European Food Research and Technology
https://doi.org/10.1007/s00217-024-04562-4

ORIGINAL PAPER

Almond *(Prunus dulcis)* varieties classification with genetic designed lightweight CNN architecture

Mustafa Yurdakul[1] · İrfan Atabaş[1] · Şakir Taşdemir[2]

Received: 29 January 2024 / Revised: 15 April 2024 / Accepted: 20 April 2024

Abstract

Almond *(Prunus dulcis)* is a nutritious food with a rich content. In addition to consuming as food, it is also used for various purposes in sectors such as medicine, cosmetics and bioenergy. With all these usages, almond has become a globally demanded product. Accurately determining almond variety is crucial for quality assessment and market value. Convolutional Neural Network (CNN) has a great performance in image classification. In this study, a public dataset containing images of four different almond varieties was created. Five well-known and light-weight CNN models (DenseNet121, EfficientNetB0, MobileNet, MobileNet V2, NASNetMobile) were used to classify almond images. Additionally, a model called 'Genetic CNN', which has its hyperparameters determined by Genetic Algorithm, was proposed. Among the well-known and light-weight CNN models, NASNetMobile achieved the most successful result with an accuracy rate of 99.20%, precision of 99.21%, recall of 99.20% and f1-score of 99.19%. Genetic CNN outperformed well-known models with an accuracy rate of 99.55%, precision of 99.56%, recall of 99.55% and f1-score of 99.55%. Furthermore, the Genetic CNN model has a relatively small size and low test time in comparison to other models, with a parameter count of only 1.1 million. Genetic CNN is suitable for embedded and mobile systems and can be used in real-life solutions.

Keywords Almond classification · Convolutional neural networks · Genetic algorithm · Deep learning · Optimization

Introduction

The almond *(Prunus dulcis),* with an origin that dates back approximately 5.88 million years [1], is one of the nuts with the highest protein content [2]. It is also rich in essential nutrients such as vitamin E, calcium, iron, magnesium, phosphorus, and zinc [3]. Almond supports heart health [4] with beneficial fats and helps regulate blood sugar, contributing to diabetes management. In addition to its nutritional benefits, it has a range of applications across different industries. In traditional medicine, the almond offers natural solutions for various health issues [5]. It is utilized as a moisturizer in cosmetics and skincare products, acts as fillers and sweeteners in the pharmaceutical industry, and is employed as active ingredients in natural cleaning products. With these usages, almonds have become a sought-after product in international export markets [6]. The classification of almond varieties is crucial for quality assessment and market value determination. Traditional methods for classifying almond varieties often rely on morphological characteristics and genetic analysis. However, these methods can be time-consuming, labor-intensive and prone to errors. Since morphological traits may vary and genetic tests are expensive, there is a demand for faster, cheaper and innovative methods for the classification of almond varieties.

In the existing literature, one of the methods used to automatically classify almonds is based on feature extraction. Initially, characteristics such as shape, color, and texture are extracted from images, and then the features are classified using a machine learning algorithm. Teimouri et al. [7], captured images of five almond types [Normal (NA), Broken (BA), Double (DA), Wrinkled (WA), Shell (SA)] and extracted a comprehensive set of features. The features were classified with Artificial Neural Network (ANN) and achieved accuracies: 98.92% for NA, 99.46% for BA, 98.38% for DA, 98.92% for WA, and 100% for SA. Halac et al. [8] classified five almond types using shape and RGB

✉ Mustafa Yurdakul
mustafayurdakul@kku.edu.tr

[1] Computer Engineering Department, Kırıkkale University, Kırıkkale, Turkey

[2] Computer Engineering Department, Selcuk University, Konya, Turkey

Published online: 16 May 2024

color features. They achieved an accuracy rate of 92.72% with Principal Component Analysis (PCA) and Support Vector Machine (SVM) algorithms. Nasirahmadi et al. [9] classified almonds as sweet or bitter, extracting various features. The Chi-Square Support Vector Machine (Chi-SVM) was used as the classifier, achieving an accuracy rate of 91% (± 1.6). Narendra et al. [10] proposed a feature-extraction based method for classifying almond varieties as Nonpareil (NP), Mission (MI), Carmel (CR), and California (C). They achieved an accuracy of 97.13% with Decision Tree (DT) algorithm. Table 1 summarizes the literature on feature-extraction based almond classification studies.

CNN is another method used for the automatic classification of almonds. Benarous [11] proposed a novel CNN model based on AlexNet to classify 16 almond varieties and achieved an accuracy rate of 94% accuracy. Vidyarthi et al. [12] classified almond kernels into five categories using pre-trained CNN models (Inception-V3, ResNet50, VGG-16) and a custom model. ResNet50 achieved the highest accuracy at 98.4% and the custom model achieved 92.2% accuracy but the custom model had fewer parameters compared to the pre-trained models.

Table 2 provides a summary of CNN-based variety classification studies in the literature. When Table 2 is examined, CNNs have shown promising results in almond classification. However, the number of studies is limited. CNNs have already proven effective in various agricultural applications, such as classifying dry beans [13], hazelnuts [14], lentil varieties [15], and corn [16]. Thus, the use of CNN based methods for almond classification is important. However, CNN models must be designed to be light-weight and fast to ensure their feasibility in embedded systems. When designing a CNN model, it is necessary to determine various hyperparameters, such as the number and size of convolutions, the pooling method to be used, and the number of neurons in fully connected layers.

The determination of the hyperparameters is an optimization problem. The values of CNN hyperparameters, such as the size and number of convolutional filters, the size and type of pooling filter, and the number of neurons, are all exact numbers. As a result, they can be solved using optimization approaches that are suitable for a discrete structure. In this context, the Genetic Algorithm (GA), by its very nature, is well suited to a discrete structure.

GA is a search and optimization algorithm developed inspired by Darwin's theory of evolution [17]. The GA has achieved successful results in solving optimization problems [18]. Rodrigues et al. [19] used ResNet50V2 model to classify Acute Lymphoblastic Leukemia. They applied transfer learning and determined the dropout, momentum, and learning rate parameters using GA. Das et al. [20] developed a CNN model for detecting the stages of diabetic retinopathy. The authors employed a GA to determine the parameters such as the number of convolution and pooling layers, the quantity and size of kernels in the convolutional layers. Ayan [21] used DenseNet121, MobileNetV2, and InceptionResNetV2 models for classifying insect types using the transfer learning method. During the transfer learning

Table 1 Feature-extraction based almond classification studies in the literature

References	Author(s) and year	Varieties	Features extracted	Classifier	Accuracy
[7]	Teimouri et al. (2016)	Normal (NA), broken (BA), double (DA), wrinkled(WA), shell (SA)	Shape, color, texture	ANN	NA: 98.92%, BA: 99.46%, DA: 98.38%, WA: 98.92%, SA: 100%
[8]	Halac et al. (2017)	Raw, blanched, roasted, roasted blanched, unknown	Shape and color histogram	SVM	92.72%
[9]	Nasirahmadi et al. (2017)	Sweet or bitter	Harris, harris-laplace, hessian, hessian-laplace, maximally stable extremal regions (MSER)	Chi-SVM	91%
[10]	Narendra et al. (2021)	Nonpareil (NP), Mission (MI), Carmel (CR), California (C)	Geometric, shape, color, texture	DT	97.13%

Table 2 CNN-based almond classification studies in literature

References	Author(s) and year	Varieties	Classifier	Accuracy
[11]	Benarous (2020)	16 different almond varieties	AlexNet and AlexNet V2	AlexNet: 89%, AlexNet V2: 94%
[12]	Vidyarthi et al. (2021)	Almond kernels categorized by percentage of brokenness (0%, 5%, 15%, 25%, 35%)	ResNet50	98.4%

process, the ratio of layers to be frozen (freeze ratio), the number of layers, neuron count in the layers, and the dropout rate were determined using GA.

Table 3 provides a summary of GA applications in the literature. When Tables 1–3 are examined, it is seen that using GA together with CNN can potentially yield effective results in classifying almonds. In light of the literature, in this study, an almond dataset containing 1556 images of four distinct almond varieties was created. The almond varieties were classified using five well-known and light-weight and pre-trained CNN models. In addition, a new CNN model called 'Genetic CNN' was proposed. The model's hyperparameters were determined by GA. The rest of the paper is structured as follows: Sect. "Material and methods" details the methodology including data collection, CNNs, Transfer Learning, Genetic Algorithms, and Genetic CNN. Sect. "Experimental setup" covers the experimental setup and evaluation metrics. Sect. "Experimental results" presents the experimental results. Sect. "Discussion" discusses the findings, and Sect. "Conclusion" concludes the study and provides details about future works.

Material and methods

The details of the methods used in the study are explained in this section. The first subsection describes the system setup for data collection. The 'Dataset' subsection provides a detailed explanation of the data obtained. The 'Convolutional Neural Networks' subsection explains the basic concepts of CNNs. The 'Transfer Learning' subsection discusses the process of transferring features (weights) from previously trained CNN models. The section on the 'Genetic Algorithm' explains the algorithm's working principles. The proposed Genetic CNN architecture is then described.

Data collection

Almond varieties named Ak, Nurlu, Kapadokya and Sıra collected from different regions of Turkey were used in the study. To capture images of the almonds, a special chamber with homogeneous lighting was designed. The chamber is lighted with string circuits with LEDs on it operating with 5 Volt. The Samsung NX300 camera was used to capture the almond images. The camera has a 20.3 Megapixel CMOS sensor and a variable lens ranging from 24–55 mm. The lens was fixed at 45 mm and images were captured. The images captured at an ISO setting of 100 and a shutter speed of 1/400 s. Images are saved in the memory in JPG format. Figure 1 presents a schematic representation of the system.

Dataset

The Ak, Nurlu and Sıra almond varieties are grown in the Datça district of Muğla in the Aegean region of Turkey, while the Kapadokya almond is grown in Nevşehir in Central Anatolia. Both regions have suitable climate and soil properties for almond cultivation. A total of 1556 images of the four varieties were captured. Sample images selected from the dataset are shown in Fig. 2.

Table 4 provides details on the quantity values of the almond varieties. The dataset is publicly available on the Kaggle platform (https://github.com/ymyurdakul/datasets/blob/main/almond/archive.zip).

Convolutional neural network

CNN is a deep learning algorithm inspired by the visual cortex system [27]. CNN is extensively utilized in image analysis for tasks like image classification, detection, or

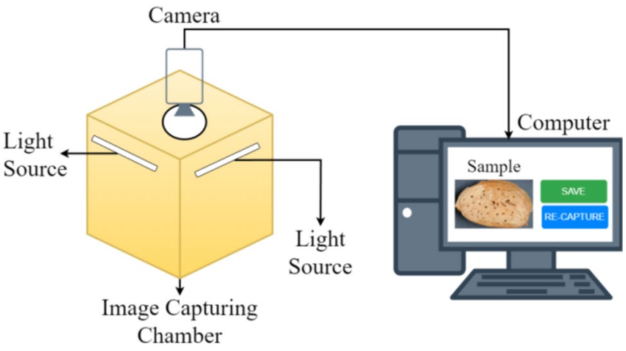


Fig. 1 Data collection system

Table 3 GA applications in the literature

References	Author(s) and year	Application	Model	Key focus of GA
[19]	Rodrigues et al. (2022)	Classification of acute lymphoblastic leukemia	ResNet50V2	Tuning dropout, momentum, learning rate parameters
[20]	Das et al. (2022)	Detection of diabetic retinopathy stages	CNN model	Determining convolution/pooling layers, kernel size and number
[21]	Ayan (2023)	Classification of insect types	DenseNet121, MobileNetV2, Inception-ResNetV2	Determining freeze ratio, layer/neuron count, dropout rate

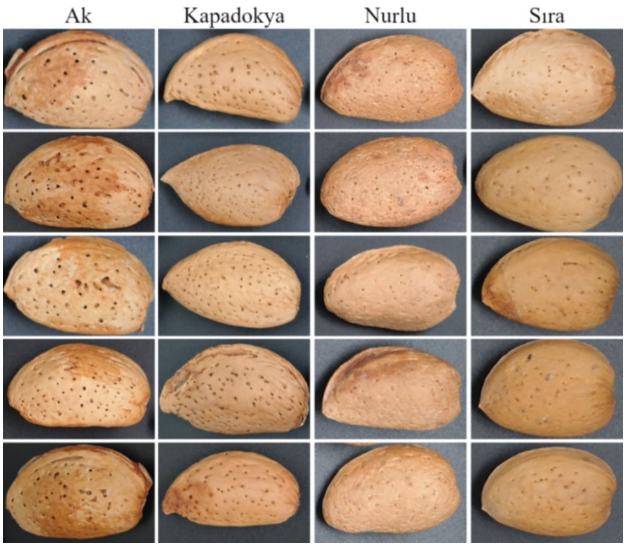


Fig.2 Almond dataset sample images

Table 4 Almond varieties distribution

Variety	Ak	Kapadokya	Nurlu	Sıra	Total
Count	401	465	306	384	1556

segmentation and has achieved successful outcomes in various fields, such as defense [28, 29], medicine [30–33], agriculture [34–36] and education [37, 38]. A CNN comprises three main components: convolution, pooling, and a fully connected layer. The convolution layer extracts features from images, while the pooling layer reduces the feature matrix size obtained by the convolution layer. The fully connected layer consists of multilayer artificial neural network architecture. At the end of the fully connected layer, a softmax activation function is typically used for classification. Various CNN architectures have been proposed using these three core components. The number of parameters and complexity of models increases with the usage of components. For real-time classification tasks, light-weight models are necessary. In this work, five well-known and light-weight CNN models have been trained on the dataset, considering the parameter and size values. The details of the models are shown in Table 5.

EfficientNetB0 contains Mobile Inverted Bottleneck Convolution (MBConv) blocks, which have an inverted residual structure that improves scaling. DenseNet121 contains dense blocks to ensure full information sharing across layers. MobileNet is designed with depth-wise separable convolutions to reduce computation. MobileNetV2 builds upon this with inverted residuals and linear bottlenecks for enhanced performance. Lastly, NASNetMobile, created through Neural Architecture Search, employs Normal and Reduction Cells to effectively manage spatial dimensions and feature complexity.

Transfer learning

Transfer learning (TL) is the process of transferring previously learned knowledge to new and different situations [39]. In deep learning, TL enables a model trained on a large dataset to be effectively and quickly trained on a smaller, specialized dataset [40]. TL enables the model to focus on the specific features of the new dataset instead of re-learning the general features. The five light-weight models used in this study were previously trained on the ImageNet dataset. To successfully implement transfer learning, the final classification layers of these models have been removed and replaced with a Global Average Pooling (GAP) layer, and a four-output softmax layer was added for almond classification. The purpose of using the GAP layer was to enable the model to average the feature maps in the spatial dimension to obtain a denser representation. Also, the GAP layer aims to reduce the computational load and minimize the risk of overfitting by decreasing the number of parameters in the model.

Genetic algorithm

Genetic algorithm (GA) is an optimization algorithm based on natural selection, developed in 1975 at the University of Michigan by John Holland and his students [23]. Inspired by Darwin's theory of evolution, Holland and his students designed an algorithm to emulate genetic processes in living beings. In GA, potential solutions are represented by data structures called chromosomes, which consist of genes

Table 5 The details of light-weight CNN models [22]

Model	Parameter count(M)	Size(Mb)	Depth	Block type
EfficientNetB0	5.3	29	132	Mobile inverted bottleneck convolution
DenseNet121	8.1	33	242	Dense connection
MobileNet	4.3	16	55	Depthwise separable convolution
MobileNet V2	3.5	14	105	Depthwise separable convolution
NASNetMobile	5.3	23	389	Normal cell and reduction cell

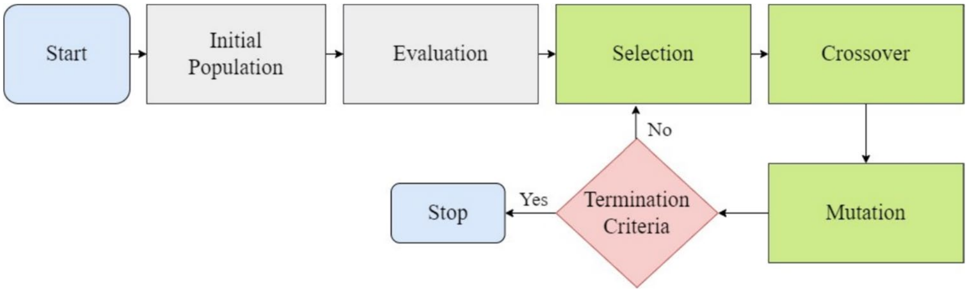


Fig. 3 Flowchart of the GA

typically expressed as a series of numbers or a string of characters, symbolizing a solution to a problem [18]. Each iteration of GA, known as a 'generation,' contains a population of solutions evaluated by a specific fitness function [24]. The fitness function measures the quality of a chromosome's solution. The solution process begins with a randomly created initial population, and then the best chromosomes are selected. In this process, methods such as 'roulette wheel' selection are commonly used [25]. The roulette wheel determines the chance of each chromosome being selected based on its fitness level [25]. Chromosomes with higher fitness values are more likely to be included in the genetic pool for future generations. The selected chromosomes generate offspring chromosomes using genetic operators known as crossover and mutation. The crossover operator combines the features of two chromosomes to generate offspring [24]. The mutation operator makes random changes in chromosomes to increase genetic diversity [24]. Typically, the values of a few genes on a chromosome are altered. The primary purpose of mutation is to enhance the diversity in the search space, preventing the algorithm from focusing only on specific solutions. The new offspring generated through crossover and mutation operations are evaluated based on their fitness values. The most effective solutions are then selected for the subsequent generation, thereby enhancing the overall quality of the population. The process is repeated until a predetermined stopping criterion is met or a satisfactory solution is found. The flowchart illustrating the process of the GA is presented in Fig. 3.

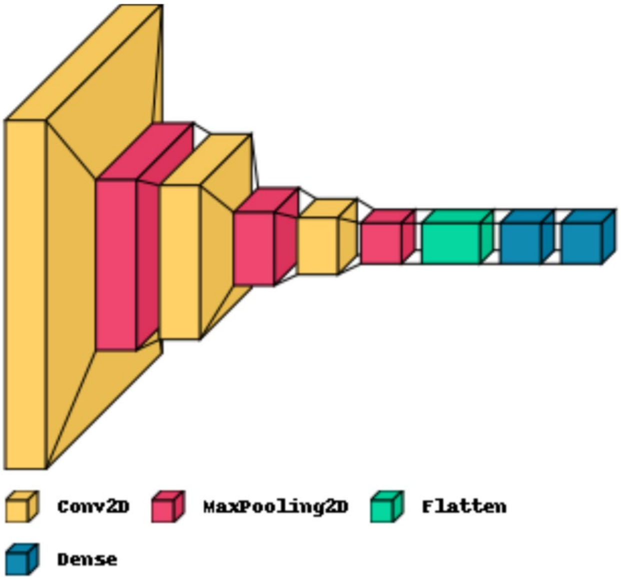


Fig. 4 Genetic CNN architecture

Table 6 CNN architecture hyperparameters and range

Hyperparameter	Range
Filter Size	[1×1 (0), 3×3 (1), 5×5 (2)]
Filters in conv layer	[1–64]
Activation function	[Relu(0), Tanh(1), Selu(2), Elu(3)]
Pooling type	[Max(0),Average(1)]
Pooling size	[2×2 (0), 3×3 (1)]
Neurons count of dense layer	[1–2048]

Genetic CNN architecture

The Genetic CNN model consists of three convolution layers and three pool layers. The choice of three layers is because additional layers would increase the number of parameters and the size of the model. Also increasing the count of the layer can degrade accuracy due to over-complexity [26, 27]. After the pooling layer, the multidimensional feature map is flattened and then fed into a dense layer. The last layer is a classification layer that uses the softmax activation function to classify different types of almonds. Figure 4 visually represents the layers of the proposed architecture.

The proposed model's filter size, number of filters, and activation function in the convolution layers, type and size of pooling in the pooling layers, and number of neurons in the dense layer were determined using the GA. Table 6 provides the parameter ranges.

To limit the number of parameters, the number of filters in the convolution layers was chosen between 1 and 64. The 1×1 and 3×3 filters were used to extract local and simple features from the images, while the 5×5 filters were chosen to capture features in larger areas [28, 29]. Additionally, a variety of activation functions were implemented in the layers. For the pooling methods, the commonly used Average and Max pooling methods were chosen. The pooling sizes were chosen to be either 2×2 or 3×3. The number of neurons in the dense layer was kept within a broad range to enhance the model's flexibility and adaptability for diverse data sets and tasks.

Population initialization

A chromosome containing 16 genes was created to represent candidate solutions. The first three genes of the chromosome determine the hyperparameters of the model's first convolutional layer. The next two genes contain the hyperparameters of the first pooling layer. This pattern continues for the second and third convolutional and pooling layers, so the first 15 genes define the hyperparameters of these layers. The last gen determines the number of neurons in the model's dense layer. Figure 5 shows an example of a chromosome with randomly generated genes, which represents CNN architecture.

The size of the population must be greater than the length of the chromosome [30]. Additionally, the population size is related to the computational complexity. In many studies in the literature, successful results have been achieved with a 50 population size [30–32]. For all these reasons, the first population was created with 50 chromosomes consisting of gene values randomly selected from the ranges in Table 6.

Fitness function

The quality of the solutions in the population was determined using the F1-Score values of CNN models on validation data. The F1 Score is the harmonic mean of the precision and recall rates of the model on the data. The reason for using the F1-Score is that it balances both precision and recall rates, thus allowing the algorithm to make a more balanced selection. The F1-Score is calculated using eq. 1.

$$Fitness = F1 - Score = \frac{2\text{xPrecisionxRecall}}{\text{Precision+Recall}} \tag{1}$$

Selection

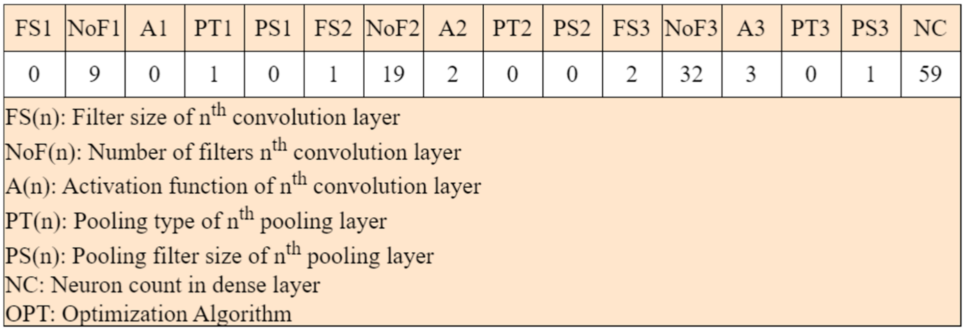
In GA, during the selection phase, chromosomes are chosen from the population to produce new offspring. There are various selection methods such as Tournament, Roulette Wheel, and Rank Selection. In this study, the Roulette Wheel Selection method was employed, which assigns selection probabilities proportional to the fitness values of each chromosome, thereby favoring those with higher fitness. Consistently selecting the fittest chromosomes through the Roulette Wheel method enhances the overall quality of the population. The Roulette Wheel is in alignment with the 'survival of the fittest' principle, fundamental to GA. The Roulette Wheel method ensures the evolution of the most optimal solutions, while maintaining genetic diversity within the population [33].

Crossover and mutation

The crossover operator in GAs is used to change the genes of parent chromosomes to create new and potentially better gene combinations. It is conducted by intersecting chromosomes at a specific point and exchanging gene segments. In this study, single-point crossover is used, where genes are mixed from a randomly chosen point along both parental chromosomes. The crossover operator ensures that offspring inherit genes from both parents. For example, if the 6th gene is the crossover point in Fig. 6, the first child inherits the first 6 genes from one parent and the rest from the other, while the second child inherits in a similar way but from the other parents.

After the crossover operation, mutation processes are typically applied. Mutation enhances genetic diversity by altering random genes in the newly formed chromosomes. In this study, for the mutation process, a random gene from the chromosome was selected, and a value was randomly assigned from the range specified in Table 6. Figure 7 illustrates an example of a mutation process where a gene selected at random is replaced with a random value.

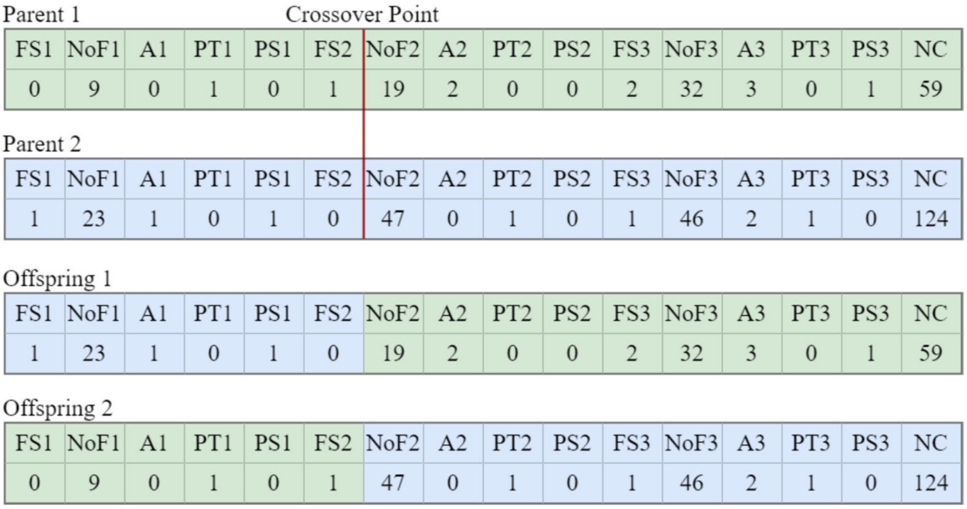
After crossover and mutation, the fitness values of the offsprings were determined. Based on the fitness values, it



FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
0	9	0	1	0	1	19	2	0	0	2	32	3	0	1	59

FS(n): Filter size of n^{th} convolution layer
NoF(n): Number of filters n^{th} convolution layer
A(n): Activation function of n^{th} convolution layer
PT(n): Pooling type of n^{th} pooling layer
PS(n): Pooling filter size of n^{th} pooling layer
NC: Neuron count in dense layer
OPT: Optimization Algorithm

Fig. 5 Chromosome representation with randomly generated values



Parent 1

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
0	9	0	1	0	1	19	2	0	0	2	32	3	0	1	59

Parent 2

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
1	23	1	0	1	0	47	0	1	0	1	46	2	1	0	124

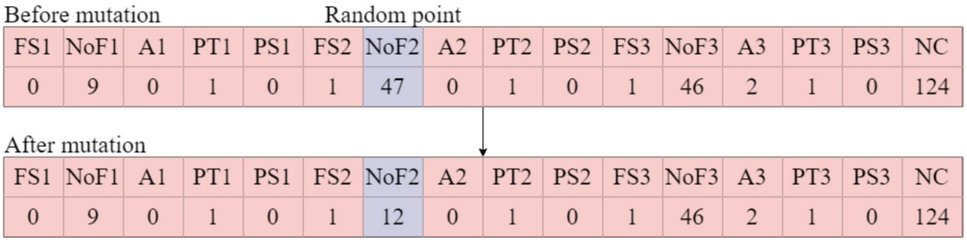
Offspring 1

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
1	23	1	0	1	0	19	2	0	0	2	32	3	0	1	59

Offspring 2

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
0	9	0	1	0	1	47	0	1	0	1	46	2	1	0	124

Fig. 6 The single-point crossover operation



Before mutation

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
0	9	0	1	0	1	47	0	1	0	1	46	2	1	0	124

After mutation

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
0	9	0	1	0	1	12	0	1	0	1	46	2	1	0	124

Fig. 7 The mutation operation

is decided whether they will proceed to the next generation. Then, depending on the number of elites in the population, poor candidates are eliminated. The selection, crossover, and mutation processes are repeated until the termination criterion is met.

Experimental setup

All experiments were performed on a computer configured with an Nvidia GTX 3060Ti GPU, 128 GB RAM, and an Intel(R) Core(TM) i9-10920X CPU. The CNN models and optimization algorithm was developed utilizing the Keras 2.9.0 API, which operates on a Tensorflow 2.9.2 backend with Python 3.8.10. CUDA V11.2 was used for GPU acceleration. In the experimental studies, pre-trained CNN models were used and retrained for the classification of almond varieties. The learning rate was set at 0.01, with a mini-batch size of 32. Stochastic Gradient Descent was used as the optimizer. The training was stopped after 50 epochs to prevent the network from becoming overfitting. In the classification layer of the CNN models, categorical cross-entropy loss was utilized. The hyperparameters of the pre-trained CNN models are determined with the GridSearch method. The GridSearch finds the best-performing set of hyperparameters by trying all possible combinations over a given set of values for certain hyperparameters. Similarly, for the Genetic CNN, GA was executed for 50 iterations, employing crossover and mutation rates of 0.95 and 0.5, respectively.

Evaluation metrics

In this study, various metrics were used to evaluate the performance of the models. Accuracy measures the percentage of correct predictions among all predictions and is calculated using the formula in Eq. 2. The precision metric is used to determine the proportion of actual positives among those predicted as positive and is calculated using the formula in Eq. 3. Recall indicates how well the model identifies true positive situations and is calculated using the formula in Eq. 4. F1 Score is computed using the harmonic mean of precision and recall, as shown in Eq. 5. AUC measures the ability of a model to distinguish between classes based on the size of the area under the ROC curve. Cohen's Kappa Score quantifies inter-rater agreement, adjusting for chance. It's calculated using Eq. 6, with P_0 indicating observed agreement, and P_e representing expected agreement by chance.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 - Score = \frac{2xPrecisionxRecall}{Precision + Recall} \tag{5}$$

$$\mathrm{K} = \frac{P_0 - P_e}{1 - P_e} \tag{6}$$

The calculation of the metrics includes TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative), where TP represents instances correctly predicted as positive by the model, TN indicates instances correctly predicted as negative, FP refers to instances incorrectly predicted as positive, and FN represents instances incorrectly predicted as negative. The study evaluated the reliability and generalizability of the model on the dataset using the k-fold cross-validation method. The k-fold cross-validation in this study was set to $k = 5$, which is a common practice in the literature [42, 43].

Experimental results

Pre-trained CNN models

The mean and standard value performance metrics of the light-weight CNN models used in the study are presented in Table 7. The confusion matrices of all CNN models at the end of the testing process are given in Fig. 8.

When the results in Table 7 are analyzed, high performances were achieved overall. CNN models achieved an accuracy rate between 95.11% and 99.20%. Standard deviation values show that the values obtained are consistent. NASNetMobile was the most successful model with an accuracy of 99.20%, precision of 99.21%, recall of 99.20%, f1-score of 99.19%, kappa score of 98.92% and

Table 7 Classification performance of CNN models

Model	Classification metrics (%)					
	Accuracy	Precision	Recall	F1-Score	Kappa	AUC
DenseNet121	99.16 (±0.0060)	9917 (±0.0059)	99.16 (±0.0060)	99.16 (±0.0060)	98.88 (±0.0080)	99.99 (±0.0001)
EfficientNetB0	98.78 (±0.0038)	98.79 (±0.0037)	98.78 (±0.0038)	98.78 (±0.0038)	98.36 (±0.0051)	99.91 (±0.0015)
MobileNet	98.01 (±0.0147)	98.14 (±0.0130)	98.01 (±0.0147)	98.02 (±0.0147)	99.95 (±0.0006)	97.33 (±0.0198)
MobileNetV2	95.11 (±0.0690)	96.46 (±0.0425)	95.11 (±0.0690)	95.21 (±0.0670)	93.35 (±0.0943)	99.76 (±0.0035)
NASNetMobile	**99.20 (±0.0036)**	**99.21 (±0.0035)**	**99.20 (±0.0036)**	**99.19(±0.0036)**	**98.92 (±0.0048)**	**99.99 (0.0001)**

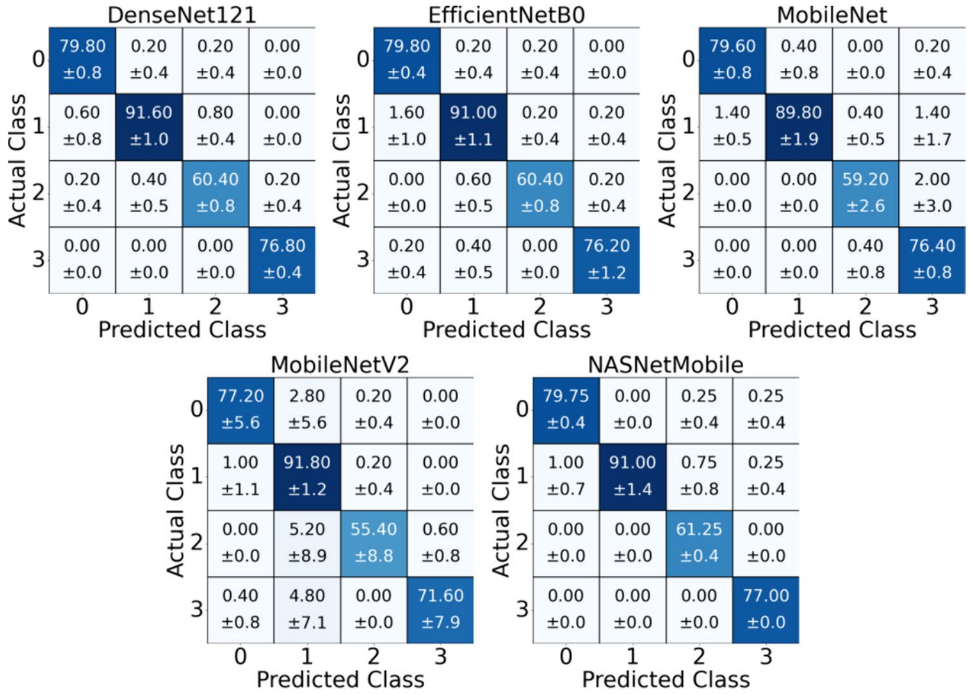


Fig. 8 Confusion Matrices of CNN models (0: Ak, 1: Kapadokya, 2: Nurlu, 3: Sıra)

AUC of 99.99. NASNetMobile achieved high performance with the advantages of blocks such as separable convolution and pooling improved with reinforcement learning. DenseNet121 was the second most successful model with 99.16% accuracy. MobileNetV2 had the lowest accuracy rate with 95.11%.

When Fig. 8 is analyzed, the DenseNet121, EfficientNetB0, MobileNet and NASNetMobile models had low standard deviation values and minimal incorrect predictions in distinguishing almond species. However, the MobileNetV2 model had higher standard deviation values and made more errors in identifying the almond varieties. Figure 9 shows the mini-batch accuracy and mini-batch loss values of the models during training. There is no significant difference between the accuracy and loss values during the training and validation process using the training and validation data. This indicates that the model does not overfit and memorize the training data. From this perspective, it can be assumed that the model has better generalization ability on unseen data.

Proposed genetic CNN

The details of the hyperparameters for the optimal solution are shown in Table 8. The first convolutional layer uses 3×3 size filters, which contain 9 filters in this layer. The ReLU activation function is chosen for the same layer. The pooling method is average pooling with a 2×2 pooling window. In the second convolutional layer, 3×3 size filters are used, but the number of filters is increased to 19. The ReLU activation function is also used in this layer. The pooling type is automatically set to maximum pooling, and the pooling size is set to 2×2. In the third convolutional layer, 5×5 filters are determined, with 32 filters in this layer. The more advanced ELU (Exponential Linear Unit) activation function is selected. Maximum pooling is also chosen in this layer,

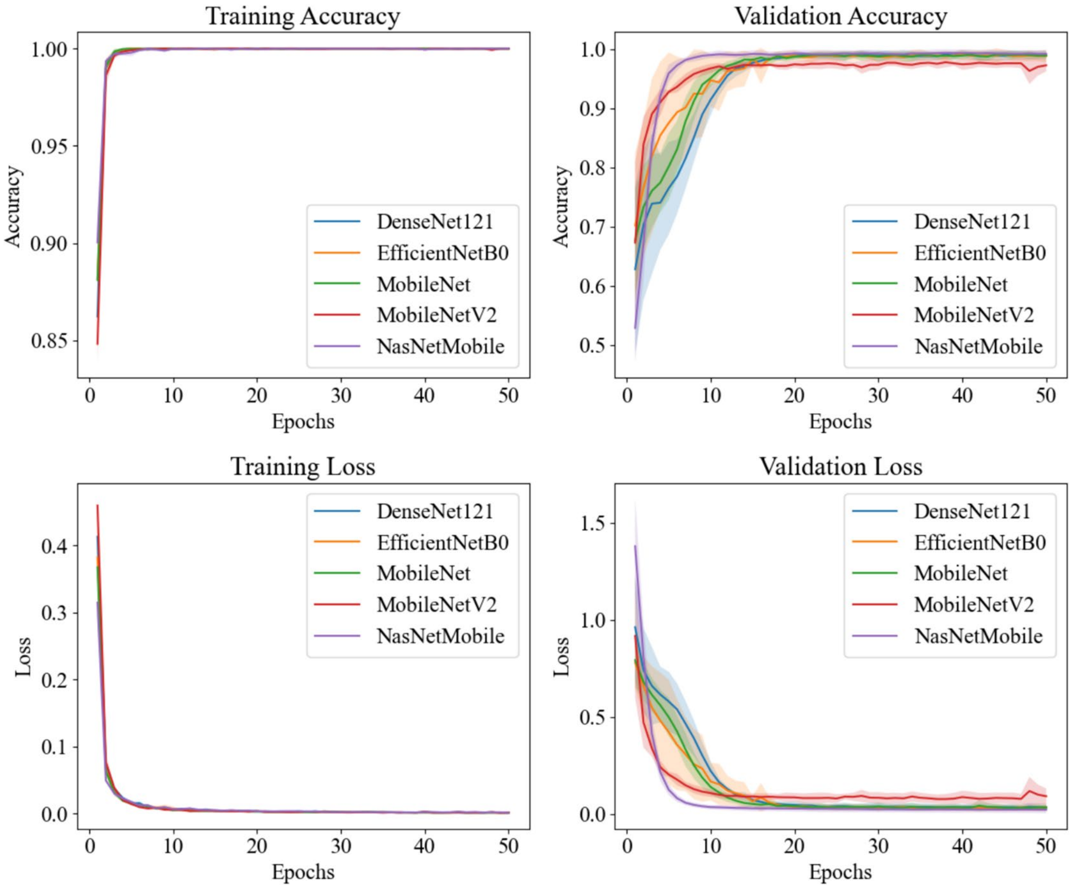


Fig 9 The mini-batch accuracy and loss graphs of the CNN models

Table 8 CNN architecture hyperparameters

FS1	NoF1	A1	PT1	PS1	FS2	NoF2	A2	PT2	PS2	FS3	NoF3	A3	PT3	PS3	NC
1	9	0	1	0	1	19	0	0	1	2	32	3	0	1	59

Table 9 Classification performance of genetic CNN

Model	Classification metrics (%)					
	Accuracy	Precision	Recall	F1-Score	Kappa	AUC
Genetic CNN	99.55 (±0.0044)	99.56 (±0.0043)	99.55 (±0.0044)	99.55 (±0.0044)	99.40 (±0.0059)	99.99 (±0.0001)

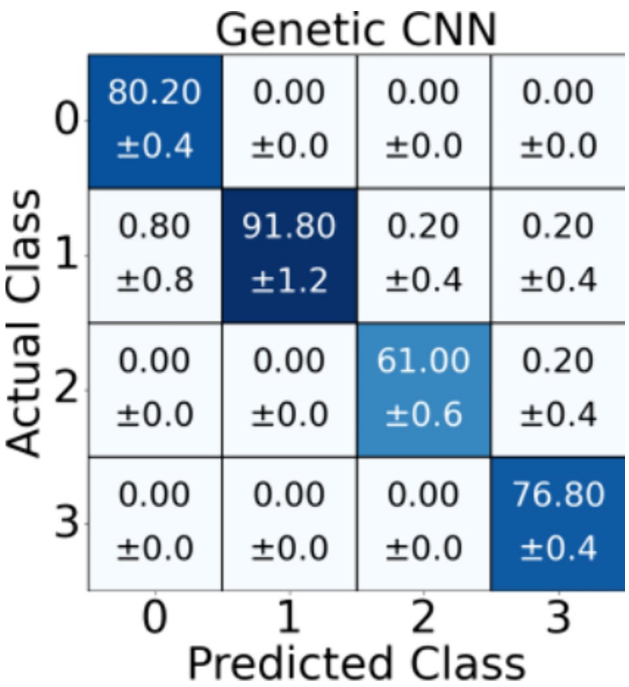


Fig. 10 Confusion matrix of Genetic CNN (0: Ak, 1: Kapadokya, 2: Nurlu, 3: Sıra)

with a pooling size of 3×3. Finally, a total of 59 neurons in the dense layer are also determined by the GA.

The classification performance of the Genetic CNN model is presented in Table 9 and the complexity matrix is presented in Fig. 9. The Genetic CNN has a very high performance with an accuracy rate of 99.55%. Also, the low standard deviation values indicate that the results are consistent.

The confusion matrix in Fig. 10 visually represents the performance of Genetic CNN in classifying the varieties of almonds. The high rates on the main diagonal for each variety reflect the classification success and prediction reliability of the model, while the low standard deviation values indicate the consistency and stability of the model across experiments. When Fig. 10 is analyzed, it is seen that Kapadokya is correctly predicted with ± 1.2 standard deviation. Although the standard deviation is quite low, it shows that it is more difficult for the model to distinguish Kapadokya from other varieties. It was also confused with other classes at minimal rates. The main reason for this is that Kapadokya is similar in pattern to Ak and similar in shape to Nurlu and Sıra.

Figure 11 shows the mean and standard deviation of the mini-batch accuracy and loss values of the Genetic CNN. When the training and validation accuracies are examined, the difference between the values is quite minimal. Similarly, the difference between training and validation loss values is quite minimal. This indicates that the model does not overfit and does not memorize the training dataset. From this point of view, it can be thought that the model has a good generalization ability on unseen data.

With an accuracy of 99.55%, precision of 99.56%, recall of 99.55%, f1-score of 99.55, kappa score of 99.40% and AUC of 99.99, Genetic CNN outperformed the well-known CNN models in terms of classification performance. In embedded systems, factors such as the number of parameters, size and estimation time are also important. Table 10 shows the comparison between pre-trained CNN models and Genetic CNN in terms of the parameter count, size and average testing time metrics.

Fig. 11 The mini-batch accuracy and loss graphs of the Genetic CNN

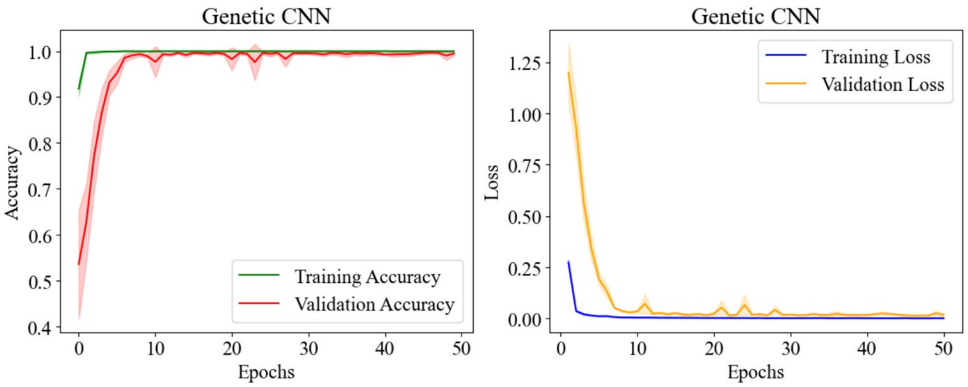


Table 10 Details of pre-trained and genetic CNN models

Model	Parameter count (M)	Size (Mb)	Avg. testing time (s)
DenseNet121	8.1	33	1.90 (±0.08)
EfficientNetB0	5.3	29	3.86 (±0.07)
Genetic CNN	**1.1**	**5.8**	**0.94 (±0.05)**
MobileNet	4.3	16	5.23 (±0.05)
MobileNet V2	3.5	16	9.33 (±0.07)
NASNetMobile	5.3	23	1.59 (±0.06)

The Genetic CNN, with 1.1 million parameters and a size of 5.8 Mb, has a lower parameter count and size compared to other models and achieves the shortest test time of 0.94 s (± 0.05). These characteristics indicate that the Genetic CNN is suitable for almond classification tasks, especially in embedded devices or mobile devices during quality control stages for almond variety detection.

Comparison with literature studies

The Genetic CNN outperformed other light-weight models in terms of parameters, size, test time and classification performance. Furthermore, the performance of the Genetic CNN was also compared with similar studies in the literature in the field of almond classification. A summary of related work in the literature is presented in Table 11.

In the literature, various classifiers such as SVM, Chi-SVM, Decision Tree, AlexNet and ResNet50 were used and accuracy rates between 91% and 98.4% were achieved. However, the Genetic CNN model proposed in this study outperformed traditional feature extraction-based classifiers with its automatic feature extraction architecture. It also outperformed other CNN-based studies in the literature in terms of both parameter size and classification performance.

Discussion

Almonds have a significant role in agriculture and food sectors due to its high nutritional value and wide industrial utilization. Accurate classification of almonds is critical for both quality control and determination of the market value. However, traditional classification methods of almond cultivars are based on morphological characteristics and genetic analysis. These methods are both time-consuming and error-prone, which creates limitations in terms of efficiency and accuracy. In this study, a dataset was created using images of four different almond cultivars in Turkey. The aim of the study is to develop a model that provides faster and more accurate results than traditional methods. In this context, five well-known CNN models (DenseNet121, EfficientNetB0, MobileNet, MobileNetV2, NASNetMobile) were tested for almond classification. The models were evaluated in terms of classification accuracy and efficiency. In this context, the highest classification performance was achieved with NASNetMobile with an accuracy of 99.20%, precision of 99.21%, recall of 99.20%, f1-score of 99.19%, kappa score of 98.92% and AUC of 99.99. A new model called Genetic CNN was also proposed. The Genetic CNN outperformed well-known and pre-trained CNN models by achieving an accuracy of 99.55%, precision of 99.56%, recall of 99.55%, f1-score 99.55%, kappa score 99.40% and AUC of 99.99. The box plot diagram of the results obtained on the test data as a result of the experimental studies is shown in Fig. 12. When Fig. 12 is analyzed, it can be seen that all models are successful in all metrics with a value close to 100. It is also seen that the standard deviation of all the performance metric values is very low in all models. As a result, it is seen that the models have statistical stability.

In addition to classification performances, the Genetic CNN outperformed other well-known models in terms of the number of parameters (1.1 million), model size (5.8) and test time (0.94). These factors are important indicators of how efficiently the model works in practical applications. In this context, Genetic CNN is ideal for real-time

Table 11 Comparison of genetic CNN with similar studies

	Author(s) and year	Classification metrics(%)							
References		Classifier	Accuracy	Precision	Recall	F1 Score	Kappa	AUC
[8]	Halac et al. (2017)	SVM	92.7	–	–	–	–	–
[9]	Nasirahmadi et al. (2017)	Chi-SVM	91	–	–	–	–	–
[10]	Narendra et al. (2021)	Decision Tree	97.1	–	–	–	–	–
[11]	Benarous (2020)	AlexNet	89	–	–	–	–	–
[11]	Benarous (2020)	AlexNet V2	94	–	–	–	–	–
[12]	Vidyarthi et al. (2021)	ResNet50	98.4	96.0	96.0	96.0	–	–
–	**Yurdakul et al. (2024)**	**Genetic CNN**	**99.5**	**99.5**	**99.2**	**99.1**	**98.9**	**99.9**

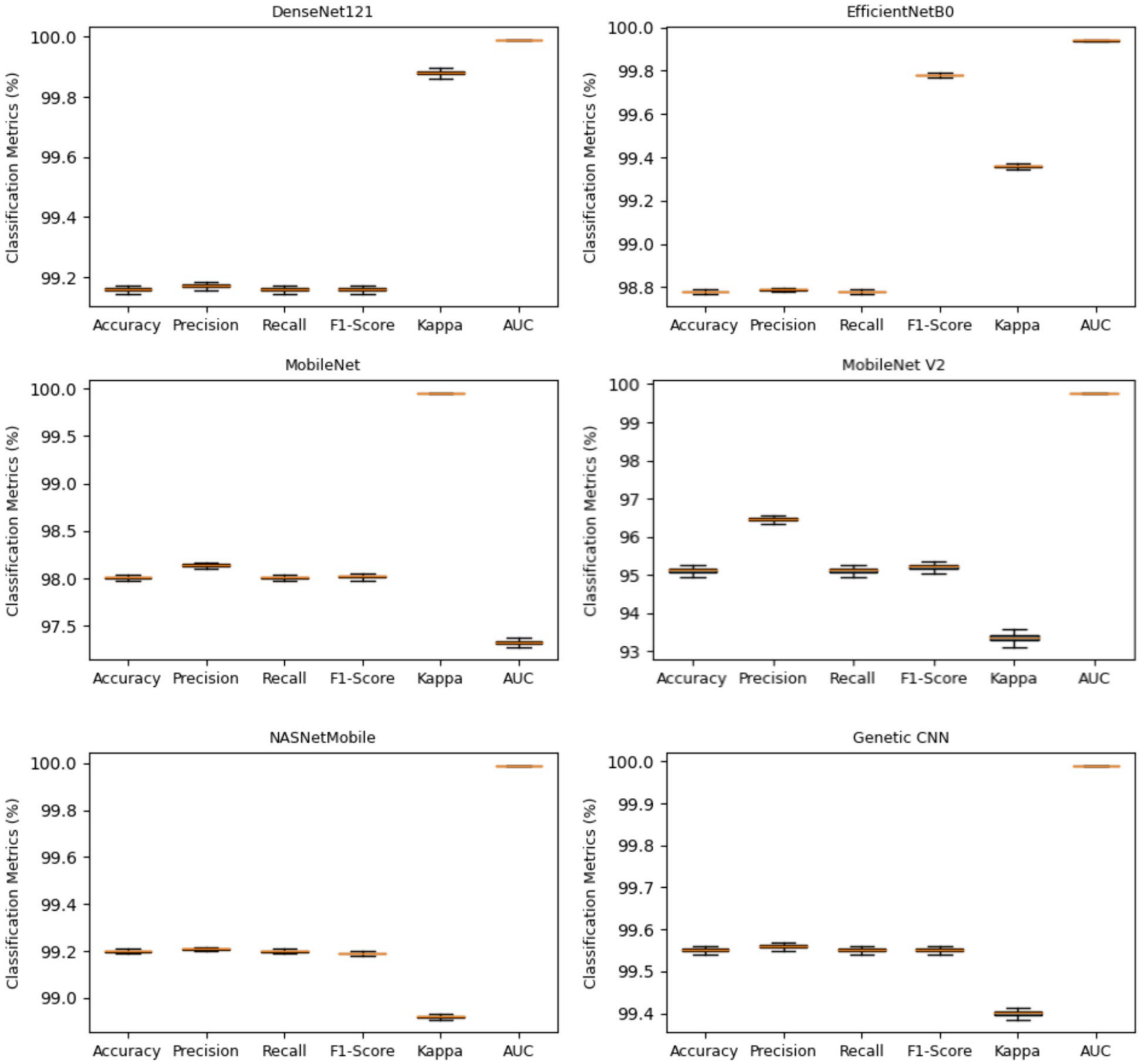


Fig. 12 Box plot diagrams illustrating the classification metrics of cnn models

applications and embedded systems. When compared with other studies on almond classification in the literature, it is seen that the Genetic CNN model is more successful than other studies (Table 10). Furthermore, Genetic CNN was compared with the CNNs designed using GA in the literature; Table 12 shows the literature studies.

When Table 12 is analyzed, it is seen that genetic-based CNNs achieve successful results. It is seen that there is an increasing trend in model performances over the years. The Genetic CNN has surpassed related studies in the literature in terms of classification performance, contributing to the trend.

Conclusion

Almonds are a nutritious food containing a variety of essential nutrients. In addition to being a nutritious food, it is also important in various industrial applications. These characteristics have made almonds a sought-after product in international export markets. Classifying almond varieties is vital for quality assessment and determining their market value. In this study, images of four different almond varieties were collected from Muğla and Nevşehir provinces of Turkey. The images were classified using

Table 12 Comparison of genetic CNN with other genetic-based CNN studies

References	Author(s) and year	Accuracy	Precision	Recall	F1 Score	Kappa	AUC
[34]	Rikhtegar et al. (2016)	94.6	–	–	–	–	–
[35]	Sun et al. (2020)	96.7	–	–	–	–	–
[36]	Dangi et al. (2022)	96.9	–	–	–	–	–
[20]	Das et al. (2022)	98.6	–	–	–	–	99.3
–	Yurdakul et al. (2024)	99.5	99.5	99.2	99.1	98.9	99.9

well-known and light-weight CNN models. Additionally, a new CNN model called Genetic CNN was developed and the classification process was performed. Among the well-known CNN models, the highest performance was obtained with NASNetMobile with an accuracy of 99.20%, precision of 99.21%, recall of 99.20%, f1-score of 99.19%, kappa score of 98.92% and AUC of 99.99%. The Genetic CNN achieved an accuracy of 99.55%, precision of 99.56%, recall of 99.55%, f1-score 99.55%, kappa score of 99.40% and AUC of 99.99. The Genetic CNN model has fewer parameters and size compared to other models and short testing time. In future studies, we will increase the number of almond varieties, design CNN architectures with various optimization models, develop machine-based classification systems, and develop mobile application-based classification systems.

Author contributions Mustafa Yurdakul: Conceptualization, Methodology, Review & editing, Software, Validation, Visualization, Writing, İrfan Atabaş: Conceptualization, Data Collection, Software, Visualization, System Setup, Şakir Taşdemir: Conceptualization, Methodology, Supervision, Review & Editing,

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Data availability The dataset prepared in the study can be accessed via the following link. https://github.com/ymyurdakul/datasets/blob/main/almond/archive.zip

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.

Compliance with ethical requirements The study does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication Not applicable.

References

1. Zeinalabedini M, Khayam-Nekoui M, Grigorian V, Gradziel T, Martínez-Gómez P (2010) The origin and dissemination of the cultivated almond as determined by nuclear and chloroplast SSR marker analysis. Sci Hortic 125(4):593–601
2. Sze-Tao K, Sathe S (2000) Functional properties and in vitro digestibility of almond (*Prunus dulcis* L.) protein isolate. Food Chem 69(2):153–160
3. Barreca D et al (2020) Almonds (*Prunus dulcis* Mill. DA webb): A source of nutrients and health-promoting compounds. Nutrients 12(3):672
4. Banjanin T et al (2021) Physicochemical properties, fatty acids, phenolic compounds, and mineral contents of 12 Serbia regional and commercial almond cultivars. J Food Process Preserv 45(1):e15015
5. Ghafarzadeh J, Sadeghniiat-Haghighi K, Sadeghpour O, Akbarpour S, Amini-Behbahani F (2019) Investigating the prevalence of sleep disorder and the impact of sweet almond on the quality of sleep in students of Tehran, Iran. Iran J Public Health 48(6):1149
6. Sumner DA, Matthews WA, Medellín-Azuara J, Bradley A (2014). The economic impacts of the California almond industry. In A report prepared for the Almond Board of California
7. Teimouri N, Omid M, Mollazade K, Rajabipour A (2016) An artificial neural network-based method to identify five classes of almond according to visual features. J Food Process Eng 39(6):625–635
8. Halac, D. Sokic E, Turajlic E (2017). Almonds classification using supervised learning methods. In 2017 XXVI International Conference on Information, Communication and Automation Technologies (ICAT), IEEE, pp 1–6.
9. Nasirahmadi A, Ashtiani S-HM (2017) Bag-of-feature model for sweet and bitter almond classification. Biosys Eng 156:51–60
10. Narendra V, Krishanamoorthi M, Shivaprasad G, Amitkumar V, Kamath P (2021) Almond kernel variety identification and classification using decision tree. J Eng Sci Technol 16(5):3923–3942
11. Benarous AOF (2020) Almond variety detection using deep learning (Doctoral dissertation)
12. Vidyarthi SK, Singh SK, Xiao HW, Tiwari R (2021) Deep learnt grading of almond kernels. J Food Process Eng 44(4):e13662
13. Koklu M, Ozkan IA (2020) Multiclass classification of dry beans using computer vision and machine learning techniques. Comput Electron Agric 174:105507
14. Gencturk B. et al (2023). Detection of hazelnut varieties and development of mobile application with CNN data fusion

feature reduction-based models. In European food research and technology, pp 1–14.

15. Butuner R, Cinar I, Taspinar YS, Kursun R, Calp MH, Koklu M (2023) Classification of deep image features of lentil varieties with machine learning techniques. Eur Food Res Technol 249(5):1303–1316
16. Avuçlu E, Taşdemir Ş, Köklü M (2023) A new hybrid model for classification of corn using morphological properties. Eur Food Res Technol 249(3):835–847
17. Sivanandam S, Deepa S, Sivanandam S, Deepa S (2008) Genetic algorithms. Springer, Berlin
18. Gen M, Lin L (2023) Genetic algorithms and their applications. In: Pham H (ed) Springer handbook of engineering statistics. Springer, Berlin, pp 635–674
19. Rodrigues LF, Backes AR, Travençolo BAN, de Oliveira GMB (2022) Optimizing a deep residual neural network with genetic algorithm for acute lymphoblastic leukemia classification. J Digit Imaging 35(3):623–637
20. Das S, Saha SK (2022) Diabetic retinopathy detection and classification using CNN tuned by genetic algorithm. Multi Tools Appl 81(6):8007–8020
21. Ayan E (2023) Genetic algorithm-based hyperparameter optimization for convolutional neural networks in the classification of crop pests. Arabian J Sci Eng 49:1–15
22. Keras applications [Online]. Retrieved May 14, 2024, from https://keras.io/api/applications/
23. Rani D, Jain SK, Srivastava DK, Perumal M (2013) Genetic algorithms and their applications to water resources systems. Metaheuris Water Geotech Trans Eng. https://doi.org/10.1016/B978-0-12-398296-4.00003-9
24. Alhijawi B, Awajan A (2023) Genetic algorithms: theory, genetic operators, solutions, and applications. Evol Intell. https://doi.org/10.1007/s12065-023-00822-6
25. Razali NM, Geraghty J (2011). Genetic algorithm performance with different selection strategies in solving TSP. In Proceedings of the world congress on engineering, International Association of Engineers Hong Kong, China, 2, pp 1–6.
26. Lu T-C (2021) CNN Convolutional layer optimisation based on quantum evolutionary algorithm. Connect Sci 33(3):482–494
27. Köpüklü O, Babaee M, Hörmann S, Rigoll G (2019). Convolutional neural networks with layer reuse" In 2019 IEEE international conference on image processing (ICIP), IEEE, pp 345–349.
28. Khanday OM, Dadvandipour S, Lone MA (2021) Effect of filter sizes on image classification in CNN: a case study on CFIR10 and fashion-MNIST datasets. IAES Int J Artificial Intell 10(4):872
29. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv:1602.07360.
30. Kuriakose S, Shunmugam M (2005) Multi-objective optimization of wire-electro discharge machining process by non-dominated sorting genetic algorithm. J Mater Process Technol 170(1–2):133–141
31. Kim IY, De Weck O (2005) Variable chromosome length genetic algorithm for progressive refinement in topology optimization. Struct Multidiscip Optim 29:445–456
32. Kanan HR, Faez K, Taheri SM (2007). Feature selection using ant colony optimization (ACO): a new method and comparative study in the application of face recognition system. In Advances in Data Mining. Theoretical Aspects and Applications: 7th Industrial Conference, ICDM 2007, Leipzig, Germany, July 14–18, 2007. Proceedings 7, Springer, pp 63–76.
33. Lipowski A, Lipowska D (2012) Roulette-wheel selection via stochastic acceptance. Physica A 391(6):2193–2196
34. Rikhtegar A, Pooyan M, Manzuri-Shalmani MT (2016) Genetic algorithm-optimised structure of convolutional neural network for face recognition applications. IET Comput Vis 10(6):559–566
35. Sun Y, Xue B, Zhang M, Yen GG, Lv J (2020) Automatically designing CNN architectures using the genetic algorithm for image classification. IEEE Trans Cybern 50(9):3840–3854
36. Dangi D, Bhagat A, Dixit DK (2022) Sentiment analysis on social media using genetic algorithm with CNN. Comput Mater Continua 70(3):5399–5419

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