Journal of Agricultural Sciences (JAS) 32(4):1051-1061
DOI: 10.15832/ankutbd.1795811

J Agr Sci-Tarim Bili
e-ISSN: 2148-9297
jas.ankara.edu.tr

A Novel Image Filtering Approach for Image Classification Applied to Severity Assessment of Root Rot Disease in Arugula Cultivation

Ikbal Aygun[a*], Onur Umit Sener[b], Ulas Ozbent[b], Selami Beyhan[c]

[a] *Ege University, Faculty of Agriculture, Department of Agricultural Engineering and Technology, Bornova, Izmir, TÜRKIYE*

[b] *Izmir Demokrasi University, Electrical and Electronics Engineering, Uckuyular Dist., Gursel Aksel Blvd., Karabaglar, Izmir, TÜRKIYE*

[c] *Yildiz Technical University, Artificial Intelligence and Data Engineering, Electric-Electronic Faculty, Davutpasa Campus, Esenler, Istanbul, TÜRKIYE*

ARTICLE INFO
Research Article
Corresponding Author: Ikbal Aygun, E-mail: ikbal.aygun@ege.edu.tr
Received: 03 October 2025 / Revised: 12 March 2026 / Accepted: 14 May 2026 / Online: 12 October 2026

Cite this article
Aygun I, Sener O U, Ozbent U, Beyhan S (2026). A Novel Image Filtering Approach for Image Classification Applied to Severity Assessment of Root Rot Disease in Arugula Cultivation. *Journal of Agricultural Sciences (Tarim Bilimleri Dergisi),* 32(4):1051-1061. DOI: 10.15832/ankutbd.1795811

ABSTRACT
In this paper, a new image filtering method is proposed using nonlinear symbolic functions. In this way, feature extraction from images is achieved using a small number of filters and parameters. Nonlinear symbolic functions are determined by an optimization method and different approaches are proposed to increase the filtering speed. Nonlinear functions also act as a regression layer and more efficient design is achieved by moving the image to nonlinear space. In sustainable agriculture, artificial intelligence has been recently utilized to pre-detect diseases and determine their severity to perform spraying more accurately in site-specific applications for human health and soil ingredients. Therefore, a new dataset for detecting the disease root rot severity of Arugula (Eruca vesicaria) was created and labeled for the image classification such that there is a significant decrease in production yield every year due to root rot disease. The proposed CNN model with nonlinear symbolic function filtering was produced %86.92 accuracy for the classification of disease severity whereas conventional 4-layer CNN model and pre-trained large image classification models such as VGG16, VGG19, ResNet50, ResNet101, InceptionV3, DenseNet, EfficientNetV2S, MobileNet, MobileNetV2 and Xception have been fine-tuned and classification accuracy is provided between 57-92% without preprocessing of images. Between the designed models, MobileNet model with less computational time and high accuracy can be a potential for future application in drone-based spraying systems. However, the proposed model with less number of parameters is also promising for future applications. The application results of image classification facilitate variable-rate spraying for sustainable agricultural production, ultimately protecting both human and soil health.

Keywords: Symbolic functions, Nonlinear Filters, Root rot disease, Deep learning models, Image classification

1. Introduction

Artificial intelligence (AI) has recently been serving many fields of science and its application areas are expanding so that the development of theoretical methods is inevitable depending on new application areas. The development of image classification models has been a crucial field that has taken the lead and influenced other fields (LeCun et al. 2015). The success of pre-trained models in feature extraction and accuracy allows them to be used in more areas without much effort (Schmidhuber, 2015). In the field of smart agriculture, the usage of artificial intelligence is very important for the protection of both humans and nature that drives the application of various methods in this field, not only image classification. For example, AI facilitates more precise fertilization and pest control, accurate yield predictions and early disease detection by optimizing resource use, increasing crop yields and improving water efficiency (Liakos et al. 2018). A comprehensive survey study by (Liakos et al. 2018) highlights the important contributions of AI to smart agriculture and emphasizes its potential to transform agricultural practices. Many studies had been conducted to detect disease or estimate yield. One of the studies conducted by schueller et al. (1999) was a strategy and a system were generated to create yield maps for hand harvest citrus. Also, Gan et al. (2018) developed a model that combined color and thermal information for immature green citrus fruit detection. Using this method, early yield estimation could be conducted. Linker et al. (2012) developed an algorithm for green apple detection in a natural environment. The algorithm achieved correct identification rates of 85% under non-direct sunlight conditions and 95% under diffuse lighting conditions.

Today, pre-trained deep learning models provide solutions to many classification problems. Especially thanks to the ability to extract advanced features, it has become possible to obtain meaningful results from seemingly complex data. These pre-trained models provide us with acceptable results that can also solve the plant disease classification problem, which is the subject of the present work. They have pioneered many studies in the last few years. Pre-trained models provide high accuracy rates by reducing data limitations and computational costs in the detection and classification of plant diseases. MobileNet is widely used in mobile and embedded systems due to its low computational power requirement and fast operation, which allows it to be used more

frequently in real-time agricultural applications and to obtain effective results. In addition, this model has achieved high accuracy rates in plant disease detection problems in literature (Howard et al. 2017). On the other hand, InceptionV3, which has another CNN approach, provides an advantage in detecting subtle differences with its more complex architecture and has achieved successful results in large data sets (Szegedy et al. 2016). Similarly, DenseNet201 improves efficiency and accuracy by increasing the information flow between layers with their dense connection structure. As a result, better performance in plant disease detection has been widely presented in the literature (Huang et al. 2017). Similarly, ResNet50 improves the performance of deep learning models with lower error rates despite the number of deep layers and has also been effectively used in the detection of plant diseases (He et al. 2016). Since these models are pre-trained on large data sets with transfer learning techniques, they exhibit high performance when re-trained with agricultural data sets (Mohanty et al. 2016). As a result, these pre-trained models detect plant diseases more accurately, increasing efficiency and sustainability in the agricultural sector.

Arugula can be grown anywhere in the world, but it is particularly widely cultivated in Mediterranean countries and in Northwestern and Western Europe. In Europe, it is mostly cultivated in Spain, Greece, Turkey, and Italy. Rocket, which can be grown in the fields during the summer months and in greenhouses during the winter months, can be consumed in every season of the year (Morales et al. 2022). (Palumbo et al. 2024) developed algorithms for determining color differences in arugula leaves and presented an effective machine learning model. In this research, the Random Forest model, previously developed to predict chlorophyll and ammonia contents in arugula leaves of different colors in the leaves, was examined to identify a small set of visual features (colors) associated with these features of arugula. These features were used as inputs for several simple, fast, and easy-to-understand algorithms that predict with less computational complexity, classifying visual quality (VQ) and comparing chlorophyll and ammonia contents to the original Random Forest model. Precision agriculture applications, as in many fields, are used especially for environmentalism and sustainable production to optimize pesticide use in agricultural production and to increase agricultural yield. Palumbo and others (2024) created a new dataset to determine the quality of arugula leaves and developed a technique to obscure the image and pre-trained models. Deep Neural Network (DNN), Random Forest (RF) based on Decision Trees, and Support Vector Regression (SVR) are three popular supervised machine learning methods. In other work, air quality and water quality measurements, including temperature, humidity, CO2, pH and Total Dissolved Solids (TDS) were used for yield prediction (Torres-Tello et al. 2020). Six crops were studied (garlic, basil, red chard, rainbow chard, arugula, and mint). When working with DNN models, the prediction rates on the datasets have been quite high. Many studies conducted in the field of machine learning and the rapid developments that have occurred have contributed to the development of data-driven paradigms such as smart farming systems and precision agriculture systems (Aashu et al. 2024). Hyperspectral data collected from lettuce and arugula leaves as a test field were used to train a neural network that can classify plant species based on their spectral features. In the conducted studies, a neural network was trained using hyperspectral data collected from lettuce and arugula leaves. This neural network is a system that can classify plant species based on their spectral characteristics. The developed system can classify the plant species used for testing at a high frame rate and with high precision (e.g., 0.996, Neri et al. 2024). In their study, they applied different nutrient media to the D. Tenuifolia plant for a week and captured images. As a result of the research, they aimed to detect abiotic and biotic stresses early, improve plant nutrition practices, and enhance the performance of rocket plants using a digital reflection-based probe. Using artificial intelligence, image-based classification models were run within the stress range, and the hyperspectral fingerprints of the leaves were trained and used for validation. The relevant physiological conditions have validated the spectral examination (Navarro et al. 2022).

The objective of this study was to propose an approach that leverages fine-tune pretrained models to optimize pesticide application in arugula cultivation for enhanced precision agriculture. This approach aims to minimize the use of pesticide while maximizing crop yield and quality, thereby reducing the environmental impact of agricultural practices. It should be noted that the main objective of this study is not the proposal of a new deep learning architecture, but the introduction of a symbolic function based convolutional filtering paradigm for plant disease severity assessment. In addition, a new real-world arugula root rot severity dataset is introduced, and the effectiveness of symbolic filtering is demonstrated in comparison with conventional CNNs and state-of-the-art pretrained models. The proposed approach provides a low-parameter and computationally interpretable alternative that is suitable for precision agriculture and embedded deployment scenarios.

2. Material and Methods

The arugula plant, which is a member of the Brassicaceae family, has gained popularity in recent years. Arugula can be grown all seasons and be eaten raw in salads or cooked. Its seeds yield oils that are used in a variety of ways and for a range of purposes (Sabatino et al. 2021). The "Eruca sativa Mill" variety, known as "Izgın" grown in the Central Anatolia Region of Turkey, is used only as an oil plant and animal feed. Arugula is susceptible to various diseases that can negatively impact its growth, yield, and quality. Figure 1 shows a sample image for the healthy arugula. Figure 1 shows a sample image for the healthy arugula.

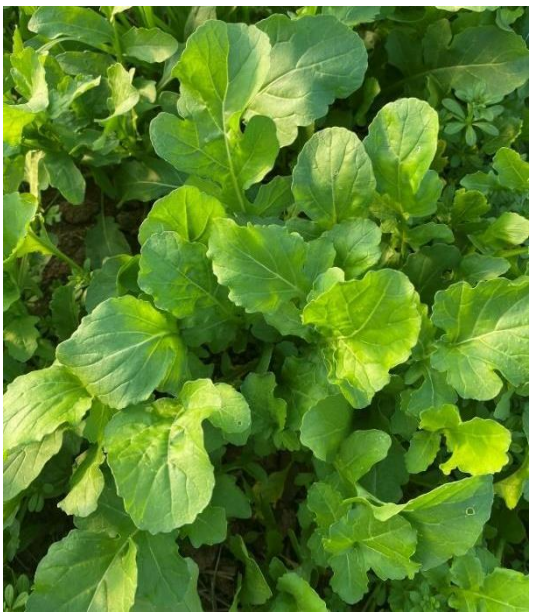


Figure 1- Arugula sample

Diseases that can be affected mostly arugula is White rust, Root rot, White rot, Gray mold, Septerio Spot Disease, alternaria blight. One of the prevalent diseases affecting arugula is Root rot, a fungal disease that can manifest at various stages of the plant's life cycle. The disease is characterized by the appearance of lesions on the leaves, stems, and even the seed-producing organs of the plant. Arugula is susceptible to a number of diseases caused by pathogens and insects. Root rot can reduce the yield and quality of arugula crops through various means such as stem, leaf and head rots. Root rot caused by Pythium spp. is a fungus that lives in the soil. Rhizoctonia spp. is a fungus that lives in the soil and can be transmitted by seed. Root rot is a disease seen in the seedling stage. At the same time, damage can occur before emergence (pre-emergence). In the seedling stage, seedlings are seen lying down from the root collars that come into contact with the soil (post-emergence). As a result of deaths that occur both before and after emergence, gaps in the form of burrows are formed in the seedling. If the seedling conditions are suitable, the disease can cause complete destruction of the seedlings. The disease has spread to all areas where seedlings are produced in our country. The most common causal agents of root rot diseases are fungi and oomycetes Legumes seem to be the most susceptible hosts, but root rots can also afflict monocots, dicots, cereals, legumes, fruit trees, and tubers (Williamson-Benavides & Dhingra, 2021). The impact of root rot on global crop production can be devastating, with losses ranging from minor economic impacts to complete crop failure, depending on factors such as the specific pathogen involved, the susceptibility of the host crop, and the prevailing environmental conditions. Pest management strategies in arugula production should focus on preventive methods to mitigate the impact of diseases such as root rot. Herbicides could be used to eliminate weeds that may serve as a host for the pathogen during the periods between arugula crops. Herbicide application must be used for the leaves of the arugula plant for root rot. In order to eliminate root rot, it should be sprayed quickly, and the disease agent should be eliminated.

2.1. Proposed symbolic filtering of images

In this section, symbolic functions, function optimization strategies and evaluation of the designed models are presented respectively. Symbolic Function Based Convolution (SFBC) method is based on the selection of 13 predefined mathematical analytic functions into a 3x3 matrix. These functions are chosen by random search or genetic algorithm from the set of functions below.

In the proposed SFBC, each convolution kernel is made up of using predefined analytic symbolic functions. Let $\mathbb{F} = \{f_1, f_2, \ldots, f_{13}\}$ be the function set, where each function $f_i(\cdot)$ is a nonlinear analytic function in the form of polynomial, trigonometric, logarithmic, or exponential forms defined as

$$\mathbb{F}=\{x,\ x^2,\ x^3,\ sin(x),\ cos(x), sin(3x), cos(3x),\ max(0,x),\ e^{-x}, tanh(x),\ e^{-2x},\ \frac{1}{1+e^{-x}}, 0\} \quad (1)$$

If $f_k \in \mathbb{F}$ is chosen using an optimization method (e.g., genetic algorithm), and x_i denotes the corresponding pixel intensity or derived image statistic. Then, the convolution operation is conducted with this nonlinear kernel, effectively projecting the image into a symbolic nonlinear feature space prior to standard CNN processing. The outputs are subjected to machine learning steps and disease classification is performed. Different strategies were tried to improve the performance of the SFBC method.

2.1.1. Function optimization with genetic algorithm (SFBC-1)

Genetic algorithms (GAs) are search-based algorithms developed using the ideas of genetics, heredity, evolution theory and natural selection. GA is a subfield of evolutionary computation, which is a considerably larger field of study (Ayan et al. 2020). In GAs, there are many different solutions to a given problem. Similar to biological genetics, the resulting solutions undergo recombination and mutation, producing new offspring and replicating the process over several generations. The value of each individual's objective function determines their unique fitness value, and those who are fitter are more likely to mate and generate more fit individuals. GAs are frequently employed to generate excellent answers for search and optimization issues. In this method, symbolic functions are determined using GA. Classification accuracy is obtained as 85.3% with the kernel sequence obtained by GA. This is a high success rate such that Table 2 shows the comparison of all designed models. The optimized final kernel function matrix is given as follows.

$$\begin{bmatrix} e^{-x} & \tanh(x) & 0 \\ 0 & \sin(x) & \cos(x) \\ e^{-2x} & 1/(1+e^{-x}) & x^2 \end{bmatrix} \quad (2)$$

At this stage, since each image takes approximately 40-45 seconds at the test stage, the search for a model with a shorter classification time but a higher success rate was continued. Apart from these, there are 4 convolution layer CNN models that are not pre-trained and have few parameters. However, its performance is lower than the two proposed models. In the discussion section, the performance and computational burdens are discussed.

2.1.2. Entropy based kernel approach (SFBC-2)

In this method, the function matrix is based on random search, and the entropy value of the image is calculated, and this value is given as input to the functions. In other words, a constant convolution matrix is obtained. While the use of entropy can help to understand the information density in the image, the classification accuracy of this method yielded results around 80%. Therefore, the expected improvement was not achieved, and it was decided to work on alternative methods.

2.1.3. Standard deviation based kernel approach (SFBC-3)

This approach uses a random search to find the functions, but instead of using the x value in the functions, the standard deviation value is provided as input after the total standard deviation of the image's three channels is computed. In this way, a fixed kernel specific to each image is created. This method was accelerated with the help of the PyTorch library by using optimized convolution operations and training times were reduced to approximately 3 minutes and 30 seconds. However, due to the high output standard deviation, the classification performance is low. The approach in methods 2 and 3 was taken as a basis by adding α parameter to the functions and this parameter was used to calculate the loss function during model training. In this way, the model was optimized and a more efficient learning process was achieved. Classification accuracy was 83.85%, which is similar to previous methods.

2.2. Overview of pretrained models

Convolutional Neural Networks have become a standard in multi-class data classification thanks to the features they can adaptively extract from the data and the ability to make sense of the hierarchy between these features. Thanks to their multi-layered structure, CNNs can extract many different features specific to each level from the input data, and due to their hierarchical structure, they aim to weigh this information flow and solve the classification problem, which makes them very valuable for use in multi-class classification problems. A simple CNN model was designed with 4 convolution layers shown in Figure 2 and applied to a dataset for multiclass classification (Shanthini at al. 2025).

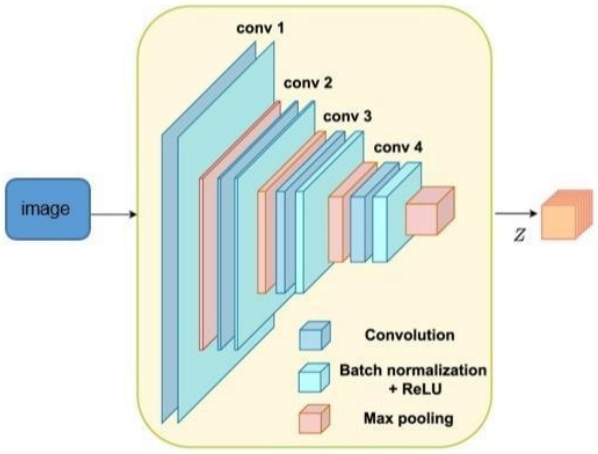


Figure 2- CNN Model (Shanthini at al. 2025)

Over time, different approaches have been developed on CNNs, and these approaches have developed methods based on parameters such as success, resource management, and time. For example, Traditional CNNs, which are among the most frequently used of these approaches today, exhibit a standard approach in terms of the ability to extract features from the input data and solve the classification problem thanks to their convolution layers. The key factor in solving the classification problem here is the depth of the layers of the models with this approach and the high number of parameters they obtain thanks to the feature map in each layer.

One of the different approaches developed on Convolutional Neural Networks is Depthwise Separable Convolution and this approach is different from traditional convolutional neural networks due to its architecture. In the method applied in this approach, the convolution step is divided into two as depthwise and pointwise. This approach is also called group convolution. Depthwise Convolution, as its name suggests, takes a different approach on the depth of the layers, which is that instead of passing all input channels through the kernel filter, this filter is applied to the input channels separately, the filters do not interact with different input channels, each channel has its own convolution. Then, as the second step, in Pointwise Convolution, a 1x1 kernel filter is used and a linear combination of feature maps is obtained thanks to this filter. In order to adjust the depth of the convolution layers of the model, it is sufficient to determine how many channels the pointwise convolution will have, and this provides great flexibility in adjusting the complexity of the CNN model. CNN models trained with the DSC approach provide advantages in terms of computational cost while keeping the predictive success of the model high, which is often preferred, especially when efficiency is taken into account in mobile and embedded systems.

Another CNN approach is Residual Networks. The reason for developing this approach is the "residual" solution it brings to the problem of gradient vanishing in artificial neural networks, which is the problem of the gradient becoming too small in the backpropagation part of the training, causing the weights not to change and the model not to learn. In this solution, structures such as skip connections or shortcut connections are used, and this means directly adding the input of the previous layer to the next layer, so what the model needs to learn is actually how much deviation should be made from the input of the previous layer in order to obtain the output in the next layer. This "residual" difference is added to the input, allowing the model to move on to the next step, and thus the model reinforces its ability to overcome complex problems by learning these differences at each step. Another CNN approach, Inception Networks, is also frequently included in the literature. The structural feature that distinguishes this CNN approach from others is the simultaneous, parallel convolution operations. In this approach, which allows extracting features at different levels in the data by passing the data through different filters in parallel and simultaneously, the model is deepened with the 1x1 kernel filter, one of the filters used in parallel convolutions, and it allows representing the block with a high channel number in the previous layer with a single channel, which reduces the computational load of the model and contributes to learning by increasing the depth. More advanced features can be extracted in parallel with other filter sizes. This is the reason why Inception Networks are especially popular in image processing areas.

In this study, we use various advanced convolutional neural network architectures to visualize and compare their performance in plant disease detection. We aim to provide a comprehensive evaluation of their effectiveness and computational efficiency by utilizing models such as VGG16, VGG19, ResNet50, ResNet101, InceptionV3, DenseNet, EfficientNetV2S, MobileNet, MobileNetV2 and Xception. The results we will focus on in this paper are the results we obtain with the pre-trained weights of these models and we will examine the effects of different approaches.

VGG16 and VGG19 are deep convolutional neural networks that use 3×3 convolutional layers. They were used as baseline pretrained architectures for comparison. ResNet50 and ResNet101 use residual connections to build up deeper architectures.

They were selected and added to see how this learning affects the way we classify the severity of diseases. InceptionV3 uses convolutional filters of different sizes of parallel convolutional filters to capture characteristics at different scales. Also, it was tested to see how well it balanced the accuracy and speed. DenseNet201 connects convolutional layers together to reuse features and increase the flow of gradients, which results in high classification accuracy but needs more computing power. EfficientNetV2S uses compound scaling to balance depth, width, and resolution; however, its performance was limited on the proposed dataset. MobileNet and MobileNetV2 are lightweight designs that use depth-wise separable convolutions. This makes them a good choice for mobile and embedded agricultural applications. Xception replaces traditional convolution blocks with depth-wise separable convolutions and was included as a high-efficiency pretrained architecture.

These architectures are cornerstone designs in this area and provide different potency in computational efficiency, reduction of parameters, and performance. In most cases, the choice of architecture depends on the specific requirements of the application in terms of computational resources, accuracy needs, and the nature of the dataset.

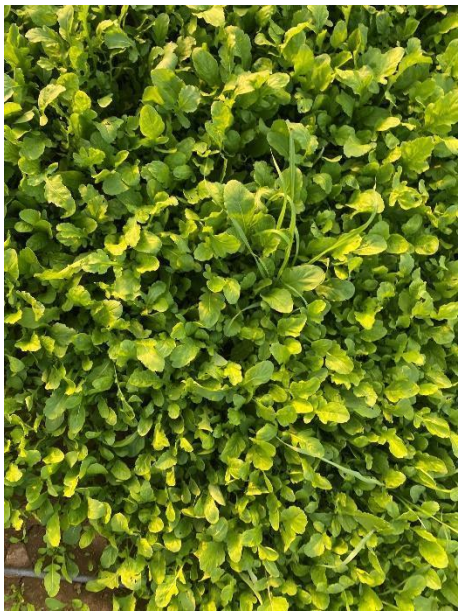
3. Image Collection and Classification Results

In this section, a dataset collected from a real greenhouse was used. The collected data were labeled into five classes based on predefined criteria. Various preprocessing steps were applied to the dataset, including masking and normalization. The experiments utilized several pretrained models, which were applied to both masked and unmasked data. The results were recorded for further analysis. These results provide insights into the effects of the masking process on the dataset and the performance of the models used.

3.1 Data collection

In the dataset creation phase, images were taken from arugula plants grown in greenhouses using a camera of a mobile phone iPhone 11 PROMAX, Apple Inc, Cupertino, California, United States). This allowed for controlled conditions, ensuring consistent lighting, orientation, and resolution of the leaf images. To create a comprehensive data set that would enable the successful application of masking techniques used in pretrained models, the researchers carefully curated a collection of healthy and affected arugula leaf images.

Arugula plant, which is widely grown around the world, was selected for use in the study. In the images taken, healthy and 3 different disease levels were obtained according to the disease levels. A total of 704 images were used to create the data set. There is no data set previously collected for this type of scientific research. This data set will form a unique basis for another future research. Figure 3 presents some images to represent the disease severity of the arugula. The data set consists of four classes: healthy plants and three severity levels of root rot disease as seen in Figure 3.



(a) No disease.



(b) Severity Level 1.



(c) Severity Level 2. (d) Severity Level 3.

Figure 3- Root to disease samples

3.2. Computational results and discussion

In the application results, to make sure the comparison was fair, all the models were trained on the same data sets. The stratified sampling method was used to divide the dataset into three groups: training (70%), validation (15%), and test (15%). The Adam optimizer was used for training where the performance results were recorded with 5-fold training and testing. The learning rate was set to 1e−4, the batch size was 32, and the maximum number of epochs was initially set to 50. However, based on validation performance, the final number of epochs was reduced to 30 since to avoid overfitting, the validation loss was stopped early. We used the accuracy, precision, recall, and F1-score metrics to test the model on the test set that was not used for training. We used the same hardware and software for all tests to ensure uniformity. There were four nodes in the output layer for each model, with each node representing one of the four disease severity classes.

SBFC-1 is a model based solely on symbolic functions. Its architecture consists of a custom convolution layer via kernel, followed by ReLU activation, standard CNN convolution, MaxPooling2D, flattening, another ReLU layer, a fully connected layer, dropout at 0.1, and a final fully connected layer.

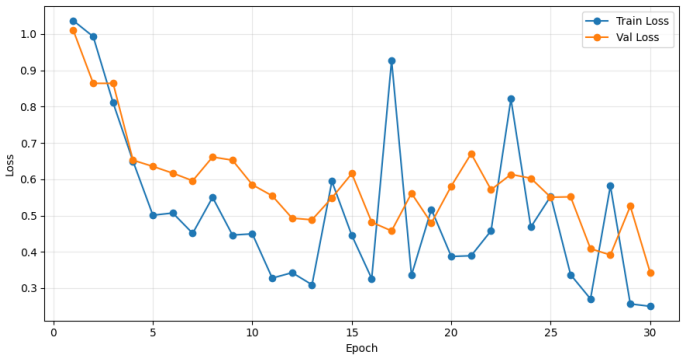


Figure 4- Training and validation loss curves of the proposed SFBC-1 model

SBFC-1 model achieved a loss of 0.251, precision of 0.840, recall of 0.853, F1-score of 0.84, and an accuracy of 85.30%, respectively. The average processing time per image is approximately 45 seconds. The training and loss function values are shown in Figure 4. After 30th epoch, the validation loss increases and training loss converges to zero. The proposed SBFC-1 model provides lower loss values and higher accuracy over epochs compared to the conventional CNN model. Table 1 presents the confusion matrix of the SBFC-1 classification results. The matrix indicates that while the model performs strongly on the majority classes (Healthy and High), there is noticeable misclassification particularly between Medium and High severity levels, as well as between Low and Medium classes. This behavior is expected due to visual similarity between Medium and High severity levels. The dataset is class-imbalanced; therefore, precision, recall, and F1-score are reported using macro-averaging to avoid bias toward the majority classes and to provide a more balanced evaluation of disease severity classification performance.

Furthermore, five-fold cross-validation was conducted to ensure robustness and stability of the proposed model across different data splits. As shown in Table 1, the model classifies healthy arugula leaves more accurately than diseased leaves. When the problem is considered as a binary classification task (healthy vs. diseased), the overall accuracy exceeds 98%, indicating that the model is highly effective at classifying healthy leaves from infected ones.

Table 1- Confusion matrix for SBFC-1 model classification results

True \ Pred.	*Healthy*	*Low*	*Medium*	*High*
Healthy	255	10	0	0
Low	0	38	11	1
Medium	0	6	78	11
High	0	0	65	215

SBFC-2 is an entropy-based symbolic function kernel model. It follows a similar architecture to SBFC-1, replacing the kernel-based convolution with one based on entropy. The rest of the pipeline remains the same: ReLU, CNN convolution, MaxPooling2D, flattening, ReLU, fully connected layer, dropout at 0.5, and a final fully connected layer. This model reported a loss of 0.302, precision of 0.724, recall of 0.711, F1-score of 0.663, and accuracy of 71.4%, completing its process in 7 minutes and 15 seconds. It processes each image in approximately 0.65 seconds.

SBFC-3 is a variance-based symbolic function kernel model. Although detailed architectural and performance metrics were not provided, it is noted for its improved efficiency with an average processing time of approximately 1.3 seconds per image, suggesting a trade-off between performance and computational speed.

The results obtained after training the four-layers (feature map numbers are: 1st 32, 2nd 64, 3rd 128, 4th 256) simple CNN model shown in the table above. Table 2 presents the numerical results for the simple CNN model and pretrained models' classification of root rot disease dataset. The attribution of the table above will be based on the metrics such as Loss, Accuracy, Precision, Recall, and F1 Score. Specific performance comparisons of some models on the basis of performance, complexity, and computational cost are as follows:

VGG19 - VGG19 had an accuracy of 80.54% with a loss of 0.4892. It had an accurate rate of 85.25%, a recall of 80.54%, and an F1 score of 77.63%. VGG19 has 19 layers in its architecture; hence, the construction of deep architecture that captured complicated features. This depth nonetheless increased computational demand and memory. A large number of used parameters further increased the time needed for training and inference. It performed slightly worse compared to VGG16, with a small margin of difference in accuracy and F1 score but performed better in terms of recall.

VGG16 outperformed VGG19 slightly, with a loss of 0.4142 and an accuracy of 81.88%. It had a precision of 83.09%, a recall of 81.88%, and an F1 score of 80.25%. VGG16, with its 16 layers, was less complex and computationally expensive compared to VGG19. That being said, it had about 138 million parameters and therefore was not lightweight. Thus, VGG16 outperforms VGG19 with respect to accuracy and F1 score; therefore, it could be applied in cases when memory and computational power were not worse. On the other hand, DenseNet201 was shown with a loss of 0.2187- the highest accuracy of 91.95% shown. Indeed, its precision, recall, and F1 score are all higher than 92%, by far the best result on these metrics. DenseNet201 had dense connections between layers, encouraging feature reuse and easing the gradient flow. However, such complexity and many parameters require heavy memory and computational resources. DenseNet201 had the highest accuracy, precision, recall, and F1 score, but it had a higher computation compared with the rest of the models.

As a different CNN approach, EfficientNetV2S performed much worse compared to the other models, providing a loss of 1.0016 and an accuracy of 57.72%. Precision and recall were 48.22%, while the F1 score was 51.9%. The model EfficientNetV2S was scaled by compound scaling, which balances network depth, width, and resolution with efficiency. It seems to struggle with this dataset probably due to not being adequately trained or large enough and which might be indicative of EfficientNetV2S being poorly suited to the task sans additional tuning of this model.

Table 2- Performance of Designed Models

Model	*Loss*	*Accuracy*	*Precision*	*Recall*	*F1 Score*
SBFC-1 (1 Conv.)	**0.251**	**0.853**	**0.790**	**0.832**	**0.799**
SBFC-2 (1 Conv.)	0.302	0.714	0.724	0.711	0.663
SBFC-3 (1 Conv.)	0.273	0.840	0.766	0.805	0.772
CNN (4 Conv.)	0.495	0.782	0.740	0.752	0.731
VGG16	0.489	0.805	0.852	0.805	0.776
VGG19	0.414	0.818	0.830	0.818	0.802
DenseNet21	0.218	**0.919**	**0.923**	**0.919**	**0.920**
EfficientNet 2S	1.001	0.577	0.482	0.577	0.519
InceptionV3	0.380	0.825	0.876	0.825	0.822
MobileNetV2	**0.175**	**0.919**	0.920	**0.919**	0.918
MobileNet	0.311	0.845	0.886	0.845	0.846
ResNet50	0.739	0.624	0.593	0.624	0.567
ResNet101	0.823	0.677	0.764	0.677	0.604
Xception	0.309	0.852	0.854	0.852	0.846

InceptionV3's model loss was 0.3807 and accuracy was 82.55%. The model reaches a precision of 87.66%, recall of 82.55%, and F1 score of 82.28%. InceptionV3 employs inception modules, which efficiently capture features with good performance at multiple scales. The total parameters of this network were approximately 23 million, which was comparatively efficient in computational cost.

MobileNetV2 had better accuracy than VGG models and most ResNet variants, while requiring much fewer resources compared to those networks on this dataset. Its loss was very low, with an exact value of 0.1753, while its accuracy had grown to be among the highest-91.95%. Precision, recall, and F1 score are all around 92%-efficient, indeed. Optimized for mobile and embedded vision applications, MobileNetV2 offers an architecture that, for best performance, leverages a very unique combination of depthwise separable convolutions, each preceded by channel shuffling and pixel unshuffling to allow reduced computation cost and memory. It contained approximately 3.4 million parameters; hence, it was highly efficient. MobileNetV2 could be used to provide a perfect balance between very high performance at low computational cost; hence, it becomes one of the best options in a resource-constrained environment with not much sacrifice in terms of accuracy and F1 score. MobileNet's loss was 0.3118 with accuracy of 84.56%. Depthwise separable convolutions were working almost exactly as in MobileNetV2, but this network lacked some optimizations that were present in MobileNetV2. It contains about 4.2 million parameters. While MobileNet did perform quite decently, it was outdone by MobileNetV2, which was even more efficient and hence a more complete model.

ResNet50, which had a residual approach among CNN models, had a 0.7399 loss, while the accuracy was at 62.42%. Precision was low, at 59.34%, and recall at 62.42%, leading to a poor F1 score of 56.73%. ResNet50 relied on a residual learning approach to ease the training of deep networks. The model possessed an overall 50 layers with roughly 25.6 million parameters. However, it was likely to overfit or not generalize well on the provided dataset. It performed poorly when compared with most of the models. It could be that fitting the current dataset went wrong. With ResNet101's loss is 0.823, and ResNet101 has 67.79% accuracy. Precision is 76.4%, recall is 67.79% while F1 score is 60.48%.

ResNet101 doubled the layers from ResNet50, and its parameters increased to around 44.5 million; thus, it was more computationally expensive. This depth surely helped in learning features that were more complicated but may have required more data and careful tuning. ResNet101 did somewhat better compared to ResNet50 but did not perform as well as the models such as DenseNet201 and MobileNetV2.

Xception had 0.309 loss and an accuracy of 85.23%. In an overall sense, the precision, recall, and F1 score were all around 85%. Xception fully exploits depthwise separable convolutions similar to MobileNet but much bigger in size, holding around 22.9 million parameters. It was found to have a great balance between high performance and computational efficiency as well. Meanwhile, Xception was an extremely good model, compared to InceptionV3 but just a little more computationally efficient if the structure is taken into consideration.

In other words, the best performance with the highest accuracy, balanced precision, recall, and F1 score was attained by DenseNet201 and MobileNetV2 from all studied architectures. Other strong performances had been reported by VGG16, InceptionV3, and Xception showed high accuracy and F1 scores. Medium performances obtained in EfficientNetV2S and ResNet50 could be due to present data and perhaps point to model size or training issues. Looking into the model complexities and the computational cost, MobileNetV2 would most likely be the most efficient model that provides high performance with low resource consumptions. This was considering the dense connectivity of DenseNet201, which was computationally very expensive, though it yielded the highest accuracy. VGG models were known for their strong performance, but they came with

high computational costs because of their deep architecture. Therefore, selecting a model should consider both performance needs and the computational resources at hand.

As a result of the research, DenseNet201 and MobileNetV2 models were determined to have the best performance with the highest accuracy, balanced precision, recall and F1 score among all the examined architectures. VGG16, InceptionV3 and Xception were determined as other strong performances showing high accuracy and F1 scores. The moderate performances obtained in EfficientNetV2S and ResNet50 can be attributed to the presented data. In terms of model complexity and computational cost, MobileNetV2 is most likely to be the most efficient model, providing high performance with low resource consumption. This contrasts with the dense connectivity of DenseNet201, which is very expensive in terms of computation but provides the highest accuracy. VGG models were known for their high performance but have a high computational cost due to their deep architecture. Therefore, both performance requirements and available computational resources should be taken into consideration when choosing a model.

Notice that the proposed SFBC framework is primarily designed to improve feature extraction in lightweight and low-parameter CNN architectures. While modern pretrained models such as MobileNetV2 already achieve strong performance through deep hierarchical representations, the symbolic filtering approach provides its main advantage in scenarios where computational resources are limited, and model simplicity is required. This study has some limitations, even though the results are promising. The dataset is relatively limited and only comes from one greenhouse environment. Future research will concentrate on extensive multi-site datasets and real-time field validation to enhance the evaluation of generalization performance. The proposed approach does not involve personal or sensitive data, as all images were collected from agricultural environments. From a sustainability point of view, being able to accurately assess the severity of a disease makes it possible to use pesticides more effectively, which helps the farmers or producers to decrease the amount of chemical usage during agricultural production. Also, site-specific application can be done by assessing the severity level of root rot disease.

4. Conclusions

In this study, we investigated the effects of different pre-trained CNN models on the data set collected from real greenhouse conditions, specifically focusing on rocket plants. The data was classified into four different classes based on predefined criteria, and the performance of the models for comparison was assessed using a number of large pre-trained models as well as basic CNN models. The performance of the models for comparison was assessed using a number of large pre-trained models as well as basic CNN models. Two pre-trained models consistently predict better results than the other selected models in each evaluation. There are important practical implications if these models are successfully applied to the arugula dataset. Due to the high accuracy of the model, pesticides can be applied more precisely and effectively, reducing waste and improving crop health. These encouraging results suggest that this approach can be successfully applied to variable-rate pesticide application under actual field conditions when combined with an embedded device. This development offers a powerful way to minimize pesticide use and improve crop management, which is a step forward for precision agriculture.

Author Declarations

Authors' contributions: Ikbal Aygun: Data curation, Writing-Original Draft, Writing- Review & Editing. Onur Umit Şener: Software, Writing-Original Draft. Ulas Ozbent: Software, Writing-Original Draft. Selami Beyhan: Conceptualization, Methodology, Supervision, Writing- Review & Editing.

Conflict of interest/Competing interests: The author(s) declare no conflict of interest.

Ethics approval/declarations: No need/ Not applicable.

Funding: This research received no external funding.

Consent to participate: No need/ Not applicable.

Consent for publication: No need/ Not applicable.

Data availability: Data are available on request due to privacy or other restrictions.

References

Aashu Rajwar K, Pant M & Deep K (2024). Application of Machine Learning in Agriculture: Recent Trends and Future Research Avenues, arxiv.

Ayan E, Erbay H & Varçın F (2020). Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks,Computers and Electronics in Agriculture,179, 105809, ISSN 0168-1699 https://doi.org/10.1016/j.compag.2020.105809.

Gan H, W S Lee, V Alchanatis, R Ehsani & J K Schueller (2018) Immature green citrus fruit detection using color and thermal images, Computers and Electronics in Agriculture152: 117-125. https://doi.org/10.1016/j.compag.2018.07.011

He K, Zhang X, Ren S & Sun J (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778) https://doi.org/10.1109/cvpr.2016.90

Howard A G (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Huang G & Liu Z, Van Der Maaten L & Weinberger K Q (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708)

LeCun Y, Bengio Y & Hinton G (2015). Deep learning. *nature*, 521(7553): 436-444

Liakos K G, Busato P, Moshou D, Pearson & S Bochtis (2018). Machine learning in agriculture: A review. Sensors 18(8): 2674 https://doi.org/10.3390/s18082674

Linker R, Cohen O & Naor A (2012). Determination of the number of green apples in RGB images recorded in orchards, Computers and Electronics in Agriculture 81: 45-57 https://doi.org/10.1016/j.compag.2011.11.007

Mohanty S P, Hughes D P & Salathé M (2016). Using deep learning for image-based plant disease detection. Frontiers in plant science, 7: 1419. https://doi.org/10.3389/fpls.2016.01419

Morales M & Janick J (2022). Arugula: A promising specialty leaf vegetable. Reprinted from: Trends in new crops and new uses.

Navarro A, Nicastro N, Costa C, Pentangelo A, Cardarelli M, Ortenzi L & Pane C (2022). Sorting biotic and abiotic stresses on wild rocket by leaf-image hyperspectral data mining with an artificial intelligence model. Plant Methods, 18(1): 45 https://doi.org/10.1186/s13007-022-00880-4

Neri I, Caponi S, Bonacci F, Clementi G, Cottone F, Gammaitoni L & Mattarelli M (2024). Real-Time AI-Assisted Push-Broom Hyperspectral System for Precision Agriculture. Sensors, 24(2): 344 https://doi.org/10.3390/s24020344.

Palumbo M, Cefola M, Pace B, Colelli G & Attolico G (2024). Machine learning for the identification of colour cues to estimate quality parameters of rocket leaves. *Journal of Food Engineering* 366: 111850. https://doi.org/10.1016/j.jfoodeng.2023.111850

Sabatino L, Consentino B Rouphael Y, De Pasquale C, Iapichino G, D'Anna F & La Bella S (2021). Protein hydrolysates and mo-biofortification interactively modulate plant performance and quality of 'canasta'lettuce grown in a protected environment. Agronomy, 11(6): 1023. https://doi.org/10.3390/agronomy11061023

Shanthini K S, Sudhish N George, Athul Chandran O V, Jinumol K M, Keerthana P, Jobin Francis& Sony George (2025). NorBlueNet: Hyperspectral imaging-based hybrid CNN-transformer model for non-destructive SSC analysis in Norwegian wild blueberries, Computers and Electronics in Agriculture, Volume 235,110340, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2025.110340. https://doi.org/10.1016/j.compag.2025.110340

Schmidhuber J (2015). Deep learning in neural networks: An overview. Neural networks 61: 85-117. https://doi.org/10.1016/j.neunet.2014.09.003

Schueller J K, J D Whitney T A Wheaton, W M Miller & A E Turner (1999). Low-cost automatic yield mapping in hand-harvested citrus, Computers and Electronics in Agriculture, 23(2): 145-153. https://doi.org/10.1016/s0168-1699(99)00028-9

Szegedy C, Vanhoucke V, Ioffe S, Shlens J & Wojna Z (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826). https://doi.org/10.1109/cvpr.2016.308

Torres-Tello J, Venkatachalam S, Moreno L & Ko S B (2020). Ensemble learning for improving generalization in aeroponics yield prediction. In 2020 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 1-5). IEEE. https://doi.org/10.1109/iscas45731.2020.9181283

Williamson-Benavides B A& Dhingra A (2021). Understanding root rot disease in agricultural crops. Horticulture 7(2): 33. https://doi.org/10.20944/preprints202012.0681.v1

Copyright 2026 The Author(s). This is an open-access article published by Faculty of Agriculture, Ankara University under the terms of the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium or format, provided the original work is properly cited.