Journal Software, Hardware and Information Technology
p-ISSN: 2808-3385 | e-ISSN: 2776-8961
Volume 6 Issue 2, June 2026

# Hybrid Deep Feature Extraction and Machine Learning for Potato Disease Classification

[1] Bimantyoso Hamdikatama, [2] Kusrini, [2] Arif Setyanto, [3] Julius Jeremiah Allieu
[1] Faculty of Communication and Informatics, Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
[2] Magister of Informatics, Universitas AMIKOM Yogyakarta, Yogyakarta, Indonesia
[3] Department of Mathematics and Computing Science, Punjabi University, Patiala, India
bh972@ums.ac.id, kusrini@amikom.ac.id, arief_s@amikom.ac.id, juliusjeremiahallieu@gmail.com

*Notes*

*Submitted: 08-05-2026*
*Revised: 28-05-2026*
*Accepted: 29-05-2026*
*Published: 30-06-2026*

**How to cite:** B. Hamdikatama, Kusrini, A. Setyanto, and J. J. Allieu, "Hybrid Deep Feature Extraction and Machine Learning for Potato Disease Classification," *Journal Software, Hardware and Information Technology (SHIFT)*, vol. 6, no. 2, pp. 129–149, Jun. 2026, doi: 10.24252/shift.v6i2.285.

**DOI**: 10.24252/shift.v6i2.285

Copyright 2026© the Author(s)

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

**ABSTRACT**

Accurate classification of potato leaf diseases is crucial for improving agricultural productivity and reducing crop losses. However, existing end-to-end convolutional neural network models often require high computational cost, long training time, and large labeled datasets, limiting their implementation in resource-constrained agricultural environments. Therefore, this study proposes a hybrid classification framework that combines EfficientNetB3 as a deep feature extractor with classical machine learning classifiers, namely Support Vector Machine (SVM), Random Forest (RF), and XGBoost, for potato leaf disease classification. Experiments were conducted using the PlantVillage potato leaf dataset, which exhibits class imbalance, particularly in the Healthy class. Images were preprocessed using augmentation techniques, while extracted deep features were normalized, reduced using Principal Component Analysis (PCA), and analyzed using K-Means Clustering before classification. The models were evaluated using accuracy, precision, recall, F1-score, and training time. Experimental results show that EfficientNetB3 + XGBoost achieved the highest accuracy of 96.7% with reduced training time, while EfficientNetB3 + SVM provided competitive performance with the fastest training process. These findings demonstrate that integrating deep feature extraction with machine learning classifiers provides an effective and computationally efficient approach for potato leaf disease classification in precision agriculture.

**Keywords:** EfficientNetB3, random forest, SVM, XGBoost.

## 1. INTRODUCTION

Potato (*Solanum tuberosum*) is one of the most important food crops worldwide and plays a significant role in food security and agricultural sustainability [1]. However, potato production is highly susceptible to various leaf diseases such as Early Blight and Late Blight, which can cause substantial yield loss if not detected and controlled at an early stage [2][3]. Traditional disease identification methods rely heavily on visual inspection by experts, which is time-consuming, subjective, and impractical for large-scale agricultural fields [4].

In recent years, computer vision and machine learning techniques have been widely applied to plant disease detection [5]. Conventional machine learning approaches typically require handcrafted features, which are often limited in capturing complex visual patterns of plant diseases. To overcome this limitation, deep learning-based convolutional neural networks (CNNs) have demonstrated superior performance in extracting discriminative features directly from raw images [6].

Among various CNN architectures, EfficientNet has gained considerable attention due to its balanced scaling of network depth, width, and resolution. Several studies have successfully employed EfficientNet variants for plant disease classification, achieving high accuracy on benchmark datasets such as PlantVillage [7]. Despite their strong performance, most existing approaches adopt an end-to-end deep learning framework, where CNNs are used simultaneously for feature extraction and classification [8].

However, end-to-end deep learning models present several challenges, particularly in agricultural applications with limited datasets and computational resources. These models often require extensive training, large labeled datasets, and high computational power, which may not be feasible in real-world farming environments [9]. Moreover, the decision-making process of deep neural networks is often considered a *black box*, making it difficult to interpret and analyze the classification results [10].

To address these limitations, hybrid learning approaches that combine deep feature extraction with classical machine learning classifiers have emerged as a promising alternative [11]. In this paradigm, deep learning models are utilized solely as feature extractors, while the classification task is handled by machine learning algorithms such as Support Vector Machine (SVM), Random Forest (RF), and XGBoost [12][13]. This hybrid strategy offers several advantages, including improved interpretability, reduced computational complexity, and enhanced flexibility for small to medium-sized datasets.

Therefore, this study proposes a hybrid deep feature learning framework for potato leaf disease classification, where EfficientNetB3 is employed as a deep feature extractor and multiple machine learning classifiers are used for disease classification. The proposed approach aims to achieve high classification performance while maintaining model efficiency and interpretability, making it suitable for practical plant protection applications [14].

Based on the literature review, numerous studies have applied both conventional machine learning and deep learning approaches for plant and potato leaf disease classification. In recent years, CNN-based architectures such as VGG, ResNet, and EfficientNet have demonstrated strong performance in plant disease detection due to their ability to automatically learn discriminative visual features [15][16]. Most existing studies employ end-to-end deep learning frameworks, where CNN models simultaneously perform feature extraction and classification, while conventional machine learning approaches generally rely on handcrafted feature extraction and image segmentation techniques.

The first study [17], *"Potato Plant Leaves Disease Detection and Classification using Machine Learning Methodologies"*, employed traditional image processing techniques combined with machine learning classifiers. The methodology utilized K-means clustering for image segmentation, Gray Level Co-occurrence Matrix (GLCM) for feature extraction, and a multi-class Support Vector Machine for classification, achieving an accuracy of 95.99%. Although the reported performance is relatively high, the approach heavily depends on handcrafted features and manual segmentation, which are sensitive to lighting conditions, background noise, and

variations in leaf appearance. In addition, GLCM features have limited capability in capturing complex semantic disease patterns, reducing the model's robustness and scalability in real-world agricultural applications.

Similarly, the second study [18], *"Potato Leaf Disease Classification using Image Processing and Artificial Neural Network"*, utilized K-means clustering and GLCM-based features as input to an Artificial Neural Network (ANN), achieving an accuracy of 94%. Despite its simplicity, the method suffers from similar limitations associated with traditional feature engineering approaches, including limited generalization capability and dependency on handcrafted descriptors. Furthermore, the shallow ANN architecture lacks sufficient depth to automatically learn discriminative visual representations from complex disease patterns.

In contrast, the third study [19], *"Potato Plant Disease Classification Through Deep Learning"*, applied an end-to-end Convolutional Neural Network (CNN) model and reported an accuracy of 99.84%. While this result demonstrates the strong representational capability of deep learning models, the study does not sufficiently address computational cost, training efficiency, and deployment challenges. End-to-end CNN models typically require extensive computational resources and long training durations, limiting their practical implementation in resource-constrained agricultural environments. Moreover, such models often behave as black-box systems, making feature interpretation and integration with alternative classifiers more difficult.

Despite these promising results, several limitations remain unresolved. Traditional machine learning methods still depend heavily on handcrafted feature engineering, whereas end-to-end deep learning approaches often require high computational resources and large-scale datasets. In addition, although hybrid approaches have been explored in potato disease classification, comparative evaluation of multiple classical machine learning classifiers using identical EfficientNetB3 feature representations remains limited. Recent studies such as Peyal and Mondal [15] indicate that CNN-SVM hybrid models are promising; however, systematic comparisons involving Random Forest and XGBoost using the same deep feature representation are still underexplored.

To address these limitations, this study proposes a hybrid framework that integrates EfficientNetB3 as a deep feature extractor with classical machine learning classifiers, including SVM, Random Forest, and XGBoost. The proposed approach eliminates the need for manual feature extraction and segmentation by leveraging high-level representations learned from pre-trained CNNs. At the same time, replacing the Softmax classifier with classical machine learning models improves computational efficiency while maintaining strong classification performance. Experimental results demonstrate that the proposed hybrid models achieve competitive or superior accuracy with significantly reduced training time, providing a more balanced trade-off between predictive performance and computational efficiency.

Overall, this study addresses an important research gap by systematically combining deep learning-based feature extraction with classical machine learning classification. The proposed framework is particularly relevant for real-world agricultural applications, where both classification accuracy and computational efficiency are essential for timely and reliable potato disease diagnosis.

The main contributions of this study are summarized as follows:

1. This study proposes a hybrid classification framework that combines EfficientNetB3 as a deep feature extractor with classical machine learning classifiers for potato leaf disease classification.

2. A comprehensive comparison of SVM, Random Forest, and XGBoost classifiers using deep features is conducted to evaluate their effectiveness.
3. The proposed hybrid approach is compared with an end-to-end EfficientNetB3 model to demonstrate its efficiency and performance advantages.
4. The results provide insights into the suitability of hybrid deep learning models for plant protection applications with limited datasets.

## 2. METHOD

### 2.1. Data Collection

This study utilizes the PlantVillage Potato Disease Dataset, a publicly available benchmark dataset widely used in plant disease classification research [20]. The dataset is a comprehensive collection of RGB images representing various conditions affecting potato crops and is organized into three distinct classes: Potato Early Blight, Potato Late Blight, and Healthy Potato Leaves. Each class corresponds to a specific physiological or pathological condition of the potato plant, enabling a systematic evaluation of disease classification models [21].

Potato Early Blight is caused by the fungal pathogen *Alternaria solani*, which affects potatoes, tomatoes, and other members of the Solanaceae family. This disease, also known as target spot, typically appears on older leaves and rarely affects young, vigorously growing plants. The visual symptoms include concentric lesions and discoloration, which are critical indicators for image-based disease detection [22].

Potato Late Blight, caused by *Phytophthora infestans*, is considered the most devastating disease affecting potato crops worldwide. It can lead to severe crop losses and rapid field destruction if appropriate control measures are not implemented in a timely manner. The disease manifests through irregular dark lesions and leaf tissue decay, making accurate and early detection essential for effective plant protection strategies [23].

The Healthy Potato Leaf class represents leaves without visible disease symptoms and serves as a baseline for normal plant conditions. Healthy potato leaves are nutritionally significant, as potatoes are a rich source of vitamin C and potassium, which play important roles in human health. From a classification perspective, this class provides a reference for distinguishing diseased leaves from healthy ones [24].

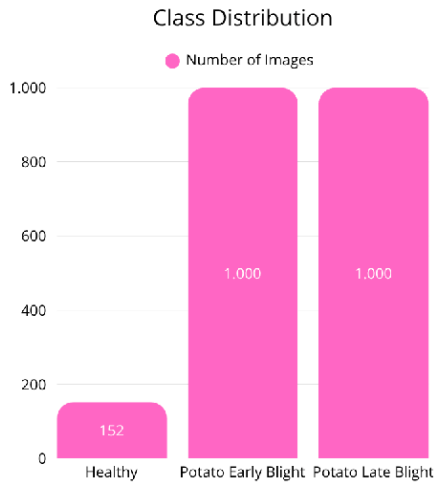



**Figure 1.** Class Distribution

Figure 1 illustrates the class distribution of the PlantVillage potato leaf disease dataset. The x-axis represents the three classes, namely Potato Early Blight, Potato Late Blight, and Healthy Potato Leaf, while the y-axis shows the number of images in each class. The figure indicates that the Early Blight and Late Blight classes contain substantially more images than the Healthy class, showing that the dataset is imbalanced. This imbalance motivates the use of class weighting during model training. The dataset contains a total of 2,152 images, distributed across the three classes. Specifically, the Potato Early Blight class consists of 1,000 images, the Potato Late Blight class also contains 1,000 images, and the Healthy Potato Leaf class includes 152 images. The dataset is organized into three corresponding folders, each representing one disease category [25].

All images in the PlantVillage dataset are captured under controlled environmental conditions, with consistent backgrounds and lighting. While such conditions may not fully reflect real-field environments, they provide high-quality and standardized images that are well-suited for evaluating the discriminative capability of deep feature extraction and machine learning-based classification models.

To ensure a fair and unbiased evaluation, the dataset is divided into training and validation subsets using an 80:20 ratio. This data partitioning strategy allows the proposed hybrid framework to be trained effectively while enabling an objective assessment of its generalization performance on unseen data.

**Table 1.** Distribution of Potato Leaf Disease Dataset

| Class | Number of Images |
|---|---|
| Healthy | 152 |
| Potato Early Blight | 1000 |
| Potato Late Blight | 1000 |
| Total | 2152 |

Table 1 summarizes the number of images in each potato leaf disease class. The Early Blight and Late Blight classes each contain 1,000 images, while the Healthy class contains only 152 images. This distribution indicates that the dataset is imbalanced, with the Healthy class being significantly underrepresented. Therefore, class weighting is required during training to reduce the risk of model bias toward the majority classes.

### 2.2. Data Preprocessing

Data preprocessing is a crucial step to enhance model generalization and improve classification performance [26]. In this study, image preprocessing and augmentation are performed using the ImageDataGenerator module from the TensorFlow Keras library [27]. This approach enables efficient loading, preprocessing, and augmentation of image data during model training and validation [28].

All images are resized to 300 × 300 pixels, which corresponds to the input size required by EfficientNetB3. Prior to feature extraction, images are normalized using the EfficientNet-specific preprocessing function, which scales pixel values and applies normalization consistent with the ImageNet pretrained weights. This preprocessing step ensures compatibility with the pretrained EfficientNetB3 model and facilitates effective transfer learning [29][30].

To increase data diversity and reduce overfitting, several data augmentation techniques are applied to the training set. These techniques include horizontal flipping, random rotation,

zooming, shearing, and width and height shifting. Such augmentations simulate variations in leaf orientation, scale, and position that may occur in real-world scenarios, thereby improving the robustness of the extracted deep features [31].

A validation split of 20% is applied to the dataset to separate training and validation data. The training subset is augmented to enhance model learning, while the validation subset is processed without augmentation to provide an unbiased evaluation of model performance. Both subsets use the same preprocessing function to ensure consistency in input representation [32].

The dataset is loaded from a directory structure where each subfolder represents a specific class. Using this approach, a total of 1,722 images are allocated to the training subset, covering all three potato leaf condition classes. The remaining images are reserved for validation purposes.

Overall, the applied preprocessing and augmentation strategy ensures that the input data are standardized, diverse, and well-suited for deep feature extraction using EfficientNetB3. This preprocessing strategy ensures that the extracted deep features are robust and informative, providing a reliable input representation for subsequent machine learning classifiers.

Table 2 presents the data preparation scheme used in this study. The dataset is divided using an 80:20 split, resulting in 1,722 images for training and 430 images for validation. The training subset is used to learn feature representations and train the classifiers, while the validation subset is used to evaluate model performance on unseen data.

**Table 2.** Data Preparation

| Data Split | Number of Images |
|---|---|
| Training Dataset | 1722 |
| Validation Dataset | 430 |

Following image preprocessing and augmentation, feature vectors were extracted from the 1,024-neuron Dense layer with ReLU activation after CNN training and before the final Softmax layer. Thus, each potato leaf image was represented by a 1,024-dimensional task-specific feature vector. These vectors were then normalized and used as the input to PCA and the downstream SVM, Random Forest, and XGBoost classifiers. This single feature source was used consistently for all hybrid models and all reported results.

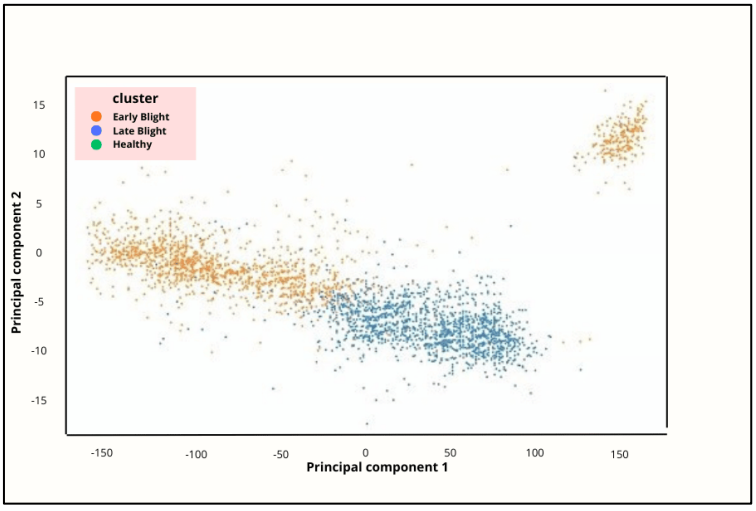



**Figure 2.** Principal component analysis (PCA) of tested dataset

To address the high dimensionality and potential redundancy of the 1,024-dimensional Dense feature vectors, Principal Component Analysis (PCA) was applied as a dimensionality reduction technique [33]. PCA was fitted only on the training feature matrix and the fitted transformation was then applied to the validation feature matrix to avoid information leakage[34]. In the final experiment, PCA was configured to retain 95% of the cumulative explained variance (n_components=0.95), resulting in 128 principal components. This reduced feature representation was used as the common input for SVM, Random Forest, and XGBoost. Figure 2 illustrates the PCA projection of the deep features extracted from the Dense-1024 representation into a lower-dimensional space. Each point represents an image sample, while the position of the point reflects the similarity of its extracted feature representation. Samples that appear close to each other indicate similar visual characteristics, whereas samples that are far apart indicate more distinct feature patterns. The separation among groups in the PCA space suggests that the EfficientNetB3-based feature extractor is able to produce discriminative features for distinguishing potato leaf disease classes [35].

After dimensionality reduction, K-means clustering is employed to further explore the intrinsic structure of the deep feature space [36]. The K-means algorithm groups feature vectors based on similarity, such that samples within the same cluster share similar feature representations, while samples in different clusters are more distinct [37]. This clustering process helps reveal latent patterns within the data that may correspond to different disease conditions. Figure 3 shows the K-means clustering result of the reduced deep feature vectors in the PCA space. Each point represents a potato leaf image, and each cluster indicates a group of samples with similar feature characteristics. The separation of clusters demonstrates that the extracted EfficientNetB3 features contain meaningful patterns that can support disease classification. This visualization also helps explain how the feature representations are grouped before being classified by the machine learning models.

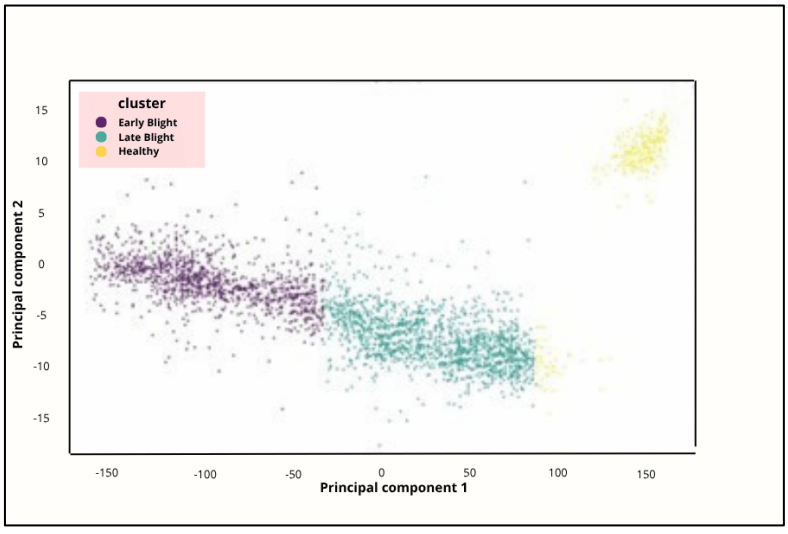



**Figure 3.** Cluster analysis (K-means) of tested dataset

Overall, the combined image-level preprocessing, data augmentation, feature normalization, dimensionality reduction, and clustering strategy ensures that the machine learning classifiers are trained on high-quality, compact, and well-structured deep feature representations[38]. This comprehensive preprocessing pipeline reduces noise, mitigates bias, enhances robustness, and ultimately improves the predictive performance of the proposed hybrid deep feature learning and machine learning framework.

### 2.3. Data Visualization

This study employs the PlantVillage Potato Disease Dataset, a publicly available benchmark dataset widely used in plant disease classification research. The dataset consists of RGB images of potato leaves categorized into three distinct classes: Early Blight, Late Blight, and Healthy potato leaves. Each class represents a specific physiological condition of the potato plant and reflects different visual disease characteristics that are critical for image-based classification.

To provide a clearer understanding of the visual differences among the disease categories, Figure 4 presents representative sample images from the PlantVillage potato disease dataset. The figure displays examples of Healthy, Early Blight, and Late Blight potato leaves. These samples show visible differences among classes, such as lesion shape, color variation, texture irregularities, and overall leaf appearance. The visual differences shown in Figure 4 justify the use of deep feature extraction, because EfficientNetB3 can capture complex disease-related visual patterns that may be difficult to describe using handcrafted features.

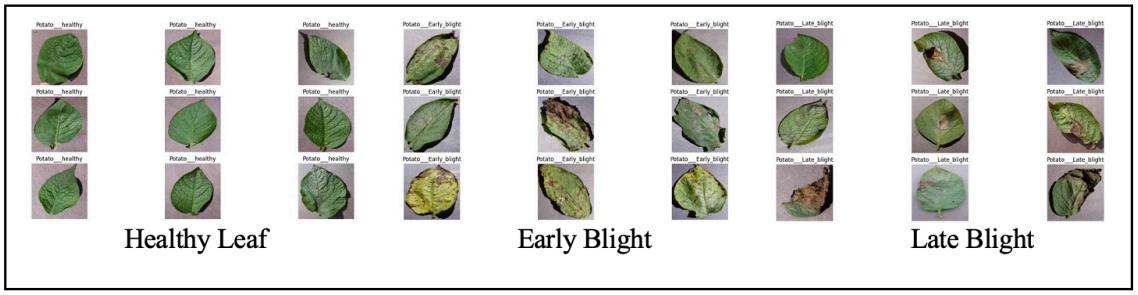



**Figure 4.** Sample images from the PlantVillage potato disease dataset

The inclusion of these representative samples supports the reliability of the dataset for evaluating image-based classification approaches and demonstrates the visual complexity that the proposed hybrid deep feature learning and machine learning framework aims to address.

### 2.4. Feature Extraction

In this study, EfficientNetB3 was used as the convolutional backbone within a fixed feature-extraction pipeline. To remove ambiguity in the hybrid stage, the feature vector used by PCA and the classical classifiers was explicitly taken from the 1,024-neuron Dense layer with ReLU activation after CNN training and before the final Softmax layer. Therefore, the GAP output and the Softmax output were not used as direct inputs to SVM, Random Forest, or XGBoost[39].

The feature extraction process is formulated in Equation (1). This equation explains how each input image is transformed into a deep feature vector using EfficientNetB3 [40].

$$\mathbf{z}_i = f(x_i), \mathbf{z}_i \epsilon\, \tau^d \tag{1}$$

In Equation (1), $x_i$ represents the $i$ input image, $f(\cdot)$ denotes the EfficientNetB3 feature extraction function, $z_i$ is the extracted deep feature vector, and d represents the dimensionality of the feature space. Each potato leaf image is therefore passed through the EfficientNetB3 backbone, GAP, and the 1,024-neuron Dense layer to produce z. The vector z is then normalized, reduced using PCA, and used as input for the downstream machine learning classifiers[41].

EfficientNetB3 is selected as the deep feature extraction backbone due to its favorable balance between representational power and computational efficiency. Unlike conventional convolutional neural networks that scale network depth, width, or input resolution

independently, EfficientNet employs a compound scaling strategy, which uniformly scales these dimensions to achieve improved performance with fewer parameters. This characteristic makes EfficientNet particularly suitable for applications involving limited datasets, such as plant disease classification[42].

### 2.5. Algorithm Application

To perform disease classification based on the extracted deep features, three classical machine learning classifiers are employed: Support Vector Machine (SVM), Random Forest (RF), and XGBoost. These classifiers are selected due to their proven effectiveness in handling high-dimensional feature spaces and their interpretability advantages.

Support Vector Machine aims to find an optimal hyperplane that maximally separates feature vectors of different classes [43]. The decision function of SVM is expressed in Equation (2). This equation shows how SVM determines the class of an input feature vector by calculating its position relative to the learned decision boundary.

$$f(z) = sign\left(\sum_{i=1}^{n} \alpha_i y_i K(z_i, z) + b\right) \quad (2)$$

In Equation (2), z is the input feature vector, $z_i$ represents the training feature vector, $y_i$ is the class label, $a_i$ denotes the Lagrange multiplier, $K(z_i, z)$ is the RBF kernel function, and $b$ is the bias term. The equation should be read as a weighted similarity calculation between the input feature vector and the support vectors. Since this study involves three classes, the SVM classifier is implemented using a multiclass strategy, such as one-vs-rest or one-vs-one classification [44].

Support Vector Machine is well-suited for classifying deep feature representations due to its effectiveness in high-dimensional feature spaces. By maximizing the margin between classes, SVM achieves strong generalization performance, particularly when the number of training samples is limited. This margin-based learning strategy also makes SVM relatively robust to overfitting, especially when combined with kernel functions such as the Radial Basis Function (RBF), which enable the modeling of non-linear decision boundaries. In addition, SVM is supported by a well-established theoretical foundation, providing strong guarantees regarding its generalization behavior [45].

However, SVM also has notable limitations. The computational cost of training can become significant as the size of the dataset increases, making it less scalable for large-scale applications. Moreover, SVM performance is highly sensitive to hyperparameter selection, including the regularization parameter and kernel-specific parameters, which require careful tuning. Compared to tree-based models, SVM also offers limited interpretability, as the resulting decision function is not easily translated into intuitive rules [46].

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the class prediction based on majority voting. Each tree is trained on a randomly sampled subset of features and data samples, which enhances generalization and reduces overfitting [47].

The prediction mechanism of Random Forest is presented in Equation (3). This equation explains how the final class label is obtained through majority voting among multiple decision trees.

$$\hat{y} = mode\{h_1(z), h_2(z), \ldots h_t(z)\} \quad (3)$$

In Equation (3), $\hat{y}$ represents the final predicted class, $h_t(z)$ denotes the prediction produced by the $t$-th decision tree, and $t$ is the total number of trees in the forest. The equation should be read as follows: each decision tree predicts a class label for the input feature vector, and the final prediction is determined by the class that receives the most votes[48].

Random Forest is an ensemble learning method that provides strong robustness against noise and overfitting through the aggregation of multiple decision trees. By training each tree on a random subset of samples and features, Random Forest is capable of capturing complex interactions among deep features without relying on strict assumptions about data distribution. One of its key strengths lies in its relatively straightforward hyperparameter tuning compared to boosting-based methods. Additionally, Random Forest offers inherent measures of feature importance, which enhance interpretability and provide insights into the contribution of different features to the classification process [49][50].

Nevertheless, Random Forest has certain drawbacks. While it performs well in many scenarios, it may be less effective in modeling highly complex decision boundaries compared to boosting-based algorithms such as XGBoost. The ensemble nature of the model also results in larger memory consumption, which may be a concern in resource-constrained environments. Furthermore, although Random Forest is more interpretable than some black-box models, its interpretability remains limited compared to that of a single decision tree [51].

XGBoost is a gradient boosting framework that builds an ensemble of decision trees sequentially by minimizing a regularized objective function. It has demonstrated strong performance in various classification tasks due to its robustness and efficiency[52].

The objective function of XGBoost is defined in Equation (4). This equation describes how XGBoost optimizes classification performance by minimizing prediction error while controlling model complexity.

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{k} \Omega(f_k) \quad (4)$$

In Equation (4), $L$ denotes the overall objective function, $l(y_i, \hat{y}_i)$ represents the loss between the true label $y_i$ and the predicted label $\hat{y}_i$ , and $\Omega(f_k)$ is the regularization term for the $k$-th tree. The first term measures classification error, while the second term penalizes overly complex models. Therefore, XGBoost aims to achieve high predictive accuracy while maintaining good generalization [53].

XGBoost is a powerful gradient boosting framework that achieves high predictive accuracy by sequentially learning from previous classification errors. This iterative learning process allows XGBoost to model complex non-linear relationships within high-dimensional deep feature spaces effectively. The inclusion of regularization terms in its objective function helps control model complexity and improve generalization performance. In addition, XGBoost is designed to be computationally efficient and scalable, making it suitable for handling large feature sets and complex classification tasks [54].

Despite its strong performance, XGBoost also presents several challenges. The model is highly sensitive to hyperparameter tuning, requiring careful optimization to achieve optimal performance. Training time can be longer compared to Random Forest due to its sequential learning nature. Moreover, the increased model complexity introduced by boosting can reduce interpretability, making it more difficult to explain classification decisions in comparison to simpler ensemble methods [55].

The combination of EfficientNetB3 and classical machine learning classifiers leverages the complementary strengths of deep learning and traditional machine learning. EfficientNetB3 excels at learning rich visual representations, while SVM, Random Forest, and XGBoost provide flexible, interpretable, and robust decision-making mechanisms. This hybrid strategy is particularly well-suited for small-to-medium-sized datasets, where end-to-end deep learning models may suffer from overfitting or limited generalization.

### 2.6. Model Training and Hyperparameter Tuning

Figure 5 presents the architecture of the proposed hybrid deep feature learning framework. The workflow begins with potato leaf image input, followed by preprocessing and augmentation to standardize image size and increase data variation. The preprocessed images are then passed through the EfficientNetB3 backbone, GAP layer, and 1,024-neuron Dense layer to extract the final fixed feature vector. These Dense-1024 features are normalized and reduced using PCA before being classified using SVM, Random Forest, and XGBoost. The figure should be read from top to bottom, showing the sequential process from image input to final disease classification output.

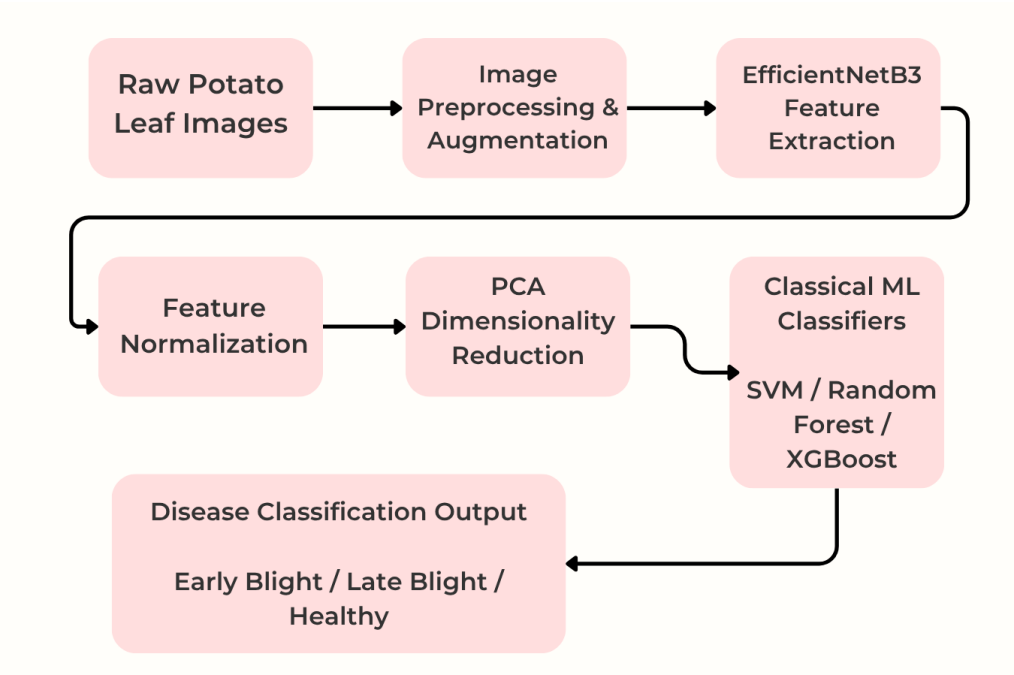



**Figure 5.** Architecture of the proposed hybrid deep feature learning

Before model training, class distribution was analyzed using the training data generator to identify potential class imbalance. The class indices and corresponding labels were obtained from the training generator, ensuring consistent mapping between class names and numerical labels. The Healthy class is substantially underrepresented compared with the Early Blight and Late Blight classes, therefore, imbalance handling was considered during both training and evaluation. Class weights were applied during the EfficientNetB3 Softmax training stage to reduce bias toward the majority classes. For the hybrid classifiers, SVM and Random Forest were trained using balanced class weights, while XGBoost used sample weights derived from the class distribution. The exported Dense-1024 feature vectors were not reweighted during PCA, imbalance handling was applied at the classifier training stage and evaluated using class-wise precision, recall, F1-score, and the confusion matrix.

**Table 3.** Hyperparameter Settings of the Classical Machine Learning Classifiers

| Classifier | Hyperparameter | Value used in this study |
|---|---|---|
| SVM | kernel; C; gamma; multiclass strategy; class_weight | RBF; C=1.0; gamma=scale; multiclass strategy=one-vs-one; class_weight=balanced |
| Random Forest | n_estimators; max_depth; min_samples_split; min_samples_leaf; class_weight | n_estimators=100; max_depth=None; min_samples_split=2; min_samples_leaf=1; class_weight=balanced |
| XGBoost | n_estimators; learning_rate; max_depth; subsample; colsample_bytree; objective; eval_metric; sample_weight | n_estimators=100; learning_rate=0.1; max_depth=6; subsample=1.0; colsample_bytree=1.0; objective=multi:softprob; eval_metric=mlogloss; sample_weight=yes |

The hyperparameter settings of the downstream classifiers are summarized in Table 3. These settings are reported because classifier performance and fairness on the minority Healthy class can be sensitive to kernel, regularization, tree depth, and imbalance-handling parameters. All classical machine learning classifiers were trained using the same PCA-reduced Dense-1024 feature representation to ensure a fair comparison. No additional feature engineering was applied separately to any classifier. The SVM and Random Forest models used balanced class weighting, whereas XGBoost used sample weights derived from the class distribution to reduce bias toward the minority Healthy class. The hyperparameters in Table 3 represent the final configuration used to generate the validation results reported in the Results section all reported result.

The EfficientNetB3 model pretrained on the ImageNet dataset is adopted as the base network, with the classification head removed (include_top=False). This configuration allows the network to act as a deep feature extractor, leveraging pretrained convolutional layers that capture general visual patterns such as edges, textures, and color distributions. The input image size is set to 300 × 300 pixels, consistent with the EfficientNetB3 architecture.

To adapt the pretrained backbone to the potato leaf disease classification task, several custom layers are added on top of the base model. A Global Average Pooling (GAP) layer is used to reduce the spatial dimensions of the feature maps while preserving the most salient global information. Compared to fully connected layers, GAP reduces the number of trainable parameters and helps minimize overfitting. This is followed by a Dropout layer with a rate of 0.5, which randomly deactivates neurons during training to enhance regularization.

A dense layer with 1,024 neurons and ReLU activation is then introduced to learn task-specific feature combinations. Another dropout layer with the same rate is applied to further prevent co-adaptation of neurons. Finally, a softmax output layer with three neurons is used to produce class probabilities corresponding to Early Blight, Late Blight, and Healthy potato leaves.

To preserve the general visual knowledge learned from ImageNet and avoid overfitting on a relatively small dataset, all layers of the EfficientNetB3 backbone are frozen during training. This ensures that only the parameters of the newly added classification layers are updated. Such

a strategy is particularly effective when the dataset size is limited, as it reduces training complexity while maintaining strong feature extraction capability.

The model is compiled using the Adam optimizer with a learning rate of 0.0001, which provides an effective balance between fast convergence and training stability. Adam is chosen due to its adaptive learning rate mechanism, which adjusts parameter updates based on first and second-order moment estimates. The categorical cross-entropy loss function is employed, as the classification task involves multiple mutually exclusive classes. Model performance is evaluated using classification accuracy.

To further improve training stability and prevent overfitting, several callback mechanisms are incorporated. Early stopping is applied by monitoring the validation loss, with a patience of five epochs. Training is automatically halted if no improvement is observed, and the model weights corresponding to the best validation performance are restored. This prevents unnecessary training and reduces overfitting.

A model checkpoint strategy is used to save the best-performing model based on validation loss, ensuring that the optimal model configuration is retained for subsequent evaluation and feature extraction. Additionally, a learning rate reduction on plateau mechanism is employed, where the learning rate is reduced by a factor of 0.2 if the validation loss does not improve for 15 consecutive epochs. This adaptive adjustment allows the optimizer to fine-tune model parameters during later stages of training, while a minimum learning rate threshold is enforced to maintain numerical stability.

The model is trained for a maximum of 50 epochs, with the number of training and validation steps determined by the batch size and dataset size. Class weights are applied during training to handle class imbalance, and both training and validation generators ensure consistent preprocessing. The combination of early stopping, learning rate scheduling, dropout regularization, and class weighting results in a robust training process that produces discriminative deep features suitable for hybrid classification.

## 3. RESULT AND DISCUSSION

This section presents the experimental results and discussion of the proposed hybrid classification framework for potato leaf disease detection. In accordance with the methodology described in Section 2, the EfficientNetB3-based network is used to generate fixed Dense-1024 feature vectors, while classification is performed using traditional machine learning algorithms, namely Support Vector Machine (SVM), Random Forest (RF), and XGBoost.

The dataset is divided into training and validation subsets using an 80:20 split. Feature extraction is conducted after image preprocessing and augmentation, and the resulting Dense-1024 feature vectors are normalized and reduced using Principal Component Analysis (PCA) before being fed into the classifiers. To avoid data leakage, PCA is fitted on the training feature matrix only and then applied to the validation feature matrix. All classifiers are trained and tested using identical feature representations.

Model performance is evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score. Because the dataset is imbalanced and Healthy class is the minority class, both aggregate metrics and class-wise metrics are reported. The confusion matrix is also included to show whether the models correctly recognize Healthy leaves rather than relying mainly on the majority disease classes.

### 3.1. Results

This section presents the experimental results of the proposed hybrid deep feature learning framework for potato leaf disease classification. The performance of three classical machine learning classifiers, namely Support Vector Machine (SVM), Random Forest (RF), and XGBoost, is evaluated using fixed Dense-1024 features extracted from the best-performing EfficientNetB3 checkpoint.

Table 4 summarizes the aggregate classification performance of each model in terms of accuracy, precision, recall, and F1-score. The table shows that EfficientNetB3 + XGBoost achieves the highest aggregate performance in this validation split, with an accuracy of 96.7%, precision of 96.7%, recall of 96.7%, and F1-score of 96.7%. EfficientNetB3 + SVM also achieves competitive performance, while EfficientNetB3 + RF produces slightly lower aggregate results. Because the numerical differences between some models are small, the results are interpreted cautiously and supported by class-wise evaluation.

**Table 4.** Aggregate Performance Comparison

| Classifier | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| EfficientNetB3 + Softmax | 93.9 | 93.8 | 93.6 | 93.7 |
| EfficientNetB3 + SVM | 96.0 | 96.2 | 95.8 | 96.0 |
| EfficientNetB3 + RF | 93.7 | 93.5 | 93.2 | 93.3 |
| EfficientNetB3 + XGBoost | 96.7 | 96.7 | 96.7 | 96.7 |

To further validate the effectiveness of the proposed hybrid framework under class imbalance, per-class precision, recall, and F1-score are reported in Table 5. This analysis is necessary because high aggregate accuracy can mask weak recognition of the minority Healthy class. Therefore, the Healthy class is discussed explicitly rather than relying only on overall metrics.

**Table 5.** Class-wise Performance of EfficientNetB3 + XGBoost

| Class | Precision (%) | Recall (%) | F1-score (%) | Support (images) |
|---|---|---|---|---|
| Healthy | 96.7 | 96.7 | 96.7 | 30 |
| Early Blight | 96.5 | 97.0 | 96.8 | 200 |
| Late Blight | 97.0 | 96.5 | 96.7 | 200 |
| Macro Average | 96.7 | 96.7 | 96.7 | 430 |

**Table 6.** Confusion Matrix of EfficientNetB3 + XGBoost

| Actual/ Predicted | Healthy | Early Blight | Late Blight |
|---|---|---|---|
| Healthy | 29 | 1 | 0 |
| Early Blight | 0 | 194 | 6 |
| Late Blight | 1 | 6 | 193 |

Table 6 shows the corresponding confusion matrix. The model correctly classifies 29 of 30 Healthy images, 194 of 200 Early Blight images, and 193 of 200 Late Blight images. Only one Healthy image is misclassified as Early Blight, while no Healthy image is misclassified as Late Blight. Most remaining errors occur between Early Blight and Late Blight, which indicates that the main confusion is between visually similar disease symptoms rather than failure to recognize the minority Healthy class.

The classical machine learning classifiers were trained once on a fixed final feature representation extracted from the best-performing EfficientNetB3 checkpoint. They were not trained iteratively across epochs. Therefore, the comparison is reported as a single final validation-set evaluation rather than as epoch-wise performance curves for SVM, Random Forest, and XGBoost.

**Table 7.** Accuracy and Measured Training Time Comparison

| Model | Accuracy (%) | Total Training Time |
|---|---|---|
| EfficientNetB3 + Softmax | 93.9 | ~45 min |
| EfficientNetB3 + SVM | 96.0 | ~18 min |
| EfficientNetB3 + RF | 93.7 | ~22 min |
| EfficientNetB3 + XGBoost | 96.7 | ~24 min |

Table 7 compares the accuracy and measured training time of the evaluated models. In this study, training time is reported as wall-clock time for the measured stage of each approach. For the EfficientNetB3 + Softmax baseline, the reported time corresponds to CNN training for up to 50 epochs with the specified callbacks. For the hybrid models, the reported time measures the post-checkpoint hybrid pipeline after the best EfficientNetB3 checkpoint is available, namely Dense-1024 feature extraction, feature normalization, PCA fitting and transformation, and classifier training. Therefore, the hybrid training-time values exclude the initial CNN training required to obtain the feature extractor checkpoint. EfficientNetB3 + SVM achieves the fastest measured hybrid-stage time of approximately 18 minutes with an accuracy of 96.0%, whereas EfficientNetB3 + XGBoost achieves the highest aggregate accuracy of 96.7% with a measured hybrid-stage time of approximately 24 minutes.

### 3.2. Discussion

The experimental results demonstrate performance differences among the evaluated EfficientNetB3-based classification approaches in terms of predictive accuracy, precision, recall, F1-score, class-wise behavior, and computational efficiency. Tables 4, 5, 6, and 7 collectively indicate that the proposed hybrid deep feature learning models achieve competitive aggregate performance compared with the end-to-end EfficientNetB3 with Softmax classifier. Table 4 shows that EfficientNetB3 + XGBoost obtains the highest aggregate accuracy, precision, recall, and F1-score in this validation split, while Tables 5 and 6 provide a more detailed view of performance on each class, including the minority Healthy class. Table 7 clarifies the measured training-time scope for the Softmax baseline and the hybrid pipeline.

The baseline EfficientNetB3 with Softmax achieves an accuracy of 93.9%, indicating that the pretrained CNN is capable of learning discriminative representations for potato disease classification. However, this performance comes with a longer measured CNN training time of approximately 45 minutes for up to 50 epochs. Despite the use of data augmentation and regularization strategies, the end-to-end training process remains computationally expensive due to the number of trainable parameters in the added classification head and the repeated optimization process across epochs.

The hybrid EfficientNetB3 + SVM model achieves an accuracy of 96.0% with a measured hybrid-stage time of approximately 18 minutes. The precision and F1-score indicate that the margin-based learning mechanism of SVM can exploit the Dense-1024 feature representation effectively. Nevertheless, because the dataset is imbalanced, the aggregate SVM result should

be interpreted together with the class-wise metrics, especially the recall and F1-score of the Healthy class.

The EfficientNetB3 + Random Forest approach achieves an accuracy of 93.7%, which is comparable to the Softmax-based model but with a significantly lower training time of around 22 minutes. Random Forest benefits from ensemble averaging and inherent robustness to noise; however, its performance is slightly constrained when handling high-dimensional continuous deep features. While the model demonstrates stable precision and recall values, its F1-score remains marginally lower than those achieved by SVM and XGBoost, indicating less effective decision boundary optimization in complex feature spaces.

Among all evaluated models, EfficientNetB3 combined with XGBoost obtains the highest aggregate performance in this validation split, achieving an accuracy of 96.7% with a measured hybrid-stage time of approximately 24 minutes. The balanced aggregate precision, recall, and F1-score suggest that XGBoost can capture non-linear relationships within the Dense-1024 feature space. However, the difference between XGBoost and SVM is relatively small; therefore, XGBoost should be described as the best-performing model in the conducted validation experiment rather than as statistically superior without additional significance testing.

Overall, the comparative analysis suggests that replacing the Softmax classifier with classical machine learning models can improve the trade-off between classification performance and measured hybrid-stage computation. However, because the experiments are based on a single train-validation split and no statistical significance test or repeated cross-validation is reported, the observed differences among models should be interpreted cautiously. Future experiments using repeated cross-validation, McNemar testing, or paired statistical tests would provide stronger evidence regarding whether the observed performance differences are statistically meaningful.

## 4. CONCLUSION

This study presents a hybrid deep feature learning framework for potato leaf disease classification by integrating EfficientNetB3 as a deep feature extractor with classical machine learning classifiers. The proposed approach addresses key limitations of end-to-end convolutional neural networks, particularly high computational cost and limited interpretability, while maintaining strong classification performance.

Experimental results show that the hybrid models achieve competitive aggregate accuracy, precision, recall, and F1-score compared with the baseline EfficientNetB3 with Softmax classifier. Among the evaluated classifiers, EfficientNetB3 combined with XGBoost obtains the highest aggregate performance in the conducted validation split, attaining an accuracy of 96.7% with reduced measured hybrid-stage training time compared with the end-to-end CNN training stage. The EfficientNetB3 + SVM model also exhibits strong aggregate performance with the fastest measured hybrid-stage time, indicating its potential suitability for scenarios involving limited and imbalanced datasets. However, because some performance differences are small, these findings should be interpreted cautiously without statistical significance testing.

The comparative analysis indicates that decoupling deep feature extraction from classification can enhance the balance between predictive performance and computational efficiency. By leveraging pretrained CNN representations and machine learning classifiers, the proposed hybrid framework provides a practical solution for plant disease detection. This approach is particularly relevant for agricultural applications where accuracy, efficiency, and resource constraints must be carefully balanced.

This study has several limitations. First, the dataset is imbalanced, particularly in the Healthy class, which requires careful interpretation using per-class metrics and the confusion matrix. Second, experiments were conducted only on the PlantVillage dataset under controlled imaging conditions, which may limit generalization to real-world agricultural environments. Third, the proposed framework has not yet been validated using multiple datasets or real-field images. Fourth, repeated cross-validation and statistical significance testing were not conducted; therefore, small differences among classifiers should not be overinterpreted.

Future research may explore fine-tuning selected layers of the feature extractor, incorporating explainable artificial intelligence techniques to improve model interpretability, and extending the framework to multi-crop and multi-disease classification scenarios. Overall, the findings of this study demonstrate the effectiveness of hybrid deep learning and machine learning models as a robust alternative to fully end-to-end CNN architectures for precision agriculture.

## DECLARATION OF ARTIFICIAL INTELLIGENCE USE

The authors used gemini, chatGPT to assist with language editing of the Introduction. All outputs were reviewed and verified by the authors, who take full responsibility for the content of this article.

## AUTHORS' CONTRIBUTIONS

**Bimantyoso Hamdikatama:** Conceptualization, Methodology, Software, Writing – original draft.

**Kusrini:** Supervision, Validation, Writing – review & editing.

**Arif Setyanto:** Supervision, Methodology, Writing – review & editing.

**Julius Jeremiah Allieu:** Formal analysis, Writing – review & editing.

## FUNDING

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## CONFLICT OF INTERESTS

The authors declare that there are no conflicts of interest regarding the publication of this article.

## REFERENCES

[1] V. Sharma, A. Kaur, and A. Kumar, "Early blight disease management of potato," *Potato Res.*, vol. 68, no. 2, pp. 1869–1892, 2025, doi: 10.1007/s11540-024-09816-y.

[2] F. Marisa, M. A. R. Ramadhani, A. L. Maukar, P. Savitri, Saepudin, and A. Syafariah, "Physiological symptom-based classification of potato leaf diseases using K-nearest neighbor," in *Proc. 2025 19th Int. Conf. Telecommun. Syst., Serv., Appl. (TSSA)*, 2025, pp. 1–6, doi: 10.1109/TSSA68467.2025.11303840.

[3] F. M. Farag and A. Khalifa, "Biocontrol potential of endophytic *Bacillus* spp. against early blight disease in potato (*Solanum tuberosum*)," *Biol. Control*, vol. 210, p. 105897, Nov. 2025, doi: 10.1016/j.biocontrol.2025.105897.

[4] J. Li *et al.*, "Transcriptome analysis of potato (*Solanum tuberosum* L.) seedlings with varying resistance levels reveals diverse molecular pathways in early blight resistance," *Plants*, vol. 14, no. 15, p. 2422, 2025, doi: 10.3390/plants14152422.

[5] A. Kumar and V. K. Patel, "Classification and identification of disease in potato leaf using hierarchical based deep learning convolutional neural network," *Multimed. Tools Appl.*, vol. 82, no. 20, pp. 31101–31127, 2023, doi: 10.1007/s11042-023-14663-z.

[6] A. T. Mulugeta, W. Jifara, E. Bogale, T. Desiyo, and A. Mokonnen, "Early detection and classification of potato leaf disease using convolutional neural networks," *Appl. Comput. Intell. Soft Comput.*, vol. 2025, no. 1, p. 7614841, 2025, doi: 10.1155/acis/7614841.

[7] M. Shendage, K. Golekar, A. Jambhale, and A. Bang, "Potato disease classification using deep learning," in *Proc. 2025 Int. Conf. Mach. Learn. Auton. Syst. (ICMLAS)*, 2025, pp. 342–350, doi: 10.1109/ICMLAS64557.2025.10967792.

[8] K. Kayathri, P. Sasikala, S. B. Rajakumari, K. S. Priya, M. R. Ramya, and S. Mohanambal, "Enhanced tomato leaf disease classification using the EfficientNetB3 and ResNet50 fusion," in *Proc. 2025 14th Int. Conf. Syst. Model. Adv. Res. Trends (SMART)*, 2025, pp. 219–226, doi: 10.1109/SMART66937.2025.11389623.

[9] L. K. Kumari *et al.*, "A deep learning based approach for tomato leaf disease detection using EfficientNetB3," in *Proc. 2025 Int. Conf. Cogn., Green Ubiquitous Comput. (IC-CGU)*, 2025, pp. 1–6, doi: 10.1109/IC-CGU67042.2025.11338013.

[10] G. R. Panigrahi, G. S. Shiny, G. V. Londhe, S. Sharma, A. Misra, and D. Bhardwaj, "Automated tomato leaf disease detection using EfficientNetB3: A deep learning approach for agricultural diagnostics," in *Proc. 2025 Int. Conf. Decis. Aid Sci. Appl. (DASA)*, 2025, pp. 1580–1585, doi: 10.1109/DASA68193.2025.11498966.

[11] M. A. Patil, M. Manohar, C. Laxuman, K. Parane, B. M. Dodamani, and G. Sunkad, "Hybrid deep learning-based potato and tomato leaf disease classification," in *Advances in Data-Driven Computing and Intelligent Systems* (Lecture Notes in Networks and Systems). Singapore: Springer, 2024, pp. 157–174, doi: 10.1007/978-981-99-9524-0_13.

[12] T. Illakiya, A. V. Joel, N. D. Nahata, S. Sukumar, and S. P. Chelladurai, "A hybrid deep learning model for potato leaf disease classification using transfer learning and attention mechanism," in *Proc. 2025 8th Int. Conf. Comput. Methodol. Commun. (ICCMC)*, 2025, pp. 837–842, doi: 10.1109/ICCMC65190.2025.11140741.

[13] L. Priyadharshini, P. Kalaiselvi, S. Sujatha, S. Uma Maheswari, V. A. Kandaswamy, and R. Suganthi, "HNCS: A robust development of deep learning based potato plant disease identification using hybrid neural classification strategy," in *Proc. 2024 4th Asian Conf. Innov. Technol. (ASIANCON)*, 2024, pp. 1–5, doi: 10.1109/ASIANCON62057.2024.10838204.

[14] A. R. Nalla, D. K. Basha, A. Archana, and K. S. Rao, "Plant disease classification using EfficientNetB3 for tomato leaf," in *Proc. 2025 IEEE 1st Int. Conf. Recent Trends Comput. Smart Mobility (RCSM)*, 2025, pp. 1–6, doi: 10.1109/RCSM67767.2025.11507776.

[15] H. I. Peyal and M. N. I. Mondal, "Potato leaf disease classification using lightweight CNN-SVM and real-world web-based tools," in *Proc. 2025 Int. Conf. Elect., Comput. Commun. Eng. (ECCE)*, 2025, pp. 1–5, doi: 10.1109/ECCE64574.2025.11013900.

[16] S. Chauhan and K. S. Kumar, "AI-driven potato disease classification: ResNet50 for detecting late blight in crop leaves," in *Proc. 2025 3rd Int. Conf. Augment. Intell. Sustain. Syst. (ICAISS)*, 2025, pp. 793–798, doi: 10.1109/ICAISS61471.2025.11042051.

[17] A. Singh and H. Kaur, "Potato plant leaves disease detection and classification using machine learning methodologies," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 1022, no. 1, p. 012121, 2021, doi: 10.1088/1757-899X/1022/1/012121.

[18] A. H. T. Rusli, B. C. C. Meng, N. S. Damanhuri, N. A. Othman, M. H. Othman, and W. F. A. W. Zaidi, "Potato leaf disease classification using image processing and artificial neural network," in *Proc. 2022 IEEE 12th Int. Conf. Control Syst., Comput. Eng. (ICCSCE)*, 2022, pp. 107–112, doi: 10.1109/ICCSCE54767.2022.9935654.

[19] A. Baranwal, M. Mishra, A. Goyal, and Yogesh, "Potato plant disease classification through deep learning," in *Proc. 2022 Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COM-IT-CON)*, 2022, pp. 673–681, doi: 10.1109/COM-IT-CON54601.2022.9850701.

[20] K. El Moutaouakil, B. Jabir, and N. Falih, "A convolutional neural networks-based approach for potato disease classification," in *Business Intelligence (CBI 2022)* (Lecture Notes in Business Information Processing). Cham, Switzerland: Springer, 2022, pp. 29–40, doi: 10.1007/978-3-031-06458-6_2.

[21] E. Liljeroth, L. Stridh, E. Edin, and R. S. Sajeevan, "Early blight in potato: Decade-long field studies show high variation in yield losses," *Potato Res.*, vol. 68, no. 4, pp. 4909–4930, 2025, doi: 10.1007/s11540-025-09940-3.

[22] M. Dutta, D. Gupta, Y. Gulzar, M. S. Mir, C. W. Onn, and A. B. Soomro, "Leveraging Inception V3 for precise early and late blight disease classification in potato crops," *Trait. Signal*, vol. 41, no. 2, pp. 705–715, 2024, doi: 10.18280/ts.410213.

[23] R. Asakaviciute, A. Zelya, T. Andriychuk, and A. Razukas, "Evaluation of potato varieties for yield, quality, and late blight resistance," *Life*, vol. 15, no. 9, p. 1378, 2025, doi: 10.3390/life15091378.

[24] J. Rashid, I. Khan, G. Ali, S. H. Almotiri, M. A. AlGhamdi, and K. Masood, "Multi-level deep learning model for potato leaf disease recognition," *Electronics*, vol. 10, no. 17, p. 2064, 2021, doi: 10.3390/electronics10172064.

[25] 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, doi: 10.48550/arXiv.1511.08060.

[26] V. W. de Vargas *et al.*, "Imbalanced data preprocessing techniques for machine learning: A systematic mapping study," *Knowl. Inf. Syst.*, vol. 65, no. 1, pp. 31–57, 2023, doi: 10.1007/s10115-022-01772-8.

[27] A. Tawakuli, B. Havers, V. Gulisano, D. Kaiser, and T. Engel, "Survey: Time-series data preprocessing: A survey and an empirical analysis," *J. Eng. Res.*, vol. 13, no. 2, pp. 674–711, 2025, doi: 10.1016/j.jer.2024.02.018.

[28] K. M. Rao, S. Ghanta, and K. Supriya, "Data preprocessing techniques: Emergence and selection towards machine learning models – a practical review using HPA dataset," *Multimed. Tools Appl.*, vol. 82, no. 24, pp. 37177–37196, 2023, doi: 10.1007/s11042-023-15087-5.

[29] G. Verma, "Eye disease classification enhanced by EfficientNetB3: A deep learning approach," in *Proc. 2024 Int. Conf. Artif. Intell. Emerg. Technol. (Global AI Summit)*, 2024, pp. 1016–1021, doi: 10.1109/GlobalAISummit62156.2024.10947971.

[30] M. T. Ramakrishna *et al.*, "Leveraging EfficientNetB3 in a deep learning framework for high-accuracy MRI tumor classification," *Comput. Mater. Contin.*, vol. 81, no. 1, pp. 867–883, 2024, doi: 10.32604/cmc.2024.053563.

[31] A. Amato and V. Di Lecce, "Data preprocessing impact on machine learning algorithm performance," *Open Comput. Sci.*, vol. 13, no. 1, p. 20220278, 2023, doi: 10.1515/comp-2022-0278.

[32] B. Bala and S. Behal, "A brief survey of data preprocessing in machine learning and deep learning techniques," in *Proc. 2024 8th Int. Conf. I-SMAC (IoT Soc., Mobile, Anal. Cloud)*, 2024, pp. 1755–1762, doi: 10.1109/I-SMAC61858.2024.10714767.

[33] J. Hu, "High-resolution processing and feature extraction of logging curves based on wavelet transform and principal component analysis," in *Proc. 2025 IEEE 4th Int. Conf. Safe Prod. Informatiz. (IICSPI)*, 2025, pp. 389–397, doi: 10.1109/IICSPI66775.2025.11437785.

[34] C. E. Emere, C. O. Aigbavboa, W. D. Thwala, *et al.*, "A principal component analysis of sustainable building construction features for project delivery in South Africa," *J. Eng. Des. Technol.*, vol. 23, no. 5, pp. 1463–1482, 2025, doi: 10.1108/JEDT-01-2024-0015.

[35] I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philos. Trans. R. Soc. A*, vol. 374, no. 2065, p. 20150202, 2016, doi: 10.1098/rsta.2015.0202.
[36] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982, doi: 10.1109/TIT.1982.1056489.
[37] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010, doi: 10.1016/j.patrec.2009.09.011.
[38] O. Kisi, S. Heddam, K. S. Parmar, A. Petroselli, C. Külls, and M. Zounemat-Kermani, "Integration of Gaussian process regression and K means clustering for enhanced short term rainfall runoff modeling," *Sci. Rep.*, vol. 15, no. 1, p. 7444, 2025, doi: 10.1038/s41598-025-91339-8.
[39] G. Verma, K. S. Kumar, N. Sharma, S. Kaparwan, and R. Madan, "Optimized EfficientNetB3 framework for high-precision apple leaf disease detection and classification," in *Proc. 2026 Contemp. Comput. Innov. Conf. (CCIC)*, 2026, pp. 1–6, doi: 10.1109/CCIC68129.2026.11485912.
[40] M. Velamakanni, S. S. Kanumalli, A. P. Gopi, V. L. Narayana, V. Sujatha, and A. Naresh, "Tomato leaf disease detection using EfficientNetB3," in *Proc. 2025 IEEE Madhya Pradesh Sect. Conf. (MPCON)*, 2025, pp. 765–771, doi: 10.1109/MPCON66082.2025.11256726.
[41] B. Bataineh *et al.*, "Explainable hybrid deep learning with EfficientNetB3–DenseNet121 feature fusion for banana leaf spot disease classification," *Int. J. Adv. Signal Image Sci.*, vol. 12, no. 1, pp. 1120–1137, 2026, doi: 10.29284/yeveds27.
[42] E. S. Mughele, S. C. Chiemeke, and J. D. Ebube, "Tomato leaf disease detection using transfer learning with EfficientNetB3: A CNN-based approach," *Caliphate J. Sci. Technol.*, vol. 7, no. 3, pp. 454–459, 2025, doi: 10.4314/cajost.v7i3.18.
[43] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in Neural Information Processing Systems 9*. Cambridge, MA, USA: MIT Press, 1997, pp. 155–161.
[44] X. Yang, P. Meng, Z. Jiang, and L. Zhou, "Deep siamese residual support vector machine with applications to disease prediction," *Comput. Biol. Med.*, vol. 196, p. 110693, 2025, doi: 10.1016/j.compbiomed.2025.110693.
[45] J. Rybak, H. Battey, and W.-X. Zhou, "On inference for the support vector machine," *J. Mach. Learn. Res.*, vol. 26, no. 85, pp. 1–54, 2025. [Online]. Available: https://jmlr.org/papers/v26/23-1581.html
[46] W. Zheng, X. Yuan, X. Bao, and Y. Dong, "Adaptive support vector machine for time-variant failure probability function estimation," *Reliab. Eng. Syst. Saf.*, vol. 253, p. 110510, 2025, doi: 10.1016/j.ress.2024.110510.
[47] Z. Mohamadi *et al.*, "The application of random forest-based models in prognostication of gastrointestinal tract malignancies: A systematic review," *Front. Artif. Intell.*, vol. 8, p. 1517670, 2025, doi: 10.3389/frai.2025.1517670.
[48] R. O. Ogundokun, P. A. Owolawi, and E. A. van Wyk, "A lightweight deep feature fusion and PCA-optimized random forest framework for multi-class potato leaf disease classification," in *Applied Informatics* (Communications in Computer and Information Science). Cham, Switzerland: Springer, 2025, pp. 147–162, doi: 10.1007/978-3-032-07175-0_10.
[49] J. Korstanje, "The random forest," in *Advanced Forecasting with Python*, 2nd ed. Berkeley, CA, USA: Apress, 2025, pp. 191–204, doi: 10.1007/979-8-8688-2028-1_15.
[50] B. Mallala, A. I. U. Ahmed, S. V. Pamidi, M. O. Faruque, and R. Reddy M, "Forecasting global sustainable energy from renewable sources using random forest algorithm," *Results Eng.*, vol. 25, p. 103789, 2025, doi: 10.1016/j.rineng.2024.103789.

[51] B. Tamiru, T. Soromessa, B. Warkineh, and G. Legese, "Modelling selected soil chemical properties using TerrSet and random forest: A case of Hangadi watershed, Oromia, Ethiopia," *Geol. Ecol. Landscapes*, vol. 9, no. 4, pp. 1177–1193, 2025, doi: 10.1080/24749508.2024.2429842.

[52] Y. Xia, W. Lu, Z. Peng, J. Lou, J. Huang, and J. Niu, "Understanding the impacts of space design on local outdoor thermal comfort: An approach combining DepthmapX and XGBoost," *Energy Build.*, vol. 332, p. 115451, 2025, doi: 10.1016/j.enbuild.2025.115451.

[53] Y. Liu *et al.*, "Urban flood depth prediction and visualization based on the XGBoost-SHAP model," *Water Resour. Manag.*, vol. 39, no. 3, pp. 1353–1375, 2025, doi: 10.1007/s11269-024-04020-6.

[54] M. Imani, A. Beikmohammadi, and H. R. Arabnia, "Comprehensive analysis of random forest and XGBoost performance with SMOTE, ADASYN, and GNUS under varying imbalance levels," *Technologies*, vol. 13, no. 3, p. 88, 2025, doi: 10.3390/technologies13030088.

[55] H. Zhuo, T. Li, W. Lu, Q. Zhang, L. Ji, and J. Li, "Prediction model for spontaneous combustion temperature of coal based on PSO-XGBoost algorithm," *Sci. Rep.*, vol. 15, no. 1, p. 2752, 2025, doi: 10.1038/s41598-025-87035-2.