Journal of Computer Science
https://dergipark.org.tr/en/pub/bbd
Anatolian Science

ISSN,e-ISSN: 2548-1304
Volume:10, Issue:2, pp:186-200, 2025
https://doi.org/10.53070/bbd.1685740
Research Paper

Early Detection of Sunflower Leaf Diseases Using Image-Based Deep Learning Methods

Talha Burak Alakuş*[1], Bora Aslan[1], Burak Beynek[1], Dilan Onat Alakuş[1], Tugay Koç[2]

[1]Department of Software Engineering, Kırklareli University, Kırklareli, Türkiye

[2]Damal Directorate of Agriculture and Forestry, Ardahan, Türkiye

(talhaburakalakus@klu.edu.tr, bora.aslan@klu.edu.tr, burakbeynek@klu.edu.tr, onatalakus.dilan@klu.edu.tr, tgy3939@gmail.com)

Received:Apr.28,2025 Accepted:Jul.19,2025 Published:Dec.01,2025

Abstract— Sunflower is a crop type that has high economic value and is also used for ornamental purposes. However, various diseases seen on sunflower leaves can disrupt production and it is difficult for growers to identify these diseases with traditional approaches. Therefore, the need for image-based artificial intelligence approaches that can automatically identify diseases seen on leaves has arisen. In this study, a system that can detect diseases seen on sunflower leaves, both image-based and artificial intelligence-supported, has been developed. The study consists of four stages. In the first stage, a publicly available dataset was used, and additional data was collected by us. In the second stage, image processing was performed. In the third stage, CNN (Convolutional Neural Network), ViT (Vision Transformer) and CNN-ViT models were designed. In the last stage, the performances of these models were evaluated, and their success was determined by accuracy, recall, precision, F1-score, Cohen Kappa and Hamming loss metrics. To improve data diversity and robustness, the dataset was enriched with real-world images collected under varying environmental conditions. The preprocessing stage included a comprehensive pipeline involving Gaussian filtering, HSV conversion, histogram equalization, Canny edge detection, and segmentation to enhance feature clarity and reduce noise. The CNN-ViT model was designed to leverage local feature extraction through convolutional layers and global feature representation via self-attention mechanisms. All models were trained and validated using standardized conditions to ensure comparability. The experimental results demonstrated that the hybrid CNN-ViT model achieved superior performance in all evaluation metrics, suggesting its potential as an effective tool in precision agriculture for early disease diagnosis.

Keywords : *Classification, image processing, sunflower disease, deep learning, image diagnosis systems*

1. Introduction

The sunflower, scientifically known as Helianthus annus, first appeared in Mexico in 2,100 BC and has been used for both economic and ornamental purposes in many countries (David et al. 2008). Many countries, especially Turkey, grow sunflowers to meet consumer demand. Since its seeds contain oil and are also consumed as food, sunflower is an economically important crop (Vorobyov et al. 2021). Furthermore, straw is obtained from sunflower leaves and yellow dye from flowers, and these provide profit to both agriculture and industry (Yuan et al. 2022; Ghosh et al. 2023). However, diseases such as downy mildew, leaf scar and gray mold seen on sunflower leaves negatively affect sunflower production (Sara et al. 2022; Malik et al. 2022). Manual control and determination of diseases seen on sunflower leaves is a time-consuming process. To determine the diseases seen on the leaves, classification is done by observation, and this is prone to error since it depends on the experience of the observer (Sathi et al. 2023). Various approaches have been proposed to overcome this problem. The most important of these approaches is the spectrometer (Sasaki et al. 1998). Healthy and unhealthy sunflower leaves can be effectively classified with the spectrometer approach. In addition, diseases can be determined with gene bioinformatics approaches (Koo et al. 2013). However, the biggest limitation of these approaches is that they are time consuming, costly, and require expert knowledge. Therefore, the need for both image-based and artificial intelligence-supported systems that can identify diseases has arisen.

In recent years, computer vision and artificial intelligence (AI) have been used effectively to identify diseases seen on sunflower leaves. In this direction, researchers frequently resort to machine learning (ML) approaches (Wu et al. 2022; Kaur et al. 2022; Centorame et al. 2024). However, in ML, feature extraction is performed to extract more perceptible patterns and features from the data, which requires expert knowledge. This can cause the process to take time. To overcome this problem, researchers have turned to deep learning (DL) algorithms, another AI approach. In DL, no expert knowledge is required for feature extraction. The biggest difference between DL and ML is that feature extraction is done automatically, that is, by the model. DL models are used effectively in various studies, especially in agriculture, such as the detection of diseases in plants, the identification of pesticide insects, and plant classification (Li et al. 2021; Zhou et al. 2021; Islam et al. 2023; Wang et al. 2022). The successes achieved with DL in the field of agriculture have led researchers to determine diseases seen in sunflower leaves with DL models. (Rani et al. 2024), used CNN and RF (Random Forest) models to classify diseases seen on sunflower leaves and designed an approach that divided plant diseases into 10 different categories. The performance of the models was determined by accuracy scores and the average accuracy was obtained as 92.19%. The study emphasized that deep learning algorithms are effective in accurate and timely disease recognition and highlighted that plant diseases should be detected at an early stage to minimize agricultural losses. (Sirohi and Malik, 2021), developed deep learning-based hybrid model for sunflower classification. The study combined VGG-16 and MobileNet models and classified four different types of diseases. The study consisted of various stages including data pre-processing, labeling, data augmentation, model training and design of hybrid model. The performance of the model was determined by accuracy score and 89.2% accuracy was achieved with hybrid approach. The study indicated that early detection of diseases through computer image analysis contributes to the protection of sunflower crop. (Sathi et al. 2023), presented a deep learning-based approach to detect sunflower diseases. A total of 1,428 sunflower images were used in the study and segmentation was performed with K-Means. Preprocessing techniques such as resizing, contrast adjustment and color enhancement were applied to the images. Four different DL models were used in the study and the performances of these models were determined by accuracy scores. As a result, the study emphasized the importance of modern technologies in agricultural practices and offered a potential solution to help farmers in recognizing diseases. (Singh 2019) proposed an innovative approach combining image segmentation and PSO (Particle Swarm Optimization) algorithm for disease detection on sunflower leaves. K-means method was used for segmentation process and images were masked with pixel masking. The method provided a practical solution for disease monitoring and control especially in large areas. (Rajora et al. 2024), various diseases on sunflower leaves were determined using the CNN-SVM (CNN-Support Vector Machine) hybrid model. The study was carried out by analyzing 9,695 images and the performance of the model was determined with accuracy. The research result emphasized that it would allow farmers to detect and intervene in diseases in a timely manner and would help them increase agricultural productivity and health.

In line with the successes obtained with DL in the literature, both DL and hybrid model techniques were used in this study to recognize sunflower diseases. The effectiveness of CNN, ViT and CNN-ViT models was investigated, and the performance of the models was determined by various evaluation criteria. The highlights of the article are as follows:

- Both DL and hybrid (CNN-VIT) architectures were used together, and a comparative analysis was performed. The contribution of the hybrid structure to the success of classification was evaluated for the first time in the literature.
- Hybrid structure, where CNN and ViT architecture is combined, is still a novel approach in the field of agricultural image classification.
- In contrast to the fact that many previous studies are based only on publicly available datasets, both public data set and original data collected in the field were used together. In this way, the generalization ability of the model was increased and made more suitable for real field conditions.
- The effect of combining data obtained from different sources on the classification performance was systematically evaluated, and in this respect, practical inferences were presented for data set design and model training processes.

The rest of the study is organized as follows: In the Material and Methods section, the data and methods used in the study are mentioned. In the Application Results and Discussion, the findings are given, and the results are interpreted. In the Conclusion, the study is concluded, and a general summary of the study is given and suggestions for future studies are presented.

2. Material and Methods

The flow chart of the study is given in Figure 1. There are two important stages of the study. These are: feature extraction and design of deep learning models. Information about each stage is given in subsections.

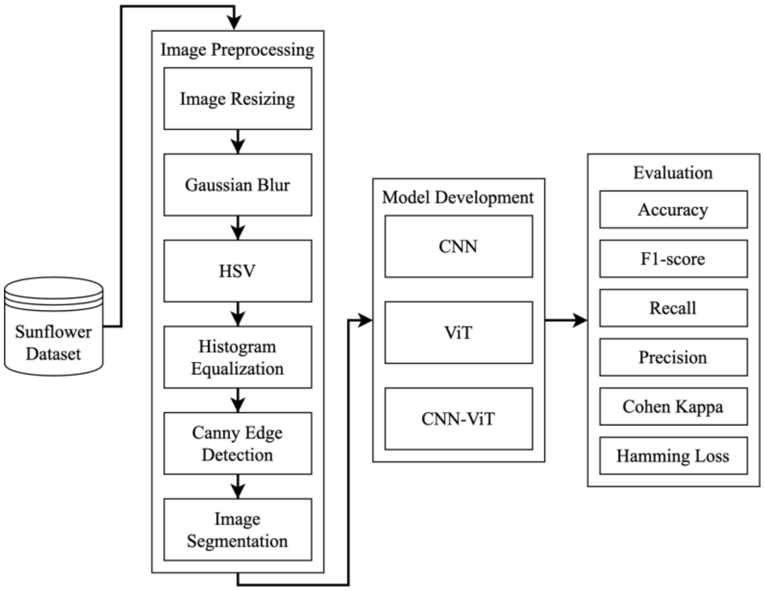


Figure 1. Flow chart of the study

2.1. Data Set

Two different data sets were used in the study. The first data set consists of images obtained from the study (Sara et al. 2022) and there are 466 images in the data set, 120 of which are downy mildew, 134 of which are fresh leaves, 72 of which are gray mildew and 140 of which are leaf marks. In addition, data augmentation was performed on the images and the number of images increased to 1,668. The second dataset was generated specifically for the study by us. Sunflower leaves were obtained from the provinces of Edirne and Kırklareli in the Thrace region with the help of a camera with high pixel resolution. There are a total of 250 images in our dataset. 100 of these images are fresh leaves, 50 are downy mildew, 70 are leaf scars, and 30 are gray mold. As in the study (Sara et al. 2022), image augmentation was performed, and the images were increased from 250 images to 875 images. During the data augmentation, rotation, scaling, and shearing, which are position-based data augmentation steps, were employed. Rotation aimed to obtain the appearance of leaf diseases from different angles. For this purpose, rotations were performed at angles of 45°, 60°, and 90° within the scope of the study. Scaling increased the image size, aiming to accurately distinguish symptoms at different proximity levels. In the scaling step, the width and height scaling were increased by 0.1 units. Finally, the shearing step distorted the image by tilting it along the axial plane. This aimed to test the robustness of the models against natural perspectives. In the shearing step, the distortion interval was set to 0.1. Various sunflower leaf images of each dataset are given in Figure 2.

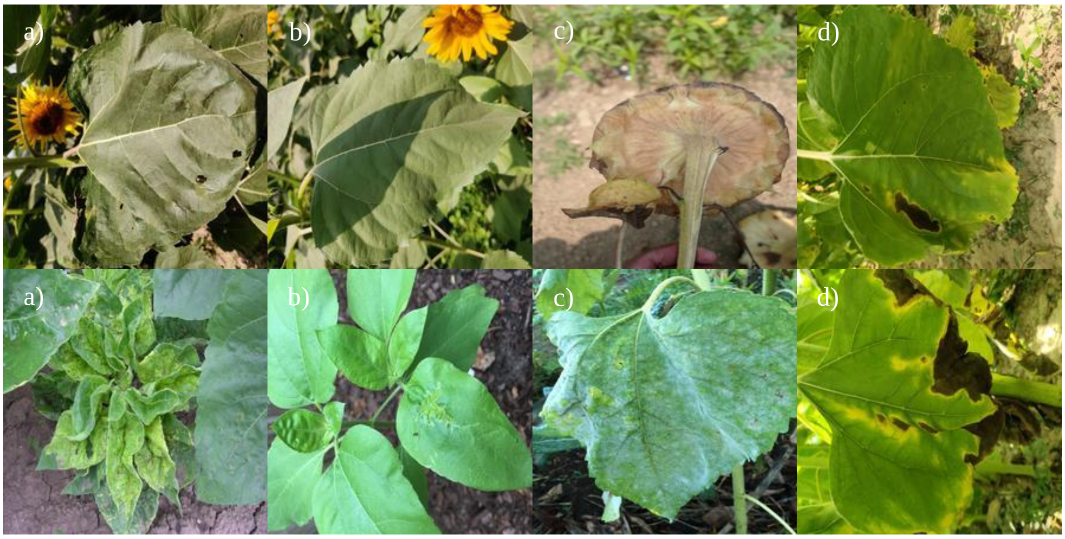


Figure 2. Leaf images belonging to the datasets. The images in the first row are from the dataset (Sara et al. 2022). The images in the second row were collected by us. The leaves are a) downy mildew, b) fresh leaf, c) gray mold and d) leaf spot, respectively

2.2. Data Preprocessing

Six different pre-processing techniques were used: image resizing, Gaussian filtering, HSV (Hue, Saturation and Value), histogram equalization, canny edge detection and segmentation. Since ViT and CNN DL models take images in 224 x 224, during the image resizing stage, the images were converted to this format. Then, Gaussian Filtering was applied to reduce noise and soften edges. After, the HSV color model, which is more suitable for the human eye's color perception and is also widely used in object detection and background separation, was employed (Andasuryani and Rasinta, 2021). Then, histogram equalization was performed to make the details in the images more distinct. In the next step, the Canny edge detection algorithm was applied to determine the edges in the images. In the last step, segmentation was performed to make it easier to recognize certain features or objects on the image. Figure 3 shows the results of applying these steps on leaf scar.

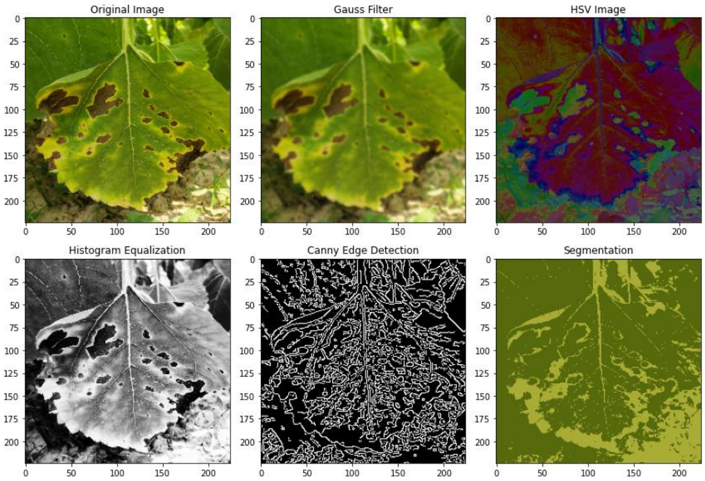


Figure 3. Implementation of image processing steps

2.3. Model Development

DL, as a sub-branch of AI and ML, aims to develop systems that can make human-like decisions using models that can process large data sets and complex structures. DL basically uses multi-layered artificial neural networks to learn. These networks are designed with inspiration from the biological nervous system and process data in layers. Each layer extracts different features from the data, producing more complex and abstract representations. DL, unlike traditional ML algorithms, has the ability to extract features directly from data (Sarhan et al. 2024). This provides a significant advantage, especially in visual data. For instance, a DL model can recognize objects, colors, and shapes in an image. This type of learning largely eliminates the need for manual feature engineering. DL has achieved success in many areas with various special network structures. CNN, one of the DL models, is widely used in image processing and computer vision fields and is preferred in tasks such as object recognition, classification, and segmentation (Çakar and Sengur 2021; Ozer 2024). Another DL model, ViT, unlike the CNN, divides the images into small-fixed size pieces and transforms them into a sequence and processes this sequence with feature extraction mechanisms specific to transformer models (Azad et al. 2024). CNNs are successful in learning local features, but they require additional layers and large datasets to model long-distance dependencies. ViT, on the other hand, can perform better, especially on large datasets, by processing the entire image simultaneously thanks to its self-attention mechanism. ViT is used in many areas such as medical image analysis, remote sensing, autonomous driving and industrial quality control and has become an important alternative in the field of computer vision in recent years (Thirunavukarasu and Kotei 2024; Alijani 2024). In addition to these models, the hybrid architecture (CNN-ViT) offers a richer and stronger classification model by combining CNN's local spatial sensitivity and ViT's transformer's global attention ability. In hybrid approach, first, 3-dimensional leaf images were fed to the CNN model by performing a series of pre-processing techniques (which were mentioned earlier). The CNN block extracted low to mid-level local features such as edge, texture, and color through a series of Conv3D, BatchNormalization, activation function and pooling layers. The final CNN feature maps were transformed via a linear dense layer into a sequence of tokens, each corresponding to a vision patch. Positional embeddings were added to these tokens, and then multiple Transformer encoder blocks were applied. These blocks include Multi-Head Self-Attention (MHSA), Layer Normalization (LN), Feed-Forward Network (FFN), and residual connections to model the spatial relationships between different parts of the image and create global feature representations. Local features from the CNN were converted to tokens and then passed to the Transformer block. The global context from the Transformer was integrated again via CNN-based advanced dense layers before being forwarded to the classification head. This enables cross-level feature fusion, preserving the local patterns of the input image while achieving broad context understanding. Finally, the Transformer output was evaluated using global average pooling (GAP) and routed to a dense layer containing Softmax. The model was trained end-to-end using various optimization algorithms with categorical cross-entropy loss for multi-class classification. This structure clearly demonstrates that the hybrid architecture, supported by both the original CNN-ViT integration mechanism and advanced modules, offers a powerful contribution to classification, both theoretically and practically.

Therefore, in addition to CNN and ViT models, a hybrid approach was used in the study. The data were separated as 80% training and 20% testing. The classification process was carried out under various scenarios. These scenarios are given in Table 1.

Table 1. Scenarios for the classification

Scenario No	Dataset	Data Augmentation
#1	(Sara et al. 2022)	N/A
#2	(Sara et al. 2022)	Available
#3	Ours	N/A
#4	Ours	Available
#5	(Sara et al. 2022) and ours	N/A
#6	(Sara et al. 2022) and ours	Available

Finally, the parameters of the developed models were determined with Grid Search (GS) algorithm. GS is a method that systematically examines all combinations in hyperparameter optimization. By definition, a grid is created with given hyperparameter values, and the model is trained for each combination, its performance is measured, and the combination that yields the best results is selected. For the CNN model, the hyperparameters that needed to be tuned were pooling size, kernel size, number of filters, batch size, optimization, and learning rate. For ViT, these were: encoder dimensionality, activation function in hidden layers, transformer encoder in hidden layers, optimization, and learning rate. For CNN-ViT, the previously mentioned parameters of the two models were evaluated together, and the best parameters were selected. Apart from these hyperparameters, one of the most important parameters is the epoch value. No optimization was performed for the epoch value; the epoch value was selected as 50 to ensure equal training for each model. The parameters of the models varied for each scenario, and they are given through Table 2-4.

2.4. Model Evaluation

The performance of the models was determined by the evaluation criteria of accuracy, F1-score, recall, precision, Cohen's Kappa and Hamming Loss. The accuracy score can be expressed as the ratio of the instances correctly classified by the models to the total number of instances. It shows the general success but can be misleading in unbalanced data sets. Therefore, other evaluation criteria were used in addition to the accuracy score. The recall shows how well the models capture true positives, while precision gives the rate at which the values classified by the models as positive are actually positive. The F1-score is the harmonic mean of the precision and recall values. It is used as a better performance indicator in unbalanced data sets. Cohen's Kappa is a metric that measures the classification success by taking into account the effect of random classifications. Values close to 1 indicate that the model is strong, while values close to 0 indicate that the model makes random classifications. Hamming Loss is an error measure that shows the rate of incorrectly classified examples. It is used effectively, especially in multi-label classification problems. A low value indicates that the model is good.

2.5. Model Development Environment

The training of the models was performed on a MacBook device with an Apple M1 chip and 7 GB of RAM. The training process was entirely CPU-based, without the use of any external GPU. Despite limited system resources, each model was trained in a reasonable amount of time. Average training times were approximately 1 hour for the CNN model, 2.5 hours for the ViT model, and approximately 3.5 hours for the hybrid CNN-ViT model. All training processes were conducted in the Anaconda Spyder environment. This information is important for demonstrating the feasibility of deep learning models even with limited resources and serves as a reference for researchers wishing to assess the technical adequacy of the study. Table 5 shows the time spent and technical information for each scenario.

Table 2. Hyperparameters of CNN model for each scenario

Scenario No	Hyperparameters	Hyperparameters Range	Selected Hyperparameters
#1	Pooling size	{2x2, 4x4, 8x8}	2x2
	Kernel size	{3x3, 5x5, 7x7, 9x9}	5x5
	Number of filters	{16, 32, 64}	32
	Batch size	{2, 8, 16, 32, 64, 128, 256}	16
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.01
#2	Pooling size	{2x2, 4x4, 8x8}	4x4
	Kernel size	{3x3, 5x5, 7x7, 9x9}	5x5
	Number of filters	{16, 32, 64}	16
	Batch size	{2, 8, 16, 32, 64, 128, 256}	32
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#3	Pooling size	{2x2, 4x4, 8x8}	2x2
	Kernel size	{3x3, 5x5, 7x7, 9x9}	3x3
	Number of filters	{16, 32, 64}	16
	Batch size	{2, 8, 16, 32, 64, 128, 256}	8
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.01
#4	Pooling size	{2x2, 4x4, 8x8}	4x4
	Kernel size	{3x3, 5x5, 7x7, 9x9}	5x5
	Number of filters	{16, 32, 64}	32
	Batch size	{2, 8, 16, 32, 64, 128, 256}	32
	Optimization	{adam, rmsprop, SGD}	SGD
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#5	Pooling size	{2x2, 4x4, 8x8}	4x4
	Kernel size	{3x3, 5x5, 7x7, 9x9}	5x5
	Number of filters	{16, 32, 64}	32
	Batch size	{2, 8, 16, 32, 64, 128, 256}	32
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#6	Pooling size	{2x2, 4x4, 8x8}	8x8
	Kernel size	{3x3, 5x5, 7x7, 9x9}	7x7
	Number of filters	{16, 32, 64}	64
	Batch size	{2, 8, 16, 32, 64, 128, 256}	32
	Optimization	{adam, rmsprop, SGD}	rmsprop
	Learning rate	{0.01, 0.001, 0.0001}	0.0001

Table 3. Hyperparameters of ViT model for each scenario

Scenario No	**Hyperparameters**	**Hyperparameters Range**	**Selected Hyperparameters**
#1	Encoder dimensionality	{1-100}	49
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	6
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.01
#2	Encoder dimensionality	{1-100}	70
	Activation function	{gelu, relu, selu}	relu
	Transformer encoder	{1-100}	66
	Optimization	{adam, rmsprop, SGD}	Adam
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#3	Encoder dimensionality	{1-100}	17
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	28
	Optimization	{adam, rmsprop, SGD}	rmsprop
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#4	Encoder dimensionality	{1-100}	33
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	18
	Optimization	{adam, rmsprop, SGD}	adam
	Learning rate	{0.01, 0.001, 0.0001}	0.01
#5	Encoder dimensionality	{1-100}	62
	Activation function	{gelu, relu, selu}	relu
	Transformer encoder	{1-100}	42
	Optimization	{adam, rmsprop, SGD}	SGD
	Learning rate	{0.01, 0.001, 0.0001}	0.001
#6	Encoder dimensionality	{1-100}	58
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	77
	Optimization	{adam, rmsprop, SGD}	SGD
	Learning rate	{0.01, 0.001, 0.0001}	0.0001

Table 4. Hyperparameters of CNN-ViT model for each scenario

Scenario No	Hyperparameters	Hyperparameters Range	Selected Hyperparameters
#1	Pooling size	{2x2, 4x4, 8x8}	2x2
	Kernel size	{3x3, 5x5, 7x7, 9x9}	3x3
	Number of filters	{16, 32, 64}	16
	Encoder dimensionality	{1-100}	44
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	14
	Optimization (ViT)	{adam, rmsprop, SGD}	adam
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.01
#2	Pooling size	{2x2, 4x4, 8x8}	8x8
	Kernel size	{3x3, 5x5, 7x7, 9x9}	5x5
	Number of filters	{16, 32, 64}	64
	Encoder dimensionality	{1-100}	49
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	25
	Optimization (ViT)	{adam, rmsprop, SGD}	rmsprop
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.001
#3	Pooling size	{2x2, 4x4, 8x8}	2x2
	Kernel size	{3x3, 5x5, 7x7, 9x9}	3x3
	Number of filters	{16, 32, 64}	16
	Encoder dimensionality	{1-100}	13
	Activation function	{gelu, relu, selu}	relu
	Transformer encoder	{1-100}	25
	Optimization (ViT)	{adam, rmsprop, SGD}	adam
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.01
#4	Pooling size	{2x2, 4x4, 8x8}	4x4
	Kernel size	{3x3, 5x5, 7x7, 9x9}	7x7
	Number of filters	{16, 32, 64}	32
	Encoder dimensionality	{1-100}	57
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	32
	Optimization (ViT)	{adam, rmsprop, SGD}	SGD
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.001
#5	Pooling size	{2x2, 4x4, 8x8}	8x8
	Kernel size	{3x3, 5x5, 7x7, 9x9}	7x7
	Number of filters	{16, 32, 64}	32
	Encoder dimensionality	{1-100}	82

	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	63
	Optimization (ViT)	{adam, rmsprop, SGD}	adam
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.001
#6	Pooling size	{2x2, 4x4, 8x8}	8x8
	Kernel size	{3x3, 5x5, 7x7, 9x9}	9x9
	Number of filters	{16, 32, 64}	64
	Encoder dimensionality	{1-100}	87
	Activation function	{gelu, relu, selu}	gelu
	Transformer encoder	{1-100}	67
	Optimization (ViT)	{adam, rmsprop, SGD}	adam
	Learning rage (ViT)	{0.01, 0.001, 0.0001}	0.0001

Table 5. General and technical information about development environment

Scenario No	Model	Training Time	Hardware Type	CPU	RAM	GPU
#1	CNN	~50 minutes	CPU	Apple M1	7 GB	None
	ViT	~2 hours	CPU	Apple M1	7 GB	None
	CNN-ViT	~3 hours	CPU	Apple M1	7 GB	None
#2	CNN	~1 hour	CPU	Apple M1	7 GB	None
	ViT	~2.5 hours	CPU	Apple M1	7 GB	None
	CNN-ViT	~3.5 hours	CPU	Apple M1	7 GB	None
#3	CNN	~45 minutes	CPU	Apple M1	7 GB	None
	ViT	~1 hour, 43 minutes	CPU	Apple M1	7 GB	None
	CNN-ViT	~2.5 hours	CPU	Apple M1	7 GB	None
#4	CNN	~1 hour	CPU	Apple M1	7 GB	None
	ViT	~2.5 hours	CPU	Apple M1	7 GB	None
	CNN-ViT	~3 hours	CPU	Apple M1	7 GB	None
#5	CNN	~1 hour, 10 minutes	CPU	Apple M1	7 GB	None
	ViT	~2.7 hours	CPU	Apple M1	7 GB	None
	CNN-ViT	~4 hours	CPU	Apple M1	7 GB	None
#6	CNN	~1.5 hours	CPU	Apple M1	7 GB	None
	ViT	~3 hours	CPU	Apple M1	7 GB	None
	CNN-ViT	~4.5 hours	CPU	Apple M1	7 GB	None

3. Results and Discussion

Developments in the field of ML and DL, when combined with image processing techniques, offer significant contributions, especially in critical areas such as agriculture and health. In this study, the performances of CNN, ViT and CNN-ViT models were compared on different datasets. The main objective of the study was to determine which data combinations and model configurations provide higher accuracy and generalization capacity. The approximate evaluation results obtained for each scenario are given in Table 6.

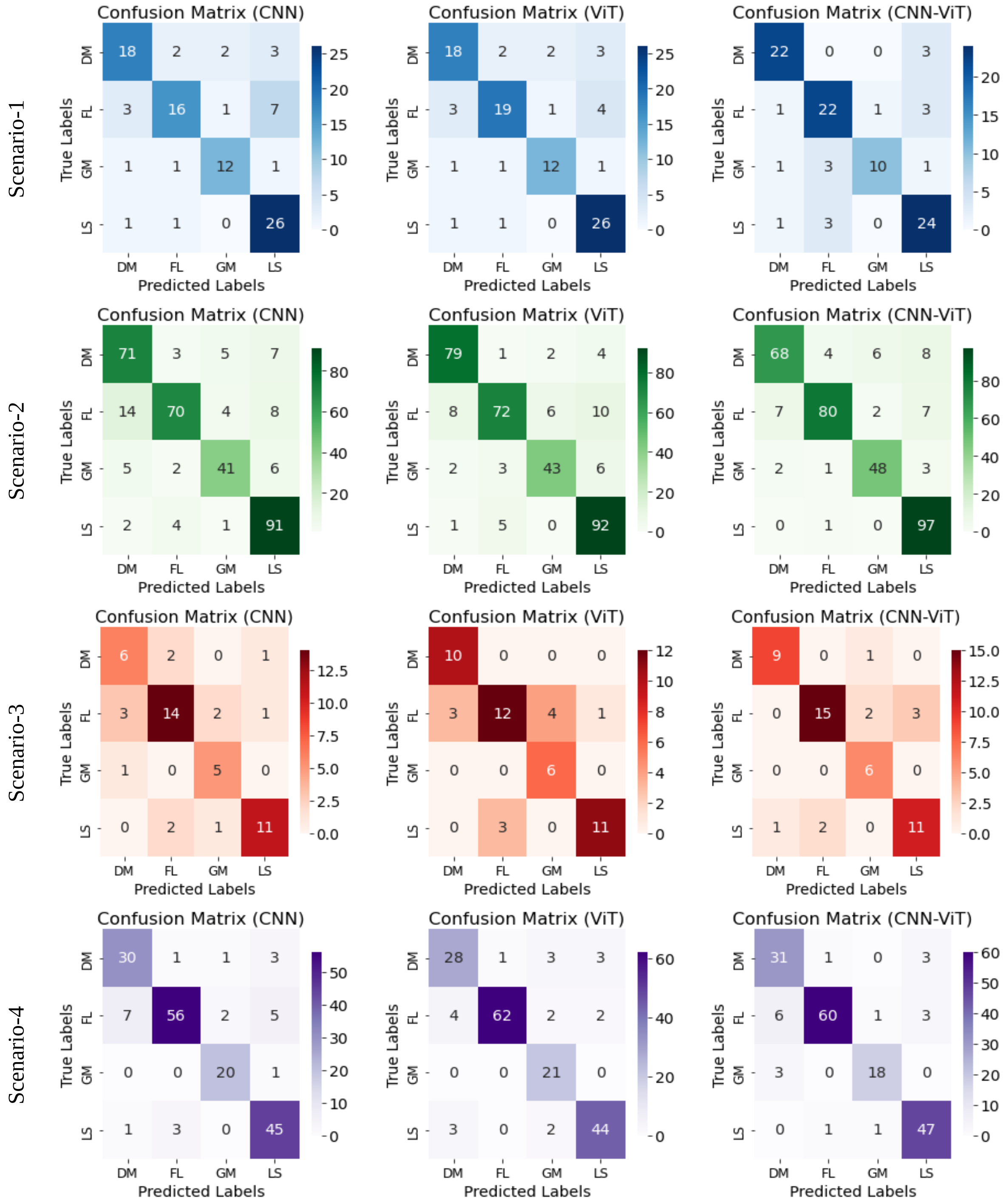
Table 6. Classification results for each scenario

Scenario No	Model	Accuracy	Precision	Recall	F1-Score	Cohen's Kappa	Hamming Loss
#1	CNN	75.8%	77.1%	76.0%	75.8%	0.52	24.2%
	ViT	77.9%	78.9%	76.5%	77.1%	0.58	22.2%
	CNN-ViT	82.1%	83.7%	80.1%	81.6%	0.67	16.2%
#2	CNN	81.7%	81.9%	81.1%	81.1%	0.63	18.1%
	ViT	85.6%	85.8%	85.1%	85.1%	0.69	14.2%
	CNN-ViT	87.7%	87.9%	87.6%	87.4%	0.72	12.1%
#3	CNN	73.5%	71.2%	74.7%	72.4%	0.49	26.3%
	ViT	78.0%	77.2%	84.6%	78.8%	0.60	21.2%
	CNN-ViT	82.0%	80.9%	85.9%	82.4%	0.64	18.3%
#4	CNN	86.3%	85.6%	88.2%	86.7%	0.71	13.2%
	ViT	88.6%	85.8%	89.6%	87.2%	0.73	11.1%
	CNN-ViT	89.1%	88.2%	89.0%	88.4%	0.75	10.2%
#5	CNN	87.5%	87.0%	88.7%	87.5%	0.72	12.2%
	ViT	90.3%	89.0%	90.5%	89.5%	0.76	9.2%
	CNN-ViT	91.7%	92.0%	90.6%	91.1%	0.77	8.3%
#6	CNN	92.0%	91.5%	91.5%	91.5%	0.78	7.3%
	ViT	94.0%	93.7%	93.5%	93.6%	0.79	5.2%
	CNN-ViT	95.7%	95.2%	95.1%	95.1%	0.81	4.1%

First, when considering Scenario 1, where only the dataset is used from (Sara et al. 2022) and no data augmentation is present, the CNN model achieved 75.8% accuracy, while the ViT model achieved a slightly higher value of 77.9% accuracy. However, one of the most striking results is that the CNN-ViT model, which is a combination of CNN and ViT models, achieved 82.1% accuracy. This showed that the combination of two different model architectures is more successful than the individual models. In Scenario 2, where data augmentation was introduced, the accuracy rate increased to 81.7%, especially in the CNN model, revealing the importance of data diversity. While the ViT model reached 85.6% accuracy in this scenario, the CNN-ViT model again achieved the best result with 87.7% accuracy. A similar trend is observed for Scenarios 3 and 4, where our own dataset is used. However, the striking point here is that the initial performances of the models are relatively lower as the dataset changes. For example, while the CNN model had an accuracy rate of 73.5% without data augmentation, it increased to 86.3% when data augmentation was applied. This showed that the raw dataset may have limited diversity and that data augmentation methods allowed the model to generalize better. The most remarkable results were obtained in Scenarios 5 and 6, where two datasets were combined. Here, it was observed that all models achieved the highest performance, especially in Scenario 6, where both data augmentation and dataset merging strategies were applied. The CNN model achieved the highest performance by reaching 92.0%, the ViT model 94.0% and the CNN-ViT model 95.7% accuracy rates. In addition, it was observed that the CNN-ViT model minimized the classification errors in this scenario, where the Hamming Loss values also decreased to the lowest levels.

One of the most important findings of this study is that data augmentation has a direct positive effect on model performance. It was observed that the generalization abilities of models trained on a single dataset were more limited, but these limitations were significantly overcome when data augmentation was used. However, it is clear that combining different datasets also provides a significant advantage. By combining two datasets, both data diversity increased and the model was able to cope better with different scenarios. In particular, the fact that the CNN-ViT model achieved the highest accuracy in all scenarios shows that hybrid models can offer a more powerful alternative in the field of image processing. Finally, when Cohen's Kappa values were examined, it was seen that the probability of random classification decreased significantly with data augmentation and data aggregation strategies. This is an important metric that supports that the models are truly learning, and their results are more reliable.

CM (Confusion Matrix) is frequently used to determine the performance of the models in each scenario and in each class. CM graphs of the models are given in Figure 4.



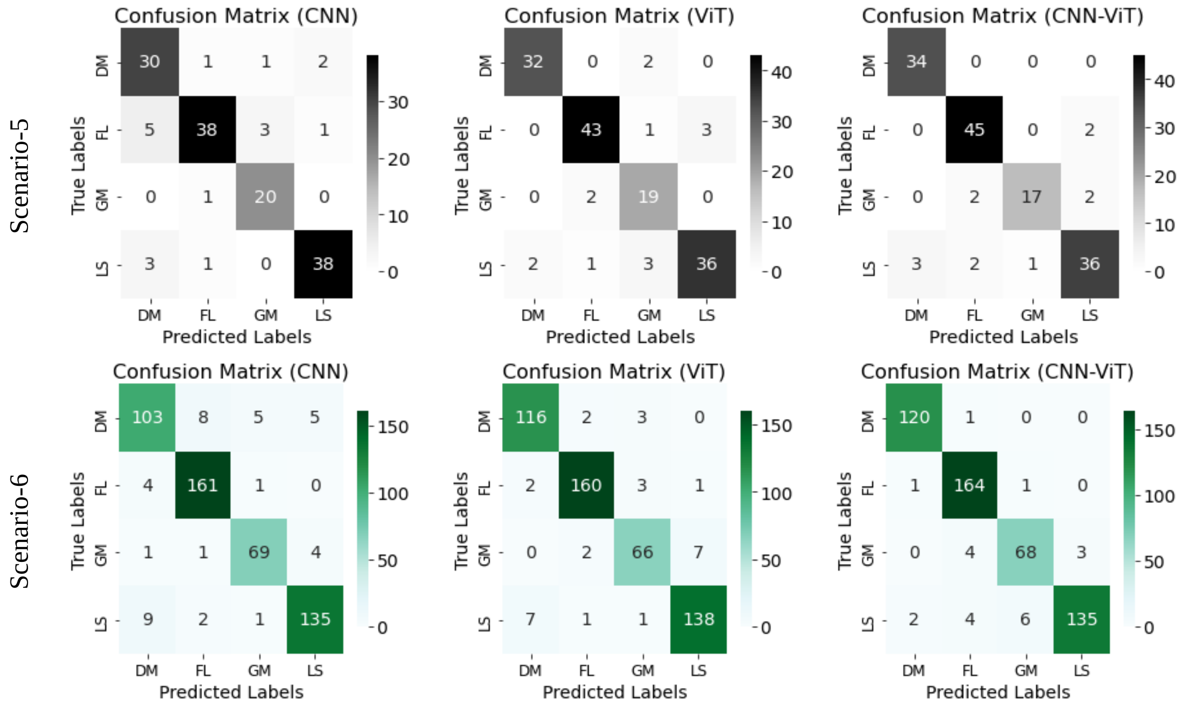


Figure 4. CM plots of models for 6 different scenarios. (DM: Downy mildew, FL: Fresh leaf, GM: Gray mold, LS: Leaf scar)

When the performances of CNN, ViT and CNN-ViT models are examined in the first scenario, it is seen that CNN and ViT models make incorrect classifications in certain classes, but the CNN-ViT model exhibits a more balanced distribution. There is a high false prediction rate especially in certain classes (e.g. LS). This situation shows that the model learns some classes better than others and therefore requires optimization. In the second scenario, it is observed that the accuracy rate increases in all models in general. While the CNN model performs better especially in large data sets, the ViT model is determined to be less efficient. The CNN-ViT model achieved the highest accuracy rate by combining the advantages of both architectures. The hybrid model provides an advantageous result, especially in terms of minimizing false positive and false negative rates. In the third scenario, it is seen that the CNN model has a low accuracy rate in certain classes and in some cases makes serious wrong classifications. In the ViT model, while the error rate decreases in certain classes, high wrong classifications continue in some classes. Although the overall accuracy rate of the CNN-ViT model is higher, the error rate is determined to be high in the LS class. In the fourth scenario, the CNN model exhibits a high accuracy rate in certain classes, while the wrong classifications are quite high in other classes. While the ViT generally shows a homogeneous distribution, the CNN-ViT model offers a better performance in all classes. In the fifth scenario, the CNN model has a very high false prediction rate in some classes, and it is observed that it overgeneralizes especially in certain classes. Although this situation is more balanced in the ViT model, errors continue in certain classes. The CNN-ViT model offers the most balanced classifications as in the previous scenarios. In the last scenario, it is observed that the CNN model has extremely high accuracy rates in some classes but makes significant false classifications in other classes. While the ViT model provides a relatively better balance, the CNN-ViT model produces the most balanced results. This shows that hybrid models have a stronger generalization ability on complex data sets. In Scenario 1, the hybrid model stood out with its high recall (0.8047) and specificity (0.9308) values, while its F1-score (0.8157) reached the highest levels. This demonstrated that the model was successful in both correctly classifying true positives and minimizing false positives, thus improving overall accuracy. In Scenario 2, CNN-ViT again had the highest recall value (0.8757), while its specificity (0.9546) and F1-score (0.8743) continued their success. This showed that the model demonstrated a balanced performance in correctly recognizing positive classes while simultaneously minimizing false positives. In Scenario 3, the CNN-ViT model's recall (0.8589) and specificity (0.9321) values were quite high, while its F1-score (0.8241) reinforced the model's balanced performance. This demonstrated that the model detected the positive class with high accuracy and also correctly classified the negative class to a large extent. In Scenario 4, although the ViT model was observed to have a higher recall value (0.8959), CNN-ViT provided the highest specificity (0.9615) and F1-score (0.8838). Here, it was observed that while the ViT model recognized the positive class well, the CNN-ViT model produced fewer false positives, providing a more balanced result. In Scenario 5, the CNN-ViT model stood out with its recall (0.9060) and specificity (0.9694) values and achieved the highest success with an F1-score (0.9113).

This specified that the model both detected the positive class with high precision and correctly classified the negative class, keeping error rates to a minimum. In Scenario 6, the CNN-ViT model achieved the highest recall (0.9512) and specificity (0.9850) values, representing its highest performance. The F1-score (0.9513) also demonstrated the model's balanced and robust performance. In this scenario, the CNN-ViT model classified both classes very accurately, minimizing false positive and false negative rates. Consequently, considering all scenarios, the CNN-ViT model achieved the highest recall and specificity values, maintaining high F1-scores. This indicated that the model achieved high accuracy in both positive classes and correctly recognized negative classes, minimizing false positives.

Some of the methods used to determine model performance in classification studies are statistical approaches. In this study, the Friedman test and one-way ANOVA (Analysis of Variance) were used to determine whether the model results were significant. The Friedman test, a nonparametric method, is used to compare the performance of multiple models on the same data blocks. In this method, the data are sorted by block, and the model mean rankings are compared. The Friedman test is more reliable, especially in cases with relatively small sample sizes, and its nonparametric structure is more robust to patterns (Inyang et al., 2024). One-way ANOVA is used to compare three or more models on the same units. It is based on the assumptions of normal distribution and homogeneity of variance (Abbas et al., 2024). The Friedman test result (x^2=12.00, p=0.0025) indicated that at least one model was statistically significantly different from the others, as $p<0.05$. This revealed that the observed differences in the performance of the three models were not random, but rather systematic and statistically significant. However, in the one-way ANOVA analysis, no significant difference was found between the model performances, as $p>0.05$ (F=1.006, p=0.389). The main reason for this may be the high variance, which may have caused the difference not to be detected by the ANOVA. When the accuracy results are examined, the performance of the CNN model varies between 73.5% and 92.0%. Similarly, the accuracy of the CNN-ViT model varies between 82.0% and 95.7%. As can be seen, the performances are in a wide range, indicating a high within-group variance.

In general, when the performances of all three models were compared, it was determined that the CNN-ViT had the best accuracy rate and a balanced classification success. The main reasons for this may be the combination of CNN's ability to learn local features and ViT's ability to capture long-range dependencies. CNN and ViT models used alone cause erroneous classifications by overgeneralizing in certain classes. The CNN-ViT produced more robust and balanced classifications by combining the strengths of both approaches. In addition, the deep learning methods used in the study can provide higher accuracy rates compared to classical machine learning approaches. In particular, the use of advanced models such as CNN and ViT increased the classification success. The use of various techniques for preprocessing the images (filtering, segmentation, etc.) allowed the model to make more accurate classifications. However, despite these successes, the study also has several shortcomings. The most important of these is the small dataset and diversity. The success of the study largely depends on the dataset used. If the dataset contains a limited number of diseases types or has low sample diversity, the generalization ability of the models may be limited. Using a larger and more diverse dataset can increase the accuracy of the models. It is also important to determine how the study will perform in real-world conditions. Different lighting conditions, weather, and natural variability in foliage can affect the accuracy of the models. Therefore, it is important to evaluate the model in outdoor tests. Finally, DL models developed in this study to detect diseases on sunflower leaves demonstrated high classification accuracy. These results are consistent with many studies in the literature demonstrating the effectiveness of deep learning in diagnosing leaf diseases in different plant species (Moupojou et al., 2023; Şener and Ergen, 2024; Toğaçar, 2002; Fan et al., 2022) These studies showed that AI is a particularly powerful tool for detecting leaf diseases through image processing. Additionally, another frequently emphasized point in the literature is that data augmentation techniques increase model generalization and reduce the risk of overfitting. Similar approaches were adopted in this study, and the results were significantly improved.

Furthermore, testing the developed models in different geographic regions is crucial for a more accurate assessment of their generalizability. While this study primarily used data from the Thrace Region, the model's adaptability should be tested by collecting data from agricultural areas with different climatic and environmental conditions.

4. Conclusion

In this study, image-based deep learning models were used for early detection of sunflower leaf diseases. CNN, ViT and hybrid CNN-ViT models were designed in the study and the performances of these models were compared with metrics such as accuracy, F1-score, precision, recall, Cohen’s Kappa and Hamming Loss. The study analyzed the success of these approaches in detail using different datasets and data augmentation techniques. The hybrid model had the highest accuracy compared to traditional DL approaches and offered a strong alternative in the detection of sunflower leaf diseases.

As a result, it has been observed that DL methods supported by image processing techniques provide high success in the detection of sunflower diseases. The hybrid CNN-ViT model has contributed to preventing diseases from reducing agricultural productivity by making accurate and timely diagnoses. The study shows that image-based deep learning models can be used in important applications such as disease detection in agriculture. It was emphasized that the developed CNN-ViT-based model can be used as a decision support tool, enabling farmers to quickly and accurately diagnose leaf diseases encountered in the field, particularly in sunflower production. Training the model with data collected under diverse environmental conditions increases its robustness and overall performance for such applications. Furthermore, in addition to the study's strengths, significant limitations such as the limited size and diversity of the datasets used, the classification of only four disease types, and the fact that the model has not yet been tested in real-time in the field are also important factors to consider. Finally, the developed CNN-ViT-based model's integration with mobile applications will enable farmers to perform real-time field diagnostics. The fact that the study was conducted on a CPU-based, low-resource system demonstrates the model's resource efficiency and its applicability to low-capacity devices. Therefore, future work plans to develop lightweight, mobile-optimized versions of the model.

Acknowledgment

This study was supported by Kırklareli University Scientific Research Projects Coordination Unit with Project Number: KLÜBAP-247.

References

Abbas JKK, Ruhaima AA, Naser OA, Hayder DM (2024) F-Test and One-Way ANOVA for Medical Images Diagnosis. *Al-Nisour Journal for Medical Sciences* 6(2): 29-38.

Alijani S, Fayyad J, Najjaran H (2024) Vision Transformers in Domain Adaptation and Domain Generalization: A Study of Robustness. *Neural Computing and Applications*36: 17979-18007.

Andasuryani I, Rasinta I. (2021). Classification of Tomato (Lycoersicon Esculentum Miil) Ripeness Levels Based on HSV Color Using Digital Image Processing. IOP Conference Series: Earth and Environmental Science, 116, 012062.

Azad R, Kazerouni A, Heidari M, Aghdam EK, Molaei A, Jia Y, Jose A, Roy R, Merhof D (2024) Advances in Medical Image Analysis with Vision Transformers: A Comprehensive Review. *Medical Image Analysis* 91: 1-66.

Çakar H, Sengur A (2021) Machine Learning Based Emotion Classification in the COVID-19 Real World Worry Dataset. *Journal of Computer Science* 6(1): 24-31.

Centorame L, Gasperinin T, Ilari A, Gatto AD, Pedretti EF (2024) An Overview of Machine Learning Applications on Plant Phenotyping, with a Focus on Sunflower. *Agronomy* 14(4): 1-23.

David LL, Pohl MD, Alvarado JL, Bye R (2008) Sunflower (Helianthus Annuus L.) As A Pre-Columbian Domesticate in Mexico. *Anthropology* 105(17): 6232-6237.

Fan X, Luo P, Mu Y, Zhou R, Tjahjadi T, Ren Y (2022) Leaf Image Based Plant Disease Identification Using Transfer Learning and Feature Fusion. *Computers and Electronics in Agriculture* 196: 106892.

Ghosh P, Mondal AK, Chatterjee S, Masud M, Meshref H, Bariagi AK (2023) Recognition of sunflower diseases using hybrid deep learning and its explainability with AI. *Mathematics* 11(10): 1-24.

Inyang EJ, Moffat IU, Clement EP (2024) Friedman Test Technique for Optimizing a Seasonal Box-Jenkins ARIMA Model Building. *Journal of Probability and Statistical Science* 22(1): 1-15.

Islam M, Adil AA, Talukder A, Ahamed KU, Uddin A, Hasan K, Sharmin S, Rahman M, Debnath SK (2023) DeepCrop: Deep Learning-Based Crop Disease Prediction with Web Application. *Journal of Agriculture and Food Research* 14: 1-11.

Kaur R, Jain A, Kumar S (2022) Optimization Classification of Sunflower Recognition Through Machine Learning. *Materials Today Proceedings* 51(1): 207-211.

Koo C, Malapi-Wight M, Kim HS, Çiftçi OS, Vaughn-Diaz VL, Ma B, Kim S, Abdel-Raziq H, Ong K, Jo YK, Gross DC, Shim WB, Han A (2013) Development of a Real-Time Microchip PCR System for Portable Plant Disease Diagnosis. *Plos One* 8(12): e82704.

Li W, Zheng T, Yang Z, Li M, Sun C, Yang X (2021) Classification and Detection of Insects from Field Images Using Deep Learning for Smart Pest Management: A Systematic Review. *Ecological Informatics* 66: 1-18.

Malik A, Vaidya G, Jagota V, Eswaran S, Sirohi A, Batra I, Rakhra M, Asenso E (2022) Design and Evaluation of a Hybrid Technique for Detecting Sunflower Leaf Disease Using Deep Learning Approach. *Journal of Food Quality* 2022: 1-12.

Moupojou E, Tagne A, Retraint F, Tadonkemwa A, Wilfried D, Tapamo H (2023) FieldPlant: A Dataset of Field Plant Images for Plant Disease Detection and Classification with Deep Learning. *IEEE Access* 11: 35398-35410.

Ozer E (2024) Brain Tumor Detection using Deep CNNs and Ensemble Algorithms over MRI Images. *Journal of Computer Science* 9(2): 142-150.

Rajora R, Banerjee D, Chauhan R, Singh M. (2024). Advanced Sunflower Leaf Disease Detection using CNN-SVM Hybrid Models. 4th Asian Conference on Innovation in Technology, pp: 1-7.

Rani L, Veeramanickam MRM, Pandey B. (2024). Innovative Fusion for Sunflower Leaf Disease Identification: CNN and Random Forest Strategies. IEEE AITU: Digital Generation, pp: 65-70.

Sara U, Rajbongshi A, Shakil R, Akter B, Sazzad S, Uddin MS (2022) An Extensive Sunflower Dataset Representation for Successful Identification and Classification of Sunflower Diseases. *Data in Brief* 42: 1-8.

Sarhan M, Layeghy S, Moustafa N, Gallagher M, Portmann M (2024) Feature Extraction for Machine Learning-Based Intrusion Detection in IoT Networks. *Digital Communications and Networks* 10(1): 205-216.

Sasaki Y, Okamoto T, Imou K, Torii T (1998) Automatic Diagnosis of Plant Disease. *IFAC Proceedings Volumes*, 31(5): 145-150.

Sathi TA, Hasan A, Alam MJ. (2023). SunNet: A Deep Learning Approach to Detect Sunflower Disease. 7th International Conference on Trends in Electronics and Informatics, pp: 1210-1216.

Şener A, Ergen B (2024) Advanced CNN Approach for Segmentation of Diseased Areas in Plant Images. *Journal of Crop Health* 76: 1569-1583.

Singh V (2019) Sunflower Leaf Diseases Detection using Image Segmentation Based on Particle Swarm Optimization. *Artificial Intelligence in Agriculture* 3: 62-68.

Sirohi A, Malik A. (2021). A Hybrid Model for the Classification of Sunflower Diseases using Deep Learning. 2nd International Conference on Intelligent Engineering and Management, pp. 58-62.

Thirunavukarasu R, Kotei E (2024) A Comprehensive Review on Transformer Network for Natural and Medical Image Analysis. *Computer Science Review* 53: 1-19.

Toğaçar M (2022) Using DarkNet Models and Metaheuristic Optimization Methods Together to Detect Weeds Growing Along with Seedlings. *Ecological Informatics* 68:101519.

Vorobyov SP, Vorobyova VV. (2021) The Ecological and Economic Effectiveness of Sunflower Oilseed Production in Russia. IOP Conference Series: Earth and Environmental Science, 670, 012057.

Wang L, Wang J, Liu Z, Zhu J, Qin F (2022) Evaluation of a Deep-Learning Model for Multispectral Remote Sensing of Land Use and Crop Classification. *The Crop Journal* 15(5): 1435-1451.

Wu L, Zeng W, Lei G, Ma T, Wu J, Huang J, Gaiser T, Srivastava AK (2022) Simulating Root Length Density Dynamics of Sunflower in Saline Solis Based on Machine Learning. *Computers and Electronics in Agriculture* 197: 1-11.

Yuan J, Wan X (2022) The Associative Effects of Sunflower Straw, Sunflower Plate, Sunflower Seed Shells Associated with Concentrate and Alfalfa Evaluated by Using An In Vitro Gas Production Technique. *Czech Journal of Animal Science* 67(7): 253-265.

Zhou J, Li J, Wang C, Wu H, Zhao C, Teng G (2021) Crop Disease Identification and Interpretation Based on Multimodal Deep Learning. *Computer and Electronics in Agriculture* 189: 1-9.