



A new hybrid approach for grapevine leaves recognition based on ESRGAN data augmentation and GASVM feature selection

Gürkan Doğan¹ · Andaç Imak² · Burhan Ergen³ · Abdulkadir Sengur⁴

Received: 2 February 2023 / Accepted: 14 January 2024 / Published online: 19 February 2024
© The Author(s) 2024

Abstract

Grapevine leaf is a commodity that is collected only once a year and has a high return on investment due to its export. However, only certain types of grapevine leaves are consumed. Therefore, it is extremely important to distinguish the types of grapevine leaves. In particular, performing this process automatically on industrial machines will reduce human errors, workload, and thus cost. In this study, a new hybrid approach based on a convolutional neural network is proposed that can automatically distinguish the types of grapevine leaves. In the proposed approach, firstly, the overfitting of network models is prevented by applying data augmentation techniques. Second, new synthetic images were created with the ESRGAN technique to obtain detailed texture information. Third, the top blocks of the MobileNetV2 and VGG19 CNN models were replaced with the newly designed top block, effectively extracting features with the data. Fourthly, the GASVM algorithm was adapted and used to create a subset of the features to eliminate the ineffective and unimportant ones from the obtained features. Finally, SVM classification was performed with the feature subset consisting of 314 features, and approximately 2% higher accuracy and MCC score were obtained compared to the approaches in the literature.

Keywords Grapevine leaf recognition · Feature selection · Deep learning · Enhanced super-resolution generative adversarial networks

1 Introduction

Plants are one of the most important basic food sources for the survival of other living things. It is estimated that there are approximately 400,000 plant species in nature [1]. In addition to being one of the most important food sources, it is also used in medicine and industry. Recognition of plant species is important not only for consumption, but also for the conservation of species in nature. Therefore, experts are needed in the determination of plant species. Botanists often make the distinction by observing the leaf characteristics of plants. Traditional methods like this are a very time-consuming, extensive, and complex process.

The grapevine leaves discussed in this article are a very useful type of leaf that is widely used in traditional Turkish and Mediterranean food cultures. Grapevine leaves are rich in low-calorie, high amounts of dietary fiber, calcium, phylloquinone (K1), phenolic compounds, phosphorus, and vitamin C. In addition, grapevine leaves can be stored frozen, canned, and in brine and therefore have a long shelf life. Also, some types of grapevine leaves provide a better

✉ Gürkan Doğan
gurkandogan@munzur.edu.tr

Andaç Imak
andacimak@munzur.edu.tr

Burhan Ergen
bergen@firat.edu.tr

Abdulkadir Sengur
ksengur@firat.edu.tr

¹ Department of Computer Engineering, Faculty of Engineering, Munzur University, Tunceli, Turkey

² Department of Electrical and Electronic Engineering, Faculty of Engineering, Munzur University, Tunceli, Turkey

³ Department of Computer Engineering, Faculty of Engineering, Firat University, Elazığ, Turkey

⁴ Department of Electrical and Electronic Engineering, Faculty of Technology, Firat University, Elazığ, Turkey

economic return than grapes. It is reported that approximately 13.5 million dollars of export revenue is obtained from Turkey's grapevine leaf exports, and 135 million dollars of export revenue is obtained from wrapping and stuffing produced from leaves [2]. The leaves have different properties in terms of thickness, thinness, hairiness, and shape. Not all types of grapevine leaves are preferred as food. Those that will be used for food, i.e., consumption, should be as fine-grained, hairless, and thin as possible. In addition, varieties without slices and with a sour taste on the palate are preferred. For this reason, it is an important need to recognize the grapevine leaf species to be used for consumption. In particular, to export a large number of grapevine leaves, it is necessary to determine the grapevine leaf types with fast, error-free, and cost-effective solutions. The most suitable of these solutions is to detect grapevine leaves autonomously with industrial machines. In this study, an approach with a very high accuracy rate is presented that will perform the recognition of grapevine leaves autonomously.

There are many studies on the classification of plant species with artificial intelligence applications developed in the field of agriculture [3–7]. First, classification studies were carried out with datasets such as Flavia [8], Swedish [9], ICL [10], and Foliage [11], which include plant species in the literature. In the past, many researchers have evaluated the performance of these manually extracted features based on shape, texture, and color features using machine learning methods. In feature-based studies, Histogram of Oriented Gradient (HOG) [12, 13], Centroid-Center Distance [14–16], Fourier Descriptor [9, 17, 18], geometric or morphological features [19–21], Gray-Level Co-occurrence Matrix (GLCM) [11, 22, 23], and local binary pattern [24] were used. In recent years, with the development of deep learning instead of machine learning techniques, low-performance problems of complex images in the field of computer vision have been solved. With the rapid progress of deep learning techniques, many studies have been carried out in the literature to automatically extract leaf features for plant species recognition [25]. Compared to traditional feature extraction models, deep learning techniques are completely data-driven and generally have higher feature possibilities for capturing distinctive features. The deeper and wider the mesh of the architecture, the more image information is obtained. In deep learning, architectures like GoogleNET [26], VGG [27], ResNet [28], DenseNet [29], MobileNetV2 [30] and Inception V4 [31] and VGG19 [27] have gotten a lot of attention. In addition, there are studies on diseases of plant leaves with deep learning methods. Ma et al. [32] carried out a study on the detection of diseases in cucumber leaves. In the deep architecture they proposed based on VGG, 4 diseases were classified in plant leaves. Dhivyaa et al. [33] proposed a

method for the detection of plant diseases that includes dense blocks, an expanded complexity network, and a bidirectional long short-term memory. Bhujel et al. [34] detected tomato leaf diseases using a light convolutional neural network model they developed. Prabu and Chelliah [35] proposed a MobileNetv2 and SVM-based method to detect mango leaf diseases. Singh et al. [36] used the AlexNet network model to detect diseases in corn leaves. Li et al. [37] developed a lightweight convolutional neural network RegNet model for apple leaf diseases. Five different apple leaf diseases were identified in this study. Anari [38] proposed a CNN-based ResNet 18 model to detect six different plant diseases, namely Apple, Corn, Cotton, Grape, Pepper, and Rice. For deep learning-based plant classification, architectures such as DeepPlant [39], Multi-Scale Convolution Neural Network (MSF-CNN) [25], ResNet-50 [40], and BLeafNet [1] have been proposed. Pawara et al. [41] proposed a new plant leaf classification method based on Inception-v3 and ResNet-50 network models. On the other hand, there are very few studies in the literature on recognizing grapevine leaf species. Carneiro et al. proposed an approach based on the transfer learning Xception Convolution Neural Network (CNN) model for the automatic classification of 12 grape varieties found in the Douro Demarked region of Portugal. They achieved an accuracy of 93% with this approach [42]. Another study, Koklu et al. by using a dataset created from grape leaf varieties collected from the Central Anatolian Region of Turkey. They used the pre-trained MobileNetV2 CNN model for feature extraction. In addition, they achieved 97.6% accuracy by using the Chi-square algorithm to select the most important features and the SVM algorithm to classify the selected features [43]. However, the performance successes obtained from the methods suggested in these studies are insufficient for recognizing the grape leaf. Therefore, in this study, we propose a new hybrid method with very high performance.

In this study, a proposed new hybrid approach for grapevine leaf recognition consists of 5 basic steps: data augmentation, training of CNN, feature extraction, feature selection, and classification. In the data augmentation step, augmentation was performed using the Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) model alongside traditional data augmentation techniques to preserve the detailed texture feature of the data. Then, CNN models such as MobileNetV2 and VGG19, which are known for their success in object recognition and pre-trained with ImageNet, were used to extract features from the data. In addition to these network models, the Top block-1 structure has been developed to provide the best learning, and these layers have been integrated into the top layer of the CNN models and training has been carried out. In the feature extraction step, the features were extracted

from the dense (160) layer of the Top block-1 structure and combined (fusion). Then, feature selection was made from these combined features with the support vector classifier (SVC) method. However, when these selected features were classified with SVM, the classification success remained at 98.86%. In this study, the Genetic Algorithm-Based Support Vector Machine (GASVM) algorithm was developed and applied to select the best feature subset. As a result, 314 important features were selected from 640 features with GASVM. In the last stage, classification was made using the SVM algorithm, whose success is known. As a result of the experimental tests, 99.57% accuracy was obtained with this proposed approach.

In summary, the main contributions of this study are:

- A new hybrid approach has been designed for the recognition of various grapevine leaf species.
- Data augmentation techniques and ESRGAN were applied to reproduce the dataset.
- With the designed top block-1 structure, MobileNetV2 and VGG19 transfer learning models were used as feature extractors.
- The readapted GASVM algorithm was applied for the selection of the best deep features.
- An excellent accuracy score of 99.57% was obtained using the proposed approach.

The remaining parts of this article are organized as follows; in Chapter 2, there is information about the dataset, data augmentation methods, block structures used by CNN models, and feature subset selection method. The approach proposed in Chapter 3 and the experimental results in Chapter 4 are evaluated and analyzed. Finally, Chapter 5 contains general conclusions.

2 Material and methods

2.1 Grapevine leaf dataset

A grapevine leaves dataset created by Koklu et al. [43] and shared as open access to researchers (<https://www.kaggle.com/datasets/muratkokludataset/grapevine-leaves-image-dataset>) were used in the study. A system has been designed to eliminate distorting factors such as shadows in the images to be obtained. The leaf images were captured with a Prosilica GT2000C camera in a specially illuminated indoor environment. Images of grapevine leave acquired before the grape harvest was used to create a dataset. There are five different classes in this dataset: Ak, Ala Idris, Büzgülü, Dimnit, and Nazli. A total of 500 images are contained in the dataset, which consists of 100 images with dimensions of 511×511 for each class.

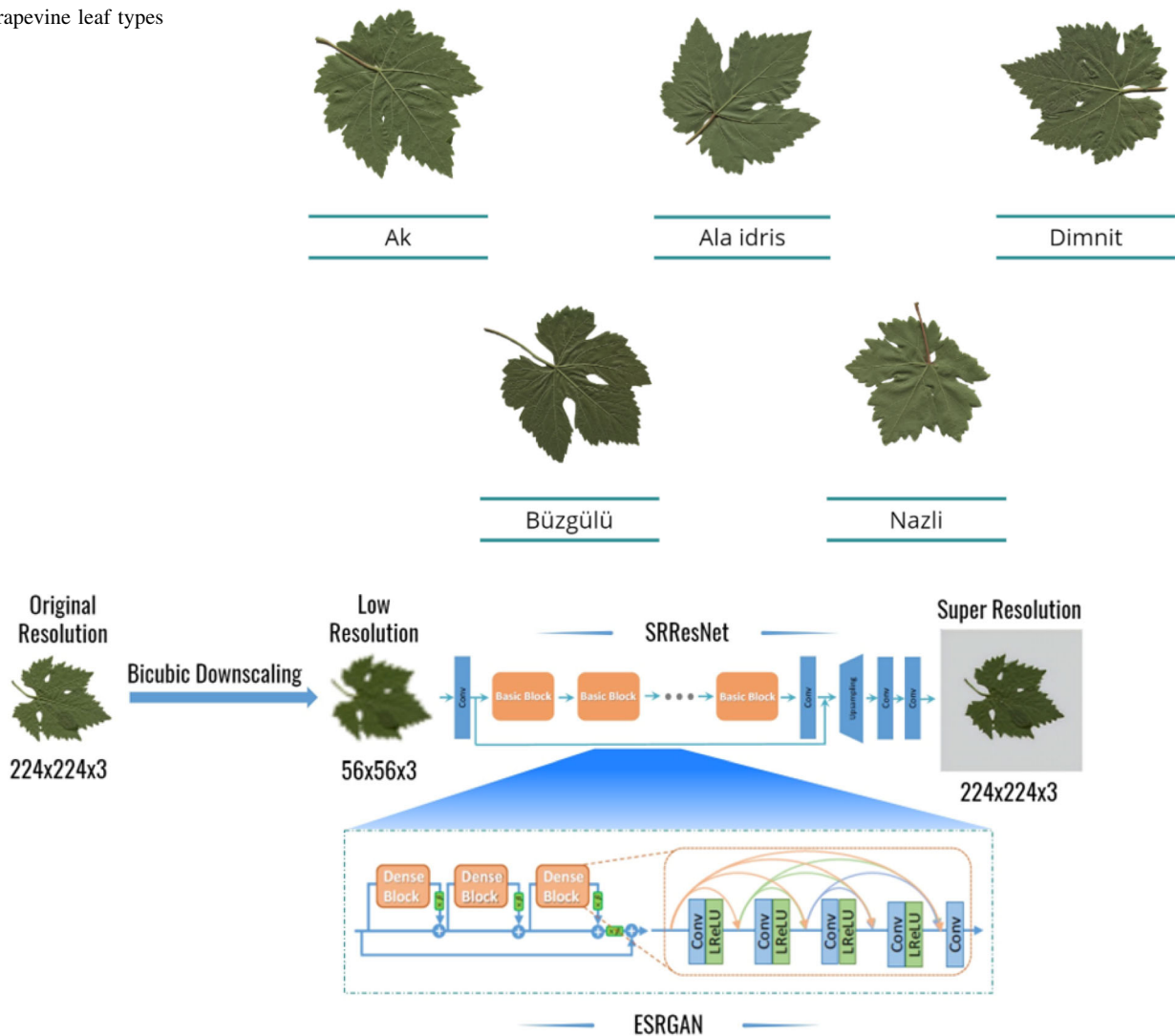
Sample images for five different classes are shown in Fig. 1.

2.2 Data augmentation

Data augmentation is a very useful method for enlarging datasets that contain very few images. The performance of the system in image classification can be significantly improved by using effective data augmentation techniques. In this study, augmentation techniques were applied to the dataset consisting of grapevine leaf images. Albumentation was utilized, which is a Python library that includes various augmentation techniques [44]. This library includes traditional data augmentation techniques such as random resizing, CLAHE, horizontal flip, and vertical flip.

ESRGAN structure, a generative modeling method, was used to extract detailed textures from augmentation images in this study. The method proposed by Wang et al. in 2018 aims to obtain higher-resolution images from low-resolution images [45]. ESRGAN, which won first place in terms of the perceptual index in the PIRM-SR Challenge, uses the basic architecture of SRResNet [46] shown in Fig. 2. Two-step changes were made to this basic architecture. As a first step, batch normalization (BN) layers are removed in all convolution blocks. It has been observed that super-resolution [47] and deblurring [48] reduce computational complexity as well as improve performance in different peak signal-to-noise ratio (PSNR)-focused tasks. Batch normalization performs normalization using the mean and variance of all inputs in a layer. Moreover, Wang et al. empirically observed that under a GAN structure, training and testing datasets are more likely to introduce artifacts if the statistics are very different or the network is deeper. For this reason, they have shown in their experimental studies that they provide stable training and consistent performance by removing the batch normalization layers in the ESRGAN structure they have developed. In the second step, as can be seen from the model in Fig. 2, the blocks are replaced with Residual-in-Residual Dense Blocks (RRDB), which combines the multi-level residual network and dense connections. Improved ESRGAN architecture with more layers and created residual blocks provided better perceptual quality.

In this study, a total of 3500 images with 224×224 dimensions were obtained using traditional data enlargement techniques such as random resizing, CLAHE, horizontal flip, vertical flip, random cropping, and random rotation. In addition, another 3500 synthetic images with 224×224 dimensions were obtained by applying the ESRGAN technique to these augmented data. As a result, 7 thousand images were created using these data augmentation methods. Figure 3 shows the new images obtained by

Fig. 1 Grapevine leaf types**Fig. 2** Obtaining a super-resolution image with the ESRGAN method from a low-resolution image belonging to Nazlı class and the general block structure of the ESRGAN model

augmentation techniques of the grapevine leaf class Nazlı in the dataset.

2.3 Transfer learning

The limited size of the dataset is not sufficient to extract distinctive features with CNN layers. Therefore, it is very difficult to achieve the high performance of a deep learning model trained from scratch using a small dataset. The transfer learning method is widely used to solve this problem [49]. This method takes information represented as weights of pre-trained network models with a large dataset. These weights are then applied to tiny datasets to learn particular features, allowing for high-performance classification. In this paper, we suggest that a layer called Top block-1 be added to the top of the main model to increase the network's performance in transfer learning. In

the proposed model, MobileNetV2 and VGG19 architectures pre-trained with ImageNet are used.

MobileNetV2: MobileNet is a frequently used lightweight convolutional neural network (CNN). The proposed technique is based on the MobileNetV2 [30] concept. Inverted residual and linear bottleneck modules from MobileNetV1 [50] were combined to form MobileNetV2. After the initial convolution layer with 32 filters, MobileNetV2 adds 19 inverted residual bottleneck layers, followed by a pointwise convolution that provides output with a size of $7 \times 7 \times 1280$ pixels. There are 3.4 million parameters in the MobileNetV2, which is less than other popular CNN models. We first used ImageNet-based MobileNetV2 in the transfer learning model. It is an efficient network model with a short execution time. As a result of data augmentation and ESRGAN proposed in the architecture, 160 features were produced from each image.

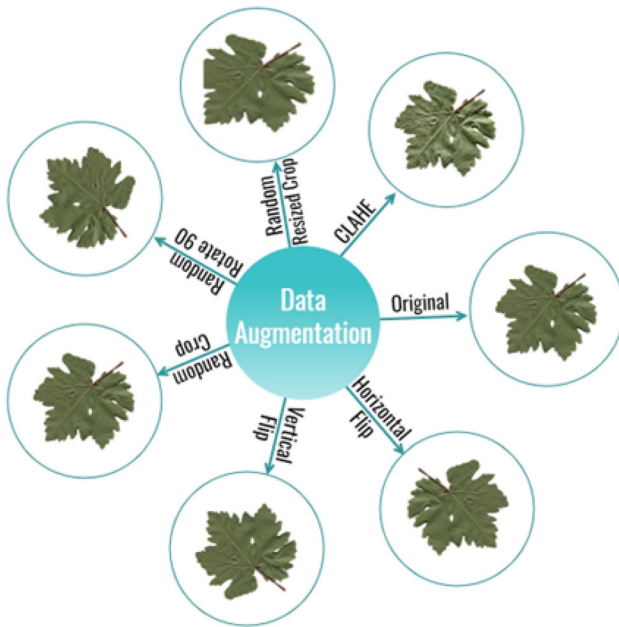


Fig. 3 Obtaining new images by applying different augmentation techniques to an image belonging to Nazli class

VGG19: In ImageNet Challenge 2014, it left its mark on the localization and classification studies of the Visual Geometry Group (VGG) team, taking first and second places, respectively. This paper utilizes Simonyan and Zisserman’s proposed 19-layer VGG19 architecture [27]. Five convolutional layers, five max-pooling layer blocks, and three fully connected layers make up the VGG19 architecture. In the architecture, the convolution layers with a 3×3 -dimensional step number of 1 and the subsequent rectified linear unit (ReLU) activation after each convolution layer are passed. As a pooling process, it consists of a maximum pooling layer with 2×2 -dimensional steps. Two fully connected layers with 4096 ReLU-activated units are employed before the final 1000 fully connected softmax layer [51]. In the proposed study, the VGG19 transfer learning method was used as a hybrid with MobileNetV2. In the ImageNet dataset, 160 features of each grapevine leaf were obtained with the pre-trained VGG19.

2.4 Top block structures

In the proposed approach, as the leaf classification task is performed by transfer learning, top block-2 is required as a minimum requirement as a standard. These top block structures are used with both VGG19 and MobilenetV2. The number of trainable parameters is 2565 when used together with the VGG19 CNN model with top block-2, while the number of trainable parameters is 6405 when used together with the MobileNetV2 CNN model. As a result of the experimental tests, it was seen that the number of learnable parameters was insufficient for the datasets used, and effective learning could not be achieved. For this reason, a new top block structure, top block-1, has been developed for the VGG19 and MobileNetV2 CNN models. When this top block structure is used together with VGG19, the number of trainable parameters is 372165, while the number of learnable parameters is 617925 when used with MobileNetV2. This means that the top block-1 structure has approximately $\times 145$ more parameters in the VGG19 CNN model and $\times 96$ times more parameters in the MobileNetV2 CNN model than the top block-2 structure. With the top block-1 structure, these CNN models can perform better learning. On the other hand, the developed top block-1 structure takes the data from the last layer of the network models as input and passes it through two convolution layers. In this way, more features are extracted. Moreover, after the first convolution layer, it is passed through the max-pooling layer, combining the most important features in the feature map with the features obtained in the second convolution layer, so that more detailed texture information is transferred.

Then, the concatenate features are passed to the fully connected layer, and first of all, 160 features are created. The number of these features was determined iteratively as a result of experimental tests. Then, these features are reduced to the class number (five) of the dataset, and classification is performed with softmax. In addition, when the top block-1 structure is used for feature extraction, features are extracted from the dense (160) layer, while all layers are used in classification.

Algorithm 1 GASVM Algorithm

```

Input:
    //Input Feature Set
     $V_{lf} \leftarrow$  Feature vector encoded in chromosome
    //Genetic Algorithm Parameters
     $Gen_{max} \leftarrow$  Maximum Generation
     $M \leftarrow$  Number of Population
     $N \leftarrow$  Number of Features
    //Fitness Function Parameters
     $W_a \leftarrow$  Classification accuracy weight
     $W_f \leftarrow$  weight of the number of features

Output:
     $F^* \leftarrow$  Optimal Features Subset

GASVM():
    1. Begin
    2. Initialize the value  $t \leftarrow 1$ , mutation rate and crossover rate
    3. Using the feature vector of the train set and test set
    4. Generate the initial population in a random fashion ( $Pop(t)$ )
    5. while (!Termination condition) begin
    6.     for each  $j \leftarrow 1$  to  $Pop_{size}$  begin
    7.         Decode  $j^{th}$  chromosome to obtain the corresponding feature subset  $V_{lj}$ 
    8.         Apply the SVM model according to  $V_{lj}$ , and compute the average
            accuracy score of  $S_a$ 
    9.         Compute fitness value,  $F_j$  of the  $j^{th}$  chromosome using  $S_a$ 
    10.        Compute total fitness value of the population;
            
$$F_s = W_a \cdot S_a + \left( W_f \cdot \frac{\sum_{i=1}^N F_i(V_{li})}{N} \right)$$

    11.     end for
    12.        The number of individuals with a high fitness value will be selected for
            the next generation.
    13.        Parent Selection using Tournament selection
    14.        Apply crossover operation
    15.        Apply mutation operation
    16.        Population Updation
    17.         $t \leftarrow t + 1$ 
    18.    end while
    19.    return  $F^*$ 
    20. end

```

2.5 Feature subset selection

Feature subset selection is basically an optimization problem that involves finding the optimum one according to a certain accuracy criterion such as accuracy and searching in the space for possible features. The selection of features in a classification task is important to achieve good generalization performance and execution time. In this study, the GASVM algorithm given in Algorithm 1 has been developed to increase the performance of the support vector machine (SVM) [52] in terms of feature subset selection. The genetic algorithm (GA) [53] is used to optimize the feature subset of the SVM.

One of the known approaches to solving the feature selection problem in the literature is the wrapper approach [54, 55]. In this approach, the classifier is trained with a

specific subset of features given as input, and the classification error is estimated using a validation set. Although this is a slower procedure, the selected features are generally more optimal for the classifier used. In the Grapevine leaf recognition task, feature subset selection plays a critical role in the performance of the prediction. It aims to improve SVM-based grapevine leaf recognition prediction with GASVM. This study uses the wrapper approach to select the most suitable subset of features with the SVM algorithm using GA.

In this context, the GASVM algorithm given in Algorithm 1 has been developed. With the GASVM algorithm, firstly, the chromosomes in the initial population containing the feature subset are randomly generated. The fitness value is calculated and evaluated for the selected feature subset in each chromosome. In the SVM classifier,

classification accuracy is evaluated by using the train and test feature samples for each class. The fitness values of individuals can be calculated by adapting the general fitness function given below:

$$\text{Fitness} = \frac{\sum_{i=1}^n F_i}{n} \quad (1)$$

In the equation here, n is the number of features, and F_i is 1 if the actual output values are equal to the predicted values; otherwise, F_i is zero. These fitness values depend on the classification accuracy and the selected features. In the genetic processing step, the number of individuals with high fitness value is selected and the genetic operation steps are continued for the next generation. If the termination criterion is fulfilled, the process is stopped and the best individuals are obtained. If the termination criterion is not fulfilled, the process continues for the next generation using genetic processing, i.e., selection, crossover, and mutation. In this study, the population size was set to 200, the number of features to 640, the number of parents to 100, the mutation rate to 10%, and the number of generations to 200 to execute the GASVM algorithm. The best individuals in the 140th generation were obtained and used for final classification.

3 Proposed approach

The proposed approach consists of five basic steps. In the first step of the model shown in Fig. 5, the dimension of the original dataset consisting of 500 images was set to 224×224 . Image augmentation techniques are then applied to the original dataset to ensure that the CNN models are best learned. The new dataset created from 3500 images is called augmentation data. Then, 80% of the augmentation data is separated as the train (2800 images) and 20% as the test (700 images) data. Then, new synthetic data are derived from this train and test data separately with the ESRGAN method. In this way, the reason why train and test data are derived separately, and the train and test data are prevented from mixing with each other. While 2800 train and 700 test data were generated in the new data derivation, this dataset was called super-resolution (SR) data. The reason for using the ESRGAN method in the derivation of new data is to take advantage of its ability to produce more natural and detailed tissue information than other super-resolution methods. In the second step, transfer learning was performed with MobileNetV2 and VGG19 CNN models to learn classes from augmentation and SR data. The reason for choosing these CNN models is because of their successful performance on small datasets. Also, the minimum requirement top block structure used with transfer learning CNN models is top block-2 in Fig. 4.

However, in the experimental tests made with this structure, it was determined that learning was insufficient. For this reason, the top block-1 structure, which can be used with these CNN models, has been developed to make learning more effective with the datasets used. In this step, top block-1 structured CNN models are trained with augmentation and SR data. In the third step, 160 features are extracted from the dense (160) layer in the top block-1 of the trained CNN models. Features extracted from each model are combined (fusion), resulting in 640 features. In the fourth step, important features are selected by using the genetic algorithm and support vector classifier together. In the fifth step, classification is done using the SVM method. The schematic view of the proposed approach is given in Fig. 5.

The hyper-parameters used in these CNN models are given in Table 1. In addition, the categorical cross-entropy loss function is used in these CNN models. As the optimization method, Adam, SGD, and RMSprop were tried and the SGD algorithm, which gave the best results, was selected. Momentum and learning rate values were tested iteratively and the values that gave the best results were selected.

4 Experimental results and discussions

In this study, performance metrics such as accuracy, precision, recall, F1-score, specificity, sensitivity, and Matthews correlation coefficient obtained from the complexity matrix were used to measure the performance of the proposed approach. For the calculation of these metrics, 4 basic parameters are needed, namely true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [43, 56, 57]. These metrics are formulated in Eqs. (2), (3), (4), (5), (6), (7), and (8).

$$\text{Acc} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FN} + \text{TN} + \text{FP}) \quad (2)$$

$$\text{Prec} = \text{TP} / (\text{TP} + \text{FP}) \quad (3)$$

$$\text{Rec} = \text{TP} / (\text{TP} + \text{FN}) \quad (4)$$

$$\text{F1-Score} = 2\text{TP} / (2\text{TP} + \text{FP} + \text{FN}) \quad (5)$$

$$\text{Sens} = \text{TP} / (\text{TP} + \text{FN}) \quad (6)$$

$$\text{Spec} = \text{TN} / (\text{TN} + \text{FP}) \quad (7)$$

$$\text{MCC} = \frac{(\text{TP} \cdot \text{TN}) - (\text{FN} \cdot \text{FP})}{\sqrt{(\text{TP} + \text{FN}) \cdot (\text{TN} + \text{FP}) \cdot (\text{TP} + \text{FP}) \cdot (\text{TN} + \text{FN})}} \quad (8)$$

Experimental studies were compiled using software technologies such as Python 3.7.13, TensorFlow 2.8.0, Keras 2.8.0, and hardware technologies such as

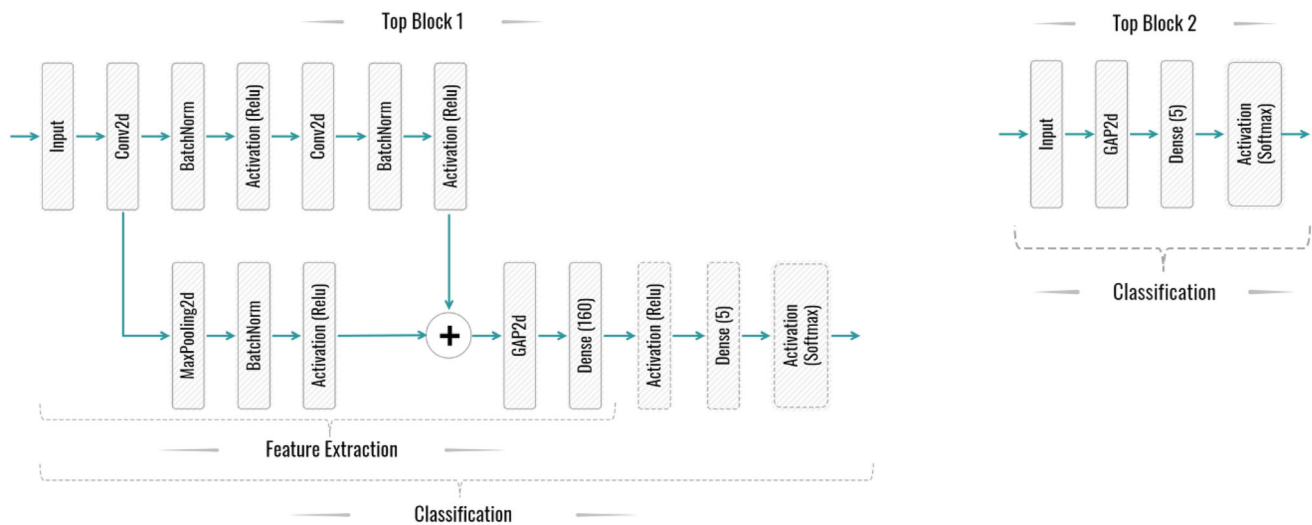


Fig. 4 Top block structures

Fig. 5 Block diagram representation of the proposed approach

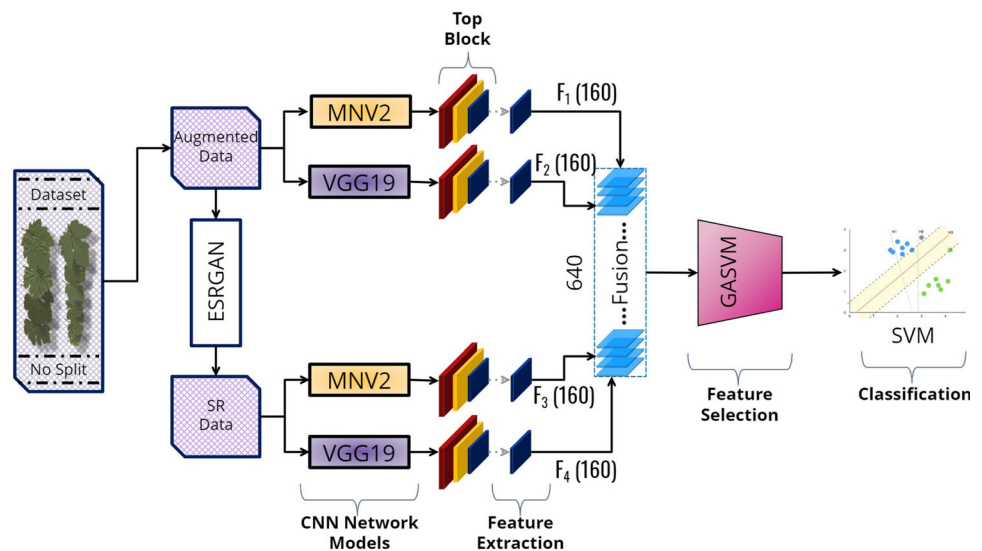


Table 1 Hyper-parameter values used in CNN models

Dataset	CNN models	Image size	Optimization methods	Momentum	Epoch	Mini batch	Learning rate
Augmentation data	MobileNetV2	224 × 224	Stochastic gradient descent	0.975	200	32	1e-3
	VGG19			0.9			1e-2
SR data	MobileNetV2			0.975			1e-2
	VGG19			0.9			1e-2

Intel(R) Xeon(R) CPU @ 2.30 GHz, 27.6 GB RAM, Tesla T4—16 GB.

In this study, experimental research is basically divided into two stages according to feature extraction or not. In this way, the reason why the experimental study is in two stages is to provide an understanding of how feature

extraction or non-extraction will contribute to the proposed approach. In the first stage, experimental results were obtained by using MobileNetV2 (MNV2) and VGG19 CNN models without extracting attributes with augmentation and super-resolution (SR) data and by classifying grapevine leaf according to the use of top block-1 or top

block-2 structures in these CNN models. In this way, it is aimed that each component used in the proposed approach, such as augmentation and SR data, MobileNetV2 and VGG19 CNN models, and top block-1 and top block-2 structures, can be evaluated separately. In the second stage, MobileNetV2 (MNv2) and VGG19 CNN models trained with augmentation and super-resolution (SR) data were used as feature extractors. The top block-1 structure provides the output of these CNN models used in the feature extraction task. The top block-1 structure aims to keep the transfer and data learning levels of these CNN models at an optimum level. In addition, feature extraction was performed with 160 features from the dense layer of the top block-1 structure. Different feature numbers such as 320, 160, and 80 were tried iteratively and the most successful results were obtained with 160 features. At this stage, the extracted features were firstly combined and 640 features were obtained. Then, ineffective and unimportant features were discarded from 640 features, and the most important features were selected by the support vector classifier (SVC) method. At this point, after the feature selection was made with the SVC method, grapevine leaf classification was made with the SVM method and the classification measurements were recorded. Then, the GASVM algorithm was developed and used to further increase the success achieved by the SVC method. In the feature selection made with the GASVM method, 314 features were selected. Finally, to measure the final classification success of the proposed approach, 314 features were classified and the results were recorded.

On the other hand, by applying data augmentation methods to the original dataset, a total of 3500 images, 500 images from each class, were derived and this derived data was called augmentation data. Then, 80% of the augmentation dataset is reserved for train and 20% for test data to be used in the training and testing of CNN methods.

In addition, this train and test dataset was passed through the ESRGAN method separately to obtain a new synthetic dataset, which is called super-resolution data (SR data). In particular, since the train and test datasets are given separately to ESRGAN, the newly derived dataset is 80% train and 20% test data. The reason for using ESRGAN in this study is to benefit from its ability to produce more natural and detailed texture information than other super-resolution GANs. The accuracy and loss graphs obtained when training the MobileNetV2 (MNv2) and VGG19 CNN models with top block-1 with augmentation and SR data are given in Fig. 6. Figure 6a and b indicates the graph of training with augmentation data, while Fig. 6c and d indicates the graph of training with SR data.

As shown in Fig. 6, the accuracy of the MobileNetV2 and VGG19 CNN models is 100% in the training made with augmentation and SR datasets, while their losses are

close to zero. However, as shown in Fig. 6a, the accuracy graph of MobileNetV2 reaches up to 97.5% with the augmentation dataset, while the accuracy graph of VGG19 reaches up to 96.5%. As shown in Fig. 6b, the loss graph of MobileNetV2 reaches up to 0.075, while the loss graph of VGG19 reaches up to 0.099. In addition, when these CNN models are trained with the SR dataset, the accuracy graph of MobileNetV2 reaches 94%, while the accuracy graph of VGG19 reaches 96%, as shown in Fig. 6c. As shown in Fig. 6d, the loss graph of MobileNetV2 reaches up to 0.184, while the loss graph of VGG19 reaches up to 0.196.

The top block structures were used in training and classification tests with MobileNetV2 and VGG19 CNN models of augmentation and SR data. The standard block structure used for training and testing CNN models in studies with transfer learning is top block-2. Since the number of learnable parameters in CNN models with this block structure was insufficient, it was necessary to develop the top block-1 structure. The comparison of these two top block structures according to the dataset and CNN models is given in Table 2. Macro-average measurement results are given in precision, recall, and F1-score metrics in this table. As seen in this table, the highest accuracy rate of 97.42% was achieved with the triple combination of Top block-1, augmentation data, and MobileNetV2. When the top block-1 and top block-2 structures are compared according to the same dataset and CNN model, the accuracy has increased by about 10%, especially with the binary combination of SR data and VGG19. When the binary combinations of the other dataset and CNN models are examined, it is observed that there is a remarkable increase in the accuracy metric of the top block-1 structure compared to the top block-2 structure.

In each CNN model, F1, F2, F3, and F4 features obtained from the dense layer of the top block-1 structure were combined to obtain 640 features. These fused features were firstly subjected to the feature selection process with the SVC algorithm and then classified by the SVM method, and the measurement results were obtained. Then, the final results were obtained by using the GASVM algorithm developed from GA and SVC algorithms to select the combined features and the SVM method for classification. The contribution of GA to the proposed approach can be seen in the tests performed in this way. In Table 3, the classification results of the combined features are given according to the feature selection method. Accordingly, the contribution made to the accuracy metric by using GA in the proposed approach is approximately 0.7%.

One of the tables that are often used to evaluate the performance of a classification model on a set of test data with known actual values is the confusion matrix. The confusion matrix table in Fig. 7a is given to evaluate the classification performance of the proposed approach.

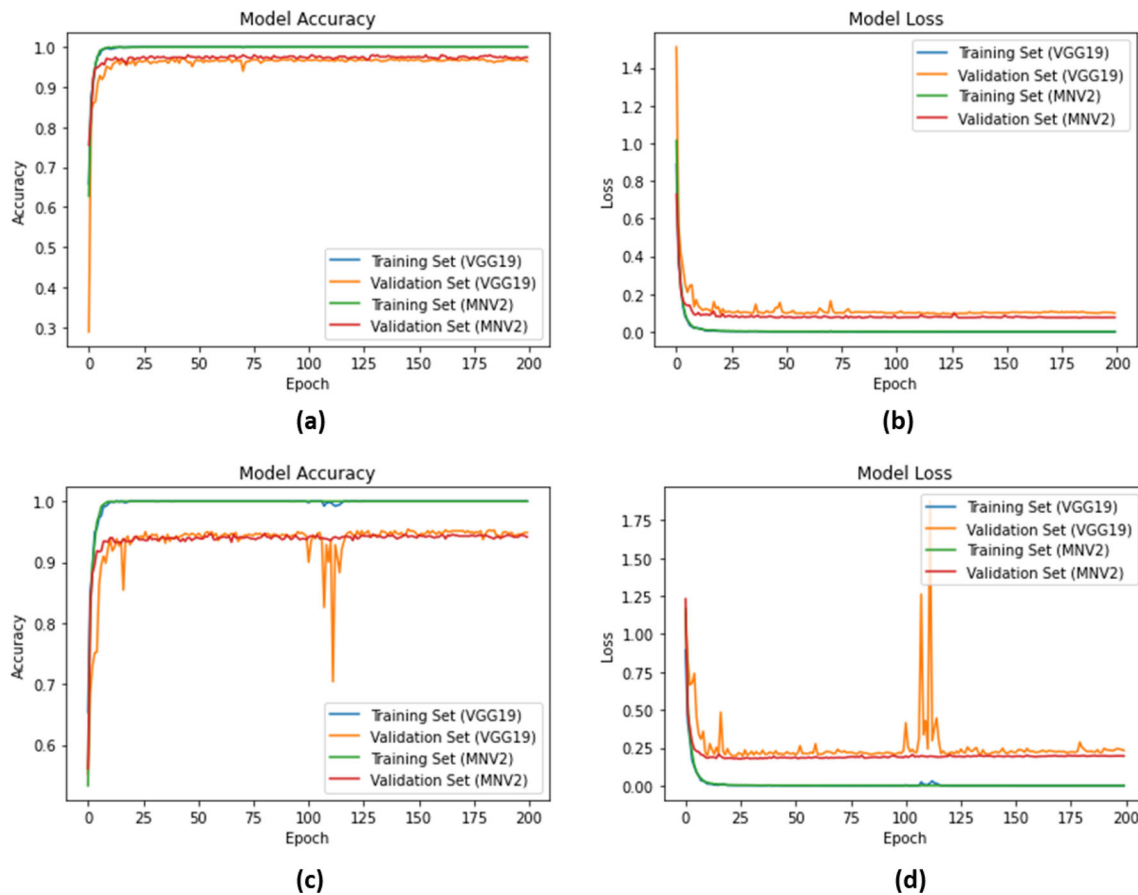


Fig. 6 Accuracy and loss graphs of MobileNetV2 and VGG19 models; **a, b** augmentation data and **c, d** SR data

Table 2 Comparison of top block structures according to the dataset and CNN models

Top block types	Datasets	Methods	Accuracy	Precision	Recall	F1-score
Top block-1	SR data	MNV2	0.9414	0.9408	0.9401	0.9404
		VGG19	0.9614	0.9612	0.9604	0.9606
	Augmentation data	MNV2	0.9743	0.9737	0.9735	0.9736
		VGG19	0.9643	0.9633	0.9637	0.9635
Top block-2	SR data	MNV2	0.9157	0.9144	0.9139	0.9139
		VGG19	0.8471	0.8644	0.8429	0.8456
	Augmentation data	MNV2	0.9671	0.9668	0.9660	0.9664
		VGG19	0.8700	0.8689	0.8673	0.8663

Table 3 Classification results of the fused features according to the feature selection method

Methods	Accuracy	Precision	Recall	F1-score
Fused features + SVC + SVM	0.9886	0.9883	0.9884	0.9883
Fused features + GASVM + SVM	0.9957	0.9956	0.9957	0.9956

Accordingly, two and one images were predicted incorrectly for the Dimnit and Nazli classes, respectively. This means that 3 images out of 700 images are incorrect and approximately 0.43% of incorrect predictions are made. Another performance evaluation method is the receiver operating characteristics (ROC) curve in Fig. 7b.

According to this curve, the distinction can be made very successfully in all classes. In addition, the class-based classification performance measurement results of the proposed approach are presented in Table 4. Accordingly, a very high classification success was achieved with an accuracy rate of 99.57%.

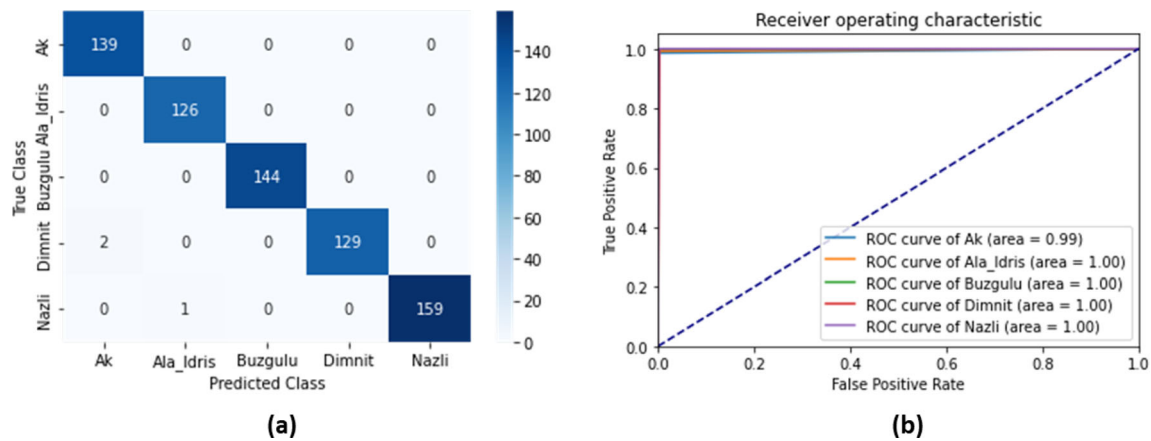


Fig. 7 Confusion matrix **a** and ROC curve **b** of the proposed approach

Table 4 Class-based classification results of the proposed approach

Classes	Accuracy	Precision	Recall	F1-score	Support
Ak	1.0000	0.9858	1.0000	0.9929	139
Ala_Idris	1.0000	0.9921	1.0000	0.9960	126
Buzgulu	1.0000	1.0000	1.0000	1.0000	144
Dimnit	0.9847	1.0000	0.9847	0.9923	131
Nazli	0.9937	1.0000	0.9938	0.9969	160
Average accuracy	0.9957				

As shown in Table 5, only Koklu et al. [43] have a study with the same dataset. The comparison of the current method using the same dataset, which includes a total of 5 classes, including Ak, Ala İdris, Büzgüli, Dimnit, and Nazlı, and 500 images, and the proposed approach is presented in this table. As shown in Table 5, while Koklu et al. obtained a 97.6% accuracy rate with 250 features, a 99.57% success rate was obtained with 314 features in the proposed study. This success rate is approximately 2% higher than the success of Koklu et al. In addition, the proposed approach was more successful in performance metrics such as specificity, precision, sensitivity, F1-score, and Matthews correlation coefficient compared to Koklu et al.'s method.

The results obtained when the test images of the grapevine leaf dataset were classified with the proposed approach are given in Fig. 8 visually and as a class. Accordingly, the class name extending along the y-axis to each image represents the actual (true) class, while the class name extending along the x-axis represents the classes predicted by the proposed approach. As can be seen, the leaf image belonging to the 2 Dimnite class was incorrectly predicted as Ak, while the leaf image belonging to the Nazli class was incorrectly predicted as Ala İdris. These incorrect predictions are considered to have occurred because the proposed approach failed to effectively extract the distinctive features of these three leaf images.

As a result, in this study, a new hybrid approach is proposed to increase the classification accuracy of grapevine leaves. In this approach, feature extraction is done with convolutional neural networks MobileNetV2 and VGG19 to provide automatic recognition. When grapevine leaves were classified using feature extraction methods and the top block-2 structure, the highest accuracy score obtained from the combination of augmentation data and MobileNetV2 was 96.71%. However, since this accuracy score was also insufficient, a new top block structure was designed. This top block structure was named top block-1, and the accuracy score of the combination of augmentation data and MobileNetv2 with this block structure was

Table 5 Comparison of the proposed approach with other methods in the literature

Study	Year	Methods	No. of Features	Acc. (%)	Spec. (%)	Prec. (%)	Sens. (%)	F1-Score (%)	MCC (%)
Koklu et al. [43]	2022	CNN + Chi-Square + SVM	250	97.60	99.40	97.62	97.60	97.60	97.01
Proposed approach	–	CNN + Fused Features + GASVM + SVM	314	99.57	99.89	99.56	99.57	99.56	99.46

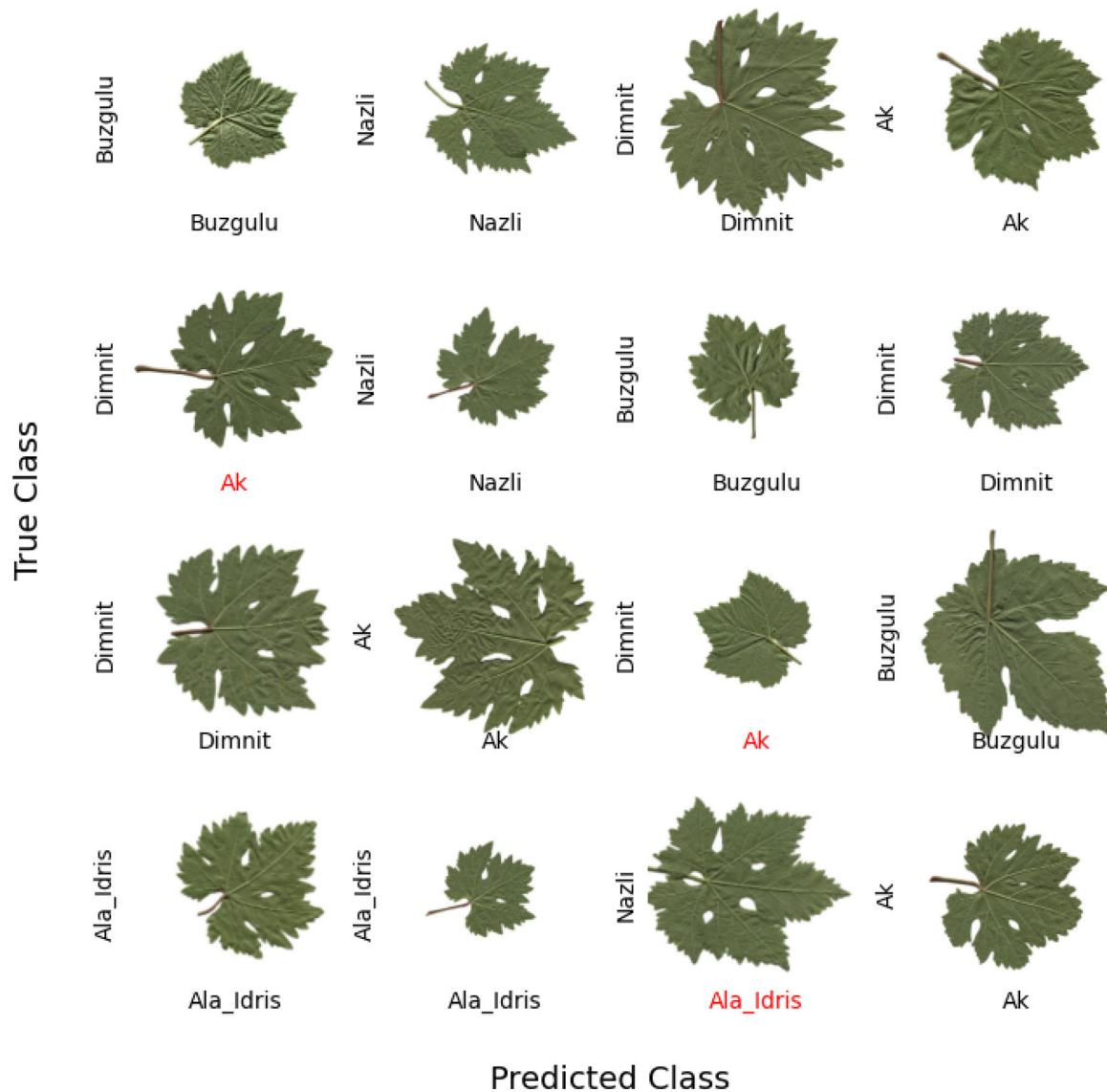


Fig. 8 A few predicted classes with the proposed approach and their test images

increased by 0.72% and reached 97.43%. Since this accuracy score is also insufficient for perfect leaf recognition, insignificant features should be eliminated. Therefore, with the GASVM method, which we adapted to this problem, we selected 314 of the most important features. Then, the selected features were classified using the SVM method and the accuracy score was increased by 2.14%, achieving a perfect accuracy of 99.57%. With this proposed approach, 697 images out of 700 test images were classified correctly, while only 3 images were misclassified. It is thought that the reason why the proposed approach misclassifies the 3 images is due to the insufficient number of leaf species in the original dataset. Even if data augmentation techniques are applied, feature extraction cannot be performed perfectly if the leaf types in the original dataset do not have a wide variety of sizes, textures, and maturity.

5 Conclusion

In this study, we propose a new hybrid approach for the automatic recognition of grapevine leaf species. In the proposed approach, data augmentation techniques are applied to provide more effective learning from the dataset. In addition, it was reconstructed with the ESRGAN model to obtain more detailed textures and high-resolution synthetic images from the newly created dataset. In our hypothesis, we designed a deep feature extractor with new detailed information by adding the top block-1 structure to the pre-trained MobileNetV2 and VGG19 CNN architectures. According to the results, the proposed block structure contributed to the increase in performance. The features taken from the datasets trained with CNN models are then combined in the following stage. The GASVM feature

selection method has been developed and applied for the selection of the most efficient deep features. To classify the features combined in the previous stage, SVM was employed. 99.57% accuracy, 99.89% specificity, 99.56% precision, 99.57% sensitivity, 99.56% F1-score, and 99.46% MCC were achieved using the designed hybrid approach model. In addition, the experimental results were compared with other studies using the relevant dataset. It has been shown that the proposed approach is approximately 2% more successful in terms of accuracy, precision, sensitivity, F1-score, and MCC metrics compared to Koklu et al., which is the only study conducted with the grapevine leaf dataset. The observed findings in the results showed both the significance of the deep features obtained by the model and its success in grapevine leaf classification.

In the future, it is aimed to further improve the proposed approach by training and testing large datasets consisting of different plant species and producing cost-effective solutions that can work on end devices.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). There is no funding source for this article.

Data availability The link to all data analyzed during this study is included in this published article.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical approval This article does not contain any data, or other information from studies or experimentation, with the involvement of human or animal subjects.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ganguly S, Bhowal P, Oliva D, Sarkar R (2022) BLeafNet: a bonferroni mean operator based fusion of CNN models for plant identification using leaf image classification. *Ecol Inform*. <https://doi.org/10.1016/j.ecoinf.2022.101585>
- Cangi R, Yağci A (2017) Bağdan sofraya yemeklik asma yaprak üretimi. *Nevşehir Bilim ve Teknol Derg*. <https://doi.org/10.17100/nevbiltek.288316>
- Lanjewar MG, Panchbhairi KG (2022) Convolutional neural network based tea leaf disease prediction system on smart phone using paas cloud. *Neural Comput Appl* 35(3):2755–2771. <https://doi.org/10.1007/s00521-022-07743-y>
- Bouguettaya A, Zarzour H, Kechida A, Taberkit AM (2022) Deep learning techniques to classify agricultural crops through UAV imagery: a review. *Neural Comput Appl* 34(12):9511–9536. <https://doi.org/10.1007/s00521-022-07104-9>
- Buyukarikan B, Ulker E (2022) Classification of physiological disorders in apples fruit using a hybrid model based on convolutional neural network and machine learning methods. *Neural Comput Appl* 34(19):16973–16988. <https://doi.org/10.1007/s00521-022-07350-x>
- Chakraborty SK, Chandel NS, Jat D, Tiwari MK, Rajwade YA, Subeesh A (2022) Deep learning approaches and interventions for futuristic engineering in agriculture. *Neural Comput Appl* 34(23):20539–20573. <https://doi.org/10.1007/s00521-022-07744-x>
- Nandhini S, Ashokkumar K (2022) An automatic plant leaf disease identification using DenseNet-121 architecture with a mutation-based henry gas solubility optimization algorithm. *Neural Comput Appl* 34(7):5513–5534. <https://doi.org/10.1007/s00521-021-06714-z>
- Wu SG, Bao FS, Xu EY, Wang YX, Chang YF, and Xiang QL (2007) A leaf recognition algorithm for plant classification using probabilistic neural network. In: ISSPIT 2007 - 2007 IEEE Int. Symp. Signal Process. Inf. Technol., pp. 11–16. <https://doi.org/10.1109/ISSPIT.2007.4458016>
- Söderkvist OJO (2001) Computer Vision Classification of Leaves from Swedish Trees. Dissertation, Linköping University
- Silva PFB, Marcal ARS and Da Silva RMA (2013) Evaluation of features for leaf discrimination. In: Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 7950 LNCS, pp. 197–204. https://doi.org/10.1007/978-3-642-39094-4_23
- Kadir A, Nugroho LE, Susanto A, Santosa PI (2011) Neural network application on foliage plant identification. *Int J Comput Appl* 29(9):15–22. <https://doi.org/10.5120/3592-4981>
- XiaQ, Zhu HD, Gan Y and Shang L (2014) Plant leaf recognition using histograms of oriented gradients, vol. 25, no. 3, pp 369–374
- Xiao XY, Hu R, Zhang SW and Wang XF (2010) HOG-Based Approach for Leaf Classification, In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 6216 LNAI, pp 149–155
- Wang Z et al (2014) Plant recognition based on intersecting cortical model. *Proc Int Jt Conf Neural Networks* 2013:975–980. <https://doi.org/10.1109/IJCNN.2014.6889656>
- Hasim A, Herdiyeni Y and Douady S (2016) Leaf Shape Recognition using Centroid Contour Distance. In: IOP Conf. Ser. Earth Environ. Sci., vol. 31, no. 1. <https://doi.org/10.1088/1755-1315/31/1/012002>
- Mahdikhanelou K and Ebrahimnezhad H (2014) Plant leaf classification using centroid distance and axis of least inertia method, In: 22nd Iran. Conf. Electr. Eng. ICEE 2014, no. Icee, pp. 1690–1694. <https://doi.org/10.1109/IranianCEE.2014.6999810>
- Kadir A (2015) Leaf identification using fourier descriptors and other shape features. *Gate to Comput Vis Pattern Recognit* 1(1):3–7. <https://doi.org/10.15579/gtcvpr.0101.003007>
- Lee KB, Hong KS (2013) An implementation of leaf recognition system using leaf vein and shape. *Int J Bio-Science Bio-Technology* 5(2):57–65

19. Shabanzade M, Zahedi M, Amin Aghvami S (2011) Combination of local descriptors and global features for leaf recognition. *Signal Image Process An Int J* 2(3):23–31. <https://doi.org/10.5121/sipij.2011.2303>
20. Chaki J, Parekh R and Bhattacharya S (2015) Recognition of whole and deformed plant leaves using statistical shape features and neuro-fuzzy classifier, In: 2015 IEEE 2nd Int. Conf. Recent Trends Inf. Syst. ReTIS 2015 - Proc., pp. 189–194. <https://doi.org/10.1109/ReTIS.2015.7232876>
21. Rojanamontien M, Sihanatkathakul P, Piemkaroonwong N, Kamales S and Watchareeruetai U (2016) Leaf identification using apical and basal features, In: 2016 8th Int. Conf. Knowl. Smart Technol. KST 2016, pp. 234–238. <https://doi.org/10.1109/KST.2016.7440521>
22. Kulkarni AH, Rai HM, Jahagirdar KA, Upparamani PS (2013) A leaf recognition technique for plant classification using RBPNN and Zernike moments. *Int J Adv Res Comput Commun Eng* 2(1):984–988
23. Kadir A, Nugroho LE, Susanto A, Insap Santosa P (2012) Experiments of zernike moments for leaf identification. *J Theor Appl Inf Technol* 41(1):82–93
24. Turkoglu M, Hanbay D (2019) Leaf-based plant species recognition based on improved local binary pattern and extreme learning machine. *Phys A Stat Mech its Appl* 527:121297. <https://doi.org/10.1016/j.physa.2019.121297>
25. Hu J, Chen Z, Yang M, Zhang R, Cui Y (2018) A multiscale fusion convolutional neural network for plant leaf recognition. *IEEE Signal Process Lett* 25(6):853–857. <https://doi.org/10.1109/LSP.2018.2809688>
26. Szegedy C et al. (2015) Going deeper with convolutions, In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 07–12-June, pp. 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>
27. Simonyan K and Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–14
28. He K, Zhang X, Ren S and Sun J (2016) Deep residual learning for image recognition, In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>
29. Huang G, Liu Z, Van Der Maaten L and Weinberger KQ (2017) Densely connected convolutional networks. In: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp. 2261–2269. <https://doi.org/10.1109/CVPR.2017.243>
30. Sandler M, Howard A, Zhu M, Zhmoginov A and Chen LC (2018) MobileNetV2: Inverted residuals and linear bottlenecks. *arXiv*, pp. 4510–4520
31. Szegedy C, Ioffe S, Vanhoucke V and Alemi AA (2017) Inception-v4, inception-ResNet and the impact of residual connections on learning. In: 31st AAAI Conf. Artif. Intell. AAAI 2017, pp. 4278–4284
32. Ma J, Du K, Zheng F, Zhang L, Gong Z, Sun Z (2018) A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Comput Electron Agric* 154(September):18–24. <https://doi.org/10.1016/j.compag.2018.08.048>
33. Dhivyaa CR, Kandasamy N, Rajendran S (2022) Integration of dilated convolution with residual dense block network and multi-level feature detection network for cassava plant leaf disease identification. *Concurr Comput Pract Exp* 34(11):1–19. <https://doi.org/10.1002/cpe.6879>
34. Bhujel A, Kim NE, Arulmozhi E, Basak JK, Kim HT (2022) A lightweight attention-based convolutional neural networks for tomato leaf disease classification. *Agric* 12(2):1–18. <https://doi.org/10.3390/agriculture12020228>
35. Prabu M, Chelliah BJ (2022) Mango leaf disease identification and classification using a CNN architecture optimized by cross-over-based levy flight distribution algorithm. *Neural Comput Appl* 34(9):7311–7324. <https://doi.org/10.1007/s00521-021-06726-9>
36. Singh RK, Tiwari A, Gupta RK (2022) Deep transfer modeling for classification of maize plant leaf disease. *Multimed Tools Appl* 81(5):6051–6067. <https://doi.org/10.1007/s11042-021-11763-6>
37. Li L, Zhang S, Wang B (2022) Apple leaf disease identification with a small and imbalanced dataset based on lightweight convolutional networks. *Sensors*. <https://doi.org/10.3390/s22010173>
38. Saber Anari M (2022) A hybrid model for leaf diseases classification based on the modified deep transfer learning and ensemble approach for agricultural aiot-based monitoring. *Comput Intell Neurosci*. <https://doi.org/10.1155/2022/6504616>
39. Lee SH, Chan CS, Wilkin P, and Remagnino P (2015) Deepplant: Plant identification with convolutional neural networks, In: 2015 IEEE International Conference on Image Processing (ICIP), pp. 452–456. <https://doi.org/10.1109/ICIP.2015.7350839>
40. Esgario JGM, Krohling RA, Ventura JA (2019) Deep learning for classification and severity estimation of coffee leaf biotic stress. *Comput Electron Agric* 169(July):2020. <https://doi.org/10.1016/j.compag.2019.105162>
41. Pawara P, Okafor E, Groefsema M, He S, Schomaker LRB, Wiering MA (2020) One-vs-One classification for deep neural networks. *Pattern Recognit*. <https://doi.org/10.1016/j.patcog.2020.107528>
42. Carneiro G, Padua L, Sousa JJ, Peres E, Morais R and Cunha A (2021) Grapevine Variety Identification Through Grapevine Leaf Images Acquired in Natural Environment, no. 1, pp 7055–7058. <https://doi.org/10.1109/igarss47720.2021.9555141>
43. Koklu M, Unlarsen MF, Ozkan IA, Aslan MF and Sabanci K (2021) A CNN-SVM study based on selected deep features for grapevine leaves classification. In: Meas. J. Int. Meas. Confed., vol. 188, no. November 2021, p 110425. <https://doi.org/10.1016/j.measurement.2021.110425>
44. “Albumentations” (2022) <https://albumentations.ai/>. Accessed 23 September 2007
45. Wang X et al. (2019) ESRGAN: Enhanced super-resolution generative adversarial networks. In: Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11133 LNCS, pp. 63–79. https://doi.org/10.1007/978-3-030-11021-5_5
46. Ledig C et al. (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp 105–114. <https://doi.org/10.1109/CVPR.2017.19>
47. Lim B, Son S, Kim H, Nah S and Lee KM (2017) Enhanced Deep Residual Networks for Single Image Super-Resolution. In: IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work, vol. 2017-July, pp 1132–1140. <https://doi.org/10.1109/CVPRW.2017.151>
48. Nah S, Kim TH and Lee KM (2017) Deep multi-scale convolutional neural network for dynamic scene deblurring. In: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp 257–265. <https://doi.org/10.1109/CVPR.2017.35>
49. Imak A, Celebi A, Siddique K, Turkoglu M, Sengur A, Salam I (2022) Dental caries detection using score-based multi-input deep convolutional neural network. *IEEE Access* 10:18320–18329. <https://doi.org/10.1109/ACCESS.2022.3150358>
50. Howard AG et al. (2017) MobileNets: Efficient convolutional neural networks for mobile vision applications, *arXiv*. <https://doi.org/10.48550/arXiv.1704.04861>

51. Carvalho T, De Rezende ERS, Alves MTP, Balieiro FKC and Sovat RB (2017) Exposing computer generated images by eye's region classification via transfer learning of VGG19 CNN. In: Proc. - 16th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2017, vol. 2017-Decem, pp 866–870. <https://doi.org/10.1109/ICMLA.2017.00-47>
52. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297. <https://doi.org/10.1007/BF00994018>
53. Whitley D (1994) A genetic algorithm tutorial. Stat Comput 4(2):65–85. <https://doi.org/10.1007/BF00175354>
54. Sun Z, Bebis G, Miller R (2004) Object detection using feature subset selection. Pattern Recognit 37(11):2165–2176. <https://doi.org/10.1016/j.patcog.2004.03.013>
55. Sukawattanavijit C, Chen J, Zhang H (2017) GA-SVM algorithm for improving land-cover classification using SAR and optical remote sensing data. IEEE Geosci Remote Sens Lett 14(3):284–288. <https://doi.org/10.1109/LGRS.2016.2628406>
56. Doğan G, Ergen B (2022) A new mobile convolutional neural network-based approach for pixel-wise road surface crack detection. Measurement. <https://doi.org/10.1016/j.measurement.2022.111119>
57. Doğan G, Ergen B (2021) İMobileNet CNN yaklaşımları ve özellik seçme yöntemleri kullanarak araç türlerini sınıflandırma. Süleyman Demirel Üniversitesi Fen Bilim Enstitüsü Derg 25(3):618–628. <https://doi.org/10.19113/sdufenbed.889715>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.