COMMUNICATIONS IN SCIENCE AND TECHNOLOGY
Homepage: cst.kipmi.or.id

A comprehensive evaluation of broad learning system for deep feature-based chili leaf disease classification

Rudi Kurniawan[a,*], Lukman Sunardi[a], Bunga Intan[a], Zulhipni Reno Saputra Elsi[b], Fatma Susilawati Mohamad[c]

[a]*Department of Informatics, Universitas Bina Insan, Lubuk Linggau 31626, Indonesia*
[b]*Department of Information Technology, Universitas Muhammadiyah Palembang, Palembang 30263, Indonesia*
[c]*Department of Information Technology, Universiti Sultan Zainal Abidin, Kuala Terengganu 21300, Malaysia*

Article history:
Received: 15 May 2026 / Received in revised form: 15 June 2026 / Accepted: 18 June 2026

Abstract

The early detection of plant leaf diseases is essential for the enhancement of crop productivity and the promotion of sustainable agricultural practices. While deep learning models have been shown to be achieve remarkable success in the recognition of plant disease, conventional classifiers commonly rely on iterative gradient-based optimization, resulting in increased training complexity. This present study investigates a hybrid framework for the classification of chili leaf disease that combines DenseNet201-based deep feature extraction with a Broad Learning System (BLS) classifier. The DenseNet201 model is employed to generate discriminative feature representations, whereas the BLS approach employs a closed-form ridge regression solution for classification. The present study involved experiments conducted by means of the publicly available Chili Plant Leaf Disease Dataset containing 1,856 original images from six different disease categories. To prevent data leakage, the dataset was initially partitioned into training, validation, and test subsets at the original-image level, with data augmentation being applied exclusively to the training set, thereby increasing it to 9,093 images. The proposed DenseNet201+BLS framework achieved a test accuracy of 99.28% and a macro F1-score of 99.00%. Furthermore, the performance of the proposed model was compared with that of Softmax, Logistic Regression, Random Forest, Multilayer Perceptron, and Support Vector Machine (SVM) classifiers using identical DenseNet201 feature representations. Among the evaluated classifiers, SVM demonstrated the highest level of accuracy (99.64%), whereas BLS exhibited a favorable balance between predictive performance and computational efficiency, requiring less than one second for training while outperforming Softmax, Logistic Regression, and Random Forest. Grad-CAM visualizations further demonstrated that the extracted deep features focus on disease-relevant regions such as lesions, discoloration patterns, and abnormal leaf structures. The findings indicate that the integration of DenseNet201 feature extraction with a Broad Learning System offers a competitive and computationally efficient alternative for the automated classification of chili leaf disease. The proposed framework facilitates accurate disease recognition with substantially reduced training costs, making it a promising solution for resource-efficient agricultural monitoring and decision-support applications.

Keywords: Plant disease classification; plant leaf disease; DenseNet201; broad learning system; deep learning

1. Introduction

Chili (*Capsicum annuum*) is a major horticultural commodity with substantial economic importance in numerous agricultural regions. Its utilization is pervasive in the domains of food processing, pharmaceutical production, and related industries. However, chili cultivation is highly vulnerable to various leaf diseases caused by fungal, bacterial, and viral pathogens. These diseases have been shown to have a significant impact on crop yield and quality, resulting in considerable economic losses for farmers. Consequently, accurate and early disease identification is essential for effective crop management and sustainable agricultural production [1,2].

In conventional agricultural practice, the diagnosis of chili leaf diseases is commonly conducted through a visual inspection. However, this method is time-consuming, subjective, and difficult to scale for large plantations, particularly in areas with limited access to plant pathology experts. The consequences of delayed or inaccurate diagnoses may include improper treatment and increased disease spread. This has motivated the development of automated image-based disease recognition systems for chili plants [1,3].

* Corresponding author.
Email: rudi.kurniawan@univbinainsan.ac.id
https://doi.org/10.21924/cst.11.1.2026.1997

This open access article is distributed under a Creative Commons Attribution (CC-BY) 4.0 license

Early studies on the classification of chili leaf disease mainly involved the utilization of handcrafted features in conjunction with conventional machine learning methods. For example, the combination of Gray Level Co-occurrence Matrix (GLCM) features with Support Vector Machine (SVM) classification has been shown to achieve the accuracy of 88% [3]. However, the efficacy of these approaches is constrained by manually engineered features, which frequently fail to capture complex variations in disease symptoms, texture, and color patterns.

With the rapid advancement of deep learning, convolutional neural networks (CNNs) have emerged as the dominant approach for plant disease recognition considering their capacity to autonomously acquire hierarchical and discriminative feature representations. In the context of the classification of chili leaf disease, several CNN-based approaches have demonstrated promising results. A squeeze-and-excitation-based CNN demonstrated a 99.12% accuracy rate on an augmented chili leaf dataset, surpassing numerous pretrained architectures, including DenseNet201, InceptionV3, MobileNetV2, ResNet101, VGG19, and XceptionNet [4]. Similarly, a customized EfficientNetB4 model demonstrated superior performance in comparison to extensively employed transfer learning architectures such as ResNet-50, DenseNet-121, MobileNetV2, and VGG-16, attaining an accuracy of 91.2% [5].

Recent studies have further enhanced the recognition of chili leaf disease through the implementation of advanced feature extraction strategies and attention mechanisms. The MCCM model incorporated multi-scale feature fusion and mixed channel-spatial attention to capture disease characteristics appearing at different scales and spatial locations, achieving an accuracy of 93.5% and demonstrating strong generalization across multiple crop disease datasets [6]. Another study proposed an optimization-based GoogLeNet framework, referred to as EGMEO, which achieved 92.806% accuracy for the classification of chili leaf disease [7]. The DBESeriesNet model, a compact deep architecture containing only 3.4 million parameters, achieved 99.80% accuracy on a chili leaf dataset [8]. Transfer learning approaches based on VGG16 and MobileNetV2 also demonstrated competitive performance, with VGG16 achieving 94% validation accuracy in a five-class classification setting [9]. In addition, ensemble-based architectures have been explored. Chili-Net, which integrates ResNet50, VGG16, and InceptionV3 for feature extraction, achieved an accuracy of 97% [10]. The collective analysis of these studies demonstrate the effectiveness of deep learning in the classification of chili leaf disease.

Beyond chili-specific applications, recent advancements in plant disease recognition across other crop species underscores the efficacy of hybrid and attention-based deep learning frameworks. A dual-network architecture integrating a Graph Attention Network with a CNN branch has been shown to achieve strong performance by leveraging both relational and spatial information for plant disease identification [11]. Lightweight attention-based CNNs have also been shown to enhance classification accuracy while substantially reducing computational complexity, making them suitable for deployment in portable agricultural systems [12]. Similarly, multi-kernel CNN architectures with attention mechanisms have demonstrated superior performance for the recognition of citrus disease by enhancing both channel-wise and spatial feature representation [13]. The findings indicate that effective disease classification increasingly depends not only on network depth but also on efficient feature representation and computational efficiency.

Among existing CNN backbones, DenseNet is particularly attractive in view of its dense connectivity structure, in which each layer receives feature information from all preceding layers [14]. This architecture promotes feature reuse, improves information flow, and mitigates gradient degradation, making DenseNet highly effective for image classification tasks [15,16]. Nevertheless, despite their strong representational capability, CNN-based models still require iterative backpropagation [17], repeated hyperparameter optimization [18-20], and substantial computational resources [21]. In practical agricultural environments, particularly where frequent retraining or deployment under limited computational resources is required, these constraints have the potential to reduce model usability [22,23].

The integration of deep feature extraction with lightweight classifiers within hybrid framework has emerged as a promising approach for reducing the computational complexity of conventional deep learning models. In these frameworks, pretrained deep networks are employed to extract discriminative visual features, while efficient classifiers perform the final prediction stage. A notable method in this regard is the Broad Learning System (BLS), which has been shown to expand the network horizontally through the utilization of feature mapping and enhancement nodes. In contrast to conventional deep neural networks, BLS utilizes a closed-form ridge regression solution to compute output weights, thereby eliminating the iterative backpropagation process and enabling efficient training [24]. Previous studies have demonstrated that BLS-based approaches can achieve competitive performance with reduced computational cost in image classification tasks [25].

Despite these advances, several research gaps remain in the classification of chili leaf disease. The majority of existing studies focus on increasingly complex CNN architectures, which frequently result in higher computational cost and complex training. Furthermore, limited attention has been given to the systematic evaluation of alternative classifiers on top of deep visual representations. Existing studies also primarily emphasize classification accuracy, while analyses involving statistical significance, robustness, computational efficiency, and classifier suitability for deep feature spaces are rarely explored.

Motivated by these limitations, this study investigates the effectiveness of the Broad Learning System as an efficient classifier for DenseNet201 feature representations in the classification of chili leaf disease. This work contributes a comprehensive empirical evaluation of BLS within a deep feature–based classification framework for the recognition of chili leaf disease. Specifically, DenseNet201 is employed as a pretrained feature extractor, while BLS is evaluated as a lightweight non-backpropagation classifier and compared with several widely used alternatives, including Softmax, Logistic Regression, Random Forest, Multilayer Perceptron, and Support Vector Machine. This comprehensive comparison is

employed by the study to examine the trade-offs among classification performance, robustness, statistical significance, training efficiency, inference cost, and model complexity. The proposed DenseNet201–BLS framework aims to retain the strong representational capability of DenseNet201 while substantially reducing classifier training time.

2. Materials and Methods

This section describes the materials and methodology employed for the classification of chili leaf disease. The proposed framework integrates DenseNet201 for the purpose of deep feature extraction with a Broad Learning System (BLS) classifier. The workflow begins with dataset preparation and image preprocessing, including resizing and normalization. Subsequently, discriminative feature representations are extracted using the pretrained DenseNet201 model. The extracted feature vectors are then utilized as inputs to the BLS classifier for the purpose of multi-class classification of chili leaf disease. To evaluate the effectiveness of the proposed framework, the dataset is divided into training, validation, and testing subsets. The performance of the model is evaluated through classification performance analysis and comparative experimental assessment.

The overall methodology adopted in this study is illustrated in Fig. 1, which presents the main stages of the proposed framework for the classification of chili leaf disease.

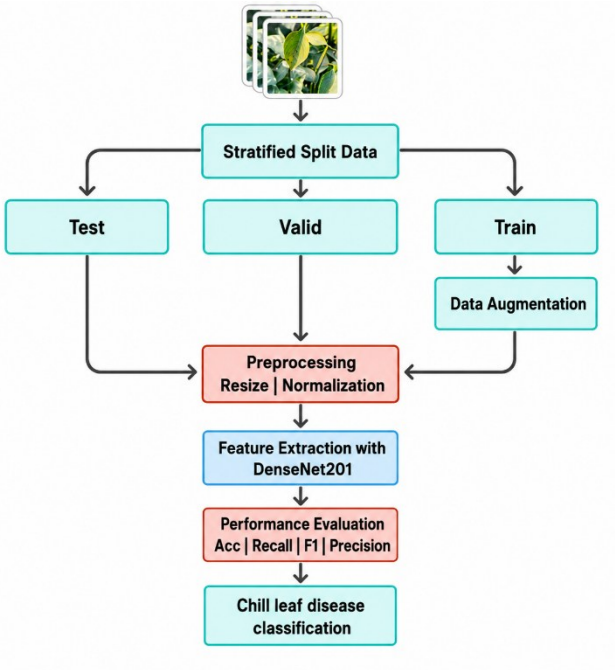

Fig. 1. Proposed framework

2.1. Data description

This present study utilizes the publicly available Chili Plant Leaf Disease and Growth Stage Dataset, which was obtained from the Mendeley Data repository [26]. For the leaf disease classification task, the original dataset contains 1,856 labeled images belonging to six classes: Bacterial Spot, Cercospora Leaf Spot, Curl Virus, Healthy Leaf, Nutrition Deficiency, and White Spot.

Table 1. Class distribution of the original chili leaf disease dataset

Class	Number of Images	Percentage (%)
Bacterial Spot	156	8.41
Cercospora Leaf Spot	156	9.70
Curl Virus	423	22.79
Healthy Leaf	458	24.68
Nutrition Deficiency	444	23.92
White Spot	195	10.51
Total	1,856	100

The images exhibit substantial variations in disease severity, leaf appearance, illumination conditions, and viewing angles, reflecting realistic agricultural environments. Table 1 provides a summary of the class distribution of the original dataset utilized in this study.

The original dataset exhibits a moderate class imbalance, with Healthy Leaf, Nutrition Deficiency, and Curl Virus containing substantially more samples than Bacterial Spot, Cercospora Leaf Spot, and White Spot. This imbalance reflects the natural distribution of disease occurrences in real agricultural environments and presents an additional challenge for multi-class disease classification. The observed variations in disease severity, leaf appearance, illumination conditions, and viewing angles further provide a realistic benchmark for evaluating the robustness and generalization capability of automated systems for the recognition of chili leaf disease.

2.2. Data preprocessing

Prior to the extraction of features and their classification, the original dataset was divided into training, validation, and test subsets through the implementation of a stratified split at the original-image level. The protocol ensured that all images derived from a particular original sample were assigned exclusively to a single subset, thereby preventing augmented-image leakage between training and evaluation data. The resulting dataset comprised 1,299 training images, 278 validation images, and 279 test images.

Data augmentation was applied exclusively to the training subset with the objective of increasing sample diversity while preserving the original validation and test sets. The augmentation pipeline included random rotation (up to 25°), width and height shifting (10%), shearing (10%), zooming (15%), brightness adjustment (0.75–1.25), and horizontal flipping. Fig. 2 presents the representative examples of the augmentation operations applied during training. Subsequent to augmentation, the number of images in the training set increased to 9,093, whereas the validation and test sets remained unchanged and contained solely original images. Consequently, model evaluation was performed exclusively on previously unseen, non-augmented images, providing a more reliable assessment of generalization performance.

Each RGB image, represented as $I \in R^{HxWx3}$, was subjected to a resizing process to 224 × 224 pixels to satisfy the input requirement of DenseNet201. This resulted in a standardized tensor representation $I \in R^{224x224x3}$. The purpose of this resizing process is to ensure uniform input dimensions across the entire dataset.

Subsequently, pixel intensity normalization was applied using the standard ImageNet mean and standard deviation values to align the image distribution with the pretrained DenseNet201 model. The normalization process is defined as follows:

$$x' = \frac{x-\mu}{\sigma} \quad (1)$$

where μ= (0.485,0.456,0.406) and σ= (0.229,0.224,0.225). This process of normalization has been shown to stabilize feature distributions and improve convergence during model training [27,28].

Furthermore, data augmentation has been demonstrated to improve model generalization and reduce overfitting by exposing the model to diverse variations in object appearance while preserving disease-related characteristics [29]. Following the implementation of preprocessing stage, all images were transformed into standardized tensors, thus making them suitable for the purpose of deep feature extraction using DenseNet201.

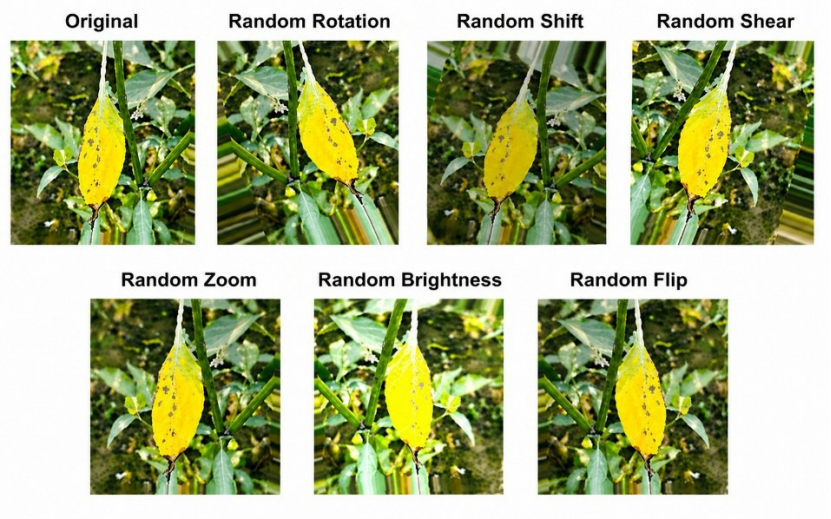

Fig. 2. Augmentation process

2.3. Deep feature extraction using Densenet201

Deep feature extraction plays a critical role in the classification of plant diseases based on images because it facilitates the automatic learning of discriminative visual representations from raw image data [30]. In this study, DenseNet201 is selected as the backbone network for feature extraction because of its strong capability for feature reuse and efficient information propagation across layers. DenseNet employs dense connectivity, in which each layer receives feature maps from all preceding layers as input [31]. This architecture mitigates the vanishing gradient problem, promotes feature reuse, and improves information flow during training.

The output of the l-th layer in DenseNet is formally defined as follows:

$$x_l = H_{l([x_0,x_1,\ldots,x_{\{l-1\}}])} \quad (2)$$

where x_l denotes the output feature map of the l-th layer, H_l represents a composite function consisting of batch normalization, nonlinear activation, and convolution operations, and $[x_0, x_1, \ldots, x_{l-1}]$ denotes the concatenation of feature maps generated by all preceding layers. This dense connectivity facilitates the ability of subsequent layers to directly access earlier feature representations, thereby enhancing feature propagation and parameter efficiency.

In this study, a pretrained DenseNet201 model initialized with ImageNet weights is utilized to extract deep visual features from preprocessed chili leaf images. Transfer learning is adopted to leverage knowledge learned from large-scale image datasets, allowing the model to capture complex texture, color, and structural patterns associated with plant diseases. Rather than training the entire network for classification, the final fully connected classification layer is removed, and the network is employed exclusively as a feature extractor.

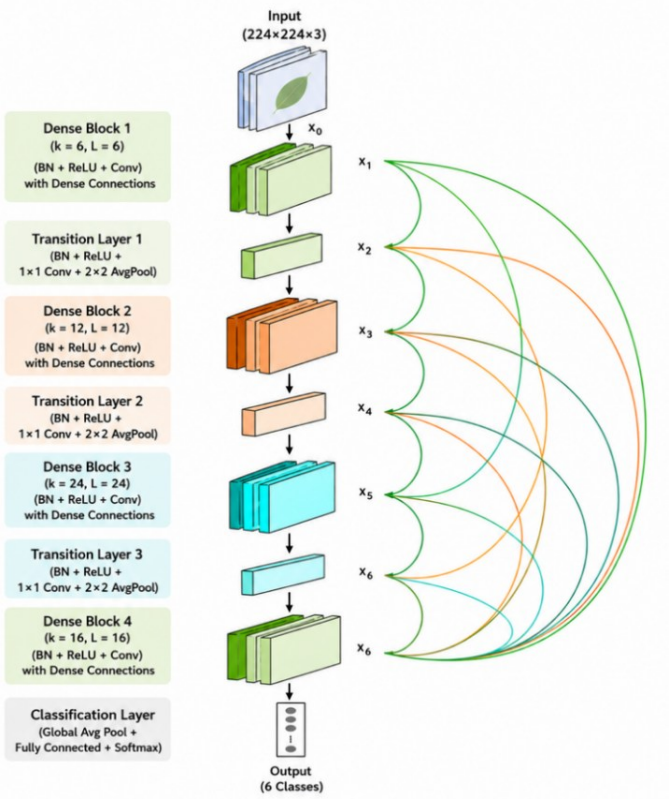

Fig. 3. DenseNet architecture

Each input image is processed through the convolutional layers of DenseNet201, followed by a Global Average Pooling (GAP) operation that aggregates spatial feature maps into a compact representation. Consequently, each image is transformed into a 1920-dimensional deep feature vector containing high-level visual information relevant to disease classification. The feature vectors are subsequently utilized as inputs for the downstream classification stage, employing the Broad Learning System (BLS).

The DenseNet201 architecture employed for feature extraction is illustrated in Fig. 3. The network consists of an initial convolutional layer followed by multiple dense blocks connected through transition layers. Each dense block contains a series of convolutional layers with dense connections that concatenate feature maps from all preceding layers. Transition layers are positioned between dense blocks with the purpose of performing dimensionality reduction through convolution and pooling operations. Finally, the Global Average Pooling layer generates a compact feature representation that is then utilized as the input to the subsequent classifier.

2.4. Broad learning system

The Broad Learning System (BLS) is an efficient learning framework that has been proposed as an alternative to conventional deep neural networks [32]. In contrast to deep architectures that improve representational capability by increasing network depth, BLS expands the model horizontally through feature mapping nodes and enhancement nodes [24]. This structure facilitates the model's capacity to accommodate nonlinear relationships while maintaining relatively low computational complexity. In addition, BLS employs a closed-form solution for the computation of output weight, rather than iterative gradient-based optimization typically employed in conventional deep learning models. This approach enables training to be conducted at a substantially faster rate [33].

In the proposed framework, the deep feature vectors extracted by DenseNet201 are used as input to the BLS classifier. Let the extracted feature matrix be defined as follows:

$$X \in R^{Nxd} \tag{3}$$

where N denotes the number of samples and $d$represents the dimensionality of the extracted deep feature vector, with $d = 1920$ in this study.

The first stage of BLS involves the transformation of input features into multiple groups of nonlinear feature mapping nodes. This process is defined as follows:

$$Z_i = \phi(XW_i + b_i) \tag{4}$$

where Z_i denotes the i-th group of feature mapping nodes, $W_i$represents the randomly initialized weight matrix, b_i is the bias vector, and $\phi(\cdot)$ denotes a nonlinear activation function, such as the hyperbolic tangent function. The outputs from all feature mapping groups are then concatenated to form the mapping feature matrix:

$$Z = [Z_1, Z_2, Z_3, \ldots\ldots, Z_n] \tag{5}$$

where n represents the number of mapping groups.

To further enhance the representational capability of the model, BLS introduces an enhancement layer that generates additional nonlinear features from the mapped features. The enhancement nodes are computed as follows:

$$H = \xi\,(ZW_e + b_e) \tag{6}$$

In this equation, Hdenotes the enhancement node matrix, W_e represents the weight matrix that has been randomly initialized, b_e denotes the bias term, and $\xi(\cdot)$ is a nonlinear activation function. These enhancement nodes serve to increase the expressive capacity of the model by capturing additional nonlinear relationships within the feature space.

The final input representation for classification is obtained by concatenating the mapping nodes and enhancement nodes:

$$H = [Z, H] \tag{7}$$

The output weights of the BLS model are subsequently computed by using ridge regression:

$$\beta = (A^T A + \lambda \mathrm{I})^{-1} A^T Y \tag{8}$$

where β denotes the output weight matrix, Y represents the target label matrix, I is the identity matrix, and λ denotes the regularization parameter that controls model complexity. This closed-form solution eliminates iterative backpropagation, thereby substantially reducing training time while maintaining competitive classification performance.

The architecture of the BLS classifier utilized in this study is illustrated in Fig. 4. The framework consists of three main components: feature mapping nodes, enhancement nodes, and an output layer that has been optimized using ridge regression. The deep features extracted by DenseNet201 are first transformed into mapping nodes, followed by enhancement nodes that expand the feature space. The resulting concatenated representation is then used to compute the output weights for multi-class chili leaf disease classification.

2.5. Experimental setup

The experiments were conducted based upon the publicly available Chili Plant Leaf Disease Dataset, which comprises 1,856 images across six classes. The dataset was subsequently divided into training, validation, and testing subsets utilizing a stratified 70:15:15 split at the original-image level with the objective of preventing data leakage. Data augmentation was implemented exclusively on the training set, thereby increasing its size to 9,093 images, while the validation and test sets remained unchanged.

Prior to classification, all images underwent a series of preprocessing steps and were processed using a pretrained DenseNet201 model for the purpose of deep feature extraction. The final fully connected classification layer was removed, and the output of the Global Average Pooling (GAP) layer was utilized as the feature representation. This process resulted in a 1,920-dimensional feature vector for each image.

To evaluate the suitability of different classifiers for DenseNet201 feature representations, six classification

methods were investigated: Softmax, Logistic Regression (LR), Random Forest (RF), Multilayer Perceptron (MLP), Support Vector Machine (SVM), and the proposed Broad Learning System (BLS). For the BLS framework, multiple combinations of mapping groups, mapping nodes, enhancement nodes, regularization coefficients, and activation functions were systematically explored using the validation set during hyperparameter optimization.

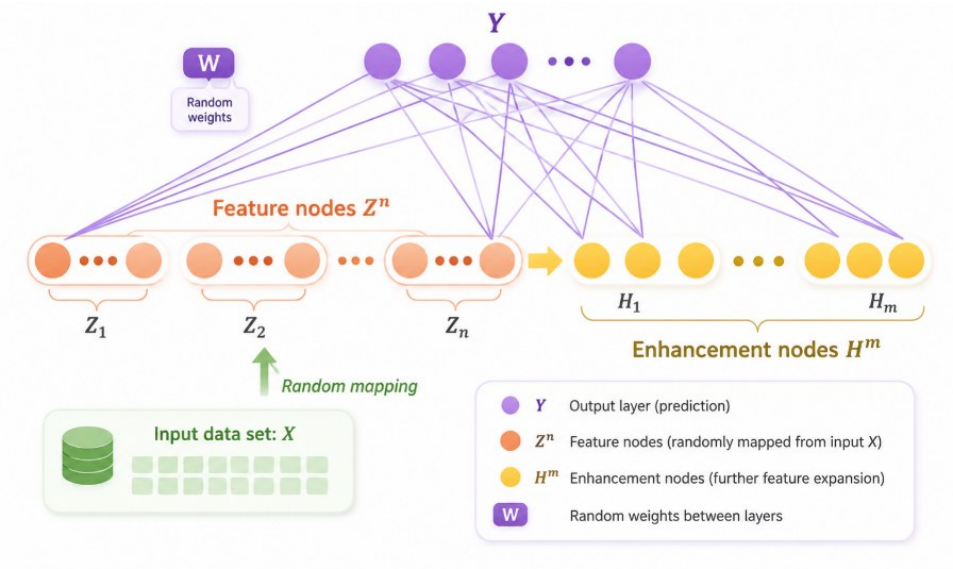

Fig. 4. BLS architecture

Model performance was evaluated using accuracy, precision, recall, and macro F1-score. To quantify prediction uncertainty, 95% bootstrap confidence intervals (CIs) were computed for classification accuracy. To ascertain whether the discrepancy in performance between the proposed framework and the baseline classifier was statistically significant, McNemar's test [34] was employed, defined as follows:

$$X^2 = \frac{(|b-c|-1)^2}{b+c} \quad (8)$$

where b represents the number of samples that were misclassified by the first model but were correctly classified by the second model. Concurrently, c denotes the opposite condition. A small p-value indicates a statistically significant difference between the compared models.

To enhance the interpretability of the model, Gradient-weighted Class Activation Mapping (Grad-CAM) technique was employed to visually represent the image regions that played the most significant role in the classification decisions made by the DenseNet201 feature extractor [35].

All experiments were conducted on a workstation that was equipped with an Intel Core i7 9th Generation processor, 64 GB DDR4 RAM, and an NVIDIA GeForce GTX 1660 Ti GPU with 6 GB memory. The implementation was developed using Python 3.10 with TensorFlow and Keras in a Jupyter Notebook environment running on Windows 11.

3. Results and Discussion

In this present study, the experiments were designed to evaluate the efficacy of the Broad Learning System (BLS) in its capacity as a classifier, with its operation being based on the extraction of deep feature representations by DenseNet201. Consequently, all reported computational efficiency measurements refer exclusively to the classification stage following feature extraction. The preprocessing pipeline and DenseNet201 feature extraction process remained consistent for all evaluated classifiers. As a result, they were excluded in the comparative training-time analysis.

It should be noted that the reported training and inference times do not represent the computational cost of the complete end-to-end pipeline. In practical deployment scenarios, the total processing cost encompasses image preprocessing, DenseNet201 forward-pass feature extraction, classifier training, inference, and model storage requirements. Consequently, the computational efficiency results presented in this study should be interpreted as a comparison of classifier efficiency under a fixed DenseNet201 feature representation rather than as a comparison of complete end-to-end disease recognition systems.

The subsequent subsections present the hyperparameter exploration, classification performance, statistical analysis, robustness evaluation, computational efficiency assessment, and explainability analysis of the evaluated classifiers.

3.1. Hyperparameter exploration of the broad learning system

To identify the most appropriate configuration for the Broad Learning System (BLS) classifier, a systematic exploration of hyperparameter was undertaken with the utilization of the training and validation sets. The evaluated parameters comprised the number of mapping groups, mapping nodes per group, enhancement nodes, regularization coefficients, and activation functions. The objective of this study was to identify a configuration that provides strong classification performance while maintaining the computational efficiency that characterizes the BLS framework. The hyperparameter search space considered in this study is summarized in Table 2, where multiple parameter values were systematically evaluated

during the tuning process.

For each configuration, the BLS classifier was trained using DenseNet201 as the feature representations extracted from the augmented training set and evaluated on the validation set. Validation accuracy and macro F1-score were utilized as the primary selection criteria, while the training time was recorded to assess computational efficiency. The most effective configurations obtained during the exploration process are summarized in Table 3.

Table 2. Hyperparameter search space for the broad learning system

Parameter	Values
Number of mapping group	5, 10, 15
Mapping node per group	100, 150, 200
Number of enhancement nodes	100, 300, 500, 700
Regularization coefficient (λ)	10^{-4}, 10^{-3}, 10^{-2}

The findings suggest that multiple configurations achieved equivalent validation performance, suggesting that the DenseNet201 feature representations are highly discriminative and relatively insensitive to moderate variations in BLS hyperparameters. Of these configurations, the model with 5 mapping groups, 150 mapping nodes per group, 500 enhancement nodes, a regularization coefficient of 0.0001, and ReLU activation was selected as the final configuration. This was done since it achieved the highest validation performance while requiring the shortest training time among the top-performing candidates.

The findings suggest that increasing the network width beyond a certain point provides minimal additional value in scenarios where highly informative deep features are already available. Consequently, a relatively compact BLS configuration was sufficient to exploit the DenseNet201 feature space effectively while preserving the computational efficiency advantages of the BLS framework.

Table 3. Top five BLS configurations obtained from hyperparameter exploration

Rank	Config. (MG–MN–EN)	λ	Act.	Test Acc.	Test F1	Time (s)
1	5–150–500	10^{-4}	ReLU	0.9964	0.9935	0.9917
2	5–150–500	10^{-3}	ReLU	0.9964	0.9935	0.9981
3	5–150–500	10^{-2}	ReLU	0.9964	0.9935	1.0198
4	10–100–300	10^{-4}	ReLU	0.9964	0.9935	1.4514
5	10–100–300	10^{-3}	ReLU	0.9964	0.9935	1.4360

Note: MG = Mapping Group, MN = Mapping Node, EN = Enhancement Node, Act. = Activation Function

3.2. Classifier performance comparison

To thoroughly assess the suitability of the Broad Learning System (BLS) for DenseNet201 feature representations, its performance was compared with that of several widely employed classifiers, including Softmax, Logistic Regression (LR), Random Forest (RF), Multilayer Perceptron (MLP), and Support Vector Machine (SVM). All classifiers were subjected to training and evaluation utilizing identical DenseNet201 feature vectors to ensure an equitable comparison. The performance results are presented in Table 4.

Table 4. Performance comparison of classifiers using DenseNet201 feature representations.

Model	Acc.	Precision	Recall	F1
DenseNet201 + SVM	0.9964	0.9975	0.9928	0.9951
DenseNet201 + MLP	0.9928	0.9903	0.9866	0.9884
DenseNet201 + BLS	0.9928	0.9952	0.9855	0.9900
DenseNet201 + LR	0.9892	0.9838	0.9793	0.9815
DenseNet201 + Softmax	0.9857	0.9809	0.9710	0.9741
DenseNet201 + RF	0.9821	0.9884	0.9648	0.9750

Of the evaluated classifiers, SVM exhibited the highest level of classification performance, obtaining an accuracy of 99.64% and a macro F1-score of 99.51%. The proposed DenseNet201–BLS framework achieved an accuracy of 99.28% and a macro F1-score of 99.00%, outperforming Softmax, Logistic Regression, and Random Forest, while demonstrating a level of performance comparable to that of MLP.

The findings suggest that BLS has the capacity to effectively utilize the highly discriminative feature representations extracted by DenseNet201. While SVM achieved the highest classification performance, the proposed DenseNet201–BLS framework attained competitive results, surpassing Softmax, Logistic Regression, and Random Forest while exhibiting performance comparable to that of MLP.

The findings also highlight the effectiveness of DenseNet201 as a feature extractor for the recognition of chili leaf disease. It has been demonstrated that the deep feature embeddings generated by DenseNet201 provide strong class separability, thereby enabling multiple classifiers to achieve high predictive performance. This observation suggests that the quality of the extracted features is a crucial factor in the overall classification performance, while BLS serves as an effective classifier capable of leveraging these rich feature representations.

3.3. Statistical significance analysis

To assess the reliability of the observed performance variations, the bootstrap confidence interval method and McNemar's test were employed. Bootstrap analysis was performed through the repeated resampling of the test set purposely to estimate the uncertainty associated with classification accuracy. Furthermore, McNemar's test was conducted to compare paired predictions generated by the DenseNet201 + Softmax and DenseNet201 + BLS classifiers on the same test samples.

Table 5. Accuracy and bootstrap confidence intervals

Model	Accuracy	95% CI Lower	95% CI Upper
DenseNet201 + Softmax	0.9857	0.9713	0.9964
DenseNet201 + BLS	0.9928	0.9821	1.0000
DenseNet201 + SVM	0.9964	0.9857	1.0000

As demonstrated in Table 5, the findings indicate that all evaluated classifiers achieved consistently high performance, with narrow confidence intervals reflecting stable predictive behavior on the test set. Although SVM achieved the highest accuracy, the confidence intervals of the three models overlapped considerably, suggesting that the observed performance differences are relatively insignificant.

To further investigate whether the enhancement in the BLS classifier in comparison to Softmax is statistically significant, McNemar's test was conducted utilizing the contingency table as presented in Table 6.

Table 6. McNemar contingency table comparing Softmax and BLS

	BLS Correct	BLS Incorrect
Softmax Correct	274	1
Softmax Incorrect	3	1

The contingency table indicates that the BLS classifier correctly classified three samples that were misclassified by Softmax, while introducing one new error. The resulting McNemar test yielded a p-value of 0.625, which exceeded the commonly applied significance threshold of 0.05. Consequently, the observed enhancement in performance of BLS in comparison to Softmax cannot be considered statistically significant at the 95% confidence level.

Although the proposed DenseNet201–BLS framework achieved higher accuracy and macro F1-score compared to Softmax, the statistical analysis suggests that the magnitude of improvement is relatively modest on this dataset. This observation is consistent with the near-ceiling performance that was achieved by all the evaluated classifiers, where only a small number of test samples remained difficult to classify.

3.4. Robustness analysis

To evaluate the robustness of the evaluated classifiers under conditions of limited training data, additional experiments were conducted using 25%, 50%, 75%, and 100% of the available training set. Fig. 5 illustrates the corresponding test accuracies, while Fig. 6 presents the macro F1-scores obtained at each training fraction.

The findings obtained demonstrated distinct learning behaviors among the evaluated classifiers. The DenseNet201 + SVM consistently exhibited the highest levels of performance across all training fractions, maintaining accuracy above 98.5% even when only 25% of the training data were available. Similarly, DenseNet201 + Softmax demonstrated relatively stable performance, achieving accuracies ranging from 98.21% to 98.57% across all training fractions.

In contrast, the proposed DenseNet201 + BLS framework exhibited greater sensitivity to limited training data. When trained using only 25% of the available samples, BLS achieved an accuracy of 92.47% and a macro F1-score of 90.43%, which are substantially lower than those obtained by SVM and Softmax. However, as more training samples became available, the system performance exhibited a steady improvement. At 50% of the training data, the accuracy increased to 96.42%, while at 75% and 100% of the training data the model achieved accuracy of 98.21% and 99.28%, respectively.

These findings suggest that BLS requires a sufficient number of training samples to fully exploit the discriminative feature representations extracted by DenseNet201. In contrast to Softmax and SVM, which demonstrated stability under the condition of reduced data, BLS appears to be more sensitive to the availability of representative training samples. This observation has important practical implications, given that a considerable proportion of real-world agricultural datasets are characterized by limited numbers of labeled images.

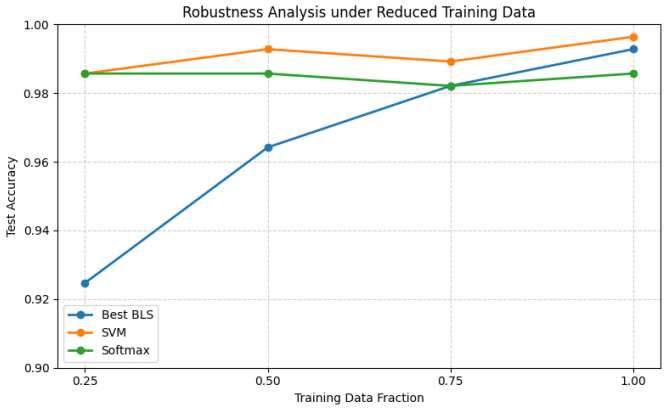

Fig. 5. Robustness analysis of testing accuracy under reduced training data

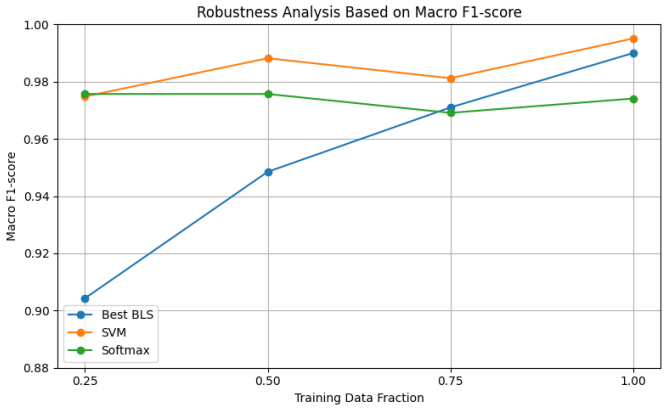

Fig. 6. Robustness analysis of F-1 score under reduced training data

Nevertheless, once an adequate number of training samples is available, the performance gap between BLS and the competing classifiers is significantly reduced. At the full training set, DenseNet201 + BLS demonstrated a performance level that was comparable to that of MLP and substantially outperformed Softmax, Logistic Regression, and Random Forest. Therefore, the robustness analysis highlights a trade-off between data efficiency and computational efficiency. BLS offers extremely fast training while requiring a sufficiently large training set in order to fully realize its classification capability.

3.5. Computational Efficiency Analysis

In addition to the predictive performance, computational efficiency is an important consideration for practical deployment. Therefore, the training time and inference time of the evaluated classifiers were measured and compared, as summarized in Table 7. It should be noted that the reported computational costs correspond exclusively to the classifier

stage following DenseNet201 feature extraction. Image preprocessing and feature extraction were identical for all classifiers and were therefore excluded from the comparison.

Table 7. Computational efficiency comparison of classifiers using DenseNet201 feature representations

Model	Training Time (s)	Inference Time (s)
DenseNet201 + BLS	0.9208	0.0192
DenseNet201 + Softmax	13.7412	0.0690
DenseNet201 + MLP	19.5788	0.0039
DenseNet201 + SVM	23.3008	1.2614
DenseNet201 + LR	31.7013	0.0030
DenseNet201 + RF	48.1433	0.1129

Although SVM demonstrated the highest classification performance, it also exhibited a substantially higher computational cost, requiring 23.30 seconds for training and 1.26 seconds for inference. Similarly, Logistic Regression, MLP, and Random Forest exhibited longer training times in comparison to that of BLS, despite achieving comparable or lower predictive performance. Softmax provided a satisfactory compromise between performance and computational cost; however, its training time remained approximately fifteen times longer than that of BLS.

The inference results reveal a different trade-off. While BLS remained considerably faster in comparison to both SVM and Random Forest during prediction process, it exhibited a lower level of speed compared to Logistic Regression and MLP. This is attributable to the additional computations associated with the feature mapping and enhancement nodes. Consequently, the primary advantage of BLS lies in reducing classifier-training cost rather than minimizing inference latency.

It is imperative to emphasize that these results should not be interpreted as the computational cost of the complete end-to-end classification pipeline. It has been shown that, given the utilization of the DenseNet201 feature extractor by all models, the preprocessing and feature extraction stages contribute equally to the overall computational burden. Therefore, the reported differences in efficiency reflect the relative cost of the classifier component operating on a fixed deep-feature representation.

The results of the study indicate that BLS provides a favorable balance between predictive performance and computational efficiency. Despite not attaining the highest classification accuracy, it exhibits competitive classification performance while requiring substantially less training time than the other evaluated trainable classifiers. Consequently, it emerges as an attractive option when rapid model training is a priority.

3.6. Explainability Analysis

To enhance the interpretability of the proposed framework and verify that the model focuses on disease-relevant regions, Gradient-weighted Class Activation Mapping (Grad-CAM) was employed. Since the Broad Learning System (BLS) operates on feature vectors that has been extracted by DenseNet201, as opposed to operating directly on image pixels, Consequently, Grad-CAM was applied to the DenseNet201 feature extractor, which is responsible for generating the visual representations that are utilized by all evaluated classifiers. Representative examples from the test set are presented in Fig. 7.

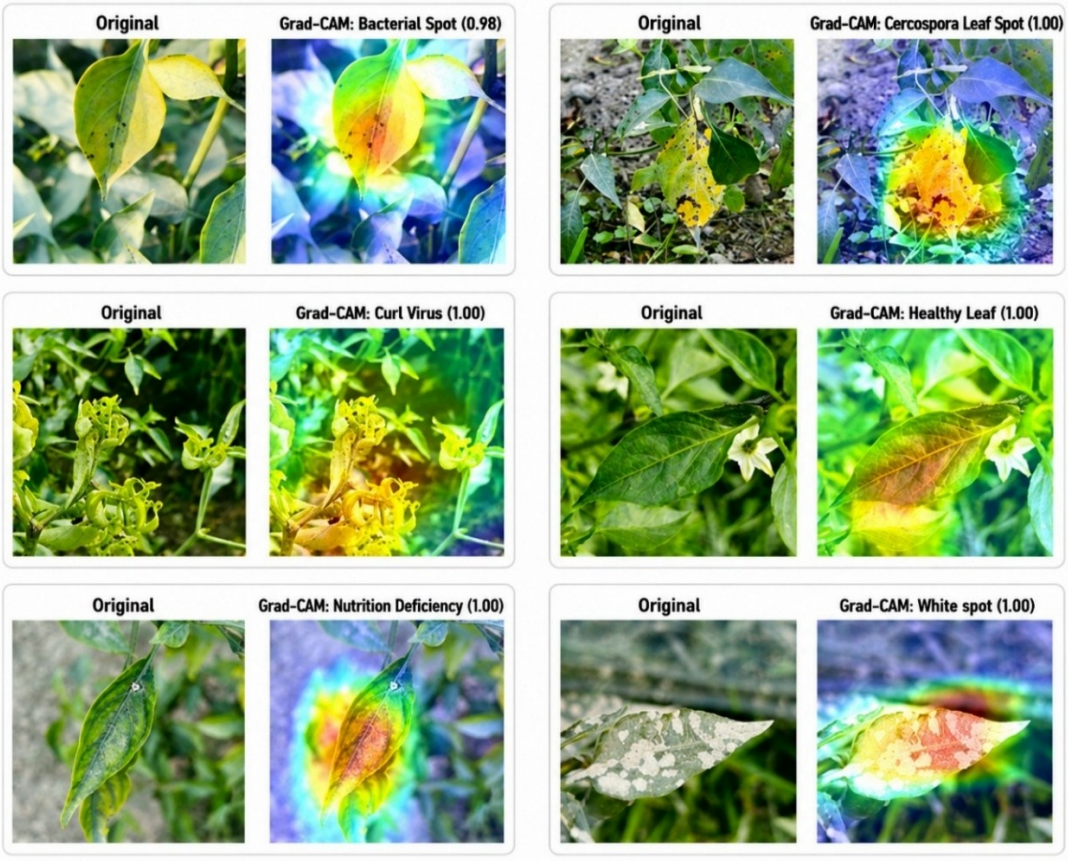

Fig. 7. Grad-CAM visualizations

The Grad-CAM visualizations demonstrate that the DenseNet201 feature extractor consistently focuses on visually meaningful regions associated with disease symptoms. In the context of the Bacterial Spot class, the highlighted regions correspond primarily to dark lesions and surrounding chlorotic tissue. In the Cercospora Leaf Spot class, the attention maps concentrate on necrotic spots distributed across the leaf surface. In the case of Curl Virus, the model focuses on distorted and curled young leaves, which are characteristic symptoms of viral infection. In the Nutrition Deficiency class, the highlighted regions correspond to chlorotic patterns and discoloration along the leaf blade and veins. Similarly, the model employed for White Spot concentrates on the white fungal lesions that predominate on the surface of the affected leaf.

In the case of healthy leaves, the distribution of attention maps is more widespread across the leaf structure and venation patterns, as opposed to being localized to specific lesions. This behavior is consistent with the absence of disease symptoms and suggests that the model relies on overall leaf appearance when identifying healthy samples.

Furthermore, the activation maps indicate that the model primarily attends to leaf regions rather than background vegetation or soil. This observation suggests that the extracted deep features are derived from biologically relevant structures rather than spurious contextual information. The Grad-CAM analysis demonstrates that the representations of DenseNet201 feature capture disease-related characteristics that are consistent with expert interpretation. This suggests that the proposed framework is reliable and interpretable.

3.7. Discussion

The experimental results demonstrate that DenseNet201 feature representations provide a highly discriminative feature space for the classification of chili leaf disease, enabling multiple classifiers to achieve near-perfect performance. Of the evaluated methods, DenseNet201 + SVM exhibited the highest level of classification performance, attaining an accuracy of 99.64% and a macro F1-score of 99.51%. The proposed DenseNet201 + BLS framework also demonstrated high level of performance, reaching an accuracy of 99.28% and a macro F1-score of 99.00%. This suggests that the proposed framework is more effective than Softmax, Logistic Regression, and Random Forest while remaining competitive with MLP.

A comparison of the evaluated framework with those of previous studies indicates that they demonstrate competitive performance. Earlier deep learning approaches for the classification of chili leaf disease, including SECNN (99.12%) [4], DBESeriesNet (99.80%) [8], and the ensemble-based Chili-Net (97.00%) [10], reported high classification accuracy on the datasets relating to chili disease. Other transfer learning approaches based on EfficientNet, VGG16, and MobileNet achieved accuracy ranging from approximately 90% to 94%, dependent upon the characteristics inherent in the dataset [5-9]. The present study is distinguished from the end-to end deep learning approaches by virtues of its investigation of the effectiveness of combining DenseNet201 feature extraction with multiple lightweight classifiers. This provides a broader assessment of classifier suitability within a deep-feature-based classification framework.

The experimental comparison indicates that the classifier choice remains important even in cases where highly discriminative deep features are available. Although SVM achieved the highest predictive performance, the proposed BLS classifier demonstrated a favorable balance between classification accuracy and computational efficiency. BLS required less than one second for training, which is substantially faster than Softmax, Logistic Regression, MLP, SVM, and Random Forest. This efficiency is primarily attributed to the closed-form ridge regression solution employed by BLS, which eliminates iterative gradient-based optimization. Consequently, BLS may be considered an attractive option in scenarios where rapid model retraining is required.

The robustness analysis further demonstrated that BLS is more sensitive to limited training data in comparison to that of Softmax and SVM. In circumstances where only 25% of the training data were available, the performance of BLS significantly decreased. In contrast, both SVM and Softmax demonstrated relatively stable performance. However, as additional training samples became available, the performance gap narrowed substantially. This finding suggests that BLS has the potential to effectively utilize deep feature representations with the availability of sufficient training data; but it may be less suitable for highly data-constrained scenarios.

The Grad-CAM visualizations provide additional insight into the learned feature representations. It has been demonstrated that the attention maps consistently highlighted disease-related regions including lesions, chlorosis patterns, discoloration, and abnormalities in leaf structures, while background vegetation and soil were largely ignored. These observations indicate that DenseNet201 extracts biologically meaningful features that support reliable disease discrimination and increase confidence in the interpretability of the classification results.

Despite the encouraging findings, it is imperative to acknowledge the limitations of the study. Firstly, the experiments were conducted using a single publicly available dataset that had been collected under a specific set of environmental conditions. Despite the implementation of a strict original-image-level data partitioning strategy to prevent augmented-image leakage, the generalization capability of the framework across different geographic regions, imaging devices, and field conditions remains to be validated. Secondly, while BLS substantially reduces classifier training time, it does not provide the highest classification accuracy and introduces additional computational overhead during inference in comparison to Logistic Regression and MLP. It is recommended that future research investigate external validation using independent field datasets. In addition, the framework should be evaluated on larger multi-crop disease datasets. Finally, there is a need to explore lightweight or incremental variants of BLS for practical agricultural monitoring applications.

4. Conclusion

This present study investigated a hybrid framework for the classification of chili leaf disease with the combination of DenseNet201-based deep feature extraction with a Broad Learning System (BLS) classifier. To facilitate a comprehensive evaluation, a comparison was conducted

between the proposed BLS classifier and Softmax, Logistic Regression, Random Forest, Multilayer Perceptron (MLP), and Support Vector Machine (SVM) utilizing identical DenseNet201 feature representations. The experimental findings demonstrated that DenseNet201 generates feature embeddings that are highly discriminative, thereby enabling all of the evaluated classifiers to achieve high classification performance. Among the classifiers evaluated in this study, SVM exhibited the highest level of predictive performance with an accuracy of 99.64% and a macro F1-score of 99.51%. Concurrently, the proposed DenseNet201+BLS framework demonstrated competitive performance with an accuracy of 99.28% and a macro F1-score of 99.00%. This performance outperformed Softmax, Logistic Regression, and Random Forest while remaining comparable to MLP. In addition to its strong predictive capability, BLS exhibited optimal training efficiency, with a training tome of less than one second, as a result of its closed-form ridge regression formulation. The robustness analysis indicated that BLS benefits substantially from increased training data availability, although it is more sensitive to limited-data conditions in comparison to SVM and Softmax. Furthermore, the Grad-CAM visualizations confirmed that the DenseNet201 feature extractor consistently focuses on disease-relevant regions such as lesions, discoloration patterns, and abnormal leaf structures, thereby supporting the interpretability of the classification decisions. The findings suggest that the integration of deep feature extraction with a Broad Learning System provides a competitive and computationally efficient alternative approach for the classification of chili leaf disease. Although SVM achieved the highest predictive performance, BLS offers a favorable balance between classification accuracy and training efficiency. This makes BLS particularly suitable for applications requiring rapid model retraining. Future work will focus on the validation of the proposed framework utilizing independent field datasets and larger multi-crop disease collections, in addition to exploring lightweight and incremental variants of BLS for practical agricultural monitoring systems.

Acknowledgements

This present research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors express their profound gratitude to the contributors of the publicly available Chili Plant Leaf Disease and Growth Stage Dataset, which served as the primary data source for this study. The availability of this dataset has had a considerable impact on research and development in the field of computer vision applications for agricultural disease diagnosis.

References

1. Y. Sambrani, Rajashekarappa and S. Bhairannawar, Chili Disease Detection and Classification Using Various Machine Learning Techniques, Proc. Int. Conf. Applied Intelligence and Sustainable Computing (2023) 1-6.
2. V. K. Velpula, S. Prasad, J. Vadlamudi, S. Gurrala, A. V. Priya and K. Srinidhi, Chilli Leaf Disease Prediction System Using Deep Learning Model, Proc. Int. Conf. Emerging Smart Computing and Informatics (2025) 1-6.
3. Y. Sari, A. R. Baskara and R. Wahyuni, Classification of Chili Leaf Disease Using the Gray Level Co-occurrence Matrix (GLCM) and the Support Vector Machine (SVM) Methods, Proc. Sixth Int. Conf. Informatics and Computing (2021) 1-4.
4. B. N. Naik, R. Malmathanraj and P. Palanisamy, Detection and classification of chilli leaf disease using a squeeze-and-excitation-based CNN model, Ecol. Inform. 69 (2022) 101663.
5. V. K. Pratap and N. S. Kumar, High-precision multiclass classification of chili leaf disease through customized EfficientNetB4 from chili leaf images, Smart Agric. Technol. 5 (2023) 100295.
6. D. Li, C. Zhang, J. Li, M. Li, M. Huang and Y. Tang, MCCM: multi-scale feature extraction network for disease classification and recognition of chili leaves, Front. Plant Sci. 15 (2024).
7. T. R., G. G. Devadhas and T. Y. Satheesha, EGMEO: Exponential Groupers and Moray Eels Optimization Based GoogLeNet for Chili Leaf Disease Classification, Proc. Int. Conf. Next Generation Computing Systems (2025) 1-7.
8. N. N. Bhookya, M. Ramanathan and P. Ponnusamy, Leaf Disease Classification of Various Crops Using Deep Learning Based DBESeriesNet Model, SN Comput. Sci. 5 (2024) 406.
9. M. -, S. Winiarti and A. Pujiyanta, Identification of Chili Plant Diseases Based on Leaves Using Hyperparameter Optimization Architecture Convolutional Neural Network, Int. J. Adv. Comput. Sci. Appl. 15 (2024).
10. S. Bandopadhyay, A. K. Gaula, I. Haider and G. Kumar, Chili-Net: An Approach for Classifying Chili Leaf Diseases Using Deep Neural Networks, Lect. Notes Comput. Sci. (2024) 45-55.
11. P. P. Kerkar, T. Veerakumar, A. D. Rahulkar, M. Kumar Panda and B. Narayan Subudhi, A Dual-Net With Graph Attention Network for Plant Diseases Identification, IEEE Sens. J. 25 (2025) 37075-37086.
12. A. Bhujel, N.-E. Kim, E. Arulmozhi, J. K. Basak and H.-T. Kim, A Lightweight Attention-Based Convolutional Neural Networks for Tomato Leaf Disease Classification, Agriculture 12 (2022) 228.
13. S. R. M, A. Gladston and K. N. H, A Multi-kernel CNN model with attention mechanism for classification of citrus plants diseases, Sci. Rep. 15 (2025) 24047.
14. G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten and K. Q. Weinberger, Convolutional Networks with Dense Connectivity, IEEE Trans. Pattern Anal. Mach. Intell. (2019) 1-12.
15. G. Huang, Z. Liu and L. Van Der Maaten, Densely Connected Convolutional Networks, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 4700-4708.
16. Y. Hou, Z. Wu, X. Cai and T. Zhu, The application of improved DenseNet algorithm in accurate image recognition, Sci. Rep. 14 (2024) 8645.
17. S. Park and T. Suh, Speculative Backpropagation for CNN Parallel Training, IEEE Access 8 (2020) 215365-215374.
18. M. Wojciuk, Z. Swiderska-Chadaj, K. Siwek and A. Gertych, Improving classification accuracy of fine-tuned CNN models: Impact of hyperparameter optimization, Heliyon 10 (2024) e26586.
19. L. Liao, H. Li, W. Shang and L. Ma, An Empirical Study of the Impact of Hyperparameter Tuning and Model Optimization on the Performance Properties of Deep Neural Networks, ACM Trans. Softw. Eng. Methodol. 31 (2022) 1-40.
20. M. A. K. Raiaan, A. Mukta, M. S. Rahman, M. A. Hossain, M. M. Hasan and M. M. Rahman, et al., A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks, Decis. Anal. J. 11 (2024) 100470.

21. B. Azam, D. Kuttichira, P. Sanjeewani, B. Verma, A. Rahman and L. Wang, A Novel Non-iterative Training Method for CNN Classifiers Using Gram-Schmidt Process, Neural Process. Lett. 57 (2025) 27.
22. E. Ayan, Genetic Algorithm-Based Hyperparameter Optimization for Convolutional Neural Networks in the Classification of Crop Pests, Arab. J. Sci. Eng. 49 (2024) 3079-3093.
23. M. Peng, Y. Liu, J. Wang, H. Zhang, X. Li and Z. Zhao, et al., Crop monitoring using remote sensing land use and land change data: Comparative analysis of deep learning methods using pre-trained CNN models, Big Data Res. 36 (2024) 100448.
24. C. L. P. Chen and Z. Liu, Broad Learning System: An Effective and Efficient Incremental Learning System Without the Need for Deep Architecture, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018) 10-24.
25. X. Zhong, S. Duan and L. Wang, An effective and efficient broad-based ensemble learning model for moderate-large scale image recognition, Artif. Intell. Rev. 56 (2023) 4197-4215.
26. M. A. S. Nirob, A. K. M. F. K. Siam, P. Bishshash and M. Assaduzzaman, Chili Plant Leaf Disease and Growth Stage Dataset from Bangladesh, Mendeley Data (2025).
27. W. Zhou, X. Ma and Y. Zhang, Research on Image Preprocessing Algorithm and Deep Learning of Iris Recognition, J. Phys. Conf. Ser. 1621 (2020) 012008.
28. S. Albert, J. Müller, T. Weber, F. Schmidt, M. Klein and A. Becker, et al., Comparison of Image Normalization Methods for Multi-Site Deep Learning, Appl. Sci. 13 (2023) 8923.
29. Y. Huang, R. Li, X. Wei, Z. Wang, T. Ge and X. Qiao, Evaluating Data Augmentation Effects on the Recognition of Sugarcane Leaf Spot, Agriculture 12 (2022) 1-19.
30. P. I. Ritharson, K. Raimond, X. A. Mary, J. Eunice and J. Andrew, DeepRice: A Deep Learning and Deep Feature based Classification of Rice Leaf Disease Subtypes, Artif. Intell. Agric. 11 (2024) 34-49.
31. N. V., Y. G., N. N. B., M. R. and P. P., Empirical Analysis of Squeeze and Excitation-Based Densely Connected CNN for Chili Leaf Disease Identification, IEEE Trans. Artif. Intell. 5 (2024) 1681-1692.
32. W. Zou, Y. Xia and W. Cao, Broad learning system based on driving amount and optimization solution, Eng. Appl. Artif. Intell. 116 (2022) 105353.
33. Y. Chu, Y. Guo, W. Ding, H. Cao and P. Ping, Broad learning systems: An overview of recent advances, applications, challenges and future directions, Neurocomputing 641 (2025) 130337.
34. T. Kavzoglu, Object-Oriented Random Forest for High Resolution Land Cover Mapping Using Quickbird-2 Imagery, in Handbook of Neural Computation, Elsevier, 2017, pp. 607-619.
35. D. N. K. Hardani, I. Ardiyanto and H. A. Nugroho, Decoding brain tumor insights: Evaluating CAM variants with 3D U-Net for segmentation, Commun. Sci. Technol. 9 (2024) 262-273.