International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 59 No. 1 September 2026, pp. 138-148

U2-Net Segmentation for Bean Leaf Disease Classification: A Statistically Validated Evaluation with DenseNet121

Noviyanto[1], Ardiansyah[1]

[1]Program Studi Teknologi Informasi, Fakultas Kesehatan dan Teknologi, Universitas Muhammadiyah Klaten

noviyanto@umkla.ac.id, ardiansyah@umkla.ac.id

Auteur correspondent: Noviyanto, noviyanto@umkla.ac.id

Abstract—Prior work by the authors proposed a modified DenseNet121 architecture for classifying bean leaf diseases and reported a test accuracy of 96.90% on the raw iBean dataset (1,295 images) using a single evaluation run. Because bean leaf images are often captured against complex, cluttered backgrounds, and because a single-run evaluation does not indicate how stable a model's performance is, this study extends that work along two directions that were not addressed previously: (1) an image segmentation stage, comparing U-Net and U2-Net, is introduced to isolate the leaf object from its background prior to classification, and (2) model performance is validated through 30 independent repeated runs and paired t-tests rather than a single evaluation. U2-Net achieved a substantially higher mean Intersection-over-Union (mIoU = 0.98) than U-Net (mIoU = 0.61) and was therefore used to produce a cleaned, class-balanced dataset of 1,233 images (411 per class). Using the same classification head configuration as the prior work (DenseNet121 backbone with two additional convolutional blocks), the model achieved a mean test accuracy of 94.63% (SD = 1.16) over 30 runs, which was significantly higher than standard DenseNet121 (92.53%, SD = 2.00, t(58) = 4.98, p < 0.0001), InceptionV3 (85.60%, SD = 3.12, t(58) = 14.84, p < 0.0001), ResNet152V2 (87.20%, SD = 2.38, t(58) = 15.36, p < 0.0001), and ResNet50V2 (86.70%, SD = 2.74, t(58) = 14.59, p < 0.0001). On its best single evaluation run, the model reached a test accuracy of 97-98%, with precision 0.97, recall 0.98, and F1-score 0.98. These results show that, on a segmented and class-balanced dataset, the previously proposed classifier maintains high and statistically stable performance across repeated trials. Because the dataset, hyperparameters, and evaluation protocol differ from the earlier study, this paper does not claim that segmentation alone caused the accuracy difference, instead, it reports the combined effect of these changes and identifies isolating the individual contribution of segmentation as a direction for future work.

Keywords—Bean Leaf Disease, Image Segmentation, U2-Net, Densenet121, Deep Learning, Statistical Validation.

1. Introduction

Beans are a widely cultivated horticultural crop valued for their nutritional content, including vitamin C, calcium, and fiber [1]. Despite these benefits, bean plants are susceptible to diseases of the leaves, stems, and seeds caused by fungi, bacteria, and viruses, which can spread rapidly and cause significant yield and quality losses [2]. Early detection of leaf disease is therefore important for maintaining bean productivity and quality.

For bean leaves specifically, three conditions are commonly of interest: Bean Rust, Angular Leaf Spot, and healthy leaves. Automating the classification of these conditions from images is complicated by factors related to image acquisition, including complex or cluttered backgrounds, variable lighting, and the angle at which images are captured [3]. In practice, healthy leaves are frequently misclassified as diseased when the background or surrounding objects share visual characteristics with disease symptoms [4], indicating that background content, and not only the leaf itself, can influence classification outcomes.

Over the past decade, both traditional machine learning and deep learning methods have been applied to plant disease recognition from images [5]. Convolutional Neural Networks (CNNs) in particular have become a common approach, owing to their ability to learn high-level feature representations directly from raw images [6], [7], often in combination with transfer learning from architectures such as InceptionV3, ResNet, and DenseNet [8].

In earlier work, the authors proposed a CNN architecture based on DenseNet121 with a modified classification head, and compared it against InceptionV3, ResNet152V2, ResNet50V2, and standard DenseNet121 on the iBean dataset released by the Makerere AI Lab (1,295 images across three classes) [9]. The proposed model achieved a single-run test accuracy of 96.90%, with precision, recall, and F1-score of 97%, outperforming the comparison architectures as well as previously reported results on the same dataset, including Elfatimi et al. (92.97%) [10] and Sahu et al. (95.31%) [11].

That earlier study, however, applied the classifier directly to raw images without addressing the complex-background problem noted above, and reported performance from a single training-and-evaluation run, which does not reveal how sensitive the reported accuracy is to the randomness inherent in model training (weight initialization, data shuffling, and augmentation sampling). The present study builds on that work by introducing two elements not previously examined: first, an image segmentation stage-comparing U-Net and U2-Net-that separates the leaf object from its background before classification is performed, and second, an evaluation protocol based on 30 independently repeated training-and-testing runs per architecture, together with paired t-tests, to assess whether observed performance differences between architectures are statistically stable rather than the result of a single favorable run.

The objectives of this study are: (1) to compare U-Net and U2-Net for separating bean leaf objects from complex backgrounds, (2) to evaluate the classification performance of five CNN architectures-InceptionV3, ResNet152V2, ResNet50V2, DenseNet121, and the previously proposed modified DenseNet121-on the resulting segmented dataset, and (3) to assess, through repeated experimentation and statistical testing, whether performance differences among these architectures are consistent across multiple independent runs.

The contribution of this paper is therefore methodological rather than architectural: it does not propose a new classification architecture, since the classification head used here is the same configuration described in the authors' prior work [9]. Instead, it contributes (a) a comparative evaluation of U-Net and U2-Net as a segmentation preprocessing stage for bean leaf disease images, and (b) a repeated-experiment, statistically validated performance evaluation-comprising 30 runs per architecture and pairwise t-tests-that was not part of the earlier single-run study.

2. Related Work

2.1 Plant and Bean Leaf Disease Classification

Several studies have applied CNN-based transfer learning to plant disease classification. Kathiresan et al. [12] and Julianto and Sunyoto [13] evaluated transfer-learning CNN architectures for rice leaf disease detection. Specific to bean leaves, Elfatimi et al. [10] used a MobileNet-based CNN on the iBean dataset and reported 92% accuracy (93.02% recall, 92.98% precision, 92.94% F1-score) after comparing several optimizers. Sahu et al. [11] evaluated GoogleNet and VGG19 on the same dataset, reporting 93.75% and comparable accuracy respectively, together with heat-map visualizations of model attention. Singh et al. [14] applied a fine-tuned EfficientNetB6 model, reporting 91.74% validation accuracy.

2.2 DenseNet-Based Approaches

DenseNet [15] connects each layer to every other layer in a feed-forward fashion within dense blocks, which encourages feature reuse and mitigates vanishing-gradient effects. Vellaichamy et al. [16] used DenseNet121 for multi-class plant leaf disease classification. In the authors' own prior work [9], DenseNet121 was used as a frozen feature extractor with an added convolutional head (two convolutional blocks with 64 and 32 filters, respectively) and achieved the best accuracy among five compared architectures on the raw iBean dataset.

2.3 Segmentation as Preprocessing for Leaf Disease Images

Removing complex backgrounds prior to classification has been explored in related work. Abed et al. [17] proposed a segmentation stage using a U-Net architecture with a ResNet34 encoder to isolate bean leaves from their background, prior to classifying disease type with several CNN backbones (DenseNet121 achieved the best result among those tested, 91.01%). This indicates that segmentation-assisted classification has previously been explored on the same underlying dataset, though with a different segmentation architecture than the one compared in this study. The original U-Net architecture was introduced by Ronneberger et al. [18] for biomedical image segmentation, and later extended by Qin et al. [19] into U2-Net, which introduces nested residual U-blocks intended to capture both local detail and global context at multiple scales without substantially increasing model depth. General reviews of CNN-based segmentation, such as Iglovikov and Shvets [20], and of automatic background handling in leaf disease imagery, such as Zhang et al. [3], further motivate the use of a dedicated segmentation stage in disease-classification pipelines.

2.4 Positioning of This Study

Relative to the authors' prior work [9], this study retains the same classification head configuration but changes the input pipeline (adding a segmentation stage) and the evaluation protocol (30 repeated runs with statistical testing, rather than a single run). Relative to Abed et al. [17], this study compares a different pair of segmentation architectures (U-Net and U2-Net, rather than U-Net with a ResNet34 encoder) and adds repeated-experiment statistical validation, which was not part of that study. As such, this paper is positioned as a methodological extension focused on preprocessing and evaluation robustness, rather than as a new classification architecture.

3. Materials and Methods

3.1 Dataset

This study uses the iBean dataset released by the Makerere AI Lab, publicly available at https://github.com/AI-Lab-Makerere/ibean. The dataset originally contains 1,295 JPG images across three classes: Bean Rust (436 images), Angular Leaf Spot (432 images), and healthy leaves (427 images). After the segmentation and cleaning procedure described in Section III-B, the working dataset used for classification in this study consists of 1,233 images, balanced at 411 images per class (Figure. 1).

(a) (b) (c)

Fig. 1. Types of bean leaf diseases (a) Bean Rust, (b) Angular Leaf Spot, (c) healthy

3.2 Image Segmentation

To address the complex-background problem noted in Section I, a segmentation stage was applied prior to classification. Manual ground-truth masks were created for 150 images (50 per class) to train and evaluate two candidate segmentation architectures: U-Net and U2-Net (Figure. 2). Both models take a 128x128x3 input image and produce a single-channel binary mask (sigmoid activation) separating the leaf object from the background.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 59 No. 1 September 2026, pp. 138-148

Fig. 2. Ground truth of green bean leaves.

The U-Net model was trained using the Adam optimizer with a learning rate of 0.0001 for 1,000 epochs, using binary cross-entropy loss. The U2-Net model, which incorporates nested residual U-blocks in its encoder-decoder structure to capture contextual information at multiple scales, was trained using the SGD optimizer with a learning rate of 0.0001 for 100 epochs, also using binary cross-entropy loss.

Segmentation quality was evaluated using mean Intersection-over-Union (mIoU) between predicted and ground-truth masks. U2-Net achieved mIoU = 0.98, compared to mIoU = 0.61 for U-Net (Table I). Based on this result, U2-Net was selected as the segmentation model applied to the full dataset. Segmentation outputs judged to imperfectly remove the background, or to no longer clearly reflect the visual characteristics of their assigned class, were discarded, this cleaning step also served to rebalance the dataset across the three classes, yielding the final set of 1,233 images described in Section III-A.

TABLE I. SEGMENTATION PERFORMANCE COMPARISON (MIOU)

Architecture	Optimizer	Epochs	mIoU
U-Net	Adam (lr=0.0001)	1000	0.61
U2-Net	SGD (lr=0.0001)	100	0.98

3.3 Preprocessing and Data Augmentation

For the classification stage, all images were resized to 300x300x3 and pixel values were rescaled to the [0,1] range. Data augmentation was applied to the training split only, using Keras' ImageDataGenerator with a rotation range of 10 degrees and a zoom range of 0.2, augmented images were generated in memory in real time during training rather than being saved as separate files. The test set consistently comprised 123 images across all five classification scenarios (Section III-F), corresponding to approximately 10% of the 1,233-image dataset, the exact training-validation split ratio used for the remaining images could not be confirmed from the available experimental records and is therefore not reported numerically.

3.4 Classification Architecture

Five CNN configurations were evaluated: InceptionV3, ResNet152V2, ResNet50V2, standard DenseNet121, and the modified DenseNet121 architecture (referred to here as Proposed Beans DenseNet121) originally introduced in the authors' prior work [9]. For all five, a corresponding pretrained backbone (ImageNet weights, top classification layers excluded) was used as a frozen feature extractor.

The Proposed Beans DenseNet121 head is unchanged from [9]: following the DenseNet121 feature extractor, a convolutional block with 64 filters (3x3 kernel, ReLU activation) is applied, followed by average pooling, a second convolutional block with 32 filters (3x3 kernel, ReLU activation) is applied, followed by average pooling, this is followed by dropout (rate 0.4) to reduce overfitting, a flatten operation, and a final dense layer with 3 units and softmax activation (figure 3). This configuration is reused here as a fixed classifier so that the effect of the segmentation preprocessing and the repeated-evaluation protocol can be examined under conditions comparable to the earlier study, rather than being presented as a new architectural contribution of this paper.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 59 No. 1 September 2026, pp. 138-148

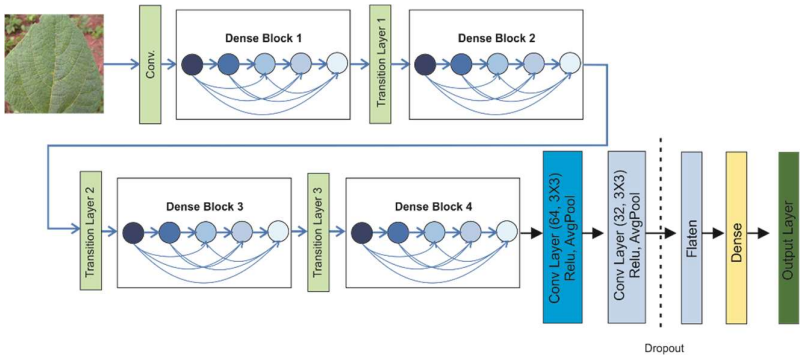

Fig. 3. Ground truth of green bean leaves.

3.5 Experimental Configuration

All classification models were trained with the Adam optimizer (learning rate 0.0001), a batch size of 32, for 30 epochs, using categorical cross-entropy loss and softmax output activation, a dropout rate of 0.4 was used in the added classification head. Experiments were conducted in Google Colaboratory using TensorFlow, with 25 GB of RAM and an NVIDIA Tesla T4 GPU.

3.6 Experimental Design

Two sets of experiments were conducted. First, each of the five classification architectures was trained and evaluated once, and its performance was summarized using training, validation, and test accuracy, together with a confusion matrix on the 123-image test set, from which precision, recall, and F1-score were computed. Second, to assess the stability of these results, each of the five architectures was independently trained and evaluated 30 times, and the mean and standard deviation of test accuracy were recorded across runs. Paired two-sample t-tests (assuming equal variances, alpha = 0.05) were then used to compare the 30-run accuracy distribution of Proposed Beans DenseNet121 against each of the other four architectures.

3.7 Evaluation Metrics

Model performance was quantified using accuracy, precision, recall, and F1-score, computed per class from the confusion matrix and averaged, following standard definitions:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

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

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

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

Segmentation quality was quantified using mean Intersection-over-Union (mIoU), the average IoU across the foreground (leaf) and background classes. Statistical significance of accuracy differences across the 30 repeated runs was assessed using the two-sample t-test, comparing the resulting t-statistic against the critical value at alpha = 0.05 (one-tail: 1.6716, two-tail: 2.0017, df = 58).

4. Results

4.1 Segmentation Results

As reported in Table I, U2-Net outperformed U-Net on the segmentation task (mIoU 0.98 vs. 0.61) and was accordingly used to prepare the classification dataset. Applying U2-Net to the full iBean dataset and discarding imperfect segmentation outputs produced the final, class-balanced dataset of 1,233 images (411 per class) used for all classification experiments.

4.2 Single-Run Classification Performance

Table II summarizes training, validation, and test accuracy for each of the five architectures from a single training-and-evaluation run on the segmented dataset. Proposed Beans DenseNet121 obtained the highest test accuracy (97%), followed by DenseNet121 (94%), ResNet152V2 and ResNet50V2 (91% and 92%, respectively), and InceptionV3 (83%).

TABLE II. SINGLE-RUN TRAIN / VALIDATION / TEST ACCURACY (%)

Architecture	Training	Validation	Testing
InceptionV3	90	82	83
ResNet152V2	99	91	91
ResNet50V2	98	92	92
DenseNet121	99	91	94
Proposed Beans DenseNet121	98	95	97

A classification report derived from the confusion matrix of the same run for Proposed Beans DenseNet121 (123 test images, 120 correctly classified) gave a test accuracy of 0.98, with precision 0.97, recall 0.98, and F1-score 0.98 (Table III). The minor discrepancy between the 97% reported in Table II and the 98% derived from the confusion matrix in Table III is attributed to differences in rounding between the two reporting methods, and both figures originate from the same evaluation run rather than from independent trials.

TABLE III. CLASSIFICATION REPORT, PROPOSED BEANS DENSENET121 (SINGLE RUN, 123 TEST IMAGES)

Metric	Value
Accuracy	0.98
Precision	0.97
Recall	0.98
F1-score	0.98

4.3 Repeated Experiment Results (30 Runs)

To assess whether the single-run results above reflect a stable pattern or a favorable individual run, each architecture was independently trained and evaluated 30 times. Table IV reports the resulting mean test accuracy and standard deviation for each architecture. Proposed Beans DenseNet121 obtained the highest mean accuracy (94.63%) and the lowest standard deviation (1.16) among the five architectures, indicating both higher average performance and greater run-to-run consistency than the comparison models.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 59 No. 1 September 2026, pp. 138-148

TABLE IV. MEAN AND STANDARD DEVIATION OF TEST ACCURACY OVER 30 INDEPENDENT RUNS

Architecture	Mean Accuracy (%)	Std. Deviation
InceptionV3	85.60	3.12
ResNet152V2	87.20	2.38
ResNet50V2	86.70	2.74
DenseNet121	92.53	2.00
Proposed Beans DenseNet121	94.63	1.16

4.4 Statistical Test Results

Table V summarizes the paired two-sample t-test results comparing the 30-run accuracy distribution of Proposed Beans DenseNet121 against each of the other four architectures. In every comparison, the t-statistic substantially exceeded the two-tail critical value (2.0017, df = 58), and the associated p-value was well below 0.0001, indicating that the observed differences in mean accuracy are unlikely to be due to chance.

TABLE V. PAIRWISE T-TEST RESULTS: PROPOSED BEANS DENSENET121 VS. COMPARISON ARCHITECTURES (30 RUNS EACH)

Comparison	Mean Diff. (%)	t Stat	df	p (two-tail)
vs. DenseNet121	2.10	4.98	58	5.95e-6
vs. InceptionV3	9.03	14.84	58	<1e-19
vs. ResNet152V2	7.43	15.36	58	<1e-19
vs. ResNet50V2	7.93	14.59	58	<1e-19

4.5 Confusion Matrix Analysis

For Proposed Beans DenseNet121 on the single evaluation run reported in Table III, 120 of 123 test images were correctly classified (figure 4). Errors were concentrated between visually similar classes (angular leaf spot and bean rust) and between healthy and diseased leaves, consistent with the pattern of misclassification reported in the authors' prior work on the unsegmented dataset [9], suggesting that some sources of visual ambiguity between classes remain even after background segmentation.

Fig. 4. Confusion Matrix Proposed Beans DenseNet121.

5. Discussion

5.1 Effect of Preprocessing

Resizing and augmentation (10-degree rotation, 0.2 zoom range) were applied identically across all five architectures in this study, so differences in Table II and Table IV reflect differences between architectures under a common preprocessing pipeline rather than differences in preprocessing itself. The dataset used in this study was not evaluated without these preprocessing steps, so their individual contribution cannot be isolated from the present results.

5.2 Role of Segmentation

U2-Net produced substantially better segmentation quality than U-Net (mIoU 0.98 vs. 0.61) and was used to prepare the dataset for all classification experiments in this study. However, this study did not include a controlled comparison of classification performance with and without segmentation under otherwise identical conditions (same architecture, same hyperparameters, same dataset split). Consequently, while the classification results reported here were obtained on a segmented and cleaned dataset, this paper does not claim that segmentation caused the observed level of classification accuracy, or that it necessarily improves classification accuracy relative to using raw, unsegmented images. Establishing this would require an ablation study evaluating the same classifier on segmented and unsegmented versions of the same dataset split, which is identified as a direction for future work in Section VI.

5.3 Classifier Performance

Among the five architectures evaluated, Proposed Beans DenseNet121 consistently obtained the highest single-run and mean repeated-run accuracy, and the lowest run-to-run variability (Table IV). Because this classification head is identical to the one used in the authors' prior work [9], this pattern is consistent with-rather than a new demonstration of-the earlier finding that this configuration performs well relative to the four comparison architectures. The contribution of the present results is not the architecture's relative ranking, which was already reported previously, but the observation that this ranking holds consistently across 30 independent runs rather than a single evaluation, and is statistically significant (Table V).

5.4 Comparison with Prior Work

The classification accuracy obtained in this study differs from that reported in the authors' prior work [9] (94.63% mean over 30 runs, or 97-98% on a single run here, versus 96.90% on a single run previously). This difference cannot be attributed to a single cause, because several aspects of the experimental setup changed simultaneously between the two studies: the dataset (1,233 segmented images here versus 1,295 raw images previously), the optimizer configuration (learning rate 0.0001 and batch size 32 here versus 0.0009 and batch size 64 previously), and the evaluation protocol (30 repeated runs here versus a single run previously). The classification architecture itself, by contrast, was not changed. As such, the accuracy difference between this study and [9] should be interpreted as reflecting the combined effect of dataset preparation, hyperparameter configuration, and evaluation protocol, not as evidence attributable to any one of these factors in isolation.

5.5 Interpretation of Repeated Experiments

The 30-run mean and standard deviation reported in Table IV provide a more complete picture of model behavior than a single evaluation run. Proposed Beans DenseNet121 had both the highest mean accuracy and the lowest standard deviation (1.16) among the five architectures, indicating that its performance advantage was consistent across independent runs rather than driven by one favorable outcome. InceptionV3, by contrast, showed the largest standard deviation (3.12), indicating comparatively less consistent performance across runs on this dataset.

5.6 Interpretation of Statistical Testing

The t-test results in Table V indicate that the difference in mean accuracy between Proposed Beans DenseNet121 and each comparison architecture, measured across 30 runs, is unlikely to have occurred by chance. This supports the claim that the observed performance ordering among architectures is statistically reliable under the conditions tested in this study. It does not,

however, indicate why the performance differences occur, nor does it isolate the effect of any single design choice (e.g., the added convolutional head, the segmentation preprocessing, or the specific hyperparameters used), the t-test evaluates whether a difference in means is statistically significant, not what causes that difference.

5.7 Limitations

This study is limited to the iBean dataset and its three disease classes, generalization to other bean leaf disease datasets, or to a larger number of disease classes, has not been evaluated and would require additional data. The exact training-validation split ratio used in the classification experiments could not be confirmed from the available records, which limits exact reproducibility of that step. Finally, as discussed in Section V-B, the individual contribution of the segmentation stage to classification accuracy was not isolated in this study and remains to be evaluated directly.

6. Conclusion

This study evaluated a segmentation preprocessing stage and a repeated-experiment evaluation protocol for bean leaf disease classification, applied to the classification architecture proposed in the authors' prior work. U2-Net was found to produce substantially better leaf-background segmentation than U-Net (mIoU 0.98 vs. 0.61) and was used to prepare a cleaned, class-balanced dataset of 1,233 images. On this dataset, using unchanged hyperparameters (Adam, learning rate 0.0001, batch size 32, 30 epochs) across 30 independently repeated runs, Proposed Beans DenseNet121 achieved a mean test accuracy of 94.63% (SD = 1.16), significantly higher than standard DenseNet121, InceptionV3, ResNet152V2, and ResNet50V2 under paired t-tests ($p < 0.0001$ in all four comparisons). On a single evaluation run, the model reached 97-98% test accuracy with precision 0.97, recall 0.98, and F1-score 0.98. These results demonstrate that the classification configuration from the authors' prior work maintains high and statistically consistent performance when applied to a segmented dataset and evaluated repeatedly, though they do not by themselves establish that segmentation improves accuracy relative to unsegmented input, since dataset, hyperparameters, and evaluation protocol were changed together rather than in isolation. Future work should include a controlled ablation comparing classification performance with and without segmentation under otherwise identical conditions, evaluation on additional bean leaf disease datasets, and extension to a larger number of disease classes.

References

[1] Y. Sahasakul et al., "Nutritional Compositions, Phenolic Contents, and Antioxidant Potentials of Ten Original Lineage Beans in Thailand," Foods, vol. 11, no. 14, 2022.

[2] E. Elfatimi, R. Eryigit, and L. Elfatimi, "Beans Leaf Diseases Classification Using MobileNet Models," IEEE Access, vol. 10, pp. 9471-9482, 2022.

[3] J. Zhang et al., "Automatic Image Segmentation Method for Cotton Leaves with Disease under Natural Environment," Journal of Integrative Agriculture, vol. 17, no. 8, pp. 1800-1814, 2018.

[4] S. H. Abed, A. S. Al-Waisy, H. J. Mohammed, and S. Al-Fahdawi, "A Modern Deep Learning Framework in Robot Vision for Automated Bean Leaves Diseases Detection," International Journal of Intelligent Robotics and Applications, vol. 5, no. 2, pp. 235-251, 2021.

[5] D. Bhatt et al., "CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope," Electronics, vol. 10, no. 20, pp. 1-28, 2021.

[6] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving Current Limitations of Deep Learning Based Approaches for Plant Disease Detection," Symmetry, vol. 11, no. 7, 2019.

[7] B. Tugrul, E. Elfatimi, and R. Eryigit, "Convolutional Neural Networks in Detection of Plant Leaf Diseases: A Review," Agriculture, vol. 12, no. 8, 2022.

[8] G. Kathiresan, M. Anirudh, M. Nagharjun, and R. Karthik, "Disease Detection in Rice Leaves Using Transfer Learning Techniques," Journal of Physics: Conference Series, vol. 1911, no. 1, 2021.

[9] Noviyanto, A. Sunyoto, and D. Ariatmanto, "Innovative Solutions for Bean Leaf Disease Detection Using Deep Learning," [in press / publication details to be confirmed by author].

[10] P. Sahu, A. Chug, A. P. Singh, D. Singh, and R. P. Singh, "Deep Learning Models for Beans Crop Diseases: Classification and Visualization Techniques," International Journal of Modern Agriculture, vol. 10, no. 1, pp. 796-812, 2021.

[11] V. Singh, A. Chug, and A. P. Singh, "Classification of Beans Leaf Diseases Using Fine Tuned CNN Model," Procedia Computer Science, vol. 218, pp. 348-356, 2023.

[12] A. Julianto and A. Sunyoto, "A Performance Evaluation of Convolutional Neural Network Architecture for Classification of Rice Leaf Disease," IAES International Journal of Artificial Intelligence, vol. 10, no. 4, pp. 1069-1078, 2021.

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in Proc. IEEE CVPR, 2017, pp. 2261-2269.

[14] A. S. Vellaichamy, A. Swaminathan, C. Varun, and K. S, "Multiple Plant Leaf Disease Classification Using Densenet-121 Architecture," International Journal of Electrical Engineering and Technology, vol. 12, no. 5, pp. 38-57, 2021.

[15] S. Abed and A. A. Esmaeel, "A Novel Approach to Classify and Detect Bean Diseases Based on Image Processing," in Proc. IEEE ISCAIE, 2018, pp. 297-302.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in Proc. MICCAI, 2015, pp. 234-241. [citation to be verified by author]

[17] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U2-Net: Going Deeper with Nested U-Structure for Salient Object Detection," Pattern Recognition, vol. 106, p. 107404, 2020. [citation to be verified by author]

[18] V. Iglovikov and A. Shvets, "TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation," arXiv:1801.05746, 2018.

[19] E. Elfatimi, R. Eryigit, and H. A. Shehu, "Impact of Datasets on the Effectiveness of MobileNet for Beans Leaf Disease Detection," Neural Computing and Applications, vol. 36, no. 4, pp. 1773-1789, 2024.

[20] J. Sharma, R. Gupta, M. Kumar, and R. Rawat, "A Deep Learning Approach to Bean Leaf Disease Classification: Evaluating ResNet152V2 Performance on Angular Leaf Spot and Bean Rust," Conference Paper, Dec. 2024. [venue/publisher to be confirmed by author]

[21] R. Agila and F. M. H. Fernandez, "Comparative Analysis of Bean Leaf Disease Detection using VGG-19, Inception, and ResNet50 Algorithms," Conference Paper, Jul. 2024. [venue/publisher to be confirmed by author]

[22] A. Kaur, V. Kukreja, D. Upadhyay, and R. Sharma, "An Integrated GoogleNet with Convolutional Neural Networks Model for Multiclass Bean Leaf Lesion Detection," Conference Paper, Mar. 2024. [venue/publisher to be confirmed by author]

[23] "Deep Convolutional Neural Network Model for Classifying Common Bean Leaf Diseases," Discover Artificial Intelligence, vol. 4, art. 96, 2024, doi: 10.1007/s44163-024-00212-6. [full author list to be confirmed by author]

[24] L. Rahunathan, D. Sivabalaselvamani, E. S. Elakkiya, M. Madhumitha, and K. Kumaresh, "Recognition of Bean Leaf Diseases Using Neural Network and Machine Learning Techniques," in Proc. 2023 3rd Int. Conf. on Smart Data Intelligence (ICSMDI), 2023, pp. 520-526.

[25] Singh et al., "Enhanced Leaf Disease Segmentation Using U-Net Architecture for Precision Agriculture: A Deep Learning Approach," Food Science & Nutrition, 2025. [full author list and volume/page numbers to be confirmed by author]

[26] F. S. Ishengoma and J. P. Telemala, "Integrating Pattern Recognition and CNN-Based Models for Improved Bean Disease Detection and Agricultural Yield Enhancement," Artificial Intelligence and Applications, vol. 3, no. 4, pp. 385-391, 2025.

[27] A. Katumba, W. S. Okello, S. Murindanyi, J. Nakatumba-Nabende, B. W. Mugalu, A. Acur, and M. Bomera, "Leveraging Edge Computing and Deep Learning for the Real-Time Identification of Bean Plant Pathologies," Artificial Intelligence in Agriculture, vol. 11, pp. 34-49, 2024.

[28] "EMSAM: Enhanced Multi-Scale Segment Anything Model for Leaf Disease Segmentation," Frontiers in Plant Science, 2025. [full author list to be confirmed by author]