Electronic Letters on Computer Vision and Image Analysis 24(1):179-194, 2025

Enhanced Bell Pepper and Grape Leaf Disease Classification Using a Depthwise Separable VGG19-Capsule Network

Midhun P Mathew[1,2], Sudheep Elayidom[3] , Jagathy Raj V P[4] and Abubeker K M[+]

[1] *Research Scholar CS, SOE-Cochin University of Science and Technology, Cochin, Kerala, India*
[2] *Assistant Professor, Amal Jyothi College of Engineering (Autonomous), Kanjirappally, Kerala, India*
[3] *Senior Professor, SMS-Cochin University of Science and Technology, Cochin, Kerala, India*
[4] *Associate Professor, Amal Jyothi College of Engineering (Autonomous), Kanjirappally, Kerala, India*

Received 18 July 2024; revised 7 January 2025; accepted 4 June 2025

Abstract

The agriculture industry and food security require accurate and timely identification of plant leaf diseases. This research presents a unique hybrid deep learning model that incorporates Depthwise Separable VGG19 with Capsule Networks (VGG19-CapsNet) and ensemble activation functions of Leaky ReLU and GELU to classify diseases on the bell pepper and grape leaves. The model employs depthwise separable convolutions to decrease the workload and maintain spatial hierarchies, while Capsule layers augment spatial perception and resilience to intra-class variability. A modified Adamax optimizer is utilized to enhance convergence stability. The framework is trained and evaluated on a hybrid dataset consisting of PlantifyDr, PlantVillage, Sravanneeli, and a custom dataset, along with the other datasets, which were boosted using various augmentation strategies. The performance is remarkable in validation process; for bell pepper, 98.11% accuracy, 98.63% precision, 97.32% recall, 98.49% specificity, and 98.18% F1 Score is achieved. For grape leaves, 98.44% accuracy, 98.46% precision, 98.20% recall, 97.40% specificity, and 98.46% F1 Score is achieved. This supports real-time use as edge solutions in agricultural settings. The architecture has a significantly higher accuracy and generalizability, greater computational efficiency, and lower overall system strain than the other CNN and hybrid models.

Key Words: Computer Vision, Image Analysis, Pattern Recognition Image segmentation, 3D Reconstruction, Active Vision, Tracking, Video and Image Sequence Analysis.

1 Introduction

The field of plant pathology fundamentally changed with the application of deep learning (DL), and artificial intelligence (AI) to automate the task of leaf disease classification during the preliminary stages of precision agriculture. This research aims to enhance the existing DL framework to achieve higher accuracy in classifying diseases in bell pepper and grape leaves, which pose significant risks to sustainable crop and economic health. Some of the major challenges of the existing model include high computational cost, loss of spatial hierarchical

Correspondence to: *kmabubeker82@gmail.com*

Recommended for acceptance by Angel D. Sappa
ELCVIA ISSN:1577-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

information, overfitting of sparse datasets, lack of robust feature representation, and difficulty in handling intra-class variations [1, 2]. Traditional convolution neural networks (CNNs), mainly deep convolutional networks (DCNN), have vast numbers of parameters, thereby causing heavy computational complexity and reduced processing speed. This cannot be applied in real-time in agricultural environments, and the processing is slow because of resource constraints on the processing devices. Standard CNNs cannot retain detailed spatial relations between features, especially when applying the model to image data containing fine details in regions such as disease-infected leaf areas. Therefore, they may misclassify subtle variations, as in distinguishing between similar diseases. The leaf disease datasets are usually small in this case, specifically for plant details like bell pepper and grape. Existing methods rely heavily on raw convolutional filters, which may not effectively capture complex feature hierarchies. Disease symptoms can vary widely, and some existing systems fail to generalise well to unseen samples due to inadequate feature representations. Leaf diseases can present diverse symptoms even within the intra-class variability, and current approaches struggle to generalise across such variations effectively, leading to lower classification accuracy. This research addresses the challenges farmers and agricultural industries face in effective disease management strategies in modern agriculture. Incorporating deep learning and computer vision (CV) technologies has greatly enhanced the identification of plant diseases. VGG19captures multi-level features through depth-wise separable convolutions (DWSC). Nevertheless, the current models fail to adapt to the distinguishing features of diseases for bell peppers and grapes. Different types of activation functions, for instance, have a profound influence on model performance. Yet, the conjunction of Leaky ReLU and GELU in plant pathology is completely undertheorised. This VGG19 framework harnesses ensemble activation to improve feature extraction by enabling the exploitation of both local and global disease patterns.

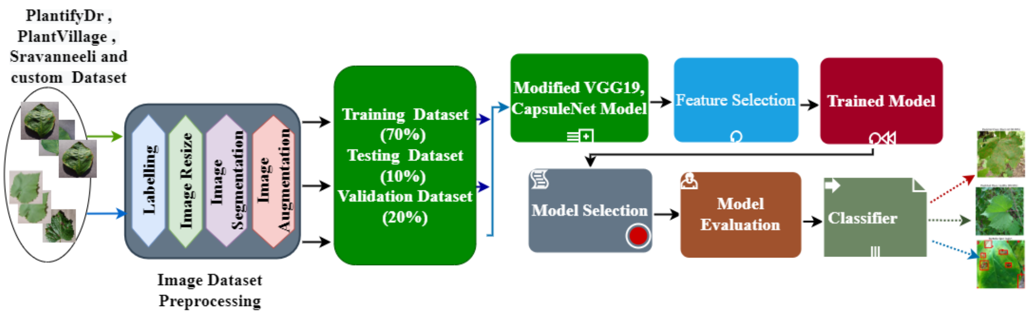


Figure 1: The architecture of Depth-wise Separable VGG19 and Capsule Network is deployed in a 128-core NVIDIA Jetson Nano single-board GPU computer.

Figure 1 illustrates the workflow of the plant leaf disease classification framework designed specifically for bell pepper and grape leaves. It starts with merging datasets acquired from various sources PlantifyDr [3], PlantVillage [4], Sravanneeli [5], and a custom one. From these datasets, a set of image preprocessing pipelines consisting of labelling, resizing, enhancement, segmentation, and augmentation is performed. These steps help in uniformity in the input dimensions and improve image quality, capturing greater diversity for the model to generalise better. After the preprocessing step, the data is split into three chunks: training data 70%, validation data 20%, and testing data 10%. The modified VGG19-CapsuleNet architecture model is provided with the split and processed data beforehand. The architecture merges DWSC with capsule layers; spatial hierarchies are captured while neutralising associated work needed, streamlining the model.

Following the construction of the model, the process continues with feature selection, model selection, and evaluation. These processes guarantee that the most pertinent attributes and classification features are retained. At the same time, the model that meets the performance requirements is selected based on the metrics defined previously, which include calculating accuracy, precision, recall, and F1-score. The model is then deployed into production for classification, with the final result being correctly identified disease labels for the grape leaves and bell pepper leaves.

Capsules preserve spatial hierarchies and inter-relationships among features, which remains particularly useful in leaf disease classification scenarios, as subtle spatial patterns define disease characteristics. Coupling Capsule Networks with the VGG19 backbone enhances spatial dependency retention during better feature representations and more precise classification of diseases for finer symptom variations. An ensemble activation strategy combining features across different layers enhances the diversity and robustness of learned features. This avoids overfitting small datasets because it uses an ensemble of learned patterns rather than relying too much on the activations of particular layers. Depth-wise separable convolutions and capsule layers can efficiently provide a multi-scale feature extraction mechanism, essential for capturing fine and broad disease features that would help the model deal well with intra-class variation. This manuscript discusses recent works concerning deep learning for leaf-based plant diseases with particular attention to CNNs, capsule networks, and other hybrids in Section 2. The ensemble of modified VGG19 capsule networks and ensemble activation functions designed for feature representation and classification enhances the detail provided in section 3. In Section 4, the rest of the experimental setup and how the data was trained and tested were discussed, analysed and compared with other existing models. The concluding section outlines the most important findings from the research, noting in particular the model's ability for real-time edge deployment on the NVIDIA Jetson Nano, and offers suggestions for future work and possible enhancements.

2 Related Work

M. Bhagat et al. [6] developed a CNN-based model for automatic detection of bacterial spots in bell pepper plants, which shows early recognition of plant diseases to improve agricultural productivity in developing countries. Zhang et al. [7] combined Inceptionv3 with ResNet50 into a unified CNN architecture that outperformed standard CNNs by adding data augmentation, early multi-network stopping, and increasing generalisation and weakening overfitting. A recent study by Diana Andrushia et al. [8] studies a grape leaf disease determination problem using convolutional capsule networks. This novel approach preserves spatial relationships across different features with much variation in the datasets utilised. Wei, H.P. et al. [9] show how transformer-based methods outperform CNNs by 4-6% in visual style transfer tasks, demonstrating better feature retention and style application. Their methodology integrates patch-based input processing, allowing for finer granularity in feature extraction and improving overall model interpretability. Zhao et al. [10] achieve state-of-the-art performance in self-supervised monocular depth estimation and visual odometry, reducing error rates by 20% compared to previous CNN models with data augmentation. Chundi et al. [11] report a 30% reduction in power consumption and a 40% increase in processing speed for FPGA-accelerated binary neural network training. Their methodology focuses on computation-in-memory techniques, significantly reducing latency and increasing throughput. In the agricultural domain, Alirezazadeh P. et al. [12] report that integrating attention mechanisms increases classification accuracy by 5-7% for plant disease detection, enhancing sensitivity to subtle leaf features. The methodology employs channel and spatial attention to amplify critical features while suppressing irrelevant background noise. Nawaz, M. et al. [13] achieved 97% accuracy in plant disease classification using VGG-19-based Faster-RCNN, significantly outperforming baseline CNNs by 8%. Their approach leverages region proposal networks (RPN) to locate diseased areas precisely, optimising detection accuracy. Parakh et al. [14] apply CNNs to detect bell pepper diseases, highlighting the practical application of convolutional networks in precision agriculture. Bhagat M. et al. [15] cla[3] ssify bell pepper leaf diseases using CNN architectures, reinforcing the role of deep learning in disease detection. Kundu, N. et.al [16] conduct comparative analyses of deep learning models for bell pepper disease classification, offering insights into the performance of different approaches. Jiang, F. et al. [17] apply deep learning and SVM techniques to classify rice leaf disease, broadening agricultural applications. Kurmi, Y. et al. [18] classify leaf images for crop disease detection, demonstrating the applicability of CNNs to diverse datasets. Thakur A. et al. [19] explore transfer learning across different bell pepper disease detection models, improving model generalisation. Altan, G. [20] evaluates Capsule Networks for plant leaf disease classification, highlighting the advantages of capsule-based architectures in capturing spatial hierarchies. Ye M. et al. [21] propose a lightweight VGG-16 model for remote sensing image classification, illustrating the potential of compact architectures. Kumar et al. [23] systematically review deep learning techniques for plant disease detection, consolidating various approaches. Dai M et al. [24] specified a lightweight GoogLeNet modified for real-time field testing that solved high computational needs. In addition, transfer learning and data augmentation approaches improve model performance, solving issues of overfitting and low training data. This research proposes a hybrid deep learning model focusing on data augmentation to solve these problems and increase accuracy.

3 Methodology

The architecture of the proposed model consists of five sequential blocks based on the classical VGG19 structure. All DWSC layers of each block significantly lower the computation load while retaining the extraction of spatial features. The batch renormalisation with convolutional layers is used to improve the stability of training and generalisation, so this architecture is core to VGG19's conception, while a leaner design tailored for edge devices. To reduce the chances of overfitting, the model is trained with a learning rate of 0.001 and a dropout of 40%. Also used ensemble activation that improves the output of a single neuron by merging several nonlinear activation functions, which ensures that learning complex decision boundaries is done more easily. The capsule layer with squash activation is added to control spatial and hierarchical relationships among features. This helps retain the essential parts and poses needed in images of leaves with diseases for the network. Furthermore, the system applies a changed version of the Adamax optimiser, which speeds up convergence and stabilises it by adjusting the learning rate based on the gradient moments. Additional regularisation techniques also improve the strength and dependability of the model, making it stronger for rapid deployment in real-time agricultural settings. Small datasets and complex intra-class variations make achieving classifiability in leaf disease classification difficult. These are leveraged to draw diverse patterns learned by each to make a more accurate and generalisable model. It enhances the robustness of the classification network against small biases in the datasets and results in a more stable output for each disease class. Incorporating the Capsule Network in the ensemble enhances its capability to learn and tackle the spatial complexity of leaf diseases. Hence, by getting features of the Capsule Network into the activation of the ensemble, it enjoys a rich set of hierarchical features, which enhance its capacity to generalise over a wide array of different disease patterns. Combining VGG19's depthwise separable convolutions with the spatially aware capsule network would make the proposed model even more effective in closely relating leaf diseases.

In Figure 2, VGG19-Capsule network with ensemble activation (Leaky ReLU & GELU) classifies bell pepper and grape leaf diseases using 256×256 input images. The model employs DWSC in five blocks (filters: 64, 128, 256, 512) with batch normalisation, max-pooling, and dropout for feature extraction. Leaky ReLU prevents dead neurons, while GELU ensures smooth activation. A Conv2D layer (64 filters, 1×1 kernel) follows each block. The capsule layer with dynamic routing captures hierarchical and spatial relationships, using the squash activation function to normalise outputs and represent feature probabilities. AveragePooling2D downsamples feature maps before flattening into a 1D vector for further processing. The dense layers (4096 neurons) employ ensemble activation, L2 regularization, and dropout (0.6, 0.4) to enhance generalisation. The softmax activation function refines features for accurate classification. Combining capsule routing, ensemble activation, and regularization ensures robust and precise plant disease detection. The bell pepper and grape leaf dataset was created by merging publicly available data with proprietary data collected from the field. Concerning publicly available pictures, image data augmentations such as rotation, flipping, colour adjustments, and gamma correction were utilised to increase the volume of data and improve model performance. Afterwards, the dataset was partitioned into 70% for training purposes, 20% for validation, and 10% for testing so that they could be trained on a Jetson Nano GPU platform.

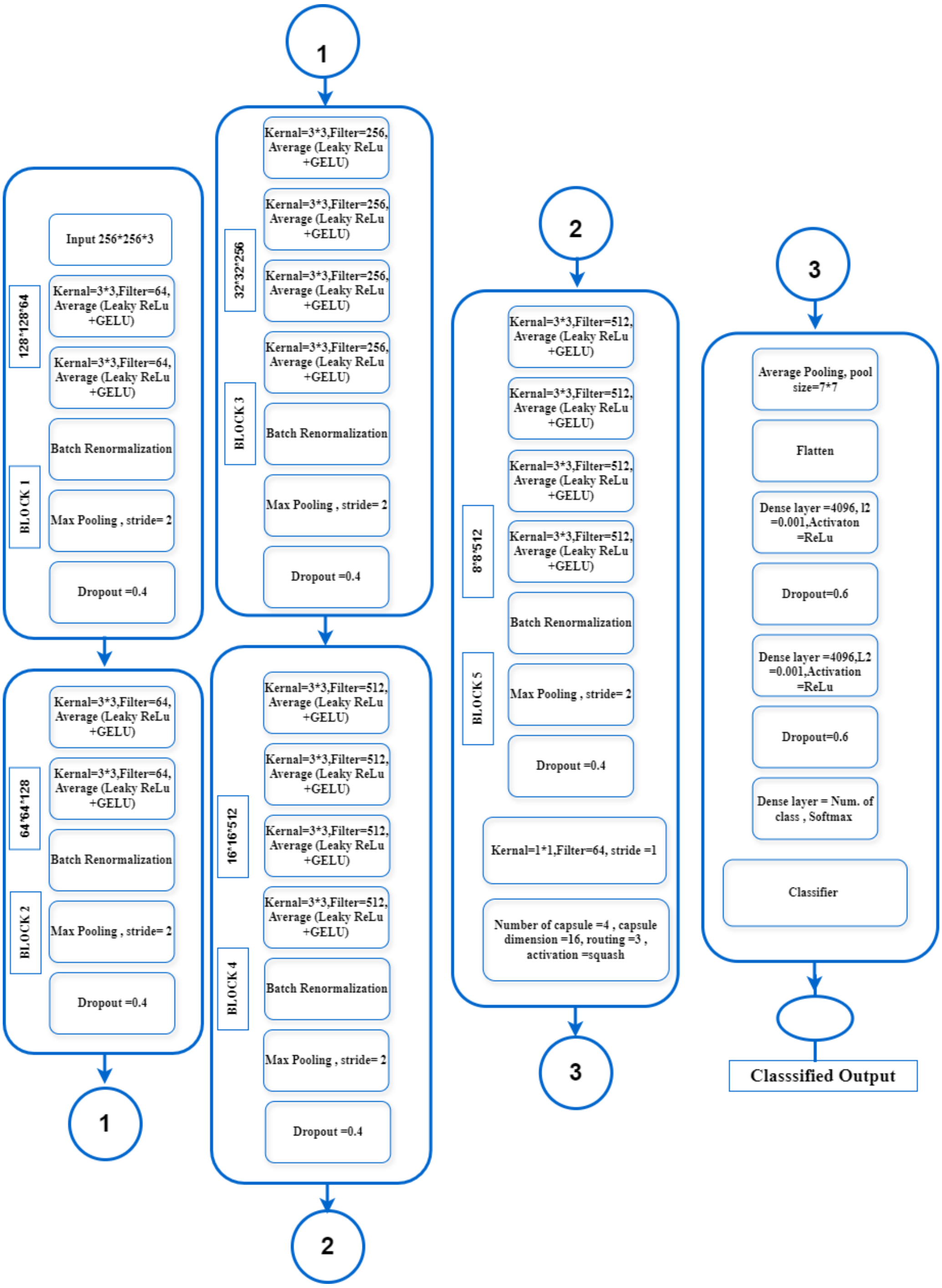


Figure 2: The architecture of Depth-wise Separable VGG19 and Capsule Networks for bell pepper and grape leaf disease classification with ensemble activation.

3.1 Dataset Overview

This research used two different datasets to classify bell pepper and grape leaf diseases, as explained in Tables 1 and 2. The bell pepper dataset contains binary class labels as 'healthy' and 'unhealthy'. In contrast, the grape leaf dataset poses a multi-class challenge with Black Measles, Black Rot, Isariopsis Leaf Spot, and healthy samples. The bell pepper dataset comprises 18,124 images, containing 6,563 healthy, 6,356 unhealthy, and 5,205 augmented images. The grape leaf dataset is more comprehensive, with 30,094 images of 4,385 healthy samples and 8,020 augmented samples.

Sl No.	Dataset	No. of healthy images	No. of unhealthy images	No. of augmented images	Total
1	PlantifyDr [3]	3449	4014	2960	10423
2	Plant village [4]	1478	997	960	3435
3	Sravanneeli [5]	850	1000	810	2660
4	Custom dataset	786	345	475	1606
Total		**6563**	**6356**	**5205**	**18124**

Table 1: Details of the bell pepper dataset used in this research, with and without augmentation.

Sl No.	Dataset	Healthy Images	Black Measles	Black Rot	Isariopsis Leaf Spot	Augmented Images	Total
1	PlantifyDr	2594	3783	3596	3228	5200	18401
2	Plant village	339	1107	944	861	1270	4521
3	Sravanneeli	1000	1383	1180	1076	1145	5784
4	Custom dataset	452	231	145	155	405	1388
Total		**4385**	**6504**	**5865**	**5320**	**8020**	**30094**

Table 2: Details of the grape leaves dataset used in this research, with and without augmentation.

In deep learning-based image classification, augmentation becomes essential, especially with an imbalanced dataset. It aids in simulating several scenarios about rotation, flipping, an increase in brightness, and even the introduction of noise, which a plant leaf might experience in real-world situations like sunlight, occlusion, and angles. In this reseach, augmentation was aimed at increasing the training sample size while reducing model overfitting and improving generalisation in field data. For example, the PlantifyDr dataset for grape leaves was supplemented with an additional 5,200 samples for class imbalance to aid the model in learning disease-specific features. Likewise, augmentation was necessary for bell pepper images to balance the healthy and unhealthy classes while preserving high classification accuracy. Using model-enhanced data, the proposed VGG19-CapsNet was trained on a well-balanced dataset representing the complexity and variability of agricultural environments, essential for reliable deployment in practical agricultural settings focused on disease prediction. The DWSC technique breaks down standard convolution operations into two distinct stages: depthwise and pointwise convolutions. Each input channel is separately filtered to extract the spatial features in the depthwise stage. This is followed by pointwise convolution, where a 1×1 kernel is applied to the channel-wise outputs to fuse inter-channel features and enable feature fusion at higher levels. This separation of DWSC reduces the number of trainable parameters and the computational burden, making it ideal for resource-constrained situations. Feature extraction is further enhanced with Capsule Network layers, which maintain the spatial hierarchies and transformations dependent on the viewpoints of the input data. These capsule layers are useful for grouping and enhancing class-specific representation of the low-level features, which the dense layers process for final classification. To promote interchangeability of the model with varying data distributions, a blend of activation functions, specifically ReLU and GELU, is employed. This approach to hybrid activation enhances the model's complexity but prevents overfitting. With the implementation of DWSC's light-weight structure, the spatial awareness of Capsule Networks, and ensemble activations for generalisation, the proposed model displays improved classification performance and optimized efficiency.

3.2 Parameters of the Proposed Plant Leaf Disease Classification Model

In this research, we have enhanced the accuracy of the real-time model for classifying plant leaf diseases Batch normalisation normalises the activations as follows;

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i \quad \text{and} \quad \sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2 \tag{1}$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \quad \text{and} \quad y_i = \gamma \hat{x}_i + \beta \tag{2}$$

In Equation 2, γ and β are learnable parameters, while batch renormalisation introduces two additional parameters, r and d, ensuring that normalised activations align with dataset statistics. The batch size determines the number of samples processed per training iteration. In this research, different batch sizes of 9, 16, and 32 were tested to evaluate their performance on the model. A batch size of 9 with a 0.001 learning rate and 60 epochs led to the highest classification accuracy. Also, some comparisons were made with different learning rates of 0.1, 0.01, and 0.001 for the model's general convergence and generalisation performance. All results from the experiments are reported in Table 3, which encapsulates the best configuration for the training of the proposed architecture.

Sl.No.	Batch Size	LR	No. of Epoch	Accuracy Bell pepper			Accuracy Grape		
				Training	Validation	Testing	Training	Validation	Testing
1	9	0.1	60	99.12	98.43	99.54	99.16	98.48	99.60
		0.01		99.23	98.65	99.78	99.26	98.70	99.83
		0.001		99.98	99.81	99.79	99.94	99.84	99.84
2	16	0.1	60	98.70	98.34	98.10	99.75	98.97	99.17
		0.01		99.10	98.40	99.20	99.23	98.16	99.34
		0.001		99.48	99.23	99.73	99.90	99.90	99.97
3	32	0.1	60	97.24	97.13	98.00	97.11	97.08	97.04
		0.01		97.54	97.24	98.40	97.23	96.86	97.70
		0.001		97.86	97.76	98.56	97.92	98.00	98.86

Table 3: Comparison of the various parameters of the proposed plant leaf disease classification model with different batch size (Bs), learning rates (Lr).

3.3 Training Progress

For the categorisation of bell pepper and grape plant leaf diseases using the VGG19-CapsuleNet model, the dataset was split meticulously into training, validation, and testing subsets with a ratio of 70:20:10. This split is necessary for balancing the learning processes and the ability to generalise with unseen data. This division guarantees that the model can utilise a sufficiently large portion of the data (70%) for training, which allows learning and feature internalisation. A portion of the validation set, which is 20 percent, assists in hyperparameter tuning and model performance evaluation during the training phase to prevent overfitting while ensuring generalisation. The additional 10 percent is allocated strictly for testing, ensuring an unbiased, independent assessment of the model's performance in practical applications. This division maintains equilibrium amongst learning, optimisation, and evaluation. Accuracy, precision, recall, and F1-score are evaluation metrics computed from this subset to determine the model's performance and trust in practical applications. Thus, the developed deep learning model achieves robustness, generalizability, and optimal performance with this structured data partitioning and evaluation scheme.

Sl No.	Dataset	Training	Validation	Testing	Total
1	PlantifyDr	7296	2085	1042	10413
2	Plant village	2405	687	343	3435
3	Sravanneeli	1862	532	266	2660
4	Custom dataset	1124	321	161	1606
Total		**12687**	**3625**	**1812**	**15454**

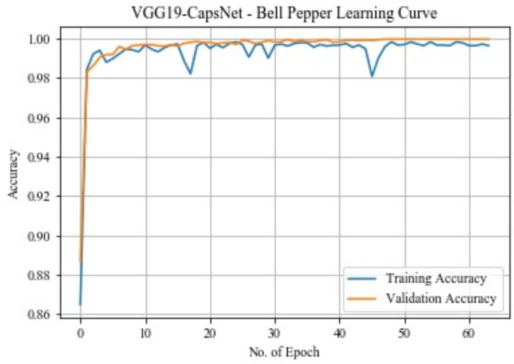
Table 4: Dataset Split Overview for Bell Pepper Leaf Disease Classification using VGG19-CapsNet.

Sl No.	Dataset	Training	Validation	Testing	Total
1	PlantifyDr	1288	3680	1840	18401
2	Plant village	3165	904	452	4521
3	Sravanneeli	4048	1156	580	5784
4	Custom dataset	972	278	138	1388
Total		**15088**	**4482**	**2240**	**22410**

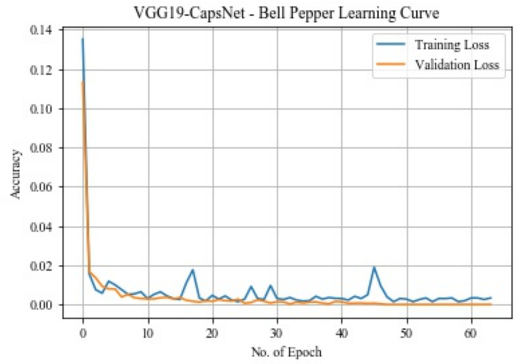
Table 5: Details of the grape leaf dataset used for training, testing, and evaluation are used in the proposed VGG19-CapsNet model.

Tables 4 and 5 shows the split ratio of training, validation, and testing dataset, which capture the methodology for evaluating the proposed VGG19-CapsNet model concerning bell pepper and grape leaf disease classification tasks. About the bell pepper dataset, 15,454 images were used, consisting of 12,687 images used for training, 3,625 images used for validation, and 1,812 images set aside for testing. The grape leaf dataset contained 22,410 images, allocating 15,088 for training, 4,482 for validation, and 2,240 for testing. This dataset allows multi-class classification, including Black Measles, Black Rot, and Isariopsis Leaf Spot. A large part of the validation and testing samples in the grape dataset needed to be collected from the PlantifyDr source, thus making it possible to evaluate the model's generalizability thoroughly. Using custom datasets is particularly important for capturing variabilities in the environment and enhancing the robustness of the model. With this thorough and detailed data partitioning approach, the VGG19-CapsNet model possesses improved binary and multi-class classification capabilities, exhibiting consistent dependability and usefulness in practical agricultural settings.

In Figure 3, the sharp increase in accuracy indicates that the model successfully trained on the data, exceeding 98% accuracy within the first 10 epochs and stabilising just above 99.5% for training and validation. This demonstrates how effectively the model extracts disease-specific input data features. During training, validation accuracy nearly tracks the training accuracy curve with very little divergence, which illustrates the strong generalisation capability of the model while suggesting that overfitting is, in fact, unlikely. The loss curve also demonstrates the same characteristics, with steep declines in the training and validation loss occurring during the first couple of epochs until the loss approaches zero and remains consistently low. Minor peaks in training loss are bound to occur due to stochastic variation and will not impact the overall directed learning trend.



(a) Accuracy Curve

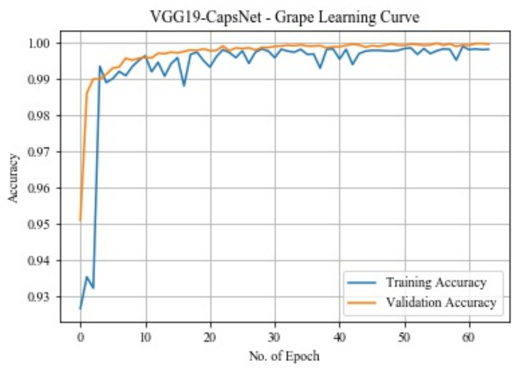


(b) Loss Curve

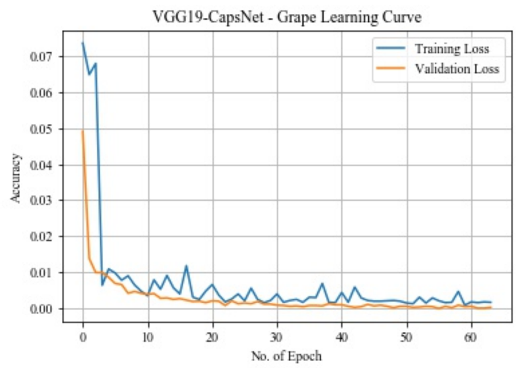
Figure 3: Performance evaluation of bell pepper leaf disease classification using VGG19-CapsNet: accuracy and loss curves.

The parallel and low-loss curves prove the model's stability and robustness during training are consistent and unchanging.

As illustrated by the training and validation accuracy curves in Figure 4, the model demonstrated an impressive level of accuracy (99%) by the 10-epoch mark; this leap forward was attributed to the rapid learning processes of the model in initial epochs, and capped accuracy was maintained for the rest of the training period. Moreover, the confirmation accuracy steadily paralleled the training accuracy, indicating assessment generalisation to new data while only a small amount of overfitting occurred. However, the small oscillation in training accuracy past epoch 20, characteristic of deep learning models, does not detract from their overall effectiveness, stability and performance. These observations are further confirmed in the loss curves. During the latter stages of the run, training and validation losses promptly decrease to near-zero values while stagnating, demonstrating effective convergence and efficient learning. In this case, learning retention is confirmed by undergoing low training loss and grade-deficient drop in validation loss after 10 epochs while maintaining alignment with training loss. Thus, the overall results indicate strong performance by the VGG19-CapsNet model on the multi-class grape leaf disease classification tasks. Practical implementation of the model can be viewed as viable due to exceptional performance indicators of accuracy, low loss, and stable curves, along with the responsiveness required by automated smart agricultural systems.



(a) Accuracy Curve



(b) Loss Curve

Figure 4: Performance evaluation of grape leaf disease classification using VGG19-CapsNet: accuracy and loss curves.

4 Model Analysis and Experimental Results

4.1 Performance Analysis

The assessment of the proposed approach encompasses metrics such as accuracy, precision, recall, specificity and F1 score. Throughout the 60 epochs, training accuracy and loss are meticulously tracked. Performance metrics for both augmented datasets are presented in Figure 5.

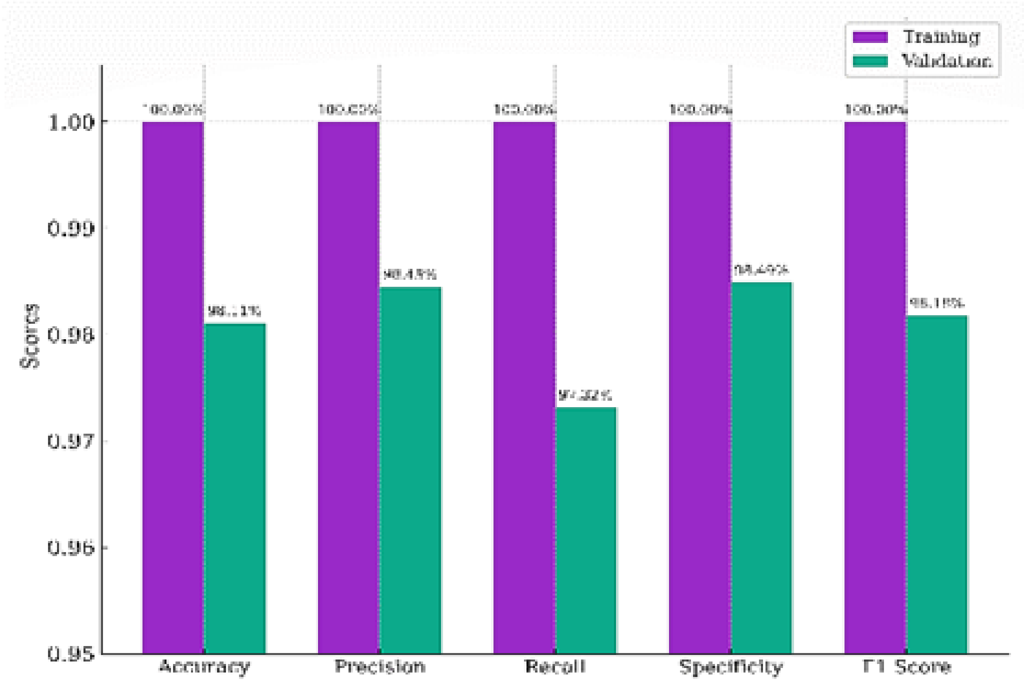


Figure 5: Performance evaluations on the training and validation dataset of bell pepper.

The classification performance of the proposed deep learning model on bell pepper disease detection is evaluated using standard metrics such as Accuracy, Precision, Recall, Specificity, and F1 Score. The Figure 5 shows that the model achieved 100% on all five metrics for training datasets, indicating that the network has learned the training data well. In this case, the validation metrics were also high: 98.11% for Accuracy, 98.63% for Precision, 97.32% for Recall, 98.49% for Specificity, and 98.18% for F1 Score. These metrics prove the model can retain good predictive performance on previously unseen data. Aside from the five metrics, Recall on the validation dataset is the lowest at 97.32%, indicating that some true positive disease instances could be misclassified. This metric is important in agricultural disease detection processes, especially for bell pepper plants, because true negatives could result in undiagnosed plant diseases and subsequent loss in yield potential. The high Precision of 98.63% shows that most positive predictions are correct, indicating minimal false positives. The Specificity 98.49% augments alongside these claims to further suggest that the model is very dependable in confirming the absence of infection in leaves. The model achieves an F1 Score of 98.18% on the validation set, which illustrates a blended score of Precision and Recall, a highlight of the model's strong performance since it achieves minimal errors in all types of classification.

In Figure 6, the deep learning model achieves perfect performance (100%) on all metrics for the training dataset, which suggests an appropriate model fit and full representative learning on the provided training samples. In contrast, the validation performance metrics are lower, illustrating the model's generalization ability towards new data. The validation Accuracy and Recall are around 98.44% and 98.41%, respectively. These values depict the model's high confidence in making accurate predictions and significantly minimizing false positive outcomes. The Recall value of 98.20% shows that the model accurately detects many diseased grape

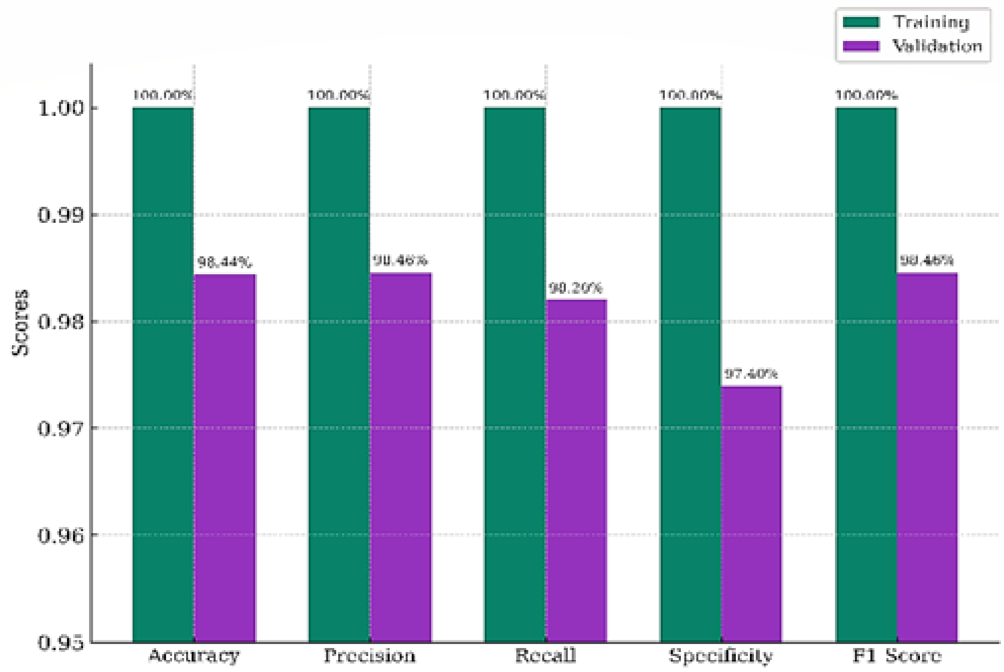


Figure 6: Performance evaluations on the training and validation data set of the grape

leaves, with very few being incorrectly classified as non-diseased. The Specificity, which assesses the correct identification of healthy leaves, scores slightly lower at 97.48%, suggesting that a small number of healthy samples may be incorrectly classified as unhealthy. The F1 Score, which combines Precision and Recall, records 98.40%, verifying that the model maintains strong sensitivity and reduced false alerts.

4.2 Testing the Model

The testing stage of the proposed VGG19-CapsNet model starts with loading the images, results visualization, and required computing libraries using the Python interface. Specifically, upon importing the necessary packages, the testing image is stored in a fixed location and is fetched using the Python Imaging Library. Matplotlib is used to visualize the output, while image operations are handled using the Python Imaging Library. Furthermore, the test image will be resized to 256x256 pixels, the standard configuration for processed images before being fed into the trained model. During this step, the pixel values will undergo normalisation to the range of [0, 1] to align with the training data distribution. Finally, one more dimension will be added to the image array to comply with the expectations of the batch input shape of the deep learning model. The final predicted label is taken from the class that scored the highest among the prediction scores, class-wise output. The selected class's prediction confidence is displayed to quantify the model's certainty. The prediction of bell pepper and grape leaf disease categories is shown and visualised in Figures 6 and 7, respectively. The model performance was assessed on test samples, where these visualisations show the ability of the system to recognise the tested samples in real-time while proving its functional intelligence for harnessed autonomous deployment. The testing stage of the proposed VGG19-CapsNet model starts with loading the images, results visualization, and required computing libraries using the Python interface. Specifically, upon importing the necessary packages, the testing image is stored in a fixed location and is fetched using the Python Imaging Library. Matplotlib is used to visualize the output, while image operations are handled using the Python Imaging Library. Furthermore, the test image will be resized to 256x256 pixels, the standard configuration for processed images before being fed into the trained model. During this step, the pixel values will undergo normalisation to the range of [0, 1] to align with the training data distribution. Finally, one more dimension will be added to the image array to comply with the expectations of the batch input shape of the deep learning model. The final predicted label is taken from the class that scored the highest among the prediction scores, class-wise output. The selected class's prediction confidence is displayed to quantify the model's certainty. The prediction of bell pepper and grape leaf disease categories is shown and visualised in Figures 6 and 7, respectively. The model performance was assessed on test samples, where these visualisations show the ability of the system to recognise the tested samples in real-time while proving its functional intelligence for harnessed autonomous deployment. The findings suggest that

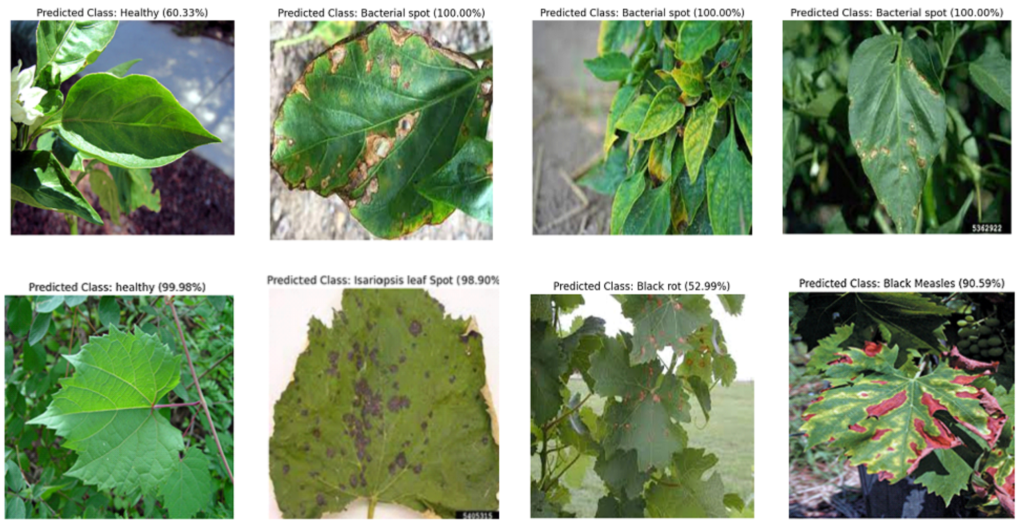


Figure 7: Experimental Evaluation of the VGG19-CapsNet Model for Bell Pepper Leaf and Grape Leaf Disease Detection on NVIDIA Jetson Nano GPU computer.

the model has high classification accuracy across all the disease classes, even when run on the GPU-enabled edge device NVIDIA Jetson Nano. To evaluate the practical robustness of the system, the model was tested against benchmark curated datasets and real-life field evaluations conducted in various environments, including daylight, nighttime, and foggy weather conditions. These results validate the system's generalisation capability under different illumination and weather conditions. The model's optimisation procedure is executed with an added layer using a Modified Adamax optimiser, which uses an exponentially weighted infinity norm. This change enhances the stability and responsiveness during training to adjust for situations where the gradients are subject to considerable change. The optimiser achieves convergence consistency by mitigating these issues and increasing overall dataset and architecture agnostic robustness. The initial image undergoes hierarchical feature extraction in the proposed framework by applying 3×3 convolutional filters with a stride of 2 in the CNN layers. Increasing feature complexity employs filter bank activation through a stacked ensemble of Leaky ReLU (LReLU) and GELU activation functions across 6 blocks, enhancing non-linearity and feature discrimination. Pooling retains important information while further downscaling the spatial resolution of the feature maps. These layers are interspersed with dropout and other regularisation techniques to manage overfitting concerns. The convolutional layer preceding classification applies L2 regularization while global average pooling provides a dropout of 40%, which allows for arbitrary sample generation during testing. The model derives classification probabilities for each input image by applying the Softmax activation function, which outputs a probability distribution over the possible disease classes for grape and bell pepper leaves. The architecture assures the model that it will yield reliable outputs for automated agricultural monitoring and diagnostics.

4.3 Comparative Analysis

The assessment of the framework implementation based on VGG19-CapsNet includes its evaluation against other deep learning frameworks that focus on detecting and classifying plant leaf diseases. This assessment evaluates current literature to ensure that only the most recent approaches are used. The review and analysis were performed to demonstrate the extent to which the method is congruous with or surpasses advances in diagnosing diseases in plants, alongside computer vision. A summary of the selected frameworks in for plant leaf disease detection using deep learning is provided in Table 6. The table lists diagnostic components including the model architecture, datasets employed, training regimens, and evaluation outcomes. This synthesises a constructive assessment of the approach proposed that is well-defined in terms of its benchmarks, discerning strengths, and examining prospects for improvement.

Methodology	**Image Class**	**Optimizer Used**	**Dataset**	**Pre-processing**	**Accuracy**	**Batch size**	**Learning rate**
Proposed VGG19-CasuleNet Model	Bell pepper, Grape	Modified Adamax	Plant Village, custom dataset	Augmented	98.11, 98.44	9	0.001
AlexNet, VGG 16, 19 [24]	Bell pepper	SGD	Plant Village	Augmented	94.86, 93.67	9	0.01
Concatenation of VGG16 and AlexNet [25]	Bell pepper and fruits	-	Collected from North Mecha Woreda	Augmented	95.82, 93.21	64	-
ResNet 50, ResNet 152 [15]	Bell pepper	Adam	Plant Village	Augmented	98.95, 98.88 (ResNet 152 with DA)	64	0.01

Table 6: Comparative evaluation of the proposed VGG19-CapsNet model for grape and bell pepper disease classification.

One of the key issues encountered while developing the VGG19-CapsNet model for disease detection in plant leaves was the inadequate diversity within the dataset. The absence of lighting variation, the leaves' orientation, and background settings were very limiting to achieving good deep learning model generalisation. A secondary issue concerned environmental shifts and the complicated nature of real-time implementation. In real-time execution, requirements like ultra-low latency, high flexibility, and strong performance against changes in context pose great challenges, particularly when the models are executed on GPU-based embedded systems. The Jetson Nano single-board computer with augmented CPU capabilities and increased memory bandwidth was able to resolve some of these challenges and improve the latency problem. Adopting the VGG19-CapsNet approach has socio-economic and environmental implications, reinforcing several United Nations Sustainable Development Goals, particularly SDG 2 (Zero Hunger), SDG 8 (Decent Work and Economic Growth), and SDG 12 (Responsible Consumption and Production). The system's early and precise diagnosis of crop diseases significantly minimises yield losses, protects harvests, and improves the efficiency of agricultural resources, including pesticides and fertilizers. This degree of accuracy reduces environmental damage and fosters greater sustainability in farming operations. In addition, farmers improve their earnings and financial health through reduced operational costs, increased crop quality, enhanced income from premium market access, and improved financial stability. The promotion of innovation and the transition to smart rural agriculture, essential in promoting rural development and sustainable food systems, are motivated by AI-powered diagnostic tools. VGG19-CapsNet enables real-time detection and classification of diseases, allowing farmers to act proactively, which directly boosts productivity and resilience. This approach enhances the economic productivity of farming enterprises. It supports sustainable development through improved food security, sustainable resource use, and economic growth using advanced agricultural technologies, enhancing overall welfare.

5 Conclusion

This research presents a new scalable and efficient deep learning framework, VGG19-CapsNet, utilizing ensemble activation functions of Leaky ReLU and GELU to classify bell pepper and grape leaf diseases. It addresses important issues related to computational burden, loss of spatial information, intra-class variability, and overfitting on small agri-data sets paradigms in agricultural disease detection. Captive layers ensure enhanced detection of subtle disease patterns by capturing higher spatial relationships. Ensemble activations offer better gradients and non-linearity, which enhances generalization. The empirical assessment conducted on the curated and augmented datasets from PlantifyDr, PlantVillage, Sravanneeli, alongside field datasets collected externally, revealed astounding results, 98.11% and 98.44% accuracy on bell pepper and grape leaf validation sets, respectively. Implementation on the edge device, NVIDIA Jetson Nano, tested real-time agricultural diagnostics and confirmed the model's accuracy for edge devices under varying environmental conditions. Comparisons against the state-of-the-art model's extensive range of cross-architecture benchmarks confirmed the model's unparalleled accuracy, speed, and overall robustness compared to the other architectures. This research sets a new standard in developing lightweight and precise frameworks for plant disease classification. In particular, attention will be directed toward adapting the model for multi-crop classification in the future, adding attention layers, and implementing real-time drone or mobile imaging systems into fully automated smart farming technology.

Declarations

Ethical Approval: Not Applicable.

Availability of Supporting Data: The datasets are available on the Kaggle repository (Reference [3, 4, 5]) and will be supplied upon request to the Corresponding Authors.

Competing Interests: There is no competing interest.

Funding: Not Applicable.

Authors' Contributions: All authors equally contributed to the work and preparation of the manuscript.

Acknowledgement: Not Applicable.

References

[1] S.F. Ahmed, M.S.B. Alam, M. Hassan et al., "Deep learning modelling techniques: current progress, applications, advantages, and challenges," *Artificial Intelligence Review*, vol. 56, pp. 13521–13617, 2023. https://doi.org/10.1007/s10462-023-10466-8.

[2] D. Koldasbayeva, P. Tregubova, M. Gasanov et al., "Challenges in data-driven geospatial modeling for environmental research and practice," *Nature Communications*, vol. 15, p. 10700, 2024. https://doi.org/10.1038/s41467-024-55240-8.

[3] `https://www.kaggle.com/datasets/lavaman151/plantifydr-dataset`, Accessed on January 1st 2024.

[4] `https://github.com/attaullah/downsampled-plant-disease-dataset`, Accessed on January 1st 2024.

[5] `https://www.kaggle.com/datasets/sravanneeli/plant-leaf-diseases-dataset-with-augmentation`, Accessed on January 1st 2024.

[6] M. Bhagat, D. Kumar, R. Mahmood, B. Pati, and M. Kumar, "Bell Pepper Leaf Disease Classification Using CNN," *Proc. of 2nd Int. Conf. on Data, Engineering, and Applications (IDEA)*, Bhopal, India, 2020, pp. 1–5. doi: 10.1109/IDEA49133.2020.9170728.

[7] M. Ji, L. Zhang, and Q. Wu, "Automatic grape leaf disease identification via UnitedModel based on multiple convolutional neural networks," *Information Processing in Agriculture*, vol. 7, no. 3, pp. 418–426, 2020. https://doi.org/10.1016/j.inpa.2019.10.003.

[8] A. Diana Andrushia, T. Mary Neebha, A. Trephena Patricia et al., "Image-based disease classification in grape leaves using convolutional capsule network," *Soft Computing*, vol. 27, pp. 1457–1470, 2023. https://doi.org/10.1007/s00500-022-07446-5.

[9] H.P. Wei, Y.Y. Deng, F. Tang et al., "A Comparative Study of CNN- and Transformer-Based Visual Style Transfer," *Journal of Computer Science and Technology*, vol. 37, pp. 601–614, 2022. https://doi.org/10.1007/s11390-022-2140-7.

[10] H. Gao, X. Liu, M. Qu, and S. Huang, "PDANet: Self-Supervised Monocular Depth Estimation Using Perceptual and Data Augmentation Consistency," *Applied Sciences*, vol. 11, no. 12, p. 5383, 2020. https://doi.org/10.3390/app11125383.

[11] P.K. Chundi, P. Liu, S. Park, S. Lee, and M. Seok, "FPGA-based Acceleration of Binary Neural Network Training with Minimized Off-Chip Memory Access," *Proc. of IEEE/ACM ISLPED*, Lausanne, Switzerland, 2019, pp. 1–6. doi: 10.1109/ISLPED.2019.8824805.

[12] P. Alirezazadeh, M. Schirrmann, and F. Stolzenburg, "Improving Deep Learning-based Plant Disease Classification with Attention Mechanism," *Gesunde Pflanzen*, vol. 75, pp. 49–59, 2023. https://doi.org/10.1007/s10343-022-00796-y.

[13] M. Nawaz, T. Nazir, M.A. Khan, V. Rajinikanth, S. Kadry, "Plant Disease Classification Using VGG-19 Based Faster-RCNN," in *Advances in Computing and Data Sciences*, ICACDS 2023, Springer CCIS, vol. 1848. https://doi.org/10.1007/978-3-031-37940-6_23.

[14] S. Parakh, M.S. Ashraf, N. Tripathi, K. Pant, M.S. Ansari, and P. Negi, "Detection of Bell Pepper Crop Diseases Using Convolution Neural Network," *Proc. of 2nd Int. Conf. on Technological Advancements in Computational Sciences (ICTACS)*, Tashkent, Uzbekistan, 2022, pp. 1–1. doi: 10.1109/ICTACS56270.2022.10705132.

[15] M. Bhagat, D. Kumar, and S. Kumar, "Bell pepper leaf disease classification with LBP and VGG-16 based fused features and RF classifier," *International Journal of Information Technology*, vol. 15, pp. 465–475, 2023. https://doi.org/10.1007/s41870-022-01136-z.

[16] N. Kundu, G. Rani, and V.S. Dhaka, ”A Comparative Analysis of Deep Learning Models Applied for Disease Classification in Bell Pepper,” *Proc. of 6th Int. Conf. on PDGC*, Waknaghat, India, 2020, pp. 243–247. doi: 10.1109/PDGC50313.2020.9315821.

[17] F. Jiang, Y. Lu, Y. Chen, D. Cai, and G. Li, ”Image recognition of four rice leaf diseases based on deep learning and support vector machine,” *Computers and Electronics in Agriculture*, vol. 179, 2020, p. 105824. https://doi.org/10.1016/j.compag.2020.105824.

[18] Y. Kurmi, S. Gangwar, D. Agrawal et al., ”Leaf image analysis-based crop diseases classification,” *Signal, Image and Video Processing (SIViP)*, vol. 15, pp. 589–597, 2021. https://doi.org/10.1007/s11760-020-01780-7.

[19] P. Thakur, A. Chug, and A.P. Singh, ”Plant Disease detection of Bell Pepper Plant Using Transfer Learning over different Models,” *Proc. of 8th Int. Conf. on SPIN*, Noida, India, 2021, pp. 384–389. doi: 10.1109/SPIN52536.2021.9565945.

[20] G. Altan, ”Performance Evaluation of Capsule Networks for Classification of Plant Leaf Diseases,” *International Journal of Applied Mathematics Electronics and Computers*, vol. 8, no. 3, pp. 57–63, 2020. doi:10.18100/ijamec.797392.

[21] M. Ye et al., ”A Lightweight Model of VGG-16 for Remote Sensing Image Classification,” *IEEE J. Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 6916–6922, 2021. doi: 10.1109/JSTARS.2021.3090085.

[22] K. Deeba, A. Balakrishnan, M. Kumar et al., ”A disease monitoring system using multi-class capsule network for agricultural enhancement in muskmelon,” *Multimedia Tools and Applications*, 2024. https://doi.org/10.1007/s11042-024-18717-8.

[23] M. Dai et al., ”Pepper leaf disease recognition based on enhanced lightweight convolutional neural networks,” *Frontiers in Plant Science*, vol. 14, 2023. doi: 10.3389/fpls.2023.1230886.

[24] M.P. Mathew, S. Elayidom, and V. Jagathyraj, ”Disease Classification in Bell Pepper Plants Based on Deep Learning Network Architecture,” *Proc. of 2nd Int. Conf. for Innovation in Technology (INOCON)*, Bangalore, India, 2023, pp. 1–6. doi: 10.1109/INOCON57975.2023.10101269.

[25] Y.A. Bezabih, A.O. Salau, B.M. Abuhayi et al., ”CPD-CCNN: classification of pepper disease using a concatenation of convolutional neural network models,” *Scientific Reports*, vol. 13, 15581, 2023. https://doi.org/10.1038/s41598-023-42843-2.