DOI: 10.2478/ata-2025-0020

Acta Technologica Agriculturae 3
Nitra, Slovaca Universitas Agriculturae Nitriae, 2025, pp. 159–168

CONFIDENCE-AWARE MULTI-MODEL IMAGE CLASSIFICATION FOR EARLY DISEASE DETECTION IN PLANTS

Tianyi ZHONG[1], Muhammad Azhar IQBAL*[2], Xu ZHANG[2]

[1]The University of Melbourne, Faculty of Engineering and Information Technology, Parkville, Victoria, Australia, zhong17@student.unimelb.edu.au

[2]University of Leeds, Faculty of Engineering and Physical Sciences, School of Computer Science, Leeds, United Kingdom, x.zhang15@leeds.ac.uk

*correspondence: M.A.Iqbal1@leeds.ac.uk

Digital agriculture is essential for enhancing crop yields by integrating modern digital methods to prevent and manage crop diseases. To address this, a deep learning-based Confidence-Aware Multi-Model Image Classification (CAMIC) framework has been developed. CAMIC incorporates FD-Net (Foliar Disease Network) to enable early detection and identification of various plant foliar diseases. Performance testing on the public PlantVillage dataset demonstrated that CAMIC can achieve a high accuracy of up to 97.91%, outperforming existing transfer learning models like ResNet, Inception, Xception, MobileNet, and EfficientNet. This solution has also been implemented as an Android application following the client-server model paradigm.

Keywords: digital plant protection; deep learning; foliar disease; convolutional neural network; transfer learning algorithms

Plant diseases pose a significant threat to crop health that ultimately affects both farmers and the agricultural sector as a whole. Coupled with limited technical support, the high cost of disease management often leads to suboptimal crop protection and reduced crop productivity (Ferentinos, 2018; Saleem et al., 2019; Joseph et al., 2023). The swift spread of plant diseases can cause significant damage to crop yields in a short period, even decimate entire crops in some instances (Yadav et al., 2022). Relying on expert analysis, the traditional methods for detecting plant diseases are largely manual and take considerable time. These approaches often face challenges due to the subtle, complex, and similar symptoms of various diseases (Pujari et al., 2014). Additionally, the differences in disease spot size, shape, and colour on leaves (even within the same disease) complicate early disease detection. Environmental factors (i.e., lighting, shadows, brightness, soil conditions, etc.) can further impact detection accuracy. Due to the limitations of these methods, untrained farmers may resort to high doses of pesticides when diseases progress unchecked. This can cause a significant reduction in crop production and severe environmental pollution (Richard et al., 2022). Fortunately, the application of deep learning (e.g., Convolutional Neural Network (CNN)) in computer vision has led to significant advancements, enabling improved accuracy in diagnosing plant diseases (Shi et al., 2023; Sarkar et al., 2023; Demilie, 2024; Iftikhar et al., 2024). By training a CNN on extensive datasets containing images of both healthy and diseased plants, various crop diseases can be effectively identified, offering new potential in agricultural disease management (Toda and Okura, 2019; Chen et al., 2020; Pandian et al., 2022). Therefore, this work aims to develop a user-friendly application based on CNN to detect and identify plant diseases. The main contributions of this paper are as follows:

- Leveraged GenerateShadow, a data augmentation technique, to improve plant disease prediction while considering real-world constraints.
- Proposed Confidence-Aware Multi-Model Image Classification (CAMIC) framework based on the Confidence Filter method to enhance plant disease detection accuracy by eliminating uncertain or unreliable predictions.
- Implementation of Multi-model FD-Net which is based on CAMIC to improve crop disease detection.
- Development of an end-to-end system, providing a lightweight mobile application to automate the detection of foliar diseases in grapes.

The rest of the article is structured as follows. The Materials and Methods Section describe the dataset description, basic and proposed augmentation techniques, data splitting, and our proposed CAMIC framework. Experimental results and overall performance comparison with classical transfer learning algorithms have been elaborated in the Results and Discussion Section. The section Conclusion concludes the work with the future perspectives.

Material and methods

Dataset description

The dataset used in this project is the PlantVillage dataset (taken from the Kaggle platform). It is a publicly available image dataset containing over 54,000 images of plant leaves, stems, and roots, representing 14 plant species (Hughes and Salathé, 2015). It includes images of plants affected by various diseases and each image in the dataset is labelled with one of 38 different plant diseases or conditions

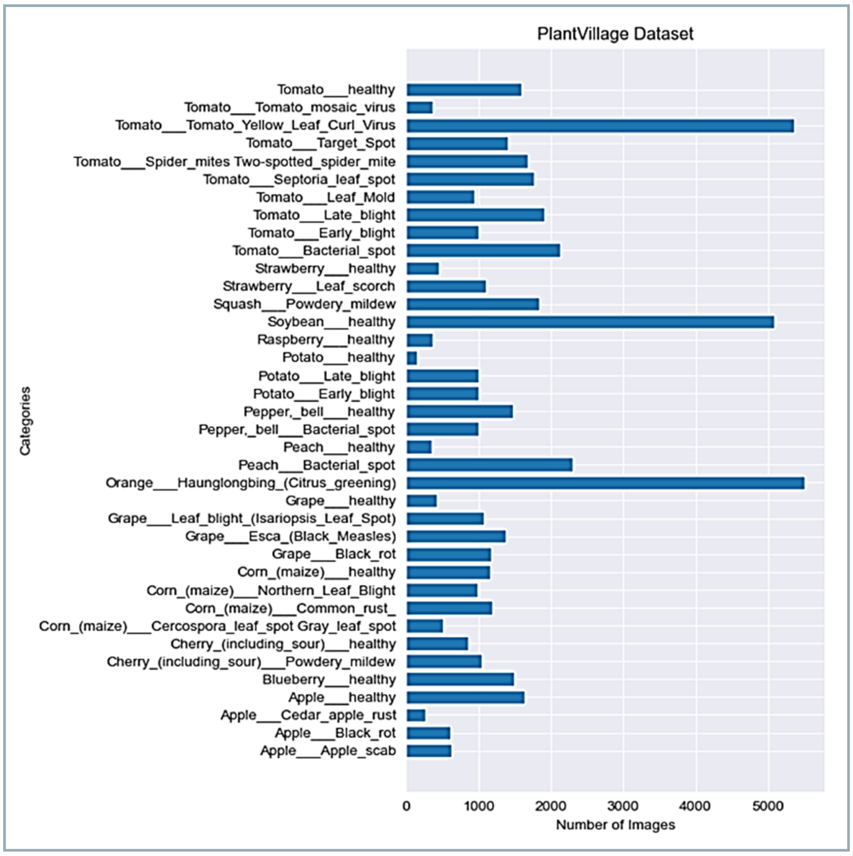


Fig. 1 Class distribution of PlantVillage dataset

(i.e., bacterial spots, yellow leaf curl virus, and powdery mildew, etc.). It also includes the images of healthy plants for each species to serve as a reference for comparison. It aims to help researchers, plant pathologists, and farmers to identify plant diseases and take necessary actions to control them. The class distribution of the PlantVillage dataset (shown in Fig. 1) is determined by domain experts who identify the specific plant species and diseases in each image. It can be seen that 31 out of 38 classes contain over 500 images, which meets the requirement to train deep learning models. However, data augmentation methods can improve the model performance.

Basic and proposed augmentation techniques

Data augmentation is a technique used in deep learning to increase the amount and diversity of training data by applying various transformations to existing data, such as flipping, rotating, cropping, adding noise, or changing the colour (Chlap et al., 2021; Maharana et al., 2022). The class distribution of the PlantVillage dataset is imbalanced. To improve the model performance, we utilised data augmentation to enhance generalisation, reduce overfitting, and better model performance on small datasets. For the designed model, we have used augmentation methods including random-flip, random-rotation, and random-crop, as shown in Fig. 2. However, in the field study, shadows can have a negative impact on computer vision tasks such as object detection and recognition that rely on CNN models. First, shadows can obscure details of an object, making it harder for the model to recognise it accurately. This can lead to a decrease in the overall accuracy of the model. Second, shadows can create patterns that resemble object features, leading the model to incorrectly classify non-object areas as objects, resulting in false positives. We developed a new data augmentation method named GenerateShadow to encounter such a phenomenon. The method selects a random rectangular region in the image and reduces its brightness to mimic shadow-like effects, as shown

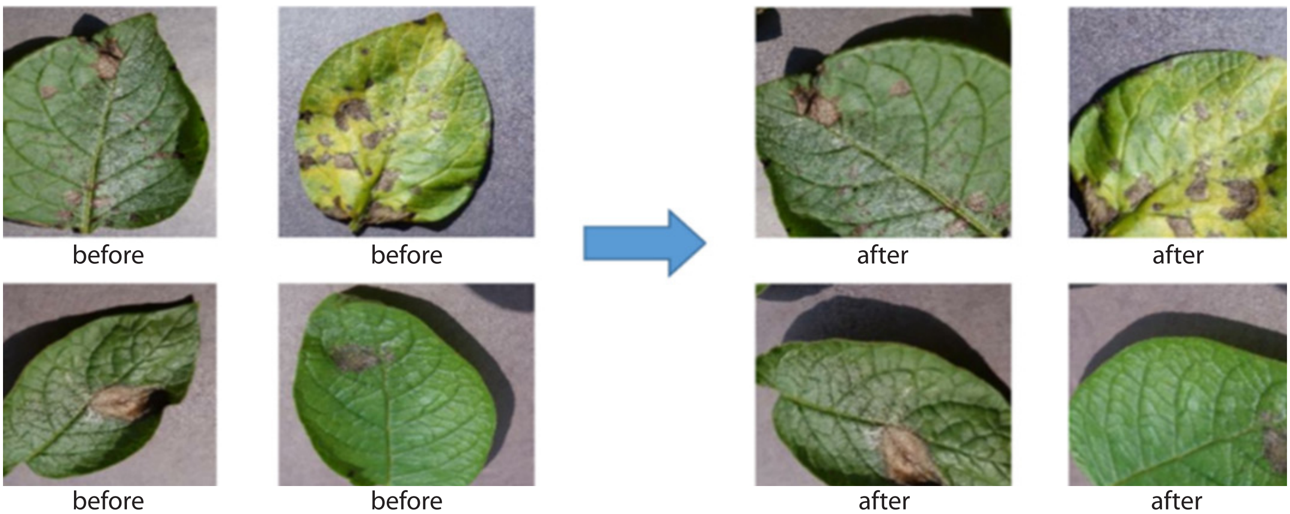


Fig. 2 Effects of traditional augmentation techniques

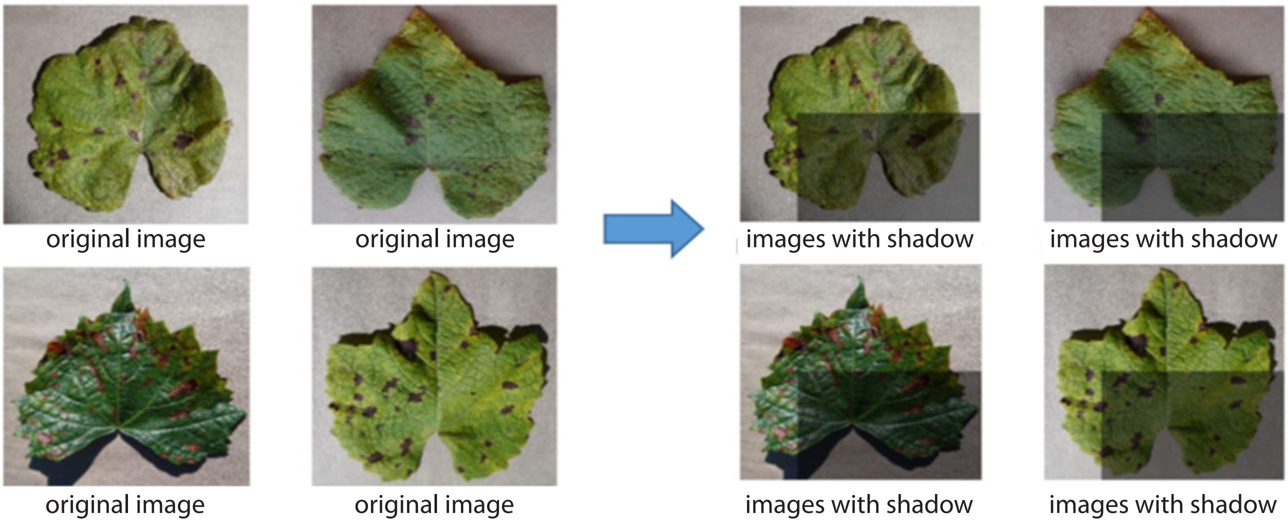


Fig. 3 Effects of GenerateShadow augmentation

in Fig. 3. The pixel values in the selected region are scaled by a random factor α (0 < α <1):

$$I'(x,y) = \begin{cases} \alpha \cdot I(x,y) & \mathit{if}\,(x,y) \in S(x,y) \\ I(x,y) & \mathit{otherwise} \end{cases} \tag{1}$$

where: $I\,(x, y)$ – represents the original image; $S\,(x, y)$ – defines the shadowed region; $I'\,(x, y)$ – the modified image

This ensures smooth integration of shadows while maintaining the overall image structure, as illustrated in Fig. 3. Integrated into the data augmentation pipeline, GenerateShadow enables the model to generate images that simulate varying lighting conditions, ultimately enhancing the dataset for training diversity.

Data splitting

In this project, we have taken Grape Foliar images as experiment objects and built the Grape dataset consisting of four categories (i.e., Grape Black Rot, Grape Black Measles, Grape Healthy, and Grape Leaf Blight). After using data augmentation for balancing, we got 1000 images for training and validation for each class, while 90% were used for training and 10% for validation. Other images (shown in Table 1) are used to test model performance.

Confidence-Aware Multi-Model Image Classification (CAMIC)

In deep learning, "confidence" generally refers to the model's certainty or probability of its prediction for a given input and is typically represented as a score between 0 and 1, where 0 indicates the model is entirely uncertain and 1 indicates the model is highly confident in its prediction. During the training process, the model is trained for the training dataset to obtain the weights of the fully connected layers. This model (based on the weights) is used for the recognition and the results are calculated through the activation function, convolution layers, and pooling layers to obtain the predicted values of each result. Finally, after applying the Softmax function, we obtain a normalised probability distribution over the possible classes (as shown in Fig. 4), where each class's probability represents the model's confidence in its prediction. The Softmax function transforms the raw logits Z_i into probability $P(y_i|x)$, which can be expressed as:

$$P(y_i|x) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{2}$$

where: $P\,(y_i|x)$ – the probability for class y_i; C denotes the total number of classes

The confidence of the prediction is then defined as the probability corresponding to the predicted class, which is the one with the highest value:

$$\hat{y} = \operatorname{argmax} P(y_i|x) \tag{3}$$

This confidence score reflects the model's certainty in its decision. As illustrated in Fig. 4, the predicted class with the highest probability is selected, and this value serves as the final confidence measure of the model's prediction.

Table 1 Data splitting results

Class	Original images	Training	Validation	Testing
Grape Black Rot	1180	900	100	195
Grape Black Measles	1383	900	100	273
Grape Healthy	423	900	100	221
Grape Leaf Blight	1076	900	100	221

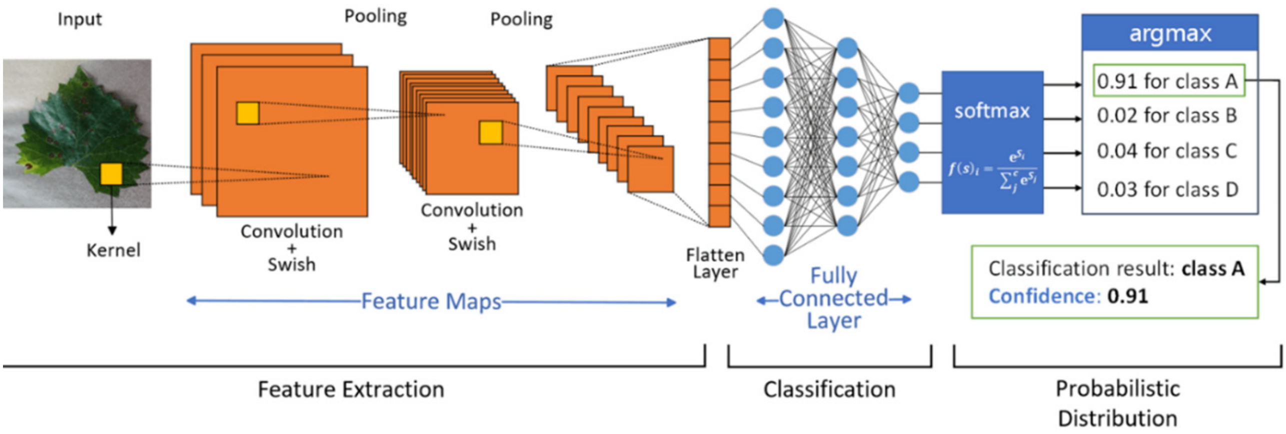


Fig. 4 Confidence illustration

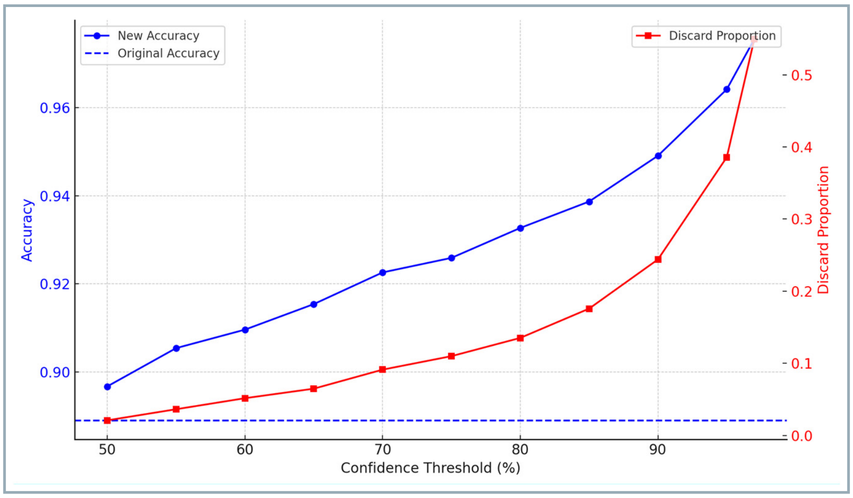


Fig. 5 Impact of confidence threshold on accuracy and discard proportion

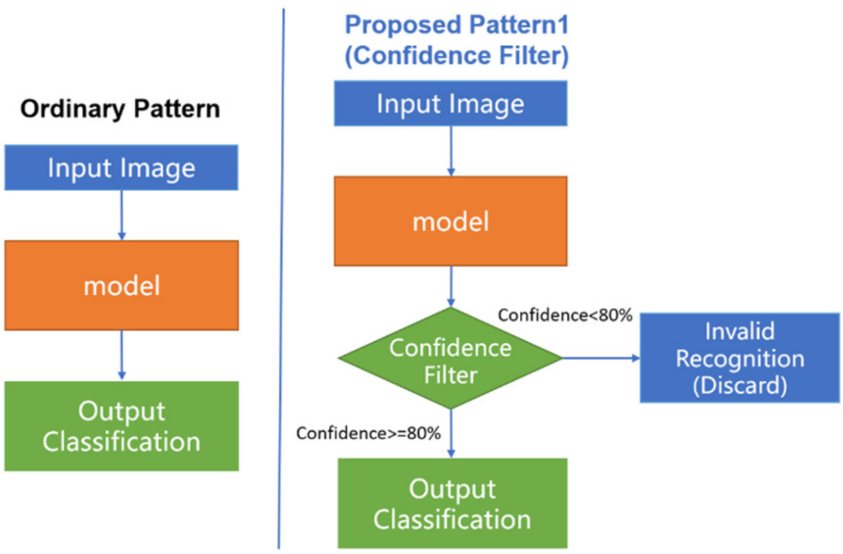


Fig. 6 Flow chart of confidence filter image classification

Concerning the analysis of the classification result, we found that wrong predictions are often in low confidence. Therefore, we developed a Confidence Filter (CF) mechanism to validate this phenomenon. In each recognition, if the confidence was lower than 80%, we supposed that the model could not identify the sample accurately, and then discarded this result.

The choice of an 80% confidence threshold was validated through a series of experiments with varying threshold values, as shown in Fig. 5. As the threshold increases, the model accuracy improves while the proportion of discarded predictions increases. At the 80% threshold, the accuracy was increased from 88.90% to 93.27%, with only 13.52% of predictions discarded. This threshold can strike a good balance between classification accuracy and discard proportion, which allows the model to retain a majority of predictions while significantly enhancing accuracy. The flowchart of this recognition pattern compared to the ordinary pattern is shown in Fig. 6. To validate the effectiveness of the CF approach, we utilised state-of-the-art transfer learning models (i.e., ResNet, Inception, MobileNet, Xception, and EfficientNet-B3) on the Grape dataset. A brief description of these transfer learning models is given below.

ResNet is a type of CNN that utilising "identity mappings" assists the training of very deep neural networks (He et al., 2016; Koonce, 2021). The Inception CNN technique is designed to extract multi-scale features efficiently (Szegedy et

Table 2 Validation of CF

Model	Original accuracy	Accuracy after CF	Confusion matrix after CF	Discard matrix	Discard proportion
ResNet50	72.20%	82.24%	[[3, 78, 0, 8], [0, 265, 0, 0], [0, 22, 103, 1], [0, 10, 0, 180]]	[[10, 67, 0, 29], [0, 7, 0, 1], [0, 37, 57, 1], [0, 9, 0, 22]]	26.37%
MobileNet-V2	79.01%	82.53%	[[182, 0, 1, 0], [140, 54, 1, 0], [0, 0, 221, 0], [0, 0, 0, 214]]	[[10, 1, 1, 0], [35, 36, 7, 0], [0, 0, 0, 0], [3, 0, 2, 2]]	10.66%
MobileNet-V3	95.16%	96.44%	[[168, 11, 0, 0], [14, 235, 0, 1], [0, 0, 219, 2], [2, 1, 0, 217]]	[[10, 6, 0, 0], [7, 16, 0, 0], [0, 0, 0, 0], [0, 0, 0, 1]]	4.5%
Inception-V3	88.79%	95.81%	[[73, 14, 0, 2], [6, 183, 0, 0], [0, 0, 221, 0], [0, 0, 6, 163]]	[[71, 32, 2, 1], [22, 59, 1, 2], [0, 0, 0, 0], [4, 1, 9, 38]]	26.59%
Xception	96.37%	98.47%	[[176, 5, 0, 0], [5, 250, 0, 0], [0, 0, 221, 0], [2, 0, 1, 187]]	[[9, 4, 1, 0], [6, 12, 0, 0], [0, 0, 0, 0], [6, 1, 2, 22]]	6.92%
EfficientNet-B3	97.47%	98.72%	[[150, 11, 0, 0], [0, 261, 0, 0], [0, 0, 221, 0], [0, 0, 0, 218]]	[[25, 9, 0, 0], [3, 9, 0, 0], [0, 0, 0, 0], [0, 0, 0, 3]]	5.38%

al., 2015; Baheti, 2023). The lightweight MobileNet CNN is specifically designed for mobile applications that utilises depth-wise separable convolutions to reduce computational complexity (Howard et al., 2017). Xception is an extension of the Inception techniques that are designed to improve the performance of image classification tasks by replacing the standard Inception modules with depth-wise separable convolutions that use fewer parameters (Chollet, 2017; Shinde and Ambhaikar, 2024). EfficientNet considers a novel scaling method that takes into account depth, width, and resolution systematically to achieve high accuracy (Tan and Le, 2019; Krishna et al., 2025).

After comparing the overall accuracy of the original models with those incorporating CF, the validation results are presented in Table 2. The axes of the confusion matrix and discard matrix in Table 2 are ordered by the four classes of grape leaf symptoms: Grape Black Rot, Grape Black Measles, Grape Healthy, and Grape Leaf Blight. The discard matrix represents the number of images for which confidence is lower than 80% and is treated as invalid recognition. The confusion matrix (after CF) added to the discard matrix equals the original confusion matrix of the models. This approach has been proven effective through our experiments and the results show that the CF can effectively filter a large quantity of error recognition and improve the model accuracy significantly, especially for the models having overall accuracy lower than 90%.

Bi-CF model

The CF can filter some error classifications, but it also means that the model cannot provide classification results for all the images since it discards a part of the results. To compensate this drawback, we developed a Bi-CF model recognition pattern. In the first step, the first model received the input image and identified it normally. Then, it checks whether the confidence of the result exceeds 80%. If it is, this recognition result will be considered valid. If the confidence is lower than 80%, the recognition will be treated as invalid, and the image will be delivered to model-2 to decide. Model-2 is trained using the same dataset and method as model-1, but with either a different model architecture or a different training batch. This provides diversity in predictions, improving the overall reliability of the system. Then, the second model will provide the final judgement. The Bi-CF model recognition process can be expressed mathematically as:

$$y = \begin{cases} M_1(x) & \text{if } C\left(M_1(x)\right) \geq \tau \\ M_2(x) & \text{if } C\left(M_1(x)\right) < \tau \end{cases} \tag{4}$$

where: x – represents the input image; M_1 (x) and M_2 (x) denote the classification results of model-1 and model-2, respectively; C ($M_1(x)$) – the confidence score of model-1's prediction; τ – the confidence threshold (set at 80% in this study)

The flow chart of Bi-CF model recognition is shown in Fig. 7, and the related experiment results are shown in Table 3.

This approach has been proven effective through our experiments, as shown in Table 3. Table 3 contains the name and accuracy of the two models that are set

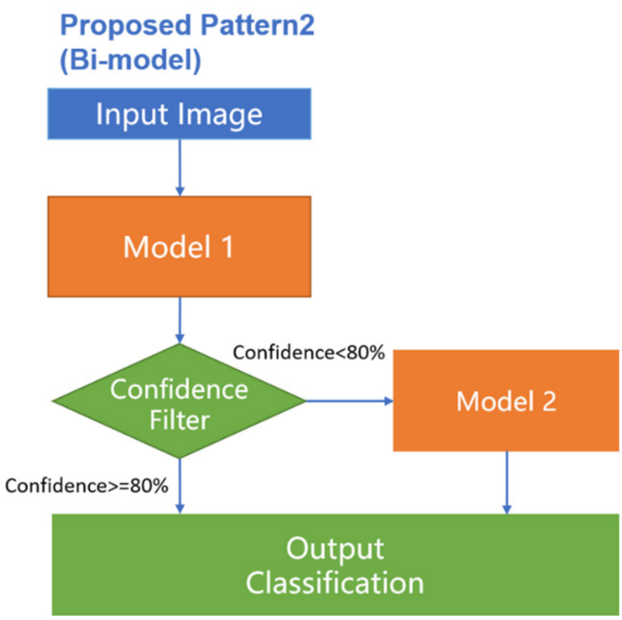


Fig. 7 Bi-CF model image classification flowchart

Table 3 Validation and comparison of bi-model recognition pattern

Model 1	Model 1 accuracy	Model 2	Model 2 accuracy	Bi-model accuracy
Inception-V3	88.79%	**Inception-V3**	88.90%	89.34%
Xception	96.37%	**Xception**	93.63%	96.59%
EfficientNet-B3	96.59%	**EfficientNet-B3**	97.41%	97.80%
Inception-V3	88.79%	**Xception**	93.63%	92.64%
Xception	96.37%	**EfficientNet-B3**	96.26%	97.91%
MobileNet-V2	92.53%	**Xception**	93.63%	95.16%

in the architecture, compared to the whole pattern recognition accuracy. Bi-CF model detection accuracy outperforms either of the two models. This architecture complements the shortcomings of the first pattern, as all the images will be included in the results. Moreover, it can be observed that the Bi-CF model architecture leads to better accuracy when two models have a small accuracy difference.

Tri-CF model

Considering our experimental results, we also observed that the model has difficulty in identifying the Grape Black Rot and Grape Black Measles (as it shows a certain probability of recognising Grape Black Rot as Grape Black Measles). In contrast, it is easy for the model to distinguish the other two classes. Therefore, we have proposed a Tri-CF model architecture, which is shown in Fig. 8.

Getting an intermediate result after the confidence-aware modules (model 1 and model 2), if the result is Grape Black Measles, it has a certain likelihood that it is actually Grape Black Rot. Hence, we set a third model which is specifically used for binary classification of these two disease types. Here is the mathematical representation of the Tri-CF model based on the proposed architecture and its logic is:

$$y=\begin{cases} M_1(x) & \text{if } C\left(M_1(x)\right)\geq\tau \\ M_2(x) & \text{if } C\left(M_1(x)\right)<\tau \text{ and } M_2(x)\neq grape-black-measles \\ M_3(x) & \text{if } C\left(M_1(x)\right)<\tau \text{ and } M_2(x)\neq grape-black-measles \end{cases} \qquad (5)$$

where: x – represents the input image; $M_1(x)$, $M_2(x)$, $M_3(x)$ – denote the classification results of model-1 model-2, and model-3 respectively; $C(M_1(x))$ – the confidence score of model-1's prediction; τ – the confidence threshold (set at 80% in this study)

The effectiveness of the model is shown in Table 4. In most cases, the Tri-CF model outperforms the original models, as long as the three models have close accuracy. The best combination is Xception with EfficientNet-B3 which achieves an accuracy of 97.80%.

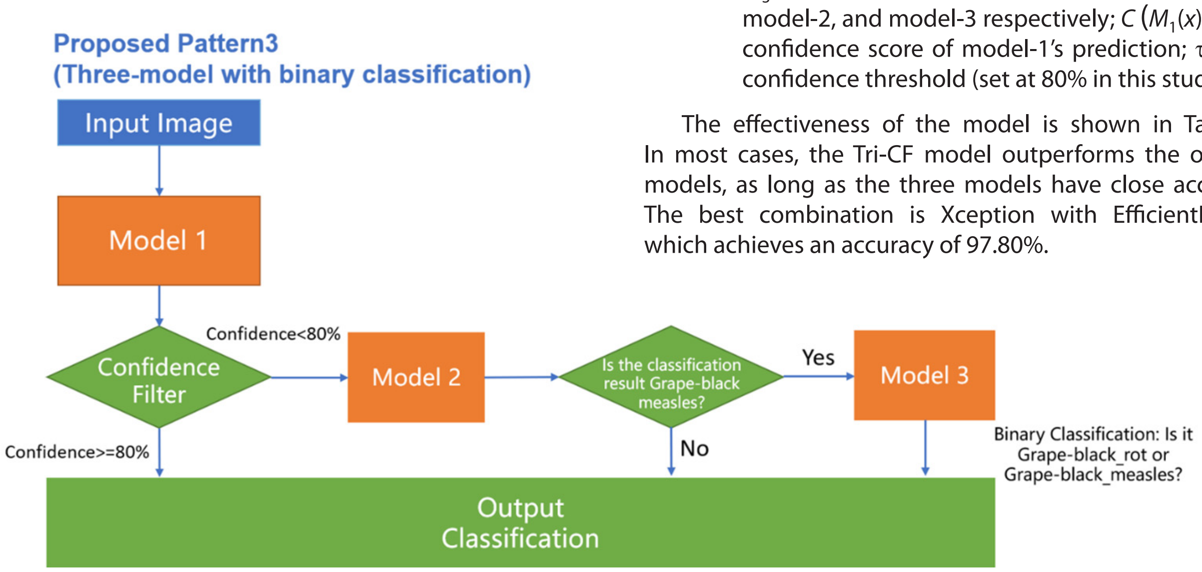


Fig. 8 Tri-CF model image classification flowchart

Table 4 Validation and comparison of Tri-model recognition pattern

Model 1	Model 1 accuracy	Model 2	Model 2 accuracy	Model 3	Model 3-Bi-classification accuracy	Three-model accuracy
Inception-V3	88.79%	**Inception-V3**	88.90%	**Xception**	86.75%	91.21%
Xception	96.37%	**Xception**	93.63%	**Xception**	86.75%	96.59%
EfficientNet-B3	97.47%	**EfficientNet-B3**	96.26%	**EfficientNet-B3**	94.02%	97.58%
Inception-V3	88.79%	**Xception**	93.63%	**Xception**	86.75%	91.76%
Xception	96.37%	**EfficientNet-B3**	96.26%	**EfficientNet-B3**	94.02%	97.80%
MobileNet-V2	92.53%	**Xception**	93.63%	**EfficientNet-B3**	94.02%	95.82%

Foliar Disease Neural Network (FD-Net)

This section presents a model named Foliar Disease Neural Network (FD-Net), which is an instance of CAMIC. The FD-Net model under the CAMIC architecture is based on the EfficientNet-B3 model. EfficientNet is considered one of the best models in the field due to its high accuracy on the ImageNet classification task with 66 M parameters. As already mentioned, EfficientNet uses a new scaling method that uniformly scales all dimensions of depth, width, and resolution using a simple yet highly effective compound coefficient (Tan and Le, 2019). Its architecture comprises eight layers ranging from B0 to B7. As the number of layers increases, the number of computed parameters remains relatively constant while accuracy improves significantly. In this study, the EfficientNet-B3 model was employed as the basis of FD-Net with the distinction of using Softmax activation function (replacing swish of original EfficientNet) as the best results under various experimental conditions and hardware constraints observed. The architecture of the proposed FD-Net is illustrated in Fig. 9.

The model accepts the image dataset as input. In the next step, the image data is pre-processed and augmented through a series of transformations to create the final image data. The model then receives this processed image data, along with the corresponding metadata regarding the input image dataset. Subsequently, the FD-Net model processes the collected input. A dense layer comprising four units and a Softmax activation function is employed as the final layer.

Results and discussion

The overall classification performance of the FD-Net model using the Grapes leaf dataset is evaluated in terms of precision, recall, and *F*1 score (results are shown in Table 5). Precision evaluates the accuracy of positive predictions by taking the ratio of true positives (*TP*) to all predicted positives, including true positives and false positives (*TP* + *FP*). Recall measures the model's ability to identify true positives out of all *TP* and *FN* (where *FN* stands for false negatives and represents actual positives that are misclassified as negative). The *F*1 score representing the harmonic mean of precision and recall helps to evaluate a balanced assessment of classification performance. The formulae of precision, recall, and *F*1 score are given below.

$$precision = \frac{TP}{TP+FP}, \quad recall = \frac{TP}{TP+FN}, \tag{6}$$
$$F1-score = 2\frac{precision \cdot recall}{precision + recall}$$

However, the detailed performance comparison of both variants of the proposed FD-Net model (i.e., FD-Net-B (FD-Net with Bi-CF model) and FD-Net-T (FD-Net with Tri-CF model) with several state-of-the-art transfer learning CNN models (using the Grape dataset) is shown in Table 6. Table 6 illustrates that the proposed model variants (FD-Net-B

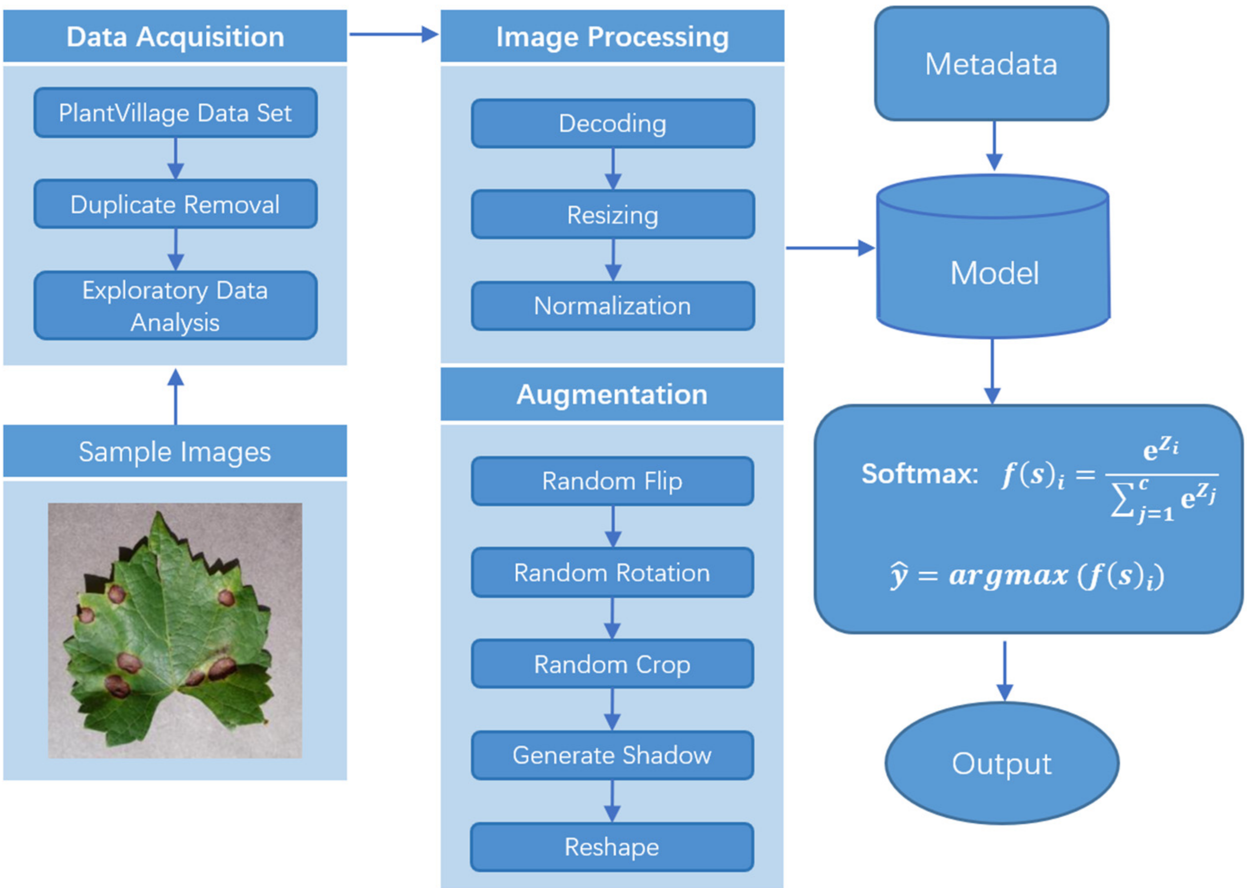


Fig. 9 FD-Net flowchart

Table 5 Classification performance

Class	Precision	Recall	*F*1 score
Grape Black Rot	0.9831	0.8974	0.9383
Grape Black Measles	0.9310	0.9890	0.9591
Grape Healthy	1.0000	1.0000	1.0000
Grape Leaf Blight	1.0000	1.0000	1.0000
Macro-average	0.9785	0.9716	0.9750

Table 6 Performance comparison of FD-Net and state-of-the-art transfer learning models

Model	Accuracy	*F*1 score	Model	Accuracy	*F*1 score
ResNet50	72.20%	0.8452	Xception	96.37%	0.9487
MobileNetV2	79.01%	0.8429	EfficientNetB3	97.47%	0.9752
MobileNetV3	95.16%	0.9518	FD-Net-B	97.91%	0.9789
InceptionV3	88.79%	0.9036	FD-Net-T	97.80%	0.9789

and FD-Net-T) outperformed existing approaches in terms of accuracy and *F*1 score.

System architecture and mobile interfaces

Based on the proposed FD-Net, a client-server system (mobile application with Windows server) has been implemented. The working principle of the system (shown in Fig. 10) is explained below.

Steps 1–3: The Android application on the mobile phone is developed to take photos or select one photo from the album and send it to the server via the gateway for further processing.

Steps 4–5: Upon receiving the photo, the server's communication module invokes the deep learning module based on the proposed FD-Net, which ultimately processes the images to identify crop/plant disease categories and is responsible for sending results back to the phone to display the results to the user. The mobile application interfaces are shown in Fig. 11.

Several key technical aspects, i.e., image preprocessing in terms of image resizing, data transmission efficiency by using Base64 encoding for image transmission, multithreading to handle simultaneous requests from multiple users, and the provisioning of precise geographical information using FusedLocationProviderClient API have been considered during the development of this client-server application.

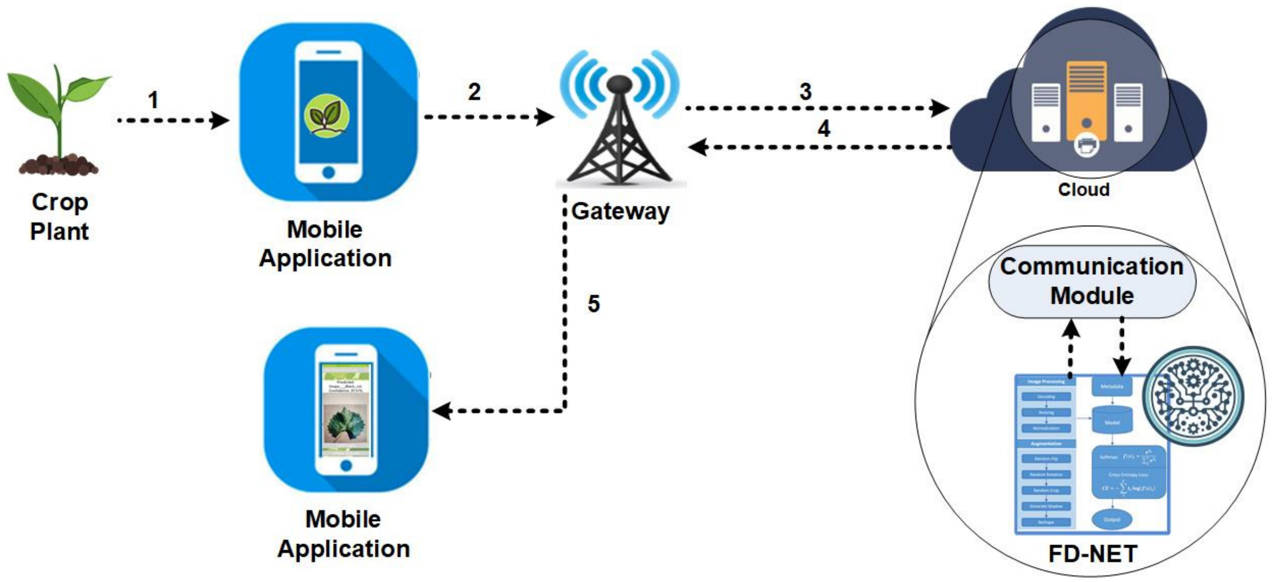
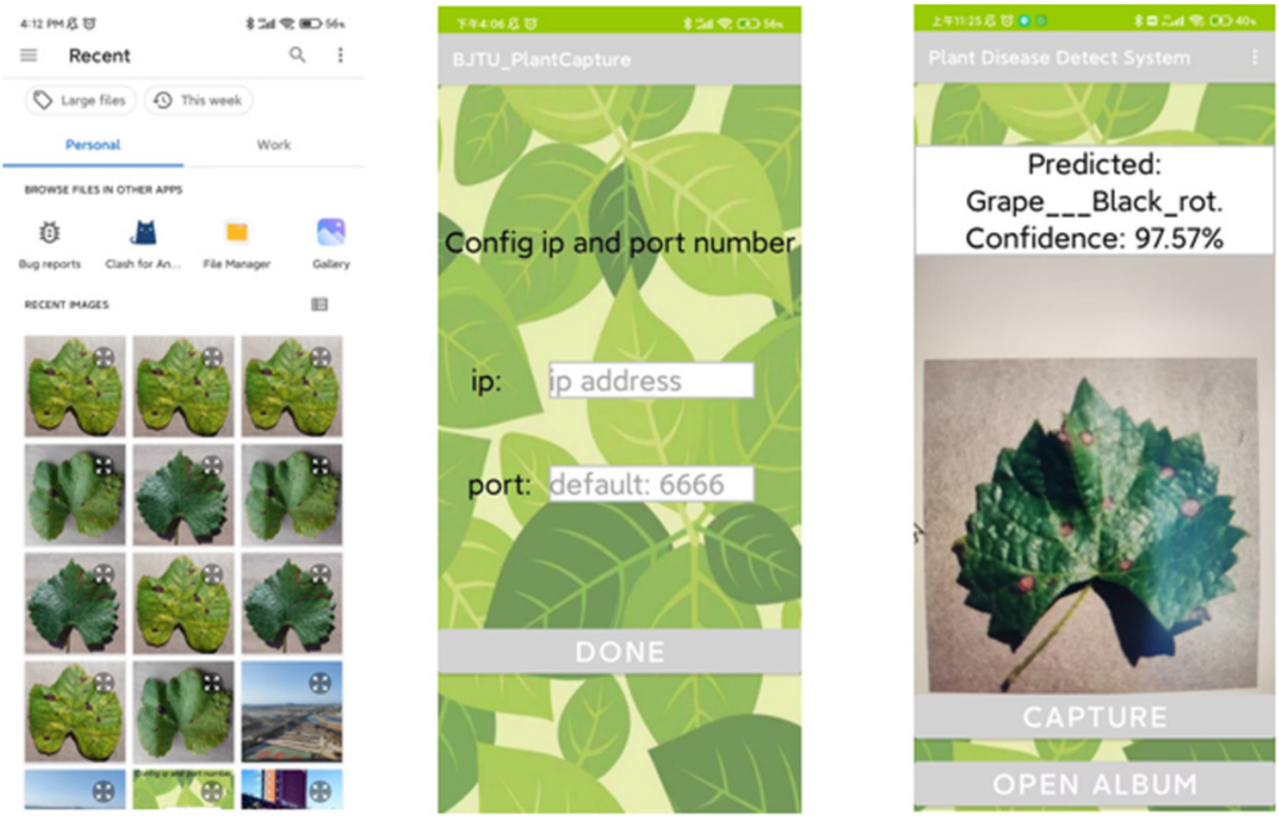


Fig. 10 End-to-end system architecture



Open Album Network Configure Result Display

Fig. 11 Mobile application interfaces

Conclusion

This study proposes the CAMIC framework, an end-to-end system designed to detect and classify plant diseases effectively. The framework uses the shadow augmentation method and Bi-CF and Tri-CF classification techniques to improve the accuracy and reliability of plant disease classification by reducing errors and improving prediction confidence. The experimental results show that the CAMIC framework performs better than existing transfer learning-based CNN models. It achieves an accuracy rate of 97.91% and an *F*1 score of 0.9789, demonstrating its strong ability to classify plant diseases accurately. Future work can explore integrating more functions, such as data analysis and treatment suggestions, to make the system even more useful and inclusive.

References

BAHETI, H. – THAKARE, A. – BHOPLE, Y. – DAREKAR, S. – DODMANI, O. 2023. Tomato plant leaf disease detection using Inception v3. In KULKARNI, A. J. – MIRJALILI, S. – UDGATA, S.K. (eds.). Intelligent Systems and Applications. Lecture Notes in Electrical Engineering, vol. 959. Singapore : Springer, pp. 49–60. eISBN 978-981-19-6581-4. DOI: https://doi.org/10.1007/978-981-19-6581-4_5

CHLAP, P. – MIN, H. – VANDENBERG, N. – DOWLING, J. – HOLLOWAY, L. – HAWORTH, A. 2021. A review of medical image data augmentation techniques for deep learning applications. In Journal of Medical Imaging and Radiation Oncology, vol. 65, no. 5, pp. 545–563. DOI: https://doi.org/10.1111/1754-9485.13261

CHEN, J. – ZHANG, D. – SUZAUDDOLA, M. – NANEHKARAN, Y. A. – SUN, Y. 2020. Identification of plant disease images via a squeeze-and-excitation MobileNet model and twice transfer learning. In IET Image Processing, vol. 15, no. 5, pp. 1115–1127. DOI: https://doi.org/10.1049/ipr2.12090

CHOLLET, F. 2017. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1800–1807. DOI: https://doi.org/10.1109/CVPR.2017.195

DEMILIE, W. B. 2024. Plant disease detection and classification techniques: A comparative study of the performances. In Journal of Big Data, vol. 11, article no. 5. DOI: https://doi.org/10.1186/s40537-023-00863-9

FERENTINOS, K. P. 2018. Deep learning models for plant disease detection and diagnosis. In Computers and Electronics in Agriculture, vol. 145, pp. 311–318. DOI: https://doi.org/10.1016/j.compag.2018.01.009

HE, K. – ZHANG, X. – REN, S. – SUN, J. 2016. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. DOI: https://doi.org/10.1109/CVPR.2016.90

HOWARD, A. G. – ZHU, M. – CHEN, B. – KALENICHENKO, D. – WANG, W. – WEYAND, T. – ANDREETTO, M. – ADAM, H. 2017. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861. DOI: https://doi.org/10.48550/arXiv.1704.04861

HUGHES, D. P. – SALATHÉ, M. 2015. An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv preprint arXiv:1511.08060. DOI: https://doi.org/10.48550/arXiv.1511.08060

IFTIKHAR, M. – KANDHRO, I. A. – KAUSAR, N. – KEHAR, A. – UDDIN, M. – DANDOUSH, A. 2024. Plant disease management: A fine-tuned enhanced CNN approach with mobile app integration for early detection and classification. In Artificial Intelligence Review, vol. 57, article no. 167. DOI: https://doi.org/10.1007/s10462-024-10809-z

JOSEPH, D. S. – PAWAR, P. M. – PRAMANIK, R. 2023. Intelligent plant disease diagnosis using convolutional neural network: A review. In Multimedia Tools and Applications, vol. 82, pp. 21415–21481. DOI: https://doi.org/10.1007/s11042-022-14004-6

KOONCE, B. 2021. ResNet 50. In Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization. Berkeley, CA : Apress, pp. 63–72. eISBN 978-1-4842-6168-2. DOI: https://doi.org/10.1007/978-1-4842-6168-2_6

KRISHNA, M. S. – MACHADO, P. – OTUKA, R. I. – YAHAYA, S. W. – NEVES DOS SANTOS, F. – IHIANLE, I. K. 2025. Plant leaf disease detection using deep learning: A multi-dataset approach. In Multidisciplinary Scientific Journal vol. 8, no. 1, article no. 4. DOI: https://doi.org/10.3390/j8010004

PANDIAN, J. A. – KUMAR, V. D. – GEMAN, O. – HNATIUC, M. – ARIF, M. – KANCHANADEVI, K. 2022. Plant disease detection using deep convolutional neural network. In Applied Sciences, vol. 12, no. 14, article no. 6982. DOI: https://doi.org/10.3390/app12146982

MAHARANA, K. – MONDAL, S. – NEMADE, B. 2022. A review: Data pre-processing and data augmentation techniques. In Global Transitions Proceedings, vol. 91, no. 1, pp. 91–99. DOI: https://doi.org/10.1016/j.gltp.2022.04.020

PUJARI, J. D. – YAKKUNDIMATH, R. – BYADGI, A. S. 2014. Recognition and classification of produce affected by identically looking powdery mildew disease. In Acta Technologica Agriculturae, vol. 17, no. 2, pp. 29–34. DOI: https://doi.org/10.2478/ata-2014-0007

RICHARD, B. – QI, A. – FITT, B. D. L. 2022. Control of crop diseases through Integrated Crop Management to deliver climate-smart farming systems for low- and high- input crop production. In Plant Pathology, vol. 71, no. 1, pp. 187–206. DOI: https://doi.org/10.1111/ppa.13493

SALEEM, M. H. – POTGIETER, J. – ARIF, K. M. 2019. Plant disease detection and classification by deep learning. In Plants, vol. 8, no. 11, article no. 468. DOI: https://doi.org/10.3390/plants8110468

SARKAR, C. – GUPTA, D. – GUPTA, U. – HAZARIKA, B. B. 2023. Leaf disease detection using machine learning and deep learning: Review and challenges. In Applied Soft Computing, vol. 145, article no. 110534. DOI: https://doi.org/10.1016/j.asoc.2023.110534

SHI, T. – LIU, Y. – ZHENG, X. – HU, K. – HUANG, H. – LIU, H. – HUANG, H. 2023. Recent advances in plant disease severity assessment using convolutional neural networks. In Scientific Reports, vol. 13, article no. 2336. DOI: https://doi.org/10.1038/s41598-023-29230-7

SHINDE, N. – AMBHAIKAR, A. 2024. Fine-tuned Xception model for potato leaf disease classification. In TANWAR, S. – SINGH, P. K. – GANZHA, M. – EPIPHANIOU, G. (eds). Proceedings of the Fifth International Conference on Computing, Communications, and Cyber-Security. IC4S 2023. Lecture Notes in Networks and Systems, vol. 991. Singapore : Springer, pp. 663–676. eISBN 978-981-97-2550-2. DOI: https://doi.org/10.1007/978-981-97-2550-2_47

SZEGEDY, C. – LIU, W. – JIA, Y. – SERMANET, P. – REED, S. – ANGUELOV, D. – ERHAN, D. – VANHOUCKE, V. – RABINOVICH, A. 2015. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9. DOI: https://doi.org/10.1109/CVPR.2015.7298594

TAN, M. – LE, Q. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the 36th International Conference on Machine Learning, vol. 97, pp. 6105–6114.

TODA, Y. – OKURA, F. 2019. How convolutional neural networks diagnose plant disease. In Plant Phenomics, vol. 2019, article no. 9237136. DOI: https://doi.org/10.34133/2019/9237136

YADAV, A. – THAKUR, U. – SAXENA, R. – PAL, V. – BHATEJA, V. – LIN, J. C. 2022. AFD-Net: Apple foliar disease multi classifcation using deep learning on plant pathology dataset. In Plant and Soil, vol. 477, no. 1, pp. 595–611. DOI: https://doi.org/10.1007/s11104-022-05407-3