leaf diseases directly from unprocessed visual data.

2.2 ResNet-Based Models (ResNet50, ResNet50V2, ResNet101)

The identity skip connections in ResNet cover the gradient vanishing issue, allowing deeper layers to acquire complicated visual features. A tomato leaf classifier utilizing ResNet50 with dual-path attention (ResNet50-DPA) showed significant improvements in accuracy and localization, underscoring the network's capacity to notice intricate lesion and texture characteristics [16]. We employed a comparable methodology: scaling images to 224×224, implementing augmentation (flips, rotations, zooms, brightness/contrast adjustments), initially freezing early layers, and then fine-tuning progressively using Adam or SGD with learning rate and early stopping mechanisms. This pipeline successfully collected both complex and simple cases' disease characteristics.

- **ResNet50:** ResNet50 is a key variant in the ResNet lineup, boasting 50 layers that include 48 convolutional layers, plus 1 Max Pool layer and 1 Average Pool layer. It primarily uses a "bottleneck block" that consists of a sequence of 1×1, 3×3, and 1×1 convolutional layers. The initial 1×1 convolution is used to reduce dimensionality, the 3×3 layer is responsible for extracting key features, and the last 1×1 layer restores the original dimensionality. Thanks to its effective balance of model complexity and computational efficiency, ResNet50 has become a popular choice for many computer vision applications.

- **ResNet50V2:** ResNet50V2 is an improved version of ResNet50, specifically designed to enhance performance by adjusting the Batch Normalization and ReLU activation layers inside the residual block. In ResNetV2, Batch Normalization and ReLU are executed before the convolution (pre-activation) on both the shortcut path and the primary branches of the residual block. This change significantly boosts gradient flow, leading to more stable and efficient training for very deep networks, often resulting in better performance than ResNetV1, especially during fine-tuning or when training extremely deep models.

- **ResNet101:** ResNet101 is a more advanced version of the ResNet architecture, featuring 101 weighted layers. The primary difference between ResNet101 and ResNet50 is the larger number of stacked bottleneck blocks at each level of the network. By adding more bottleneck blocks, ResNet101 enhances its ability to learn complex and hierarchical features, which can lead to

improved accuracy on larger and more complex datasets. However, this added depth also results in a higher number of parameters and increased computational costs, requiring more training resources and potentially raising the risk of overfitting if the training data lacks sufficient diversity. Generally, ResNet101 is considered a more robust model for capturing nuanced features that ResNet50 might overlook.

2.3 DenseNet201

The interconnected blocks of DenseNet enable rapid feature reuse and ensure a strong gradient flow, which are key for detecting multiple symptom variations. In the classification of potato leaf disease, DenseNet201 achieved approximately 95.2% validation accuracy and about 92.5% test accuracy, demonstrating its skill in recognizing complex patterns [17]. We utilized this architecture with similar preprocessing and augmentation methods, replaced the classification head, and optimized it with Adam, while carefully adjusting batch sizes and dropout rates.

2.4 EfficientNetB0

EfficientNetB0 employs compound scaling to create a well-rounded balance of depth, width, and resolution, which enhances the model's efficiency and accuracy. Earlier evaluations of different leaf-disease datasets highlighted EfficientNetB0's impressive ability to handle image noise while keeping a small footprint [18]. We started with ImageNet initialization, applied thorough augmentation, and trained the model using Adam (with a learning rate of 1×10^{-5}), utilizing early stopping and learning rate adjustments. This strategy guaranteed excellent performance and made it suitable for real-time, edge-based use.

2.5 ViT (VisionTransformer)

ViT processes images sized 224×224 by dividing them into sequences of patches, leveraging multi-head self-attention to recognize relationships that are far apart in space. While we lack direct benchmarks for leaf diseases, hierarchical transformer models such as TrIncNet have proven effective in identifying plant diseases. Additionally, comparative studies involving Vision Transformers (ViT), particularly for rice leaf diseases, show improved performance when combined with extensive augmentation and AdamW optimization [19]. We implemented these strategies, expecting that ViT would perform exceptionally well in detecting signs of diseases that are prevalent across the globe.

2.6 Swin Transformer

The Swin Transformer stands out by using shifted window-based self-attention, which cleverly combines local and global contexts within a hierarchical structure. In diagnosing ligneous leaf disease, these hierarchical Swin models have been shown to be more effective than both standard ViTs and traditional CNNs, achieving an accuracy boost of 1–2% in metrics like precision, recall, and F1-score [20]. We enhanced a pre-trained Swin model by applying our usual augmentation methods and fine-tuning it with AdamW, cosine learning-rate decay, and mixed-precision training—ideal for picking up on the intricate textures of symptoms.

2.7 Training Details

2.7.1 Loss function

Selecting a suitable loss function and optimizer is essential for directing the model's learning process, particularly in multi-class classification tasks with imbalanced datasets. In this study, we utilized the Cross-Entropy loss with logits, modified for our multi-class problem including 18 distinct disease categories by treating each class prediction independently. It is characterized as:

$$L = -\frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{C} [y_{ji} \log(\sigma(z_{ji})) + (1 - y_{ji}) \log(1 - \sigma(z_{ji}))]$$

To particularly rectify the class imbalance identified in our dataset (as outlined in Section III and Figure 2), we integrated a weighting mechanism into the BCE loss. This was accomplished by allocating a pos_weight (positive class weight) to each class, thereby scaling the loss contributions from the positive instances of that class. A greater pos_weight was assigned to underrepresented (minority) courses, whereas a pos_weight of 1 (or correspondingly lower) was utilized for overrepresented (majority) classes. The pos_weight values were established according to the inverse frequency of each class inside the training set. The pos_weight for a certain class was determined by dividing the number of samples in the most prevalent class by the number of samples in that specific class. This weighting technique guarantees that the model's optimization process prioritizes the minority classes, hence avoiding excessive bias towards the majority classes and resulting in more equitable performance across all illness categories.

2.7.2 Optimizer and Learning Rate Schedule

To optimize the parameters of our deep learning models and minimize the weighted BCE loss function, we largely employed the Adam (Adaptive Moment Estimation) optimizer. Adam is a sophisticated optimization technique that employs an adaptive learning rate, recognized for its computational efficiency and strong performance across several deep learning applications. It calculates personalized adaptive learning rates for various parameters based on estimates of the first and second moments of the gradients, rendering it very efficient in addressing sparse gradients and noisy issues. The models were initialized with a learning rate of 2×10^{-5}.

To maintain consistent training and avert overfitting, we implemented an early halting mechanism. The training was scheduled to execute for a maximum of 500 epochs. The training process would cease automatically if the validation loss failed to improve for 10 consecutive epochs (patience = 10). This technique guarantees that model training ceases when generalization performance on novel data is maximized, hence preventing superfluous computations and the risk of overfitting that may arise from extended training just on the training set.

IV.   Experiment and results

Due to the considerable class imbalance evident in our dataset, as illustrated in Figure 2, depending exclusively on overall Accuracy as the principal evaluation metric may be deceptive. Accuracy is sometimes influenced by the model's efficacy on majority classes, perhaps obscuring inadequate performance on minority classes, which frequently signify infrequent but vital disease occurrences. Consequently, we prioritize the F1-score as the principal performance metric. The F1-score, as the harmonic mean of Precision and Recall, offers a more equitable assessment of the model's proficiency in accurately classifying both positive (diseased) and negative (healthy) occurrences, especially when there is an imbalanced distribution of classes. In addition to the F1-score, metrics such as Precision, Recall, and total Accuracy are also presented to provide a thorough performance evaluation.

All experiments were performed on a specialized workstation including an NVIDIA GeForce RTX 3060 GPU, which supplied the requisite processing capacity for training intricate deep learning models. The system was additionally equipped with an Intel Core i9-12900K CPU and 64GB of DDR4 RAM. The deep learning models were executed using Python 3.10.18 and the PyTorch 2.5.1 framework, and the CUDA 12.1 library for GPU acceleration. The NVIDIA driver version was 560.94 on a Windows 10 OS. This hardware and software combination enabled the rapid training and evaluation of all proposed models within an acceptable time frame.

A.   Machine learning-based

To accurately evaluate the models' performance, we employ four distinct metrics: Accuracy, Precision, Recall, and F1-score.

- Accuracy: Quantifies the ratio of successfully classified cases to the total number of occurrences.
- Precision: Quantifies the ratio of accurate positive identifications, indicating the dependability of positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

- Recall: Assesses the ratio of true positives accurately identified by the model, emphasizing its capacity to successfully detect diseases..

$$Recall = \frac{TP}{TP + FN}$$

- F1-score: Integrates precision and recall through their harmonic mean, providing a balanced metric for assessing model efficacy.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

1. *Traditional Computer Vision Approach's Results*

Model	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors	65.46	65.67	65.46	65.07
LightGBM	**75.11**	**74.53**	**75.11**	**74.67**
Logistic Regression	73.41	72.79	73.41	72.55
Random Forest	71.86	71.02	71.86	71.05
SVM	72.98	72.62	72.98	71.67

Table 1: Results from using morphological, texture, shape features(%)

Model	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors	66.34	66.17	66.34	65.71
LightGBM	**76.42**	**76.24**	**76.42**	**76.07**
Logistic Regression	72.04	72.01	72.04	71.39
Random Forest	73.62	73.09	73.62	72.8
SVM	69.6	67.52	69.6	67.84

Table 2: Results from only using Color features(%)

Model	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors	51.05	50.73	51.05	50.36
LightGBM	**54.31**	**53.34**	**54.31**	**53.41**
Logistic Regression	44.08	42.13	44.08	39.52
Random Forest	52.85	51.62	52.85	51.43
SVM	51.69	48.86	51.69	47.57

Table 3: Results from Texture features(%)

Model	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors	78.34	78.5	78.34	78.03
LightGBM	85.81	85.34	85.81	85.46

Logistic Regression	**87.24**	**86.85**	**87.24**	**86.86**
Random Forest	82.94	82.54	82.94	82.6
SVM	84.62	84.12	84.62	83.95

Table 4: Color, Texture, Morphological, Shape Features(%)

The obtained results indicate that color cues play a crucial role in diagnosing leaf diseases, consistently yielding high results across all parameters. Utilizing color exclusively, LightGBM attained an accuracy of 76.42%, markedly surpassing the results obtained by employing solely texture features. Despite concentrating solely on shape, texture, and morphological characteristics, we attained just 75% accuracy using LightGBM. The affected leaves exhibit more pronounced and diverse color variations, and our feature extraction techniques effectively leveraged this information.

Upon integrating all feature extraction approaches, we attained a maximum accuracy of 87.24% with the LR classifier, surpassing all other models across every metric. The amalgamated feature set may have increased dimensionality and enhanced the separation of the feature space, hence facilitating the determination of decision boundaries. Moreover, LightGBM has marginally inferior performance compared to LR, achieving an accuracy of 85.81% on the complete feature set. LightGBM may have successfully identified non-linear relationships among particular feature subsets, as evidenced by its substantial performance on the alternative feature set. Another significant observation is that KNN yielded the lowest metrics; nonetheless, it performed adequately on the smaller feature set. As the dimensionality of the features increases, it becomes increasingly affected by high-dimensional spaces.

The F1 score, recall, and precision are consistently aligned with accuracy, reinforcing the conclusion that the complete collection of characteristics is essential while yet preserving robustness and generalization.

2. *Deep features approach*

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	**91.61**	**91.45**	**91.61**	**91.5**
SVM	91.67	91.23	91.67	91.32
Random Forest	88.18	87.57	88.18	87.52
K-Nearest Neighbors	90.3	90.06	90.3	90.11

Table 5: DenseNet deep features results (%)

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	**90.49**	**90.11**	**90.49**	**90.24**
SVM	90.94	90.53	90.94	90.22
Random Forest	85.74	84.94	85.74	84.93
K-Nearest Neighbors	87.96	87.63	87.96	87.71

Table 6: EfficientNet deep features results (%)

Model	Accuracy	Precision	Recall	F1 Score

Logistic Regression	89.82	89.74	89.82	89.76
SVM	**90.94**	**90.4**	**90.94**	**90.4**
Random Forest	86.5	86.03	86.5	85.79
K-Nearest Neighbors	88.24	87.89	88.24	87.98

Table 7: Resnet deep features results (%)

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	**87.36**	**86.81**	**87.36**	**86.95**
SVM	87.17	85.68	87.17	86.23
Random Forest	81.64	80.67	81.64	80.41
K-Nearest Neighbors	84.98	84.46	84.98	84.57

Table 8: VGG deep features results (%)

2.1 Overall Performance of Deep Feature Architectures:

Based on the provided tables, it is evident that DenseNet and EfficientNet yielded more effective features for classification compared to ResNet50 and VGG within our dataset. Specifically, when employing features from DenseNet, the classifiers achieved the highest overall performance, with SVM leading with an Accuracy of 0.9167 and an F1 Score of 0.9132. Logistic Regression followed closely with an Accuracy of 0.9161 and an F1 Score of 0.915.

Features from EfficientNet also demonstrated competitive performance, with SVM again achieving the highest performance in this group (Accuracy 0.9094, F1 Score 0.9040). This suggests that more modern architectures like DenseNet and EfficientNet, designed for more efficient and discriminative feature extraction, outperformed the others.

Conversely, features from ResNet50 and VGG exhibited significantly lower performance compared to DenseNet and EfficientNet. VGG, in particular, provided the least effective features, with the highest F1 Score reaching only 0.8695 (Logistic Regression) and the lowest being 0.8041 (Random Forest). This disparity could stem from the deeper, more complex architectures and the diverse feature extraction capabilities of DenseNet and EfficientNet compared to VGG or even ResNet50.

2.2 Overall Performance of Deep Feature Architectures:

When we take a closer look at traditional machine learning classifiers, it's clear that Support Vector Machine (SVM) and Logistic Regression consistently outperform most deep feature sets. Specifically:

- With the help of DenseNet features, the best performers were SVM, scoring an impressive 0.9132 on the F1 Score, and Logistic Regression, which came in slightly higher at 0.915.
- Utilizing EfficientNet features, SVM (F1 Score: 0.9040) and Logistic Regression (F1 Score: 0.9024) have continued to excel and lead the pack.
- Even though features like VGG might not be the most effective, both Logistic Regression (F1 Score: 0.8695) and SVM (F1 Score: 0.8623) still showed the best performance in their respective categories.

What this tells us is that SVM and Logistic Regression are powerful algorithms, well-equipped to learn from the complex features that deep neural networks extract.

In comparison, Random Forest and K-Nearest Neighbors (KNN) tended to perform worse than SVM and Logistic Regression across all the feature sets. Random Forest, in particular, frequently had the lowest or nearly the lowest F1 Scores in each group. Although LightGBM generally outperformed Random Forest, it still didn't quite catch up to SVM and Logistic Regression. This might imply that the features we extracted aren't perfectly aligned with what tree-based or distance-based models like Random Forest and KNN need, unless we make some adjustments.

2.3 Preliminary Conclusion

The results from the experiments indicate that merging deep features from modern CNN architectures with classic machine learning classifiers is quite an effective method. Notably, DenseNet and EfficientNet excelled at pulling out powerful features for the classification task. Among the traditional classifiers, SVM and Logistic Regression were especially effective with these features, achieving the best performance across several metrics. For future research, there's potential to explore more detailed hyperparameter tuning for each model and feature combination to enhance performance even further.

B.   *Deep Learning-based*

Model	Accuracy	Precision	Recall	F1 Score
ResNet-50	92.22	96.32	94.79	92.42
ResNet-101	91.77	91.92	92.58	91.42
ResNet-50v2	92.39	92.80	90.98	91.88
Densenet-201	**95.70**	**96.32**	**94.79**	**95.55**
EfficientNet-B0	92.37	93.13	92.02	92.57
ViT	92.29	92.21	90.94	91.57
SwinTransformer	92.09	92.57	91.30	91.93

Table 9: Deep Learning-based result(%)

Among the various convolutional networks, DenseNet-201 stood out with the best performance, achieving an impressive 95.70% accuracy and a 95.55% F1-score. Its tightly connected blocks really help in reusing features and improving gradient flow, which is crucial for picking up on subtle signs of disease, like tiny lesions and texture changes across different layers. On the other hand, EfficientNet-B0, which uses a smart scaling approach for depth, width, and resolution, strikes a great balance between being compact and having enough representational power, resulting in a solid 92.37% accuracy and a 92.57% F1-score. This makes it a fantastic choice for situations where computational resources are limited. The ResNet family also performed well: ResNet-50 recorded an accuracy of 92.22% and an F1-score of 92.42%. Its upgraded version, ResNet-50V2, managed to boost accuracy slightly to 92.39% (F1 = 91.88%). However, ResNet-101 showed a bit of a dip in performance, with 91.77% accuracy and a 91.42% F1-score, suggesting that adding more layers might not always be beneficial for moderately sized datasets.

The Vision Transformer (ViT) managed to reach an accuracy of 92.29%, proving just how powerful global self-attention can be for modeling the scattered disease patterns found on leaf surfaces. Its F1-score is pretty much in line with its accuracy, indicating a solid balance between precision and recall [4]. Meanwhile, the Swin Transformer, which takes a different approach by using hierarchical, window-based attention to combine both local and global contexts, recorded an accuracy of 92.09%. This slightly lower score suggests that, especially

with shorter leaf-disease datasets, the inherent localization biases of convolutional models—or the increased data needs of transformer models—might still give them an edge in capturing those fine details.

In summary, DenseNet-201 shines because of its multi-scale feature aggregation, which leads to impressive performance. Meanwhile, EfficientNet-B0 and ResNet-50 provide a sensible trade-off between efficiency and accuracy, making them suitable for practical use. Transformer models are still competitive, but they might need bigger datasets or a combination of inductive biases to reliably surpass CNNs in scenarios that demand intricate local feature extraction.

V. CONCLUSION

Our tests, along with a multitude of recent publications, demonstrate that end-to-end deep learning models significantly surpass classical machine learning techniques (e.g., SVM, KNN) in the classification of plant leaf diseases. Convolutional neural networks usually attain accuracy exceeding 98%, but classical models sometimes falter with intricate visual patterns, typically achieving scores below 90% under comparable settings. A comparative analysis of the PlantVillage dataset showed that a CNN model significantly surpassed SVM, Decision Tree, Logistic Regression, and KNN methods, particularly in its ability to generalize across varied leaf conditions [21]. Systematic reviews encompassing more than 160 papers affirm that deep learning methodologies provide enhanced scalability, robustness, and end-to-end adaptability in comparison to conventional pipelines [22]. The improvements arise from automatic hierarchical feature learning, effective data augmentation techniques, and optimized architectures that collectively diminish generalization error in noisy, real-world data [23]. The amalgamation of improved CNN and transformer-based models, proficient in local symptom identification and global pattern recognition, enhances the efficacy, adaptability, and deployability of deep learning models compared to conventional machine learning systems for dependable plant disease detection.

This research thoroughly examined the utilization of artificial intelligence, particularly machine learning and deep learning, for the automated identification and categorization of plant leaf diseases. Our data clearly indicate the superior efficacy of end-to-end deep learning models compared to conventional machine learning methods. Traditional methods, despite extensive feature engineering, attained a maximum accuracy of 87.24% using Logistic Regression. Additionally, when integrated with deep feature extraction, these classical classifiers demonstrated enhanced performance, with accuracies of up to 91.61%, thereby underscoring the advantages of incorporating deep representations within conventional frameworks. On the other hand, modern deep learning architectures, particularly DenseNet-201, reached an exceptional accuracy of 95.70%. This notable advancement underscores deep learning's exceptional capacity to autonomously acquire complex and highly distinctive features directly from unprocessed image input.

The strong performance of models such as DenseNet-201 and EfficientNet-B0 highlights its applicability for agricultural use, particularly in resource-limited areas. By facilitating early and precise disease identification, these AI-driven solutions enable farmers to execute timely interventions, reduce crop loss, and eventually improve global food security and sustainable agricultural practices. This study advances AI in precision agriculture and offers a scalable, efficient framework for real-world application, bridging the divide between innovative research and urgent agricultural demands.

REFERENCES

[1] Y. Zhang, G. Zhou, A. Chen, M. He, J. Li, et al., "ERCP-Net: a channel extension residual structure and adaptive channel attention mechanism for plant leaf disease classification network," *Scientific Reports*, vol. 14, no. 1, pp. 1-18, 2024. doi: 10.1038/s41598-024-54287-3.

[2] A. C. P. Calma, J. D. M. Guillermo, and C. C. Paglinawan, "Cassava disease detection using MobileNetV3 algorithm through augmented stem and leaf images," in *Proc. 17th Int. Conf. Ubiquitous Information Management and Communication (IMCOM)*, Seoul, Republic of Korea, Jan. 3–5, 2023, pp. 1-6, doi: 10.1109/IMCOM56909.2023.10035648.

[5] A. Dolatabadian, T. X. Neik, M. F. Danilevicz, S. R. Upadhyaya, J. Batley, and D. Edwards, "Image-based crop disease detection using machine learning," Plant Pathology, vol. 74, no. 1, pp. 18–38, 2025.

[6] C. Jackulin and S. Murugavalli, "A comprehensive review on detection of plant disease using machine learning and deep learning approaches," Measurement: Sensors, vol. 24, p. 100441, 2022.

[7] Y. H. Maofu Liu and B. Ye, "Image Zernike moments shape feature evaluation based on image reconstruction," Geo-spatial Information Science, vol. 10, no. 3, pp. 191–195, 2007.

[8] A. Khotanzad and Y. Hong, "Invariant image recognition by Zernike moments," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 5, pp. 489–497, 1990.

[9] Y. H. Maofu Liu and B. Ye, "Image Zernike moments shape feature evaluation based on image reconstruction," Geo-spatial Information Science, vol. 10, no. 3, pp. 191–195, 2007.

[10] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, no. 6, pp. 610–621, 1973.

[11] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," The American Statistician, vol. 46, no. 3, pp. 175–185, 1992.

[12] V. N. Vapnik, The Nature of Statistical Learning Theory. Springer, 2nd ed., 2000.

[13] L. Breiman, "Random forests," Machine Learning, vol. 45, pp. 5–32, Oct 2001.

[14] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," 12 2017.

[15] C. Zhang et al., "ICS-ResNet: A Lightweight Network for Maize Leaf Disease Classification," *Agronomy*, vol. 14, no. 7, 2024.

[16] L. Liang and J. Jiang, "A ResNet50-DPA Model for Tomato Leaf Disease Identification," *Frontiers in Plant Science*, 2023.

[17] R. A. Charisma and F. D. Adhinata, "Transfer Learning with DenseNet201 for Potato Leaf Disease Classification," *arXiv*, Jan. 2024.

[18] W. Smith et al., "Plant Leaf Disease Detection Using GreenViT on Multiple Datasets," *Sensors*, 2024.

[19] S. Mehnaz and M. T. Islam, "Rice Leaf Disease Detection: A Comparative Study Between CNN, Transformer and Non-neural Network Architectures," *arXiv*, Jan. 2025.

[20] "Automatic Classification of Ligneous Leaf Diseases via Hierarchical Vision Transformer and Transfer Learning," *Frontiers in Plant Science*, 2023.

[21] T. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.*, vol. 145, pp. 311–318, 2018, doi:10.1016/j.compag.2018.01.009.

[22] S. U. Nagaveni et al., "A systematic review of deep learning techniques for plant disease detection," *Artif. Intell. Agricult.*, published Jun. 2024.

[23] R. Sujatha et al., "Advancing plant leaf disease detection integrating machine learning and deep learning," *Sci. Rep.*, vol. 15, Art. No. 11552, 2025