Neural Computing and Applications (2023) 35:12073–12087
https://doi.org/10.1007/s00521-023-08340-3

ORIGINAL ARTICLE

Classification of apple images using support vector machines and deep residual networks

Sevim Adige[1] · Rifat Kurban[2,5] · Ali Durmuş[3] · Ercan Karaköse[4]

Received: 31 August 2022 / Accepted: 25 January 2023 / Published online: 21 February 2023

Abstract

One of the most important problems for farmers who produce large amounts of apples is the classification of the apples according to their types in a short time without handling them. Support vector machines (SVM) and deep residual networks (ResNet-50) are machine learning methods that are able to solve general classification situations. In this study, the classification of apple varieties according to their genus is made using machine learning algorithms. A database is created by capturing 120 images from six different apple species. Bag of visual words (BoVW) treat image features as words representing a sparse vector of occurrences over the vocabulary. BoVW features are classified using SVM. On the other hand, ResNet-50 is a convolutional neural network that is 50 layers deep with embedded feature extraction layers. The pre-trained ResNet-50 architecture is retrained for apple classification using transfer learning. In the experiments, our dataset is divided into three cases: Case 1: 40% train, 60% test; Case 2: 60% train, 40% test; and Case 3: 80% train, 20% test. As a result, the linear, Gaussian, and polynomial kernel functions used in the BoVW + SVM algorithm achieved 88%, 92%, and 96% accuracy in Case 3, respectively. In the ResNet-50 classification, the root-mean-square propagation (rmsprop), adaptive moment estimation (adam), and stochastic gradient descent with momentum (sgdm) training algorithms achieved 86%, 89%, and 90% accuracy, respectively, in the set of Case 3.

Keywords Support vector machines · Deep residual networks · Apple classification

✉ Rifat Kurban
rifatkurban@kayseri.edu.tr

Sevim Adige
sevim-adige@hotmail.com

Ali Durmuş
alidurmus@kayseri.edu.tr

Ercan Karaköse
ekarakose@kayseri.edu.tr

[1] Department of Graduate Education Institute, Electrical-Electronics Engineering, Kayseri University, Kayseri, Turkey

[2] Department of Computer Technologies, Vocational School of Technical Sciences, Kayseri University, Talas, Kayseri, Turkey

[3] Department of Electricity and Energy, Vocational School of Technical Sciences, Kayseri University, Kayseri, Turkey

[4] Department of Natural Sciences, Engineering and Architecture and Design Faculty, Kayseri University, Kayseri, Turkey

[5] Department of Computer Engineering, Engineering Faculty, Abdullah Gul University, Kayseri, Turkey

1 Introduction

Agriculture is a very important field worldwide due to the survival of a country's population, its contribution to national income and employment, by providing raw materials and capital to other sectors, while it also links to exports and influences the ecological balance and biological diversity. For this reason, agriculture, with its economic, social, and environmental dimensions, is closely related to all segments of society [1]. Turkey, with its prolific and environmentally diversified agricultural areas, is among the countries that produce high-quality agricultural products. One important branch of agricultural production is fresh fruit cultivation. Turkey is the leading producer in the world of some fruit species. Apple (*Malus domestica*) is a type of fruit from the rose (Rosaceae) family. Apples are healthy and widely consumed fruit that contain antioxidants, vitamins, fiber, and many different minerals. Studies have shown that apples reduce the risk of prostate and lung cancer [1]. Apples contain phenolic

compounds, which are equivalent to high amounts of vitamin C, that are valuable antioxidants that reduce the risk of cancer and DNA damage [2]. In the 2019–2020 season, Turkey ranked 4th in apple production in the world [3]. With the increase in apple production in recent years, one of the most important concerns for growers is separating apples into different types in a short amount of time without contact. Classifying apples without touching them during the classification to their types reduces the losses during export and ensures that the apple peel is more sterile. To solve this problem, using technology at the highest level in modern agriculture and benefiting from automation is a solution for producers. The processing or analysis of digitized images obtained from devices, such as cameras, video cameras, and scanners, utilizing software in the computer environment is defined as image processing. To make a decision, computer vision tasks involve methods for gathering, processing, evaluating, and understanding digital images. Recently, one of the best machine learning algorithms that can give answers to classification problems is support vector machines (SVM), which have been used to solve a large number of classification problems and have taken their location in the theory as efficient and effective machine learning algorithms that can perform with significant generalization. The most important advantages of SVM are that it transforms the classification problem so that it is a squared optimization problem and solves it. Therefore, in the learning phase of the solution of the problem, there is a decrease in the number of operations, and a quicker solution is achieved compared to other techniques/algorithms [4]. This feature provides a significant benefit, especially when considering datasets with large volumes. As it is built on optimization, when it is compared to other techniques, it has increased success when considering classification performance, computational complexity, and usefulness [5]. Another application area of the SVM algorithm is image processing. SVM and second-order discriminant analysis (QDA) methods have been used to classify apples as healthy or rotten [6]. In another study, apples are classified according to their color such as orange, stripe, and dark red using the SVM and Otsu's thresholding method with a segmentation error of less than 2% by using the adjustable SVM [7]. Patel et al. [8] used machine learning techniques, such as KNN, SVM, and Naive Bayes, to classify fruits according to their features, such as image and color, and they found that SVM gave better results. Naik et al. [9] briefly discussed and explained the basic process flows in some machine learning approaches, such as SURF, HOG, LBP, KNN, SVM, ANN, and CNN, for fruit classification and grading. Cellular nonlinear networks (CNN) are a type of artificial neural network developed as a precursor to deep learning. The key difference between CNNs and other methods is their ability to capture the local interactions and nonlinearities within a system, which makes them well-suited for modeling and analyzing complex systems with spatiotemporal patterns and dynamics [10]. They can be used for a variety of tasks, including image classification. In the case of apple image classification, a CNN would be trained on a dataset of labeled images of apples, where each image is assigned a class label indicating the type of apple it represents. The CNN would learn to recognize patterns and features in the images that are indicative of different types of apples, and it would use this knowledge to classify new, unseen images of apples. This process involves training the CNN on the labeled dataset, adjusting the weights and biases of the network to minimize classification errors, and then evaluating the performance of the trained network on a separate test set of images. Overall, CNNs can be effective at apple image classification due to their ability to learn and recognize complex patterns in data. Koklu et al. [11] made a classification based on deep learning using grape leaf images. To do this, 500 grape leaf images from five species were taken using a system with special lighting, and the system's successful classification was found to be 97.6%. Li et al. [12] suggested different methods to classify peanuts according to their size by neural networks. They determined the accuracy rate of the aspect ratio + SVM algorithm from the analyzed images to be 96.72%.

There are several studies in the literature on image processing and classification with deep learning methods. Jiang et al. [13] used an image classification-based method with deep learning to detect infections in apple fruit and to prevent other diseases caused by environmental factors promptly. Tahir et al. [14] utilized the CNN method, which is a deep learning method that was used for the classification of apple infections. Yang Liu et al. [15] proposed a model using the deep learning method to detect different types of ores and minerals and successfully detected the types of ores and minerals. Among the deep learning algorithms, residual networks, particularly the ResNet-50 architecture, are a type of CNN structure that reduces the complexity while maintaining high performance [16, 17]. A residual neural network (ResNet) is a form of artificial neural network (ANN) that creates a network, and its performance on image classification is the best compared to other network structures.

In this study, two different machine learning methods are used to classify apple varieties according to their types. Images obtained from six different species from the Yahyali region of Turkey were divided into training and test datasets. The results obtained by the BoVW + SVM method were compared with the results obtained by the ResNet-50 deep learning method.

The paper has the following organization. Machine learning methods used for classification purposes are

summarized in Sect. 2. In Sect. 3, experimental results are evaluated. Concluding remarks are given in Sect. 4.

2 Classification using machine learning

Apples have similar textural features that are not easy to distinguish with a visual inspection. For this reason, a solution is needed to classify apples that do not depend on color, shape, and texture characteristics. The machine learning methods used in this study to solve the classification problems are illustrated in Fig. 1.

2.1 Bag of visual words and support vector machines

Bag of visual words (BoVW) is an information retrieval technique that was first developed in the field of text analysis [18]. Each occurrence of a word is identified as a feature in such applications, and a word bag is represented as an irregular document representation of the vocabulary [19]. When the BoVW model learns a term from all documents, the number of occurrences (frequency) of that word can be used to classify each document. Image classification uses the same process and concept. Each visual word taken from the image is considered a visual word, and a BoVW model is built based on its recurrence.

Local identifiers extracted by Scale Invariant Feature Transform (SIFT) [20] or Speeded Up Robust Features (SURF) are viable ways of solving classification challenges and meeting performance requirements [21]. SURF descriptors are three times faster than the SIFT descriptor and can be used for applications like object detection, picture registration, classification, or 3D reconstruction. SIFT and SURF are both insensitive to light and color while having excellent noise resistance.

BoVWs are utilized to build a histogram of visual word appearances that characterize an image. An image category classifier is trained using these histograms. The steps given in Figs. 2 and 3 show how images are prepared; a bag of visual terms are generated, and an image category classifier is trained and applied.

Image datasets must be partitioned into training and test subsets, as shown in Fig. 2. The training set is used to feed the machine learning algorithm to create a generalized model. The test set's utilization is to determine the success of the artificial model against unknown data.

When the feature descriptors are obtained from the images that represent a specific category, it is possible to create a bag of features that comes from the visual vocabulary. Feature descriptors extracted from the training set are grouped by K-means clustering to obtain visual words. Euclidean L2 distance is used as the distance metric, and the k value is set to the vocabulary size. This method partitions the descriptors into a specified number of clusters iteratively. Similar attributes are grouped to form clusters. Each cluster's center denotes a feature or visual word.

Features are extracted using a feature detector or using a pre-defined grid. The grid method is usually used for homogenous scenes, such as apple images. The SURF detector has a higher scale invariance. The visual words method does not depend on spatial information or on identifying specific objects in an image. This method is based on detection rather than localization. A diagram of creating the BoVW features of the apple dataset is shown in Fig. 3.

A commonly used method of machine learning that can be implemented for regression problems (such as predictive control, handwriting recognition, environmental image classification, and many agricultural applications) and classification is SVM. SVM efficiently solves classification problems in high-dimensional spaces using different kernel functions. Thus, the number of operations is reduced in the learning phase and its biggest advantage is that it reaches a faster solution compared to other algorithms [4]. SVM

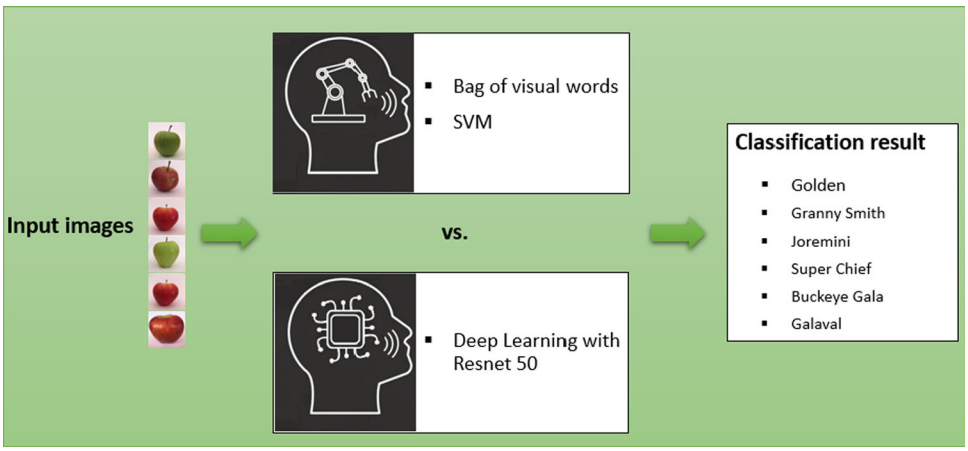

Fig. 1 The scheme of the image classification

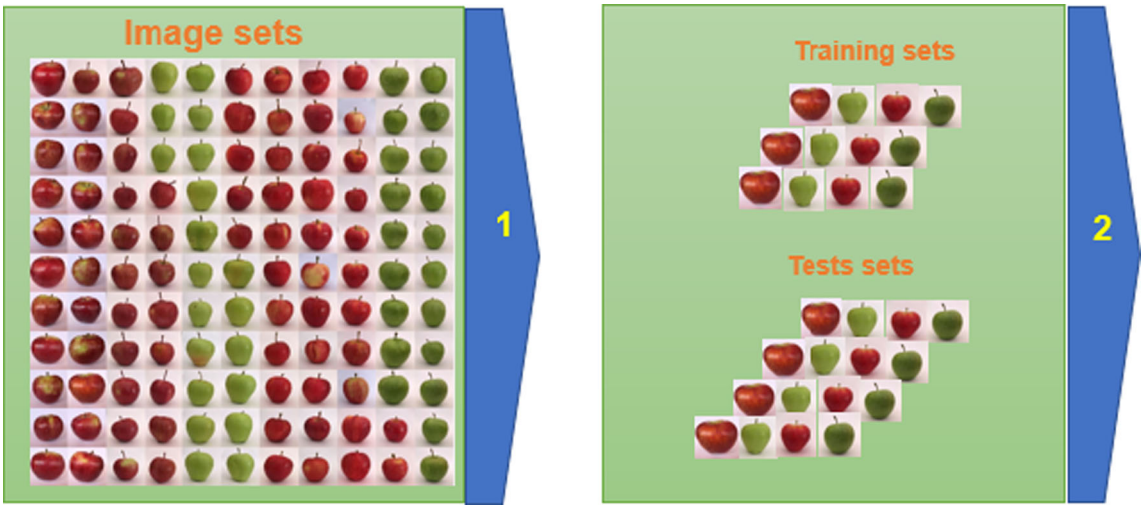

Fig. 2 Splitting image dataset into train and test subsets

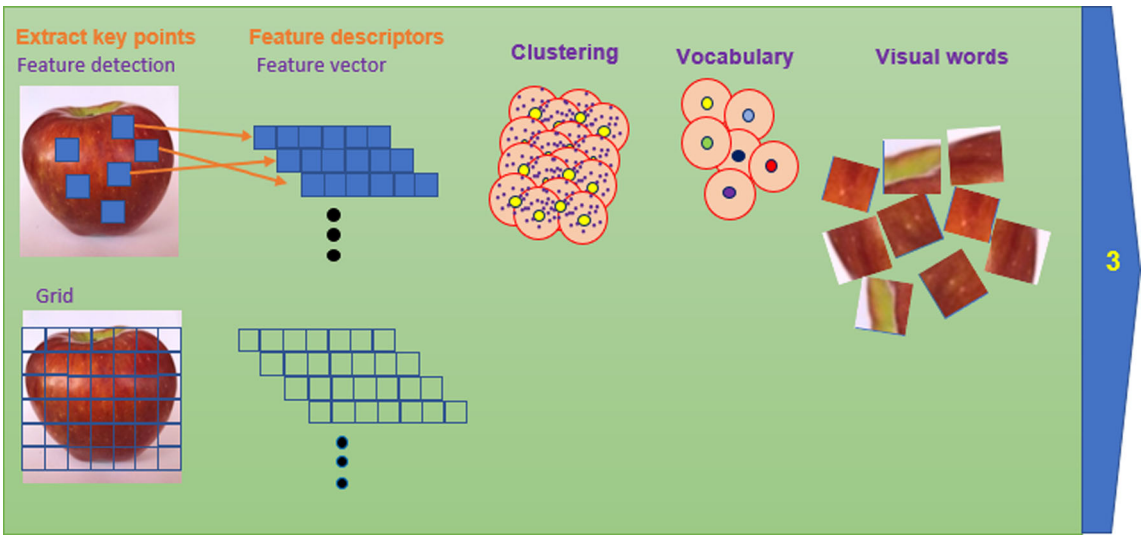

Fig. 3 Generating bag of features for apple images

balances this by minimizing the training set error and maximizing the margin to maximize its generalization ability. In addition, an important advantage of SVM is that it converges to the global solution using convex quadratic programming without getting stuck with local minimum solutions [22].

Using Lagrange multipliers, the Lagrange function may be expressed as a quadratic programming solution. For the altered feature vectors h, this may be done directly by $h(xi)$. Then it can be seen that these inner products may be calculated relatively inexpensively for certain values of h. The Lagrange function is written as:

$$L_D = \sum_{i=1}^{N} \alpha_i - 1/2 \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} \langle h(x_i), h(x_{i'}) \rangle \quad (1)$$

the solution function $f(x)$ can be written

$$f(x) = h(x)^T \beta + \beta_0 = \sum_{i=1}^{N} \alpha_i y_i h(x), h(x_i) + \beta_0 \quad (2)$$

$$\beta = \sum_{i=1}^{N} \alpha_i y_i x_i \quad (3)$$

where β is a unit vector and $h(x_i)$ the transformed feature input vectors. Given α_i, β_0 can be defined by working-out $y_i f(x_i) = 1$ in Eq. (2) for any x_i for which $0 < \alpha_i < C$ and C is the cost parameter.

A kernel function has three popular types in the SVM literature. It is defined as:

$$\text{Linear: } K(x, x') = h(x), h(x') \quad (4)$$

Polynomial: $K(x, x') = (1 + x, x')^d$ (5)

Radial basis: $K(x, x') = \exp(-\gamma x - x'^2)$ (6)

where d is the degree of polynomial, γ is a positive constant, and *exp* is the exponential function [23].

The general structure of SVM is given in Fig. 4. In the traditional approach, model complexity is controlled by keeping the number of features small. SVM provides an effective solution to the design of a learning system by controlling the model complexity regardless of size. The inputs are passed through the kernel function, weighted, and aggregated with bias.

In the proposed implementation, as shown in Fig. 5, a multiclass classifier using binary SVM is trained. Images encoded as histograms of visual words are used in the training step. A feature histogram for every image in the dataset is calculated using the approximate nearest neighbor algorithm. In a multiclass error-correcting output codes (ECOC) model using support vector machine (SVM) binary learners, each binary classifier is trained to distinguish between two classes. To classify a data sample using this model, all of the binary classifiers must be applied to the sample and record their outputs. The k-nearest neighbor (KNN) algorithm can be used as a decision rule to combine the outputs of the binary classifiers and determine the final class label for the sample. To classify a sample using the KNN decision rule, the k binary classifiers can be found that produced the strongest outputs for the sample and determine the majority class among these classifiers. The sample would then be assigned to this majority class [24]. Finally, feature vectors are formed by the corresponding feature histograms.

After the training phase, a test image set is evaluated to assess the accuracy of the machine learning model. The confusion matrix is the final output of the process that shows the accuracy of each image category. Ideally, the diagonal of the confusion matrix would be expected to have a mean accuracy of 100%.

2.2 Deep residual networks

ResNet architectures are developed by the Microsoft research team to reduce the difficulty of training deep neural networks. ResNet has several variants, consisting of 18, 34, 50, 101, and 152 weight layers [16]. ResNet stands for residual networks, a classical neural network used as the main backbone for computer vision functionality [25]. Very deep neural networks face difficulties while training due to the vanishing gradients problem (VGP). The skip connections function as key gradients, so they can flow without impediment. The element that makes the ResNets model stand out compared to other models is that its performance is very good, even though the architecture is deep. In addition, in this model, computations are performed on the computer with less performance and the ability to train artificial neural networks is quite impressive. The ResNet model is implemented using batch normalization and ReLU-enabled functionality, bypassing the links between architectures at two or three layers. He et al. [16] demonstrated that the ResNet model outperforms

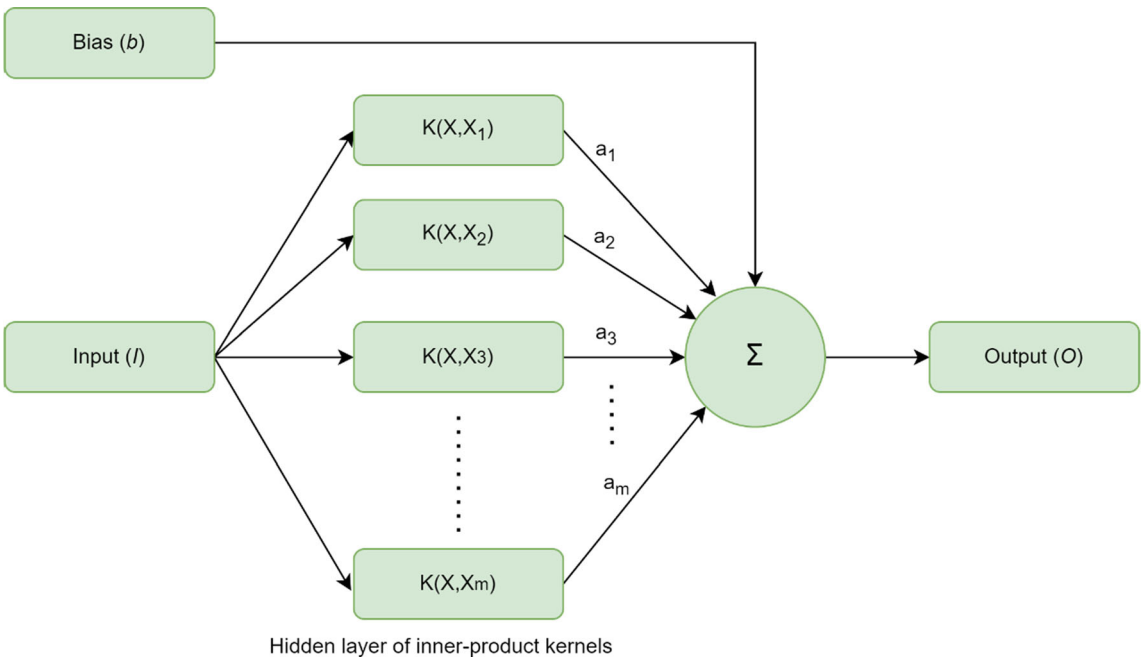

Fig. 4 General structure of SVM

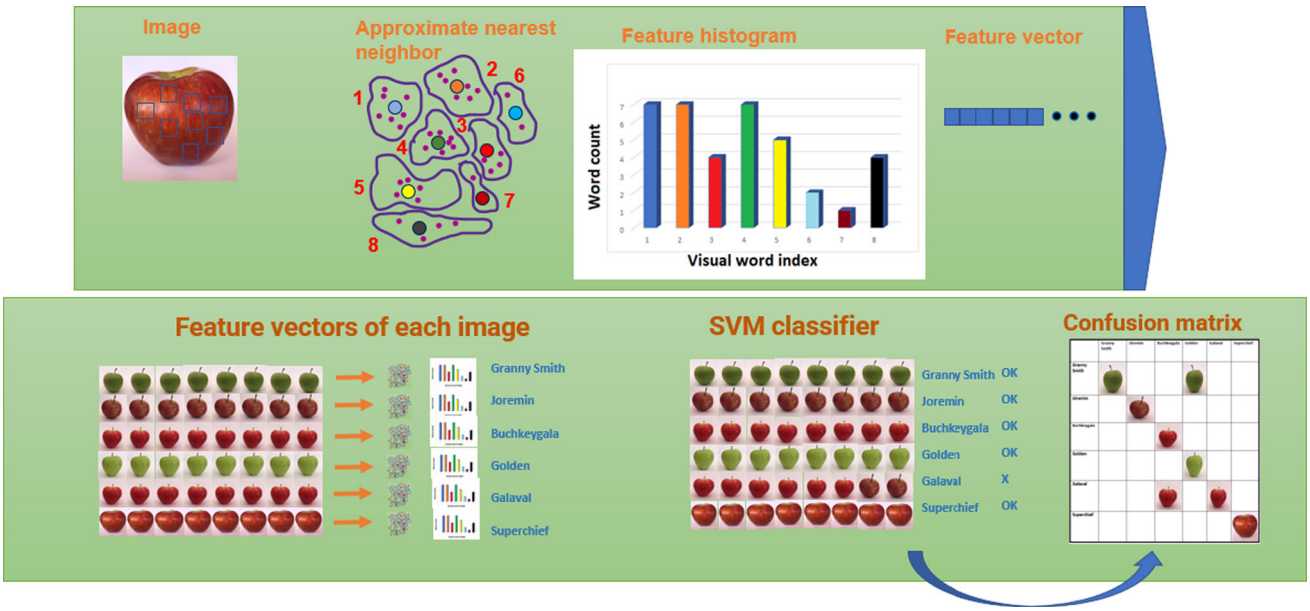

Fig. 5 Training a SVM classifier with visual words

several other models in picture classification and that image characteristics are extracted extremely effectively.

Residual learning in ResNet takes place by applying multiple layers of layers, and the residual block is written as [26]:

$$y = F(x, \{W_i\}) + x \tag{7}$$

where F, x, and y are the residual map, input–output layer expressions, respectively. The residual mapping function F $(x, \{W_i\})$ is the one that has to be learned. If the dimensions of the output data and input data are the same, the residual block in ResNet can be employed. ResNet-18 and ResNet-34 are two-layer networks, whereas ResNet-50 and ResNet-101 are three-layer networks. The ResNet-50 structure, which suggests a good balance between performance and computational complexity, was employed in this study. The training algorithms of ResNet are briefly explained:

Stochastic gradient descent with momentum (sgdm) has one learning rate that is used for all parameters. Alternative algorithms for optimization aim to increase the network training when using different rates of learning for each parameter so that they can be optimized for the loss function while including automatic adaptation. The sgdm algorithm can perform parameter updates with the momentum term, as given in the equation:

$$\theta_{\ell+1} = \theta_\ell - \alpha \nabla E(\theta_\ell) + \gamma(\theta_\ell - \theta_{\ell-1}) \tag{8}$$

where γ determines the input to the current iteration from the gradient step that came before, ℓ is the iteration number, $\alpha > 0$ is the learning rate, θ is the parameter vector, and $E(\theta)$ is the loss function. In the sgdm, the gradient of the loss function, $\nabla E(\theta)$, for the whole training set is used for the evaluation, while the complete dataset is used with the standard gradient descent algorithm.

Root-mean-square propagation (rmsprop) is an adaptive learning method that has gained popularity in recent years, as it is the extension of the stochastic gradient descent algorithm. rmsprop maintains a moving average of element squares of parameter gradients. The equations for rmsprop are given below:

$$v_\ell = \beta_2 v_{\ell-1} + (1 - \beta_2)[\nabla \mathrm{E}(\theta_\ell)]^2 \tag{9}$$

where β_2 is the moving average's rate of decay. Commonly, 0.9, 0.99, and 0.999 are used as the rates of decay. The related averaged lengths of the squared gradients were $1/(1\text{-}\beta_2)$, which is, 10, 100, and 1000 parameter updates, respectively. The rmsprop algorithm uses equation (z) to normalize updates for each parameter separately. The rmsprop algorithm uses the following equation to normalize updates for each parameter separately.

$$\theta_{\ell+1} = \theta_\ell - \frac{\alpha \nabla E(\theta_\ell)}{\sqrt{v_\ell} + \varepsilon} \tag{10}$$

where element-wise division is performed and ε is a constant with a small value that is included to avoid division by zero.

Adaptive moment estimation (adam) algorithm performs parameter updates with an additional momentum term similar to rmsprop. Both the squared values and parameter gradients, which are obtained by their element-wise

moving averages, are given by means of equation (t) and equation (k):

$$m_\ell = \beta_1 m_{\ell-1} + (1-\beta_1)\nabla E(\theta_\ell) \tag{11}$$

$$v_\ell = \beta_2 v_{\ell-1} + (1-\beta_2)[\nabla \mathrm{E}(\theta_\ell)]^2 \tag{12}$$

where β_1 and β_2 decay rates can be determined using the name-value pair arguments 'GradientDecayFactor' and 'SquaredGradientDecayFactor,' respectively. adam uses the moving averages given via equation (h) to update the network parameters [27].

$$\theta_{\ell+1} = \theta_\ell - \frac{\alpha m_l}{\sqrt{v_\ell} + \varepsilon} \tag{13}$$

ResNet-50 is a pre-trained image classification network and can be easily adapted for a certain classification purpose using transfer learning. Figure 6 illustrates the application of ResNet to the apple classification problem.

The network, trained previously on more than one million images, is loaded initially, as shown in Fig. 6 (step 1). Apple images are fed as an image data store into the ResNet-50 architecture. As shown in Fig. 6 (step 2), the learning process is accelerated by including limitations in the dataset to learn the characteristics of the apple species. Figure 6 (step 3) shows a training set created with apple photographs, and the training options are entered into the system. Optionally, it can freeze the effects of previous layers in the network by implementing the learning rates at these layers to zero. Network training can be sped up as the frozen layers' gradients are not calculated. In addition, freezing network layers can prevent these layers from overfitting the new dataset in small clusters. In ResNet, the first 10 layers make up the main body of the network. The training process is completed by removing the layers and links of the layer graph and choosing which layers to freeze. The program matrix needs $224 \times 224 \times 3$ size image data. The size parameters represent the width, height, and color channel of the input RGB images, respectively. As illustrated in Fig. 6 (steps 4 and 5), the size of the training images may be insufficient, so the size of the images can be increased. Using the fine-tuned network through training, images are classified by ResNet-50 and the classification accuracy is calculated. The test images are in a continuous loop within the system, and the loop is arranged to make the most accurate estimation.

3 Experimental results

In this study, two different machine learning methods, BoVW + SVM and ResNet-50, are used to classify apple images. While the effects of linear, Gaussian, and

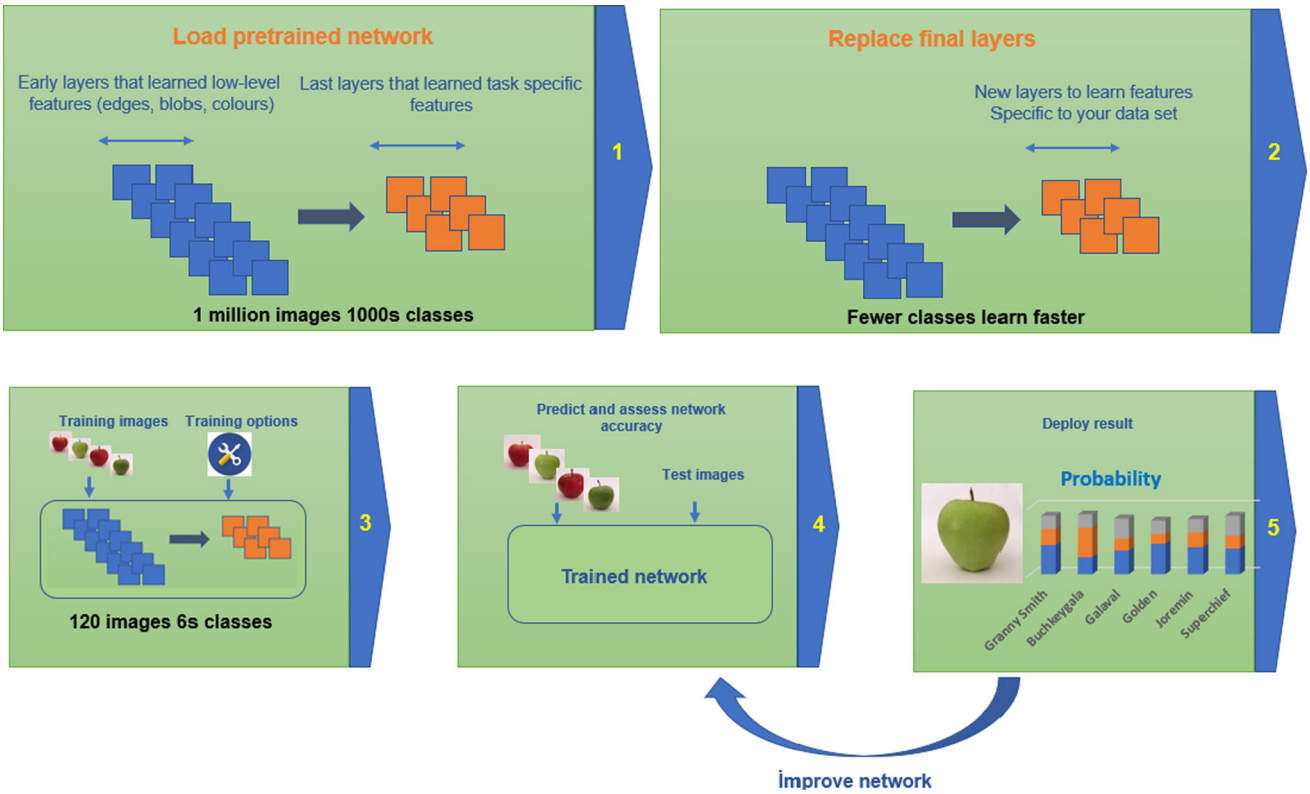

Fig. 6 Training process of ResNet-50 deep learning network to classify images

polynomial kernel functions on the result are evaluated in the BoVW + SVM method, the effects of adam, rmsprop, and sgdm training algorithms on the result are also compared in the ResNet-50 method. In addition, the results for vocabulary sizes 500, 1000, 1500, and 2000, which is an important parameter for BoVW + SVM, are compared. In the ResNet-50 method, experiments are carried out with epoch numbers 3, 6, 8, and 10. The performances of the methods are analyzed by determining the datasets into three cases: Case 1: 40% train, 60% test; Case 2: 60% train, 40% test; and Case 3: 80% train, 20% test. For a fairer comparison in the BoVW + SVM and ResNet-50 methods, experiments are repeated 30 times, and for each run, the train and test datasets are created randomly from the input dataset.

3.1 Image dataset

In this study, the classification of six different types of apples is carried out. While creating the dataset, which is the first step of the classification process, 20 apples of each type are taken from different trees at harvest time. In the dataset, a total of 120 images are obtained, including 20 Golden, 20 Granny Smith, 20 Joremini, 20 Super Chief, 20 Buckeye Gala, and 20 Galaval. The apple images used in this study are taken with a Canon EOS 70D DSLR camera. This camera has a CMOS-type sensor. The images have a resolution of 3024×3024 pixels, and the images are reduced to 512×512 pixels for BoVW + SVM and 224×224 for ResNet-50 to process the data at the optimum level. The experiments are conducted on a computer with an Intel i5 3.4 GHz processor and 16 GB RAM using MATLAB 2021b.

3.2 Classification results of BoVW + SVM

It is an essential problem to classify apple species in a hygienic manner in a short time. Within the scope of this study, BoVW is used as a feature extractor from the image dataset, and SVM is utilized to classify image features into apple species.

Case 1: Fig. 7 shows the accuracy results of the test images obtained using 40% of the dataset as the training data. Three different activation functions are used in simulations: Gaussian, linear, and polynomial. According to the results given in Fig. 7, the polynomial kernel function has a very good classification rate compared to other functions.

The best results are obtained when the vocabulary size is 2000. It is observed that as the vocabulary increased so did the accuracy rate. Table 1 shows statistical results of accuracy and CPU time (in seconds) values of Gaussian, linear, and polynomial SVM kernel functions according to different percentages in training and test dataset inputs. As can be seen from Table 1, the average CPU time for different kernels is roughly the same. The training accuracy of all cases is higher than 91%; however, for the test accuracy, the polynomial kernel performs better than the others.

Case 2: Fig. 8 shows the classification accuracy of the test dataset with 60% of the images as the training data. According to the results given in Fig. 8, it has been determined that the polynomial function has a very good learning rate compared to the other functions.

The best learning rate is obtained when using the 2000 vocabulary size. A learning rate of over 94% is achieved in this setting. It is also observed in these working conditions that the accuracy rate increased with the increase in vocabulary size in the polynomial function. This feature is also exhibited by the linear transfer function in other conditions, except for the 2000 vocabulary size. In the linear transfer function, the highest learning rate is achieved using the 2000 vocabulary size. In these settings, an accuracy rate of about 89% is achieved. The Gaussian transfer function showed good performance according to the lowest and highest vocabulary size values compared to the linear transfer function. With this function, an accuracy rate of approximately 90% is obtained when using the 2000 vocabulary size. As shown in Table 2, the average CPU time in Case 2 is higher than in Case 1, as expected due to using more data for training.

Case 3: Fig. 9 shows the results obtained with the 80% training set used for the classification of apples using BoVW + SVM. According to the results given in Fig. 9, the polynomial function has a very good learning rate compared to the other functions.

The best learning rate is realized when using the 2000 vocabulary size. With this, a learning rate of 96% is achieved. In the polynomial function, it is observed that the vocabulary size has accuracy rates of 92%, 94%, and 95% for the values of 500, 1000, and 1500, respectively. In the linear transfer function, 88% test accuracy is achieved with a 1500 vocabulary size. With the Gaussian function, an accuracy rate of approximately 92% is obtained when using a 2000 vocabulary size. As shown in Table 3, the mean test accuracy of Case 3 is better than Case 2 and Case 1, as expected due to using a greater percentage of the whole dataset for training.

3.3 Classification results of ResNet-50

The ResNet architecture solves gradient problems (disappearance and explosion) thanks to increasing the network layers of the image. ResNet-50 has very good identification accuracy and real-time performance compared to many other architectures [16]. Classification of apple images using the ResNet-50 deep learning method is performed in

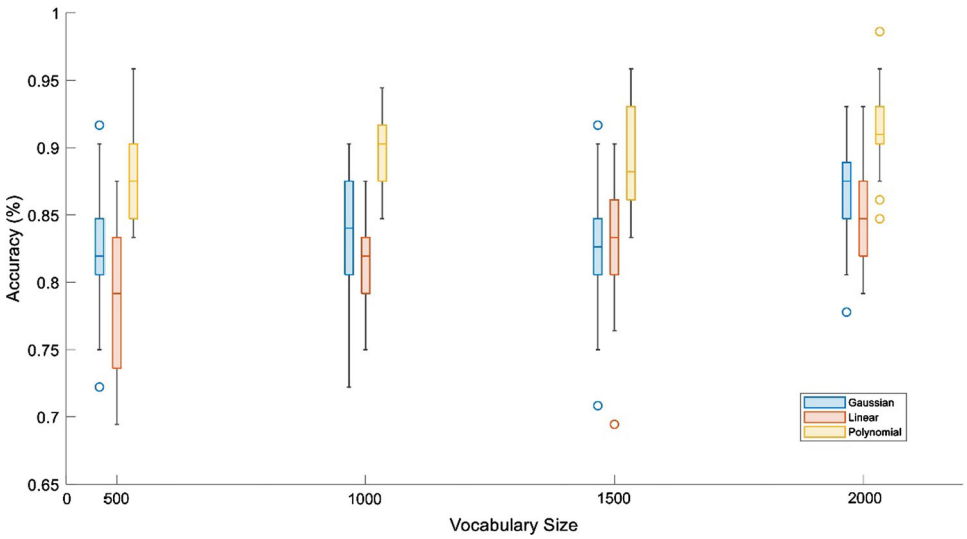

Fig. 7 Boxplot results of test dataset accuracy of apple classification using BoVW + SVM with 40% for training and 60% for testing of the dataset: Case 1

Table 1 Statistical results of accuracy and CPU time of apple classification using BoVW + SVM with 40% of the dataset in training and 60% for testing: Case 1

SVM kernel	Vocabulary size	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
Gaussian	500	0.9944	0.0094	0.8236	0.0470	107.0307
	1000	0.9993	0.0038	0.8370	0.0463	120.1018
	1500	0.9993	0.0038	0.8278	0.0436	130.7729
	2000	0.9993	0.0038	**0.8634**	0.0354	137.0979
Linear	500	0.9919	0.0162	0.7847	0.0531	108.2179
	1000	0.9938	0.0111	0.8148	0.0345	121.3457
	1500	0.9958	0.0101	0.8259	0.0457	125.0050
	2000	0.9979	0.0064	**0.8491**	0.0396	137.0353
Polynomial	500	1	0	0.8769	0.0311	103.7839
	1000	1	0	0.8954	0.0267	116.5634
	1500	1	0	0.8940	0.0364	132.0152
	2000	1	0	**0.9130**	0.0322	138.8127

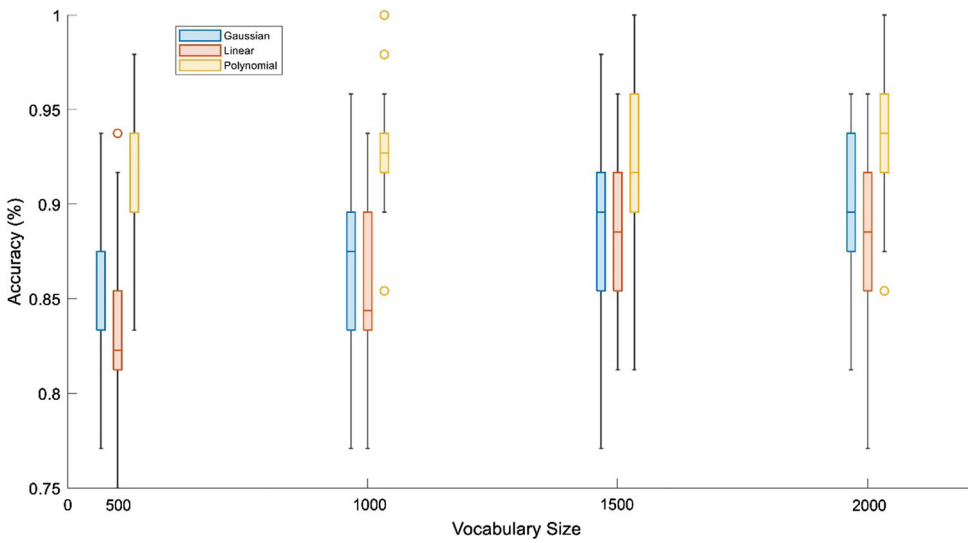

Fig. 8 Boxplot results of test dataset accuracy of apple classification using BoVW + SVM with 60% for training and 40% for testing of the dataset: Case 2

Table 2 Statistical results of accuracy and CPU time of apple classification using BoVW + SVM with 60% of the dataset in training and 40% for testing: Case 2

SVM kernel	Vocabulary size	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
Gaussian	500	0.9833	0.0152	0.8493	0.0443	142.4233
	1000	0.9912	0.0100	0.8708	0.0523	164.6093
	1500	0.9963	0.0072	0.8875	0.0443	178.3062
	2000	0.9999	0.0025	**0.8958**	0.0409	191.0347
Linear	500	0.9755	0.0181	0.8313	0.0472	141.3958
	1000	0.9866	0.0106	0.8528	0.0413	159.7241
	1500	0.9880	0.0114	0.8840	0.0362	175.9674
	2000	0.9926	0.0087	**0.8854**	0.0419	184.5520
Polynomial	500	0.9995	0.0025	0.9069	0.0374	149.7379
	1000	1	0	0.9306	0.0311	163.2641
	1500	1	0	0.9250	0.0385	186.5572
	2000	1	0	**0.9396**	0.0364	176.9773

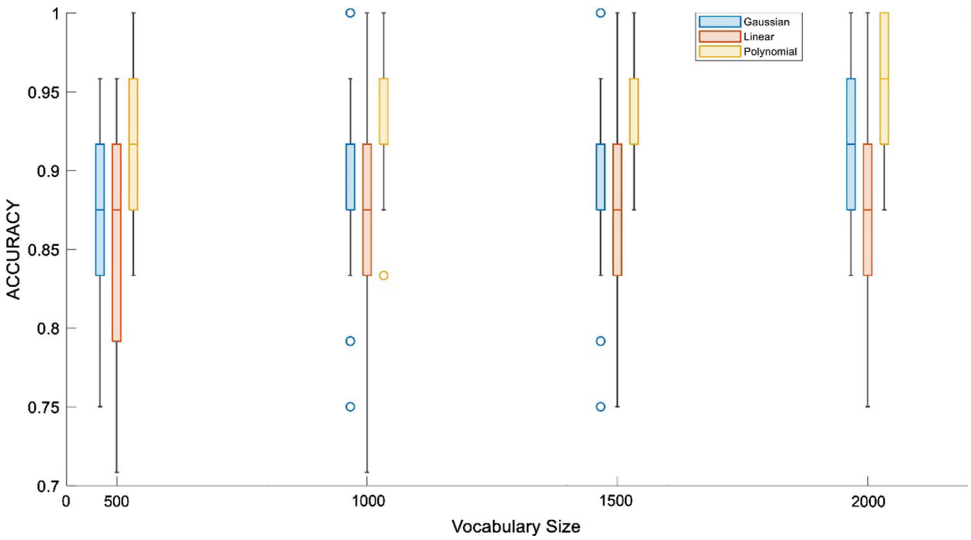

Fig. 9 Boxplot results of test dataset accuracy of apple classification using BoVW + SVM with 80% for training and 20% for testing of the dataset: Case 3

this part of the study. Three different training algorithms are used in the experiments: adaptive moment estimation (adam), root-mean-square propagation (rmsprop), and stochastic gradient descent with momentum (sgdm).

Case 1: Fig. 10 shows the results obtained using 40% for training and 60% for testing the dataset. According to the results given in Fig. 10, it is determined that the training algorithm adam at the lowest epoch number gives a higher accuracy rate than the other functions. When the epoch number is increased to 10, it is observed that the sgdm function performs better than other functions. The best learning rate for all functions occurred at 10 epochs. The accuracy rate increased with the increase of the epoch number for all functions. While the learning rate for sgdm at 10 epochs is about 88%, this rate is 83% for the adam and 86% for rmsprop functions.

Case 2: In Fig. 11, the results obtained with 60% of the dataset for training and 40% for testing in the ResNet-50 algorithm are given. In the simulations, three different training algorithms are used, as in the 60% test set. According to the results given in Fig. 11, it is determined that the adam algorithm gives a higher accuracy rate than the other functions. On the other hand, the learning performance of the sgdm function increased at 6, 8, and 10 epochs. The best learning rate for sgdm is 89% in 10 epochs. In general, it is observed that the accuracy rates of the adam and rmsprop transfer functions increase when the epoch number increases. The best values for the adam and rmsprop are approximately 86% and 85% at 10 epochs, respectively.

Case 3: The results obtained with 80% of the dataset for training data and 20% for test data are given in Fig. 12.

Table 3 Statistical results of accuracy and CPU time of apple classification using BoVW + SVM with 80% of the dataset in training and 20% for testing: Case 3

SVM kernel	Vocabulary size	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
Gaussian	500	0.9729	0.0108	0.8764	0.0614	185.1388
	1000	0.9858	0.0093	0.8972	0.0576	209.9681
	1500	0.9944	0.0066	0.8931	0.0555	214.8329
	2000	0.9986	0.0036	**0.9208**	0.0442	232.8404
Linear	500	09,587	0.0111	0.8583	0.0697	176.5363
	1000	0.9753	0.0124	0.8681	0.0610	217.8336
	1500	0.9806	0.0109	**0.8806**	0.0607	228.9741
	2000	0.9837	0.0094	0.8750	0.0547	264.8464
Polynomial	500	0.9997	0.0019	0.9181	0.0507	204.0340
	1000	1	0	0.9375	0.0434	208.7469
	1500	1	0	0.9458	0.0331	217.3534
	2000	1	0	**0.9597**	0.0387	232.4238

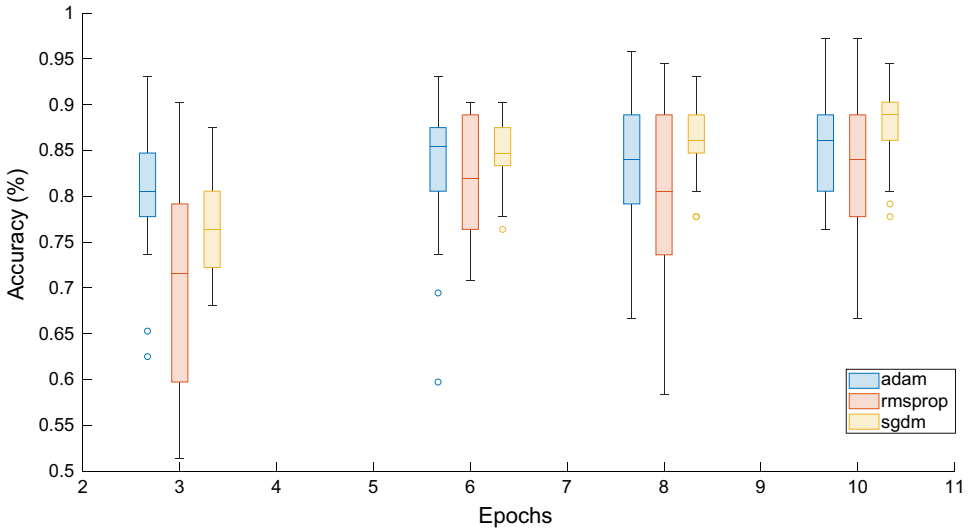

Fig. 10 Boxplot results of test dataset accuracy of apple classification using ResNet-50 with 40% for training and 60% for testing of the dataset: Case 1

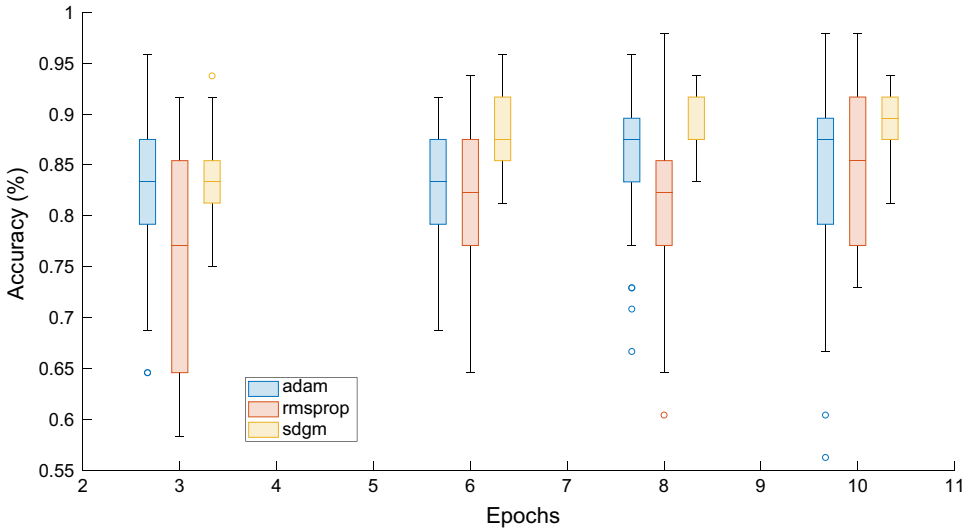

Fig. 11 Boxplot results of test dataset accuracy of apple classification using ResNet-50 with 60% for training and 40% for testing of the dataset: Case 2

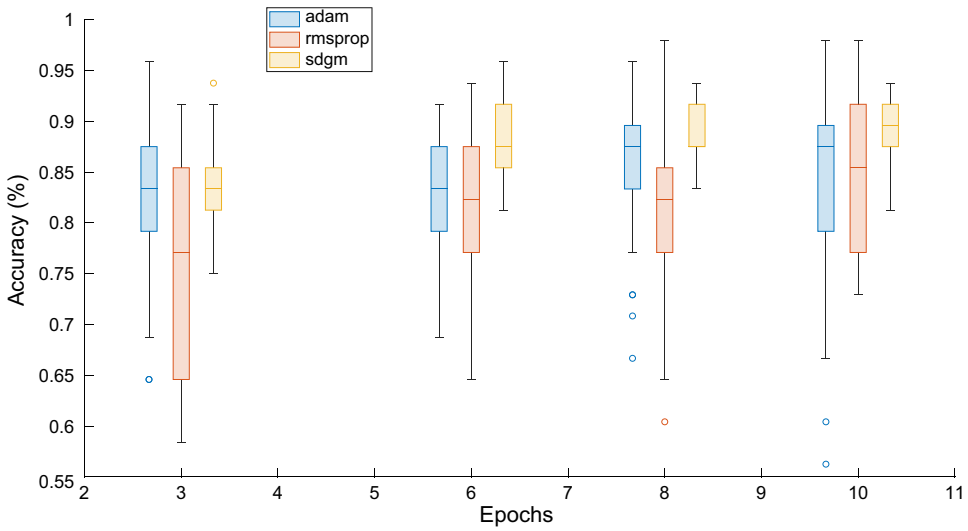

Fig. 12 Boxplot results of test dataset accuracy of apple classification using ResNet-50 with 80% for training and 20% for testing of the dataset: Case 3

According to the results given in Fig. 12, it is observed that the accuracy rates of the transfer functions generally increase with an increase in the test dataset. The best learning rates for the training algorithms sgdm, rmsprop, and adam are approximately 90%, 86%, and 89% accuracy at 10 epochs, respectively.

3.4 Comparison of BoVW + SVM and ResNet-50 methods

The polynomial kernel function reached its highest test accuracy rate in Case 1, at 91.3% for the BoVW + SVM algorithm, while the ResNet-50 algorithm reached its highest accuracy rate in the sgdm training function at 87.6% in Case 1. In the BoVW + SVM algorithm, it is determined that the highest accuracy rate increased above 93.9% in Case 2. Under the same operating conditions, it is concluded that the best accuracy rate of the ResNet-50 algorithm is approximately 89%. With an increase in the training set to 80% in Case 3, it is determined that the BoVW + SVM algorithm reached an accuracy rate of over 95.9%, but this rate remained at 90.4% in the ResNet-50 algorithm. As a result, it has been obtained that the BoVW + SVM algorithm performs very well in terms of accuracy compared to the ResNet-50 structure. When the tables are examined, performance analysis can be performed in terms of the CPU time consumption of the algorithms. According to the results of this analysis, it is determined that the ResNet-50 algorithm performs operations for a longer time than the BoVW + SVM algorithm, while it is observed that the polynomial kernel function of the BoVW + SVM algorithm performed better than the other functions under the same operating conditions, while an increase is observed in the performance of the BoVW + SVM algorithm when the vocabulary size is increased. The best classification accuracy among all operating conditions is realized in the polynomial kernel function of the BoVW + SVM algorithm, and this rate is over 95%.

Tables 1, 2, 3, 4, 5, 6 include both train and test accuracy results. However, Table 7 and Figs. 7, 8, 9, 10, 11, 12 include only the test accuracy results.

A comparison of BoVW + SVM and ResNet-50 is summarized in Table 7. In the table, accuracy is measured as the percentage of correctly classified samples and represented between 0 and 1. The experiments are repeated 30 times due to the random nature of machine learning methods. Therefore, the average of 30 independent runs is given with the standard deviation in parenthesis.

As seen from Table 7, the accuracy of Case 2 is higher than Case 1, and Case 3 is higher than Case 2. This is an expected result because 40%, 60%, and 80% of data are used in Cases 1, 2, and 3, respectively, while the training data increases the accuracy of the train and test results also increase.

The results of the BoVW + SVM and ResNet-50 methods for Case 3 are also compared with the CNN-based structures commonly used in the literature, AlexNet and GoogLeNet, in Table 8. AlexNet is a deep CNN architecture that is proposed by Krizhevsky et.al. It consists of eight layers: five convolutional layers and three fully connected layers and uses rectified linear unit (ReLU) activation function and dropout regularization to improve the generalization capability of the model. It is trained on more than one million images [28]. GoogLeNet, also known as Inception-v1, is a deep CNN architecture that is proposed by Szegedy et.al. It is characterized by the use of inception modules, which consist of multiple parallel convolutional layers of various sizes and are designed to

Table 4 Statistical results of accuracy and CPU time of apple classification using ResNet-50 with 40% of the dataset in training and 60% for testing

Training algorithm	Max epochs	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
sgdm	3	0.8708	0.0436	0.7630	0.0488	143.1643
	6	0.9757	0.0219	0.8472	0.0379	225.4943
	8	0.9889	0.0142	0.8611	0.0370	304.4231
	10	0.9972	0.0072	**0.8764**	0.0378	344.4525
rmsprop	3	0.7972	0.1157	0.7023	0.1209	141.4278
	6	0.9313	0.0629	0.8236	0.0634	232.6942
	8	0.9278	0.0863	0.8032	0.0950	292.7812
	10	0.9410	0.0582	**0.8319**	0.0795	365.9085
adam	3	0.9271	0.0578	0.8042	0.0653	146.4620
	6	0.9549	0.0413	0.8370	0.0717	221.3938
	8	0.9562	0.0442	08,347	0.0696	276.9126
	10	0.9549	0.0431	**0.8551**	0.0556	358.4349

Table 5 Statistical results of accuracy and CPU time of apple classification using ResNet-50 with 60% of the dataset in training and 40% for testing

Training algorithm	Max epochs	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
sgdm	3	0.9500	0.0229	0.8368	0.0474	165.8704
	6	0.9944	0.0086	0.8770	0.0372	280.3017
	8	0.9976	0.0064	0.8847	0.0272	370.7952
	10	0.9986	0.0042	**0.8909**	0.0287	468.0144
rmsprop	3	0.8764	0.0927	0.7590	0.1022	158.0729
	6	0.9333	0.0647	0.8257	0.0777	277.6705
	8	0.9296	0.0688	0.8181	0.0915	367.4697
	10	0.9476	0.0551	**0.8513**	0.0731	450.5992
adam	3	0.9403	0.0415	0.8243	0.0811	160.1268
	6	0.9639	0.0362	0.8292	0.0637	281.4267
	8	0.9639	0.0506	**0.8562**	0.0733	382.2819
	10	0.9634	0.0455	0.8389	0.1015	450.3780

increase the representational power of the network while maintaining computational efficiency [29].

As can be seen from Table 8, the accuracy of AlexNet and GoogLeNet is slightly lower than BoVW + SVM and ResNet-50. However, the CPU time consumption of AlexNet and GoogLeNet is better than the others.

Other neural network, SVM- and CNN-based methods in the literature can be summarized as follows: Shahin et.al. proposed a neural network model to detect bruise damage in apples. They used line-scan X-ray images of red delicious (RD) and golden delicious (GD) types of apples. To select the salient features, stepwise discriminant analysis is used. The classification accuracy is 90% and 93% for RD and GD apples, respectively [30].

Song et.al. developed a low-cost sensor to distinguish organic and conventional apples. Several machine learning algorithms are evaluated on rainbow images produced from the apple image dataset. SVM and locally weighted partial least squares classifier (LW-PLSC) obtained 93–100% classification accuracy [31].

Li et.al. proposed a model to analyze the fast grading of apple quality by using CNN with 95.33% accuracy. The results are better than Google Inception v3, gray-level co-occurrence matrix (GLCM) features merging, histogram of oriented gradient (HOG), and support vector machine (SVM) classifier [32].

Table 6 Statistical results of accuracy and CPU time of apple classification using ResNet-50 with 80% of the dataset in training and 20% for testing

Training algorithm	Max epochs	Mean train accuracy	Std. of train accuracy	Mean test accuracy	Std. of test accuracy	Avg. CPU time (s)
sgdm	3	0.9521	0.0260	0.8444	0.0557	180.0895
	6	0.9913	0.0099	0.8792	0.0442	335.5029
	8	0.9958	0.0070	**0.9042**	0.0248	401.4973
	10	0.9990	0.0032	0.9028	0.0228	487.4055
rmsprop	3	0.8896	0.0754	0.8139	0.0907	180.0703
	6	0.9288	0.0590	0.8194	0.0993	329.8573
	8	0.9278	0.0774	0.8528	0.1031	423.9877
	10	0.9497	0.0641	**0.8597**	0.0822	482.9436
adam	3	0.9170	0.0515	0.8278	0.0663	178.0338
	6	0.9406	0.0506	0.8542	0.0824	319.2763
	8	0.9646	0.0392	**0.8889**	0.0632	427.4709
	10	0.9778	0.0273	0.8736	0.0925	438.3448

Table 7 Comparison of BoVW + SVM and ResNet-50 methods with the test datasets

		Case 1	Case 2	Case 3
BoVW + SVM	Accuracy	0.9130 (0.0322)	0.9396 (0.0364)	0.9597 (0.0387)
	Parameters	Polynomial, 2000	Polynomial, 2000	Polynomial, 2000
	CPU time (s)	138	176	232
ResNet-50	Accuracy	0.8764 (0.0378)	0.8909 (0.0287)	0.9042 (0.0248)
	Parameters	sgdm, 10	sgdm, 10	sgdm, 8
	CPU time (s)	344	468	401

Table 8 Comparison of BoVW + SVM and ResNet-50 results with AlexNet and GoogLeNet CNN methods for Case 3

	BoVW + SVM	ResNet-50	AlexNet	GoogLeNet
Accuracy	0.9597	0.9042	0.8566	0.9033
CPU time (s)	232	401	123	160

4 Conclusion

In this study, the BoVW + SVM and ResNet-50 algorithms, which are machine learning methods, are used to classify apple images according to their varieties. In both methods, the best results are observed in Case 3 (80% used for training and 20% for testing of the dataset). In the BoVW + SVM method, the training accuracy in the 2000 vocabulary size polynomial kernel function is 100% and the best test rate is 95.9%. In the ResNet-50 method, it is determined that the training accuracy is 99.5% and the best test rate is 90.4% in the 8 epoch sgdm training algorithms. In general, it has been observed that the prediction abilities of both algorithms increase with an increase in the training rates. In the BoVW + SVM method, it is observed that the average time increased as the vocabulary size increased, and in the ResNet-50 method, the average time increased as the epoch number increased. In the best mean test results, there is a processing time of 232 min for 30 replicates of the BoVW + SVM algorithm, while it has a processing time of 401 min for 30 iterations of the ResNet-50 algorithm. It is determined that the BoVW + SVM method is more successful than the ResNet-50 method in terms of both test rate and mean time. In future studies, the sample size of the dataset may be increased to increase the overall accuracy of the classifier. Also, the dataset can be widened by adding other apple species to cover a greater variety. In addition, since the deep learning field is very popular, results can be obtained by conducting new experiments with new models.

Acknowledgements This research was financially supported by Kayseri University Scientific Research Projects Unit (Project No. BAP, FYL-2022-1059).

Data availability The datasets generated during and/or analyzed during the current study are available in the apple_dataset repository, github.com/rifatkurban/apple_dataset.

Declarations

Competing interests 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.

References

1. Yahia EM, García-Solís P, Celis MEM (2019).Contribution of fruits and vegetables to human nutrition and health. In: Postharvest physiology and biochemistry of fruits and vegetables. Woodhead Publishing, pp 19–45
2. Kaur C, Kapoor HC (2001) Antioxidants in fruits and vegetables–the millennium's health. Int J Food Sci Technol 36(7):703–725
3. Ahmad R, Hussain B, Ahmad T (2021) Fresh and dry fruit production in himalayan Kashmir, sub-Himalayan Jammu and trans-himalayan Ladakh. India Heliyon 7(1):e05835
4. Osowski S, Siwek K, Markiewicz T (2004) MLP and SVM networks-a comparative study. In: Proceedings of the 6th nordic signal processing symposium, 2004. NORSIG 2004. IEEE, pp 37–40
5. Nitze I, Schulthess U, Asche H (2012) Comparison of machine learning algorithms random forest, artificial neural network and support vector machine to maximum likelihood for supervised crop type classification. In: Proceedings of the 4th GEOBIA, Rio de Janeiro, Brazil, 79, 3540
6. Kafle GK, Khot LR, Jarolmasjed S, Yongsheng S, Lewis K (2016) Robustness of near infrared spectroscopy based spectral features for non-destructive bitter pit detection in honeycrisp apples. Postharvest Biol Technol 120:188–192
7. Mizushima A, Lu R (2013) An image segmentation method for apple sorting and grading using support vector machine and Otsu's method. Comput Electron Agric 94:29–37
8. Patel CC, Chaudhari VK (2020) Comparative analysis of fruit categorization using different classifiers. In: Advanced engineering optimization through intelligent techniques. Springer, Singapore, pp 153–164
9. Naik S, Patel B (2017) Machine vision based fruit classification and grading-a review. Int J Comput Appl 170(9):22–34
10. Chua LO, Yang L (1988) Cellular neural networks: theory. IEEE Trans Circuits Syst 35(10):1257–1272
11. Koklu M, Unlersen MF, Ozkan IA, Aslan MF, Sabanci K (2022) A CNN-SVM study based on selected deep features for grapevine leaves classification. Measurement 188:110425
12. Li Z, Niu B, Peng F, Li G, Yang Z, Wu J (2018) Classification of peanut images based on multi-features and SVM. IFAC-PapersOnLine 51(17):726–731
13. Jiang H, Li X, Safara F (2021) IoT-based agriculture: deep learning in detecting apple fruit diseases. Microprocess Microsyst 104321
14. Tahir MB, Khan MA, Javed K, Kadry S, Zhang YD, Akram T, Nazir M (2021) WITHDRAWN: recognition of apple leaf diseases using deep learning and variances-controlled features reduction
15. Liu Y, Zhang Z, Liu X, Wang L, Xia X (2021) Ore image classification based on small deep learning model: evaluation and optimization of model depth, model structure and data size. Miner Eng 172:107020
16. 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
17. Raikar MM, Meena SM, Kuchanur C, Girraddi S, Benagi P (2020) Classification and grading of okra-ladies finger using deep learning. Procedia Comput Sci 171:2380–2389
18. Bosch A, Munoz X, Marti R (2007) Which is the best way to organize/classify images by content? Image Vis Comput 25(6):778–791
19. Salton G, McGill MJ (1983) Information retrieval: an Introduction. Introduction to modern information retrieval, pp 1–23
20. Lowe G (2004) Sift-the scale invariant feature transform. Int J 2(91–110):2
21. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (SURF). Comput Vis Image Underst 110(3):346–359
22. Shen L, Chen H, Yu Z, Kang W, Zhang B, Li H, Liu D (2016) Evolving support vector machines using fruit fly optimization for medical data classification. Knowledge-Based Syst 96:1–75
23. Hastie T, Tibshirani R, Friedman JH, Friedman JH (2009) The elements of statistical learning: data mining, inference, and prediction, Vol 2, pp 1–758. Springer, New York
24. Mathworks Inc. (2022) Fit multiclass models for support vector machines or other classifiers. https://www.mathworks.com/help/stats/fitcecoc.html
25. Feng X, Jiang Y, Yang X, Du M, Li X (2019) Computer vision algorithms and hardware implementations: a survey. Integration 69:309–320
26. Cao Y, Wu Z, Shen C (2017) Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE Trans Circuits Syst Video Technol 28(11):3174–3182
27. Mathworks Inc. (2022) Options for training deep learning neural network. https://www.mathworks.com/help/deeplearning/ref/trainingoptions.html
28. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
29. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
30. Shahin MA, Tollner EW, McClendon RW, Arabnia HR (2002) Apple classification based on surface bruises using image processing and neural networks. Trans Am Soc Agric Eng 45(5):1619–1627
31. Song W, Jiang N, Wang H, Guo G (2020) Evaluation of machine learning methods for organic apple authentication based on diffraction grating and image processing. J Food Compos Anal 88:103437
32. Li Y, Feng X, Liu Y, Han X (2021) Apple quality identification and classification by image processing based on convolutional neural networks. Sci Rep 11(1):1–15