J. Bangladesh Acad. Sci. 49(1); 73-85: June 2025 DOI:https://doi.org/10.3329/jbas.v49i1.78209

Journal of Bangladesh Academy of Sciences

Journal homepage: http://www.bas.org.bd/publications/jbas.html

Research Article

CNN-Based crop disease and pest detection systems: Enhancing accuracy and efficiency in agricultural sustainability

Noshin Un Noor[1] and Faria Solaiman[2]

Department of Information and Communication Technology, Bangladesh University of Professionals, Dhaka, Bangladesh

ARTICLE INFO

Article History

Received: 08 December 2024
Revised: 13 Febraury 2025
Accepted: 10 April 2025

Keywords: Convolutional neural network (CNN), diseased crop dataset, graphical user interface (GUI), image classification, local binary pattern (LBP), support vector machine (SVM).

ABSTRACT

This paper highlights the effectiveness of classifying crop diseases and pests using different Convolutional Neural Network (CNN) models, utilizing images of diseased and pest-infested leaves. By combining CNNs with classification techniques like Support Vector Machines (SVM), accuracy improved to an impressive 98-99% across various plant disease categories. CNNs automate feature extraction and classification, outperforming traditional methods in complex tasks. The paper compares Local Binary Pattern (LBP) with a total of 18 CNN architectures, including Residual Network such as ResNet-101, Google Net, DarkNet-19, and others, demonstrating that CNNs consistently exceed 99 % accuracy in datasets for rice, corn, and jute. Our model achieved 99.9 % accuracy for Rice, 99.87 % for Jute, and 99.01% for Corn datasets when utilizing the ResNet-101 and DarkNet-19 CNN models. In agricultural areas like Bangladesh, this emerging method has the capability to completely transform crop disease management by facilitating early identification and prompt response, which would increase crop yields and food security. The work also makes real-time disease prediction accessible to farmers by introducing a Graphical User Interface (GUI). The potential of CNN-based systems to revolutionize precision agriculture, maximize resources, lower expenses, and empower farmers is highlighted in this study.

Introduction

Plant diseases significantly impact global agriculture, reducing crop yields, putting food security at risk and resulting in losses (Islam, 2020). Conventional detection techniques, which are frequently prone to errors, lead to inaccurate diagnoses and losses (Naidu et al., 2021). The accuracy of disease detection is increased by deep learning, particularly Convolutional Neural Networks (CNNs), which outperform conventional techniques (50–70%) with an accuracy of over 90% (Rajendran and Islam, 2017). While Panigrahi et al.'s CNN model achieved 98.78% for maize disease diagnosis, models such as Enhanced K-Nearest Neighbors (EKNN) and EfficientNet achieve 95% accuracy in pest identification (Panigrahi, 2020). The efficiency of detection is further improved by transfer learning and image preprocessing (Srestha, 2020). A Deep Convolutional Neural Network (DCNN) model that classified five diseases and healthy leaves with 96.08% accuracy used a modified Visual Geometry Group 19 (VGG-19)-based transfer learning strategy for rice leaf disease identification. This strategy reduces production losses and lowers rice prices by integrating drone and IoT technology to offer real-time, affordable diagnostics (Latif et al.,

*Corresponding author:<neha68219@gmail.com, fariasolaiman22@gmail.com>

Department of Information and Communication Technology, Bangladesh University of Professionals Dhaka, Bangladesh

2022). Moreover 60% of Bangladesh's workforce works in agriculture, which accounts for one-third of the country's GDP (Hassan, 2021). Crop yields decrease by 10–20% because of poor disease management, even with extensive pesticide use (Bharate and Shirdhonkar, 2017). Farmers have benefited from increased disease diagnosis accuracy brought forth by technology use (Gandhi et al., 2018). Deep learning improves food security and sustainability by providing quick, scalable, and affordable solutions (Falaschetti et al., 2022).

Pests and diseases are major biological risks to rice, corn, and jute, as shown in Fig. 1. Jute has the highest pest-related losses (20%), followed by rice (15%) and corn (10%). Similarly, disease losses consistently affect all three crops, ranging from 10% to 15% (Islam, 2020). These trends highlight the urgent need for integrated crop protection to improve yield stability and food security.

The objective of this study is to create an efficient detection system that improves accuracy in detecting diseases and pest infestations by using multiple CNN models as feature extractors and Support Vector Machine (SVM) as a classifier. A range of datasets of rice, corn, and jute crops with various pest infestations and diseases are gathered for the study, which then pre-processes the data and extracts features for accurate classification. (Hasan, 2019)

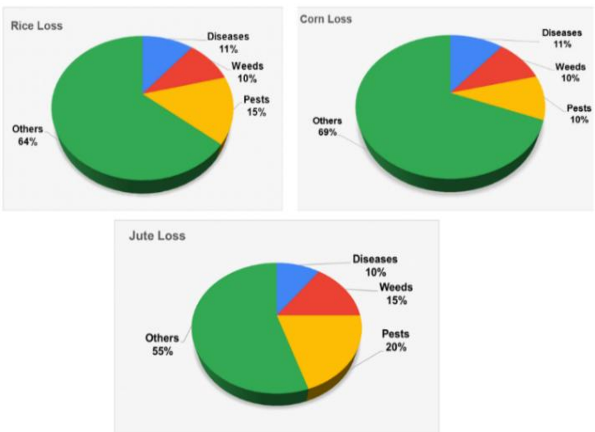


Fig. 1. Yield loss share of pests, diseases, and weeds in Rice, Corn, and Jute crops (Islam, 2020).

In order to identify the top-performing model, it compares 18 CNN architectures, such as Residual Network 101 (ResNet-101) for rice and jute and DarkNet-19 for maize. Additionally, traditional LBP methods are contrasted with CNN-based feature extraction, especially when it comes to detecting pest infestations and disease symptoms based on texture patterns. A Graphical User Interface (GUI) that is easy to use is also designed for real-time validation, making it practical for farmers and agricultural specialists. This study hypothesizes that when it comes to identifying pest infestations and diseases in rice, corn, and jute crops, CNN-based feature extraction combined with SVM classification will perform better than conventional Local Binary Pattern (LBP) techniques.

This approach facilitates precision agriculture, maximizes resource use, and improves large-scale crop monitoring by utilizing CNNs for accurate diseases and pest identification, thus contributing to sustainable farming and worldwide food security.

Materials and Method

In this study, we have separately classified crop diseases for rice and corn, and pest infestation for jute, which are three significant crops in Bangladesh. For this classification, we used images of disease-infected leaves from rice and corn, and pest-infested leaves from jute. LBP and CNN were used as feature extractors, while SVM was used for classification. The workflow of the classification model is shown in Fig. 2.

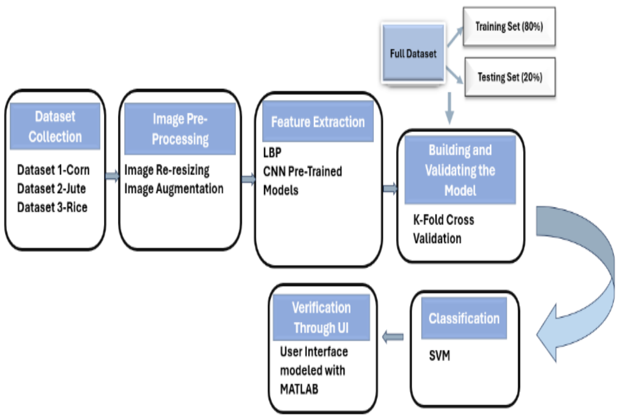


Fig. 2. Workflow of the crop disease and pest detection model using LBP for texture analysis, CNN for feature extraction, and SVM for classification.

The process begins with collecting image datasets. In this case, three datasets were used such as

Dataset 1: Corn images,

Dataset 2: Jute images and

Dataset 3: Rice images.

The collected images undergo pre-processing to prepare them for feature extraction. Images were resized to a uniform size to ensure consistency across the dataset. Images are typically converted to grayscale when using LBP, which extracts texture information based on intensity levels (Sharma et al., 2020). This phase reduces computing complexity while concentrating on texture information. Image augmentation was used to increase dataset diversity for CNN models that had already been trained. Rotation, resizing, scaling, flipping, and brightness adjustments were typically required for this (Hassan et al., 2021). Important features that can be used for categorization are taken out of the images at the feature extraction step. Features that are known to perform well for image classification tasks were extracted using LBP and CNN models that had already been trained (Panchal et al., 2019). Building and validating the classification model came next. Each dataset has been split into five folds, or subsets of which 20% was the testing dataset and 80% was the dataset for training. The K-Fold Cross-Validation technique was employed to validate the model on various subsets of the dataset, enhancing its robustness (Tugrul et al., 2022). The overall performance of the model was determined by averaging the results across all folds. Once the model is trained and validated, the SVM classifier was applied to classify the images based on the extracted features (Agarwal et al., 2020). The final step involves verifying the model's performance through a GUI that is built using MATLAB. This allows users to interact with the model and test its predictions.

Dataset Description

Dataset 1 consists of 4,800 images of corn plants, distributed across four classes: Common Rust, Gray Leaf Spot, Northern Leaf Blight, and Healthy, with each category containing 1,200 images. Dataset 2 contains 660 diseased jute plant images categorized into three classes: Stem Weevil, Jassid, and Aphid - each with 220 images. Dataset 3 comprises 4800 diseased rice plant images divided into four classes: Bacterial Blight, Leaf Blast, Brown Spot, and Tungro, with 1200 images per class. (Fig. 3). Fig. 4. presents sample images from the three datasets used in the experiment. Each image serves as a visual representation of plant health conditions or diseases.

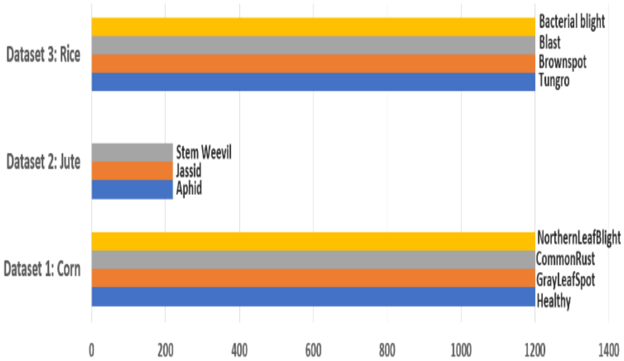
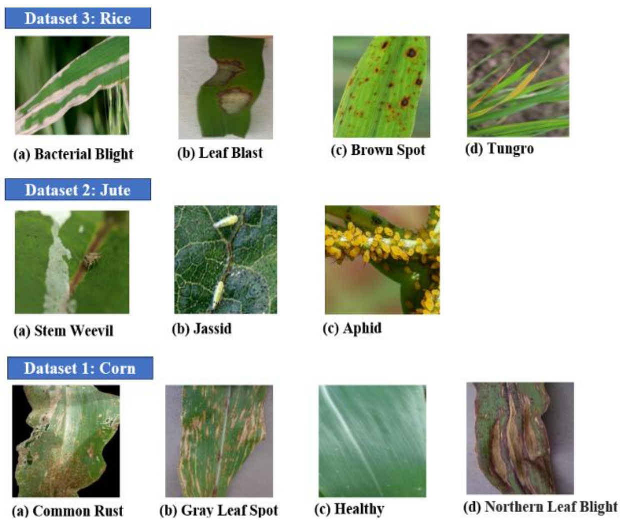


Fig. 3. Datasets: (1) 4,800 corn images across 4 classes, (2) 660 jute images across 3 classes, (3) 4,800 rice images across 4 classes.



Datasets: (Dataset 3: (a) Bacterial Blight, (b) Leaf Blast, (c) Brown Spot, (d) Tungro), (Dataset 2: (a) Stem Weevil, (b) Jassid, (c) Aphid), (Dataset 3: (a) Common Rust, (b) Gray Leaf Spot, (c) Healthy, (d) Northern Leaf Blight).

Image Pre-processing

To enhance the consistency and quality of the dataset, several pre-processing steps were applied before inputting the images into the neural network. This involved applying data augmentation methods including rotation and flipping, as well as scaling every image to a standard size for pre-trained CNNs and turning it to gray-scale for LBP (Noola and Basavaraju, 2022). LBP works with the grayscale pixel intensity values of an image, which typically range from 0 to 255 in an 8-bit grayscale image (Rajeena et al., 2023). In CNNs, pixel values are often normalized by dividing them by 255, transforming the range from 0–255 (for 8-bit images) to 0–1. This normalization can improve model performance and accelerate training. Additionally, the sizes of the images are changed to 224×224 pixels when used with CNN models.

Extraction of Image Features with LBP Operator

The LBP is a simple yet powerful texture operator that labels pixels in an image by thresholding the

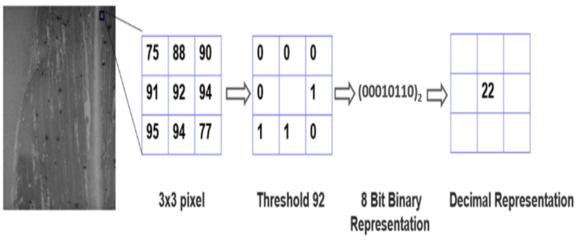


Fig. 5. Feature extraction in LBP.

neighbourhood of each pixel and interpreting the result as a binary number. (Huang et al., 2019).The image is initially transformed to grayscale in order to get the LBP texture description. After choosing an *r*-sized neighborhood around the central pixel for each pixel in the grayscale image, the LBP value is computed and saved in a 2D array that corresponds to the image's dimensions (Fig. 5). When comparing the center pixel to its eight surrounding neighbors, there are 256 possible LBP code permutations. If center pixel's intensity is greater than or equal to a neighbor's, the corresponding value is assigned 1; otherwise, it is set to 0 (Kamilaris and Prenafeta-Boldú, 2018). The comparison proceeds in a consistent clockwise or counterclockwise direction, starting from any adjacent pixel (Sourav and Wang, 2023). An 8-bit binary number is produced by binary comparisons throughout the 3×3 neighborhood and then transformed to a decimal value. By properly utilizing its capacity to acquire fine-grained information, the original LBP implementation performs exceptionally well at capturing complex details inside the image. This feature, however, significantly restricts the algorithm's ability to identify features on scales other than the fixed 3×3 neighborhood (Kaur and Kang, 2015).

Extraction of Image Features with CNN

This network is a highly effective and commonly used tool for extracting image features through deep learning techniques. In CNN-based image feature extraction, the network learns to detect and represent different image elements across multiple layers.

Convolutional layers use small filters (or kernels) to identify fundamental features such as edges, corners, and textures (Latif et al., 2022). The convolution operation includes sliding of the filter across the image and calculating the dot product between the filter and local image patches. The result is a feature map that emphasizes regions containing specific patterns of interest. (Khirade and Patil, 2015).

Table 1. List of 18 pre-trained CNN models with their respective feature layers used for performance evaluation.

Pre-Trained CNN	Feature Layer
AlexNet	fc8
ResNet-18	fc1000
ResNet-50	fc1000
VGG-19	fc8
DarkNet-19	avg1
ResNet-101	fc1000
Inception-v3	predictions
VGG-16	fc8
Xception	predictions
NasNet-Mobile	predictions
DarkNet-53	conv53
InceptionResNet-v2	predictions
DenseNet-201	fc1000
SqueezeNet	pool10
ShuffleNet	node202

GoogleNet	loss3-classifier
MobileNet-v2	Logits
EfficientNet-b0	MatMul

Table 1 lists the pre-trained CNN that was employed in this investigation as well as the fully-connected feature layer that was used to extract the image feature vectors. The CNN model is composed of multiple layers performing convolution, rectified linear units (ReLU), and pooling layers (Islam, 2020). To illustrate, a brief overview of the layered architecture of two popular CNN architectures, ResNet-101 and Darknet-19, is provided below. Fig. 6 displays the ResNet-101 structure (Kulkarni and Shastri, 2024). The design includes an input layer for feeding input images, a convolution layer, a pooling layer, and an output layer, as shown in Fig. 7.

The ResNet-101 architecture as shown in Fig. 6. begins with an input layer where images of a fixed size (typically 224×224 pixels) are fed into the network. The initial convolutional layer uses a large kernel (7×7) for feature extraction, followed by ReLU activation for normalization and non-linearity (Padilla, 2020).The network then passes through a series of stacked residual blocks, organized into four stages, each with multiple blocks. In order to mitigate the vanishing gradient issue, residual blocks in ResNet use skip connections to avoid layers and allow gradients—partial derivatives that define how weights are updated during training—to flow directly backward (Tejaswini et al., 2022).

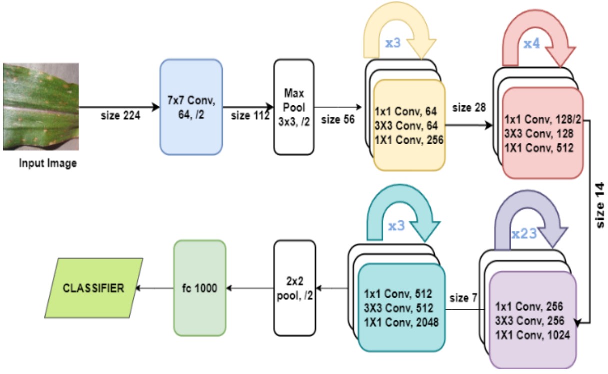


Fig. 6. The layered Architecture of Pre-Trained ResNet-101 CNN model.

Each block contains convolutional layers, batch normalization, ReLU, and a skip connection to facilitate learning residual functions. After the residual blocks, global average pooling aggregates spatial information into a vector, which is then passed through fully connected layers for classification or regression. The network uses ReLU activations throughout and ends with a SoftMax layer for image classification tasks, producing class probabilities. (Mukti and Biswas, 2019)

Darknet-19 is a CNN architecture tailored for real-time object detection. It comprises 19 convolutional layers and 5 max-pooling layers, incorporates Leaky ReLU activations, and is esteemed for its effectiveness and precision in identifying objects within images.

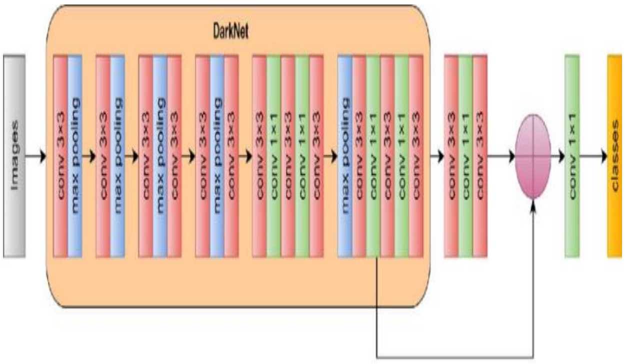


Fig. 7. The layered Architecture of Pre-Trained DarkNet-19 CNN model.

The Darknet-19 network starts with an input layer that accepts an image of size 224×224. It then applies 3×3 convolutional filters with a stride of 1, followed by max-pooling layers to reduce spatial dimensions as shown in fig. 7 (Padilla et al., 2020).To optimize computation, the network alternates between 1×1 and 3×3 filters, ensuring accuracy while reducing computational complexity. After passing through the convolutional layers, the feature maps are flattened and fed into a fully connected layer for the final classification. The output layer is a Softmax layer, which computes class probabilities for the classification task. (Panigrahi et al., 2020).

Classification of Leaf Disease Using SVM

This study utilized two feature extraction techniques: LBP and CNN. For classification tasks, a SVM, a supervised machine learning algorithm, was employed. SVM is effective for both linear and non-linear classification problems, as it identifies the optimal hyperplane that maximizes the margin between classes, ensuring robust generalization to unseen data (Awad and khanna, 2015). In this study, SVM was applied for multi-class classification within the CNN framework using the "one-vs-all" strategy. The fitcecoc function in MATLAB was used to train multiple binary classifiers, facilitating the distinction between different classes. (Shrestha et al., 2020)

In the instance of LBP, texture information is captured by comparing the intensity of a central pixel with its neighbors in order to derive feature vectors from leaf images. After that, SVM classifiers are trained using these LBP features; for complicated, non-linear problems, we used different kernels such as Radial Basis Function (RBF) for producing better accuracy.

Evaluation Metrics

Analyzing various prediction outputs and the metrics that go along with them is necessary to evaluate model performance in classification jobs. True Positives (TP), False Positives (FP), and Precision are important components that show how accurate and reliable the model is in making predictions. TP indicates that the positive class was accurately identified, whereas a True Negative (TN) happens when the model accurately predicts the negative class. On the other hand, a FP happens when a negative occurrence is mistakenly classified as positive, while a False Negative (FN) happens when a positive instance is misclassified as negative (Sun et al., 2022). The accuracy of positive classifications is indicated by precision, which is the ratio of accurately predicted TPs to all positive predictions.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} * 100\%$$

The ratio of correctly predicted true positives to all positive predictions. It measures how accurate positive predictions are.

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

Sensitivity: The ratio of true positive predictions to the total number of actual positive instances. This metric evaluates the model's capability to identify all relevant examples.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1 Score: The precision and recall harmonic mean. It offers an ideal balance between sensitivity and precision.

$$\text{F1 Score} = 2 * \text{Precision} * \frac{\text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}$$

AUC-ROC (Area Under the Receiver Operating Characteristic Curve): This metric computes the area under the ROC curve, which ranges from 0 to 1, in order to assess the effectiveness of classification models. Better model performance is indicated by a higher AUC. Perfect class separation is shown by an AUC of 1, while no discriminative ability is indicated by an AUC of 0.5. When the model's AUC is zero, it frequently misclassifies both positive and negative targets.

Confusion Matrix: A classification model is assessed using a confusion matrix, which is a tabular representation that contrasts the model's predictions with the actual results. Predictions are divided into four sections, with accurate and inaccurate classifications for both positive and negative classes provided. The confusion matrix indicates areas that require development and offers insights into the accuracy of the model.

Performance Evaluation of LBP-based SVM

First, the feature vectors of leaf image of a particular crop (i.e., rice, jute or corn) is extracted using LBP operator. Then, three LBP-based SVM classifiers are implemented separately each for leaf images of rice, jute and corn. For this, we have utilized both RBF and Linear kernels due to their effectiveness in handling complex datasets. The investigation revealed that the RBF kernel consistently provided an average accuracy

30% higher than the Linear kernel. Consequently, the RBF kernel was chosen as the candidate for the grid search process. To optimize SVM, the process is conducted using the grid search method. This in-depth investigation covers differences in the gamma and regularization parameters (C), all of which are essential to attaining the best possible model performance. The following hyperparameters were considered for fine-tuning the SVM as shown in Fig. 7.:

Kernel: [Linear, RBF]

C: [0.1, 1, 10, 100]

Gamma: ['scale', 'auto', 0.1, 1, 10]

As for the gamma parameter, when it is set to 'scale' the algorithm modifies each data point's influence according to the reciprocal of the dataset's feature count. Mathematically, this adjustment is represented as 1 / (n features * X.var ()) were X.var () denotes the variance of the feature values. Alternatively, when gamma is set to 'auto', the algorithm automatically adjusts the gamma parameter based on the inverse of the number of features in the training dataset, calculated as 1 / n features. By making the SVM model learn the unique features of the dataset, these gamma shifts help to ensure a more efficient and accurate classification method.

Table 2 presents the tuning of hyperparameters γ (gamma) and C (regularization parameter) for the RBF kernel in SVM to optimize classification performance. The best performance for corn was observed with γ = scale and C = 1, while γ = scale and C = 10 yielded the best results for jute and rice, indicating that model performance is highly sensitive to appropriate hyperparameter selection.

The values for corn dataset classification using features extracted using LBP are represented in Fig. 8. With a large number of incorrect classifications across multiple classes, the confusion matrix demonstrates the model's difficulty in correctly classifying the data. Particularly for the Healthy and Grey Leaf Spot categories, numerous instances have the wrong labels.

Table 2. Tuning Hyperparameters (γ and C) for RBF Kernel for achieving optimal classification performance using SVM.

Tuning Hyperparameters for RBF Kernel	Regularization Parameter (C)			
Gamma (γ)	C=0.1, γ= scale	C=1, γ= scale *(Best for corn)*	C=10, γ= scale *(Best for jute & rice)*	C=100, γ= scale
	C=0.1, γ= auto	C=1, γ= auto	C=10, γ= auto	C=100, γ=auto
	C=0.1, γ=0.1	C=1, γ= 0.1	C=10, γ= 0.1	C=100, γ= 0.1
	C=0.1, γ= 1	C=1, γ= 1	C=10, γ= 1	C=100, γ= 1
	C=0.1, γ= 10	C=1, γ= 10	C=10, γ= 10	C=100, γ= 10

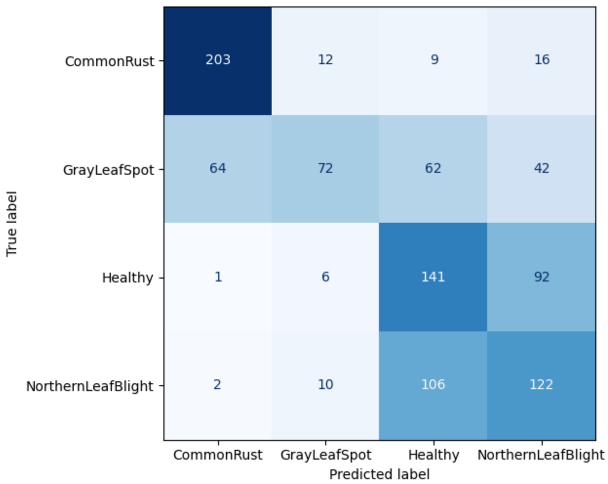


Fig. 8. Confusion matrix of the corn dataset using LBP features, with TP, TN, FP, and FN values.

Performance Evaluation of CNN-based SVM

Feature vectors were extracted separately for three crops rice, jute, and corn using a total of 18 CNN models listed in Table 1. The extracted feature vectors were then used as input to a SVM for classification.

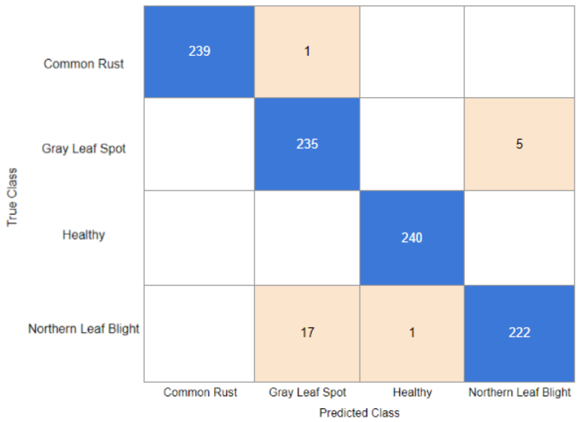


Fig. 9. Confusion matrix of corn dataset using pre-trained ResNet-101 model, with TP, TN, FP, and FN values.

The classification performance of all the 18 CNN models was analyzed using various metrics, including accuracy, precision, sensitivity, specificity, F1 score and AUC. To illustrate the performance of the approach, confusion matrices were generated for each crop using all 18 models. A sample of corn dataset's confusion matrix is given in Fig. 9. The confusion matrix shows misclassifications are minimal, indicating strong overall performance and reliability.

Result

In our demonstration, we extracted feature vectors separately from leaf images of rice, jute, and corn using the LBP operator and 18 different CNN models. These features were then classified using the widely-used SVM. The process was performed separately for each crop's image dataset. For classification, 80% of the images from each class were used to train the SVM, while the remaining 20% were used for testing. We evaluated the performance of both LBP-based and CNN-based SVMs using accuracy, precision, sensitivity, specificity, and F1 score. Finally, a User Interface (UI) was developed for real-time model validation.

Training and Validation of the Models using 5-Folds Cross Validation

Every dataset was divided into five folds, or subsets, with the training dataset comprising 80% and the testing dataset 20% of each fold. Each of the model's k training and evaluation cycles used a different fold as the validation set. The model's overall performance has been determined by averaging the performance across the folds as depicted in fig. 10. For Rice, Corn and Jute datasets the best performing CNN models were respectively Resnet-101 and DarkNet-19.

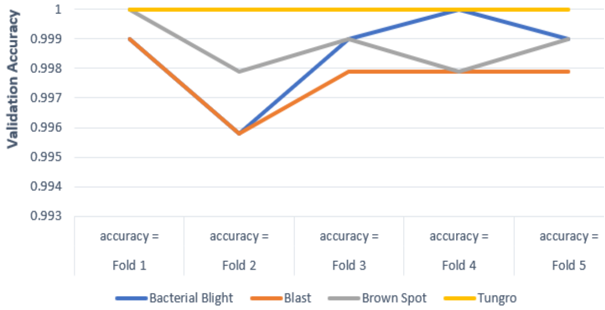


Fig. 10. 5-Fold Cross Validation of Rice Dataset for the model ResNet-101.

Result Evaluation of LBP-based SVM Model

The investigation revealed that the RBF kernel consistently provided an average accuracy 30% higher than the Linear kernel. Consequently, the RBF kernel was chosen as the candidate for the grid search process.

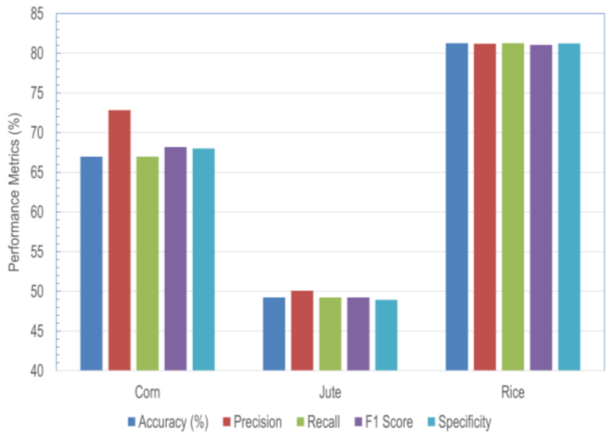


Fig. 11. Performance comparison showing accuracy, precision, recall, f1 score, and specificity of the LBP model.

Despite hyperparameter tuning of the SVM applied to features extracted through local binary pattern, the highest observed accuracy averaged at 81.25% across all datasets. Due to the suboptimal performance of LBP even with optimal SVM parameters, the research shifted to CNN-based feature extraction.

Result Evaluation and Comparison of CNN-based SVM Models

The efficiency of the network in large-scale image recognition tasks is well known, particularly when it comes to precision agriculture. In order to investigate a more reliable method of deriving features and classification with the target of much increased accuracy, a strategic move towards CNNs was made. As we can see from Fig. 12. to Fig. 17. the ResNet 101 model for Jute and Rice Datasets and Darknet-19 model for Corn Dataset fared better than any of the other models, outperforming them all in terms of precision, recall, sensitivity, specificity, f1 score, and validation accuracy. We obtained an accuracy of 99.01% for corn datasets using DarkNet-19 and an accuracy of 99.9% and 99.87% for Rice and Jute datasets respectively using ResNet-101. Figures display the findings following the validation of the models on Datasets respectively. The ResNet-101 model and DarkNet-19 outperformed the current models in every evaluation, as can be seen from the performance comparison shown in these figures.

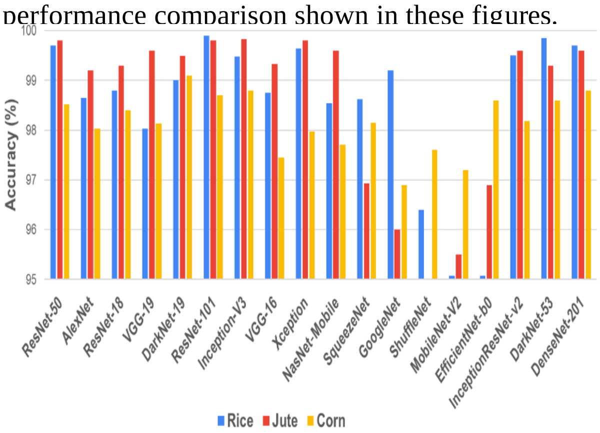


Fig. 12. Accuracy comparison of the pre-trained models considering independent training and validation dataset.

The comparative analysis of CNN models for agricultural datasets highlights the effectiveness of specific architectures for different crops. The Resnet-101 model, known for its deep residual learning frame work, has shown superior performance for Rice and Jute datasets, indicating its robust feature extraction capabilities in diverse conditions. Meanwhile, the Darknet-19 model, with its inception modules, has proven to be the most accurate for the Corn dataset, suggesting that its architecture may be better suited for the particularities of corn's image data. The close performance of ResNet-101 for the Corn dataset also suggests that it is a versatile model that can be considered for various agricultural applications. This information can guide future research and practical applications in precision agriculture, optimizing crop monitoring and management through tailored CNN model selection.

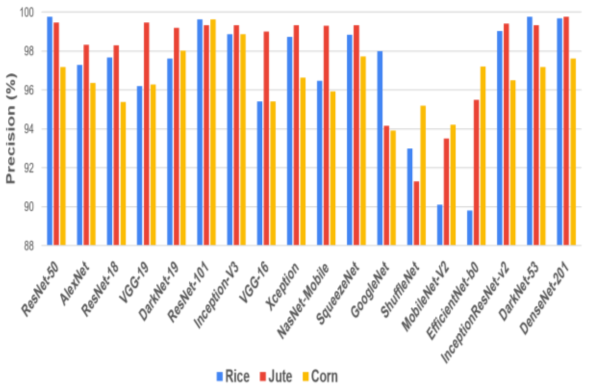


Fig. 13. Precision comparison of the pre-trained models considering independent training and validation dataset.

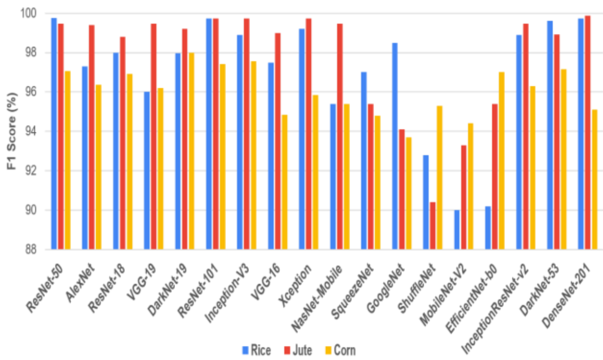


Fig. 14. F1 Score comparison of the pre-trained models considering independent training and validation dataset.

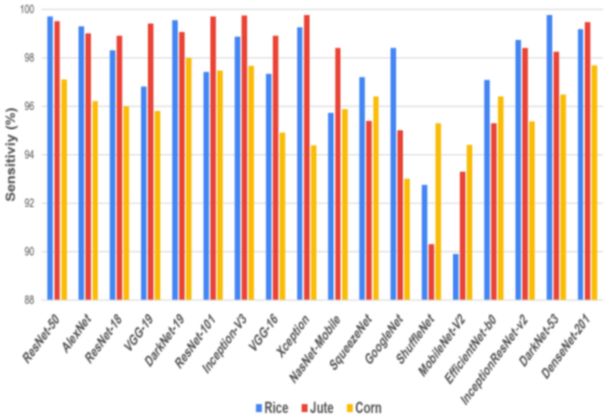


Fig. 15. Sensitivity comparison of the pre-trained models considering independent training and validation dataset.

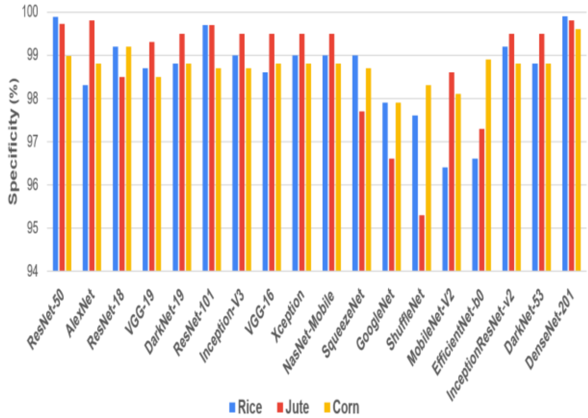


Fig. 16. Specificity comparison of the pre-trained models considering independent training and validation dataset

Furthermore, its adaptability to different crops highlights the potential for scalable solutions in disease detection and overall crop health assessment. Future studies can further refine model architectures to improve accuracy and efficiency in diverse agricultural environments.

Table 3. Performance metrics comparison of CNN models across various studies, including accuracy (A %), precision (P %), sensitivity (S %), specificity (sp %), and F1 score (F %).

Author	Model	Total Image	Class	A %	P %	S %	Sp %	F %
Pangrahi et al.	CNN	3823 (Corn)	3	98.7	98.6	98.9	-	98.7
Daneshwari et al.	EKNN	3820 (Corn)	4	99.8	-	-	99.6	-
Rajeena et al.	Efficient Net	3188 (Corn)	4	98.8	88	-	98	-
Sourav and Wang	CNN	1535 (Jute)	4	-	95	95	-	95
Latif et al.	DCNN	2187 (Rice)	6	96	96.2	96.1	99.2	96.1
This Paper	Resnet101 + SVM	4,800 (Rice)	4	99.9	99.6	99.6	99.7	99.7
This Paper	Resnet101 + SVM	660 (Jute)	3	99.8	99.4	99.7	99.7	99.7
This Paper	Darknet19 + SVM	4,800 (Corn)	4	99	98	98	98.8	98

Comparison Analysis of Different Studies

A comparative analysis between this paper and other relevant studies demonstrates that our modified CNN + SVM model exhibited superior performance. Specifically, our model achieved 99.9% accuracy for Rice, 99.87% for Jute, and 99.01% for Corn datasets when utilizing the ResNet-101 and DarkNet-19 CNN models.
Table 3 compares the performance metrics of various CNN models used in prior studies and the proposed models in this paper. The proposed ResNet101 + SVM and Darknet19 + SVM models outperformed existing methods across all metrics, achieving accuracy rates up to 99.9% and F1 scores of 99.7%, demonstrating superior precision, sensitivity, and specificity in crop disease classification for rice, jute, and corn.

Prediction in Real Time with Graphical User Interface

A GUI application for disease detection with pre-trained models has been created through the “Browse” button. The “Detect” button displays the predicted label to the users. Using the MATLAB R2021a version, users can explore and choose an image for disease classification using the GUI. The “Train” button causes the model to be trained on a predefined dataset. Training outcomes, such as accuracy and a confusion matrix, are then displayed. It applies the trained model to extract features and predict the disease class, displaying overall accuracy through the “Accuracy” button. Users can then input new test data using the “Browse” button.

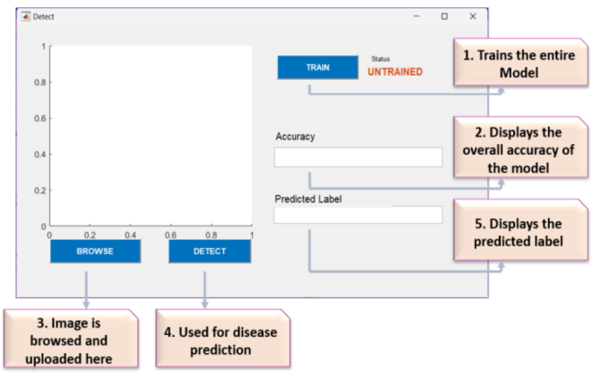


Fig. 17. Labelling of UI Components.

The disease detection system can be interacted with through an easy-to-use interface thanks to the GUI’s text labels, axes, and buttons. The deep learning model’s information is stored by the code using global variables.

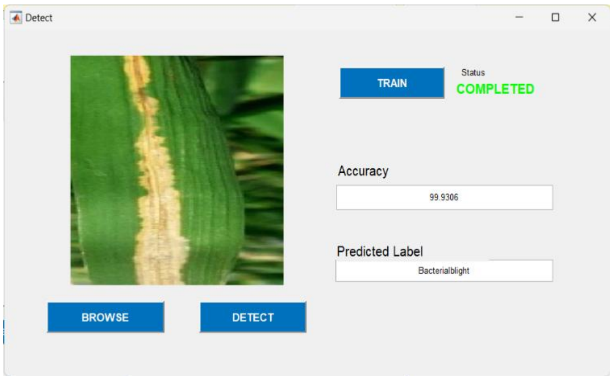


Fig. 18. Validation of the model using UI.

A sample image from rice dataset have been uploaded for prediction using pre-trained ResNet-101 as shown in Fig. 18. The model accurately predicted the label of the given image as “Bacterial Blight” with an accuracy of 99.9% respectively.

Discussion

With distinct advantages over more conventional techniques like LBP-SVM, this study highlights the potential of CNNs in the detection of plant diseases and pests. ResNet-101 and DarkNet-19 singled out for their performance across the rice, jute, and corn datasets out of the 18 CNN models that were assessed in the study. Accurate crop monitoring depends on these models' ability to recognize intricate leaf patterns, discolouration, and insect structures. Furthermore, their applicability extends beyond disease detection to pest identification due to their capacity to identify structural characteristics of pests, such as shape and segmentation patterns.

By beating conventional deep learning models in terms of accuracy and classification measures, a comparison with previous research showed how successful the modified CNN-SVM method was. Computational demands are still a problem, though, particularly in settings with restricted resources. Looking ahead, there are several avenues for improvement and expansion. Efforts to optimize the model for lower computational costs such as portable mobile phones without compromising accuracy will make the technology more accessible.

Conclusion

With an accuracy of up to 99%, this study shows how CNN-based models can revolutionize the identification of plant diseases and pests. These models provide an efficient instrument for precision agriculture by precisely detecting pest structures, textural alterations, and leaf discolouration. The incorporation of these models into simple user interfaces may enable early diagnosis, lowering crop losses and enhancing yield management in Bangladesh, where agriculture is essential to the country's economy and food security. Farmers in rural areas need this information because they experience limited resource availability together with restricted access to knowledge.

Further development of the technology should tackle current computational constraints in order to expand accessibility and enable national adoption. Deep learning developments combined with the use of mobile and edge devices may close this gap and make these technologies usable even in settings with limited resources. The advancement of agricultural technologies would help sustain farming practices while securing national food supply and increasing resource effectiveness throughout Bangladesh's agricultural sectors.

Acknowledgement

The authors are grateful to Bangladesh University of Professional's Department of Information and Communication Engineering and and their supervisor Dr. Abul Kalam Azad for providing the tools and assistance needed to complete this research. They also acknolwedge the support of colleagues and mentors, who provided direction and encouragement throughout the study process. Lastly, the authors extend their heartfelt gratitude to friends and family for their unwavering support and encouragement, which served as a const ant source of motivation throughout this journey.

Author Contributions

All authors have a substantial contribution to the research and is responsible for the content. Noshin Un Noor helped conceptualize the study, supervised the research process, coparative analysis and edited the manuscript. The methodology and result analysis was developed by Faria Solaiman. Authors dealt with validation of results. Both authors contributed as data curators, performed formal analysis and were involved in the investigation and interpretation of the findings. The authors drafted the original manuscript, prepared the visualizations, managed necessary resources and have read and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declare that they have no conflicts of interest regarding the publication of this article. All authors are affiliated with the Bangladesh University of Professionals, and no financial support was received for this research.

References

Agarwal M, Gupta SK, and Biswas KK. Development of efficient CNN model for tomato crop disease identification. *Sustain. Comput. Inform. Syst.* 2020; 28(1): 100407.

Naidu A, Majji V, and Kumaravelan G. A review of machine learning approaches in plant leaf disease detection and classification. *Proc. IEEE 3rd Int. Conf. Integr. Commun. Technol. Virtual Mobile Networks (ICICV*), 2021.

Awad M, and Khanna R. Support vector machines for classification. *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers* 2015:39-66.

Bharate AA, and Shirdhonkar MS. A review on plant disease detection using image processing. *Proc. IEEE Int. Conf. Intelligent Sustainable Systems (ICISS),* 2017.

Falaschetti L, Manoni L, Di Leo D, Pau D, Tomaselli V, and Turchetti C. A CNN-based image detector for plant leaf diseases classification. *HardwareX*, 2022; 12:e00363.

Gandhi R, Panda K, Sahoo AK, and Das H. Plant disease detection using CNNs and GANs as an augmentative approach. Proc. IEEE Int. Conf. Innovative Research Development (ICIRD), 2018.

Hassan SM, Maji AK, Jasiński M, Leonowicz Z, and Jasińska E. Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics*, 2021; 10(12): 1388.

Huang S, Liu W, Qi F, and Yang K. Development and validation of a deep learning algorithm for the recognition of plant disease. *Proc. IEEE 21st Int. Conf. High Performance Comput. Commun. IEEE 17th Int. Conf. Smart City IEEE 5th Int. Conf. Data Science Systems HPCC/SmartCity/DSS*, 2019; 1951-1957.

Islam MT. Plant disease detection using CNN model and image processing. *Int. J. Eng. Res. Technol.*, 2020; 9(10):291-297.

Kamilaris A, and Prenafeta-Boldú FX. A review of the use of convolutional neural networks in agriculture. *J. Agric. Sci.*, 2018; 156(3): 312-322.

Kaur R, and Kang SS. An enhancement in classifier support vector machine to improve plant disease detection. *Proc. IEEE 3rd Int. Conf. MOOCs Innov. Tech. Educ. (MITE*), 2015.

Khirade SD, and Patil AB. Plant disease detection using image processing. *Proc. IEEE Int. Conf. Computing Communication Control Automation*, 2015.

Kulkarni P, and Shastri S. Rice leaf diseases detection using machine learning. *J. Sci. Res. Technol.* 2024:17-22.

Latif G, Abdelhamid SE, Mallouhy RE, Alghazo J, and Kazimi ZA. Deep learning utilization in agriculture: Detection of rice plant diseases using an improved CNN model. *Plants*, 2022; 11(17): 2230.

Hasan MJ. Rice disease identification and classification by integrating support vector machine with deep convolutional neural network. *Proc. IEEE 1st Int. Conf. Advances Science Engineering Robotics Technol. (ICASERT)*, 2019.

Mukti IZ, and Biswas D. Transfer learning based plant diseases detection using ResNet50. *Proc. IEEE 4th Int. Conf. Electrical Info. Commun. Technol. (EICT)* , 2019.

Noola DA, and Basavaraju DR. Corn leaf image classification based on machine learning techniques for accurate leaf disease detection. *Int. J. Electr. Comput. Eng.*, 2022; 12(3): 2509-2516.

Padilla DA, Pajes RAI, and De Guzman JT. Detection of corn leaf diseases using convolutional neural network with OpenMP implementation. *Proc. IEEE 12th Int. Conf. Humanoid Nanotechnol. Inf. Technol. Commun. Control Environ. Manage. (HNICEM*), 2020.

Panchal P, Raman VC, and Mantri S. Plant diseases detection and classification using machine learning models. *Proc. IEEE 4th Int. Conf. Computational Systems Information Technol. Sustainable Solutions (CSITSS*), 2019.

Panigrahi K, Panda K, Sahoo AK, and Das H. A CNN approach for corn leaves disease detection to support digital agricultural system. *Proc. IEEE 4th Int. Conf. Trends Electron. Informatics*, 2020; 48184.

Rajeena PP, Fathimathul, Aswathy SU, Moustafa MA, and Ali MAS. Detecting plant disease in corn leaf using approach. Electronics, 2023; 12(8): 1938.

Rajendran G, and Islam R. Plant clinics in Bangladesh: are farmers losing less and feeding more? *CABI Case Study*, 2017; 19.

Sharma P, Singh Berwal YP, and Ghai W. Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. *Inf. Process. Agric.*, 2020; 7(4): 566-574.

Shrestha G, Das M, and Dey N. Plant disease detection using CNN. *Proc. IEEE Appl. Signal Process. Conf. (ASPCON)*, 2020.

Sourav MSU, and Wang H. Intelligent identification of jute pests based on transfer learning and deep convolutional neural networks. *Neural Process. Lett.*, 2023; 55(3): 2193-2210.

Sun X, Li G, Qu P, Xie X, Pan X, and Zhang W. Research on plant disease identification based on CNN. *Cognitive Robotics*, 2022; 2: 155-163.

Tejaswini P, Singh P, Ramchandani M, Rathore YK, and Janghel RR. Rice leaf disease classification using CNN. *IOP Conf. Ser. Earth Environ. Sci.*, 2022; 1032(1):012017.

Tugrul B, Elfatimi E, and Eryigit R. Convolutional neural networks in detection of plant leaf diseases: A review. *Agriculture*, 2022; 12(8): 1192.