Gümüşhane University Journal of Science

GUFBD / *GUJS* (2024) 14(4): 1247-1258 doi: 10.17714/gumusfenbil.1498303 Research Article

Guava fruit classification system design with convolutional neural networks

Evrişimsel sinir ağları ile guava meyvesi sınıflandırma sistemi tasarımı

Buket TOPTAŞ*[1], Sara ALTUN GÜVEN[2]

[1]*Bandırma Onyedi Eylül University, Engineering and Natural Science Facult, Software Eng. Dept, Bandırma/Balıkesir*
[2]*Tarsus University, Engineering Faculty, Computer Eng, Tarsus/Mersin*

• Received: 09.06.2024 • Accepted: 29.11.2024

Abstract
For the rapid and precise advancement of agriculture, artificial intelligence applications are of significant importance. Processes such as disease detection in the agricultural field, identification of soil types, and classification of plants and fruits are currently performed manually. Artificial intelligence enables the automation of these processes, leading to cost reduction and the minimization of human errors. In this study, a system for classifying the species of Guava fruit has been proposed. The proposed system is designed using four pre-trained convolutional neural networks. The convolutional neural networks used are GoogLeNet, Vgg19, ResNet50, and DenseNet201 architectures. The Guava fruit dataset was classified by both k-fold-stratified and an 80:20 split. All experimental studies were evaluated using six different performance metrics. The best result was achieved with the DenseNet201 architecture in the proposed method. The performance results for the DenseNet201 architecture in terms of accuracy, sensitivity, specificity, F1-score, MCC, and kappa are as follows: accuracy - 0.9658, sensitivity - 0.9677, specificity - 0.9954, F1-score - 0.9681, MCC - 0.9640, and Kappa - 0.8268.

Keywords: Convolutional neural network, Classification, Guava classification

Öz
Tarımın hızlı ve hassas bir şekilde ilerlemesi için yapay zeka uygulamaları büyük önem taşımaktadır. Tarım alanında hastalık tespiti, toprak türlerinin belirlenmesi ve bitki ile meyvelerin sınıflandırılması gibi süreçler şu anda manuel olarak gerçekleştirilmektedir. Yapay zeka, bu süreçlerin otomasyonunu sağlayarak maliyetleri düşürmekte ve insan hatalarını en aza indirmektedir. Bu çalışmada, Guava meyvesinin türlerini sınıflandıran bir sistem önerilmiştir. Önerilen sistem, dört ön eğitimli evrişimli sinir ağı kullanılarak tasarlanmıştır. Kullanılan evrişimli sinir ağları GoogLeNet, Vgg19, ResNet50 ve DenseNet201 mimarileridir. Guava meyvesi veri seti, hem k-katmanlı stratifiye hem de 80:20 bölme ile sınıflandırılmıştır. Tüm deneysel çalışmalar altı farklı performans metriği kullanılarak değerlendirilmiştir. Önerilen yöntemle en iyi sonuç DenseNet201 mimarisi ile elde edilmiştir. DenseNet201 mimarisinin performans sonuçları şu şekildedir: doğruluk - 0.9658, hassasiyet - 0.9677, özgüllük - 0.9954, F1-puanı - 0.9681, MCC - 0.9640 ve Kappa - 0.8268.

***Anahtar kelimeler**: Evrişimsel sinir ağlar, Sınıflandırma, Guava sınıflandırma*

*Buket TOPTAŞ; btoptas@bandirma.edu.tr

ISSN: 2146-538X e-ISSN: 2146-538X https://dergipark.org.tr/tr/pub/gumusfenbil

1. Introduction

Agriculture is an important industry that faces challenges such as increasing food demand and decreasing agricultural labor. In this industry, data obtained through monitoring, measuring, and analyzing various physical variables and events must be evaluated constantly. Correct evaluations will ensure that precautions can be taken against possible difficulties. Thus, studies have begun on issues such as increasing the efficiency of agriculture, ensuring sustainability, increasing food security, and minimizing environmental impacts. However, it is difficult for studies to both protect the natural ecosystem and provide a sustainable food supply worldwide. Considering the increasing amount of data and the success of modern technology in different fields, deep learning technologies have become an important tool to ensure sustainability in other sectors and in the field of agriculture.

Deep learning has various applications in the agriculture sector and is contributing to the transformation of agriculture through modern technology. Deep learning has proven itself in applications such as disease detection, crop counting, yield prediction, classification, and segmentation in the agricultural sector. The classification and identification of agricultural products is among the first and crucial steps in making agriculture more efficient. Particularly, classification applications are of great importance in identifying rare agricultural products and distinguishing between very similar agricultural products. There is a substantial body of literature on the classification of agricultural data.

Kapila et al. (2021) conducted a study on classification and damage detection of apple fruit. In this study, they used the features taken from the last layer of weighted conventional convolutional neural networks. These features are given to classifiers such as support vector machines (SVM), linear regression, k-nearest neighbor and random forest. Ultimately, they achieved the highest classification success in the SVM classifier trained with features from the ResNet50 network (Kapila et al., 2021). Loddo et al. proposed a study that classifies plant seed data. In their study, they used a convolutional neural network called SeedNet. SeedNet is built as an architecture consisting of six layers, and a maxpooling layer is added before each layer. The overlearning issue has been managed by the usage of maxpooling. Two distinct seed data sets were used in the proposed method's trials, and a comparison of the SeedNet architecture with conventional convolutional networks was given (Loddo et al., 2021). Adige et al. have developed a method for classifying apple fruit. The method employs machine learning algorithms, specifically utilizing SVM and ResNet-50 architecture. The dataset in this method was tested at three different split ratios for training and testing purposes (Adige et al.,2023). Huang et al. (2022) used a deep learning-based approach to analyze soybeans. Their method made use of SNet architecture with Mask R-CNN. Deep Adjustable Convolution is a sophisticated convolution that was used in the construction of the SNet architecture. This design also included an MFR module, which is a particular module. Mixed Feature Recalibration Module is what MFR stands for (Huang et al., 2022) Conversely, (Türkoğlu et al., 2020) used convolutional neural networks (CNN) architecture to classify diseases in apricot fruit. A proposal was made for an eighteen-layer architecture to classify diseases. It was compared how well the suggested approach and conventional convolutional networks classified data. (Doğan et al., 2023) created a system that uses extreme learning machines (ELM) and transfer learning architecture to categorize dry beans. A pre-trained technique was presented by (Singh et al., 2022) for identifying distinct species of pistachios. AlexNet, VGG16, and VGG19 were used to distinguish between two different varieties of pistachios, "siirt" and "red," after the dataset was partitioned in an 80:20 ratio. A transfer learning-based approach was presented by Alsirhani et al. (2023) to categorize the date fruit dataset. In their proposed method, Vgg19, Vgg16, DenseNet121, Inception, ResNet152V2, InceptionResNetV2, DenseNet169, EfficientNetV2M and DenseNet201 architectures were used. The success of the transfer learning architectures is compared with traditional machine learning methods. Ultimately, the date fruit was divided into 27 classes (Alsirhani et al., 2023). In addition to these studies, classification processes have been carried out in various agricultural domains, such as classifying plant diseases (Khan et al., 2022; Chen et al., 2020), classifying grapevine leaves (Koklu et al., 2022), and classifying coffee species (Pinto et al., 2017).

In this article, the species classification of guava fruit harvested in the Larkana district of Pakistan is presented. Guava fruit is a tropical fruit that resembles pear fruit in appearance but tastes like a mixture of pineapple, pear, banana, and grapefruit. Moreover, this fruit contains a significant number of phytochemicals important for health. These compounds have a broad therapeutic spectrum, including the ability to regulate blood sugar and cholesterol levels, and possess antioxidant, antibacterial, anti-inflammatory, antitumor, and anticancer properties. (Jamieson et al.,) These characteristics highlight guava's role not only as a nutritious fruit but also as a potential therapeutic agent. Guava fruit has species such as Local Sindhi, Thadhrami, and Riyali. Each species is divided into Green, Mature Green, and Ripe according to the level of ripeness. Traditional convolutional neural network architectures have been used to classify guava fruit species. A comparison of the traditional CNN architectures has been made and a system has been developed for the classification of guava fruit for industry. The developed model has both commercial and agricultural potential. Commercially, it can create a model that determines the approximate prices of guava fruit. From an agricultural point of view, it can be used to classify guava fruit by informing farmers.

The aim of this study is to classify guava fruit species quickly and efficiently. Within the scope of the study;

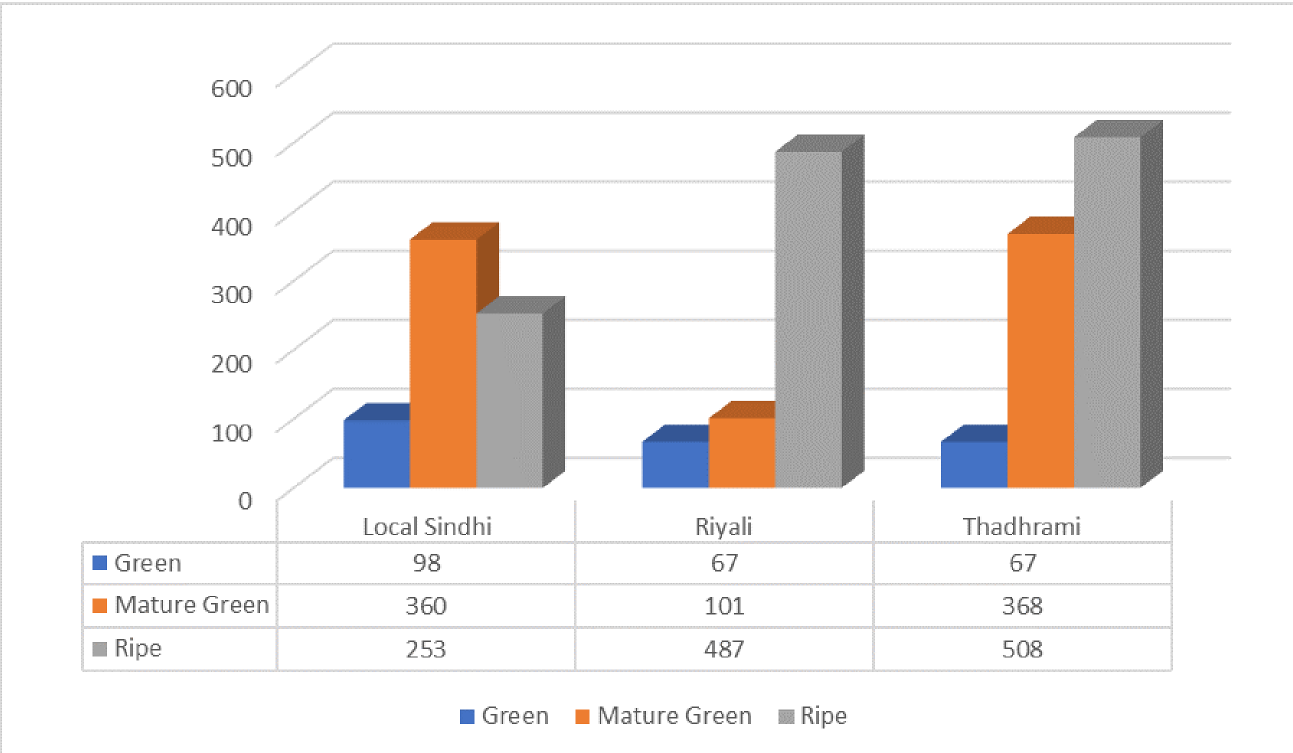
- Classification of guava fruit species realized
- In the classification phase, the most successful model was determined by using different CNN architectures with transfer learning.
- A comprehensive analysis of CNN architectures in Classification with transfer learning was performed.
- A preliminary preparation for future studies was created for the data set used.

The rest of the organization of this paper is as follows: Section 2 is Materials and Methods, which describes the dataset used, the classification architectures used, and the proposed methodology. Section 3 is Experiments and Results, which describes the performance metrics used, the experimental work done, and the results of these studies. The last section is the Discussion and Conclusion section where conclusions and discussions are made

2. Material and method

2.1. Used dataset

The dataset used is a publicly shared dataset from 2023 (Maitlo et al., 2023). The dataset includes three types of guava fruit (Local Sindhi, Thadhrami, and Riyali). Furthermore, each type is categorized into three types according to its ripeness level: Green, Mature Green, and Ripe. In total, there are 2309 images. Figure 1 shows the distribution of the data in the dataset. In this study, three guava fruit species and three ripeness levels are considered for 9 different classes.



	Local Sindhi	Riyali	Thadhrami
Green	98	67	67
Mature Green	360	101	368
Ripe	253	487	508

Figure 1. Details of the used dataset

The article provides a table, Table 1, that includes the labels for the 9 classes in order, along with their meanings. For example, the label "Ls_Green" is the name of the first class, representing the Local Sindhi type Green maturity level.

Table 1. Types and class labels of guava fruits in the dataset

Class	Label name	Types	Maturity level
1.class	Ls_Green	Local Sindhi	Green
2.class	Ls_MatureGreen	Local Sindhi	Mature Green
3.class	Ls_Ripe	Local Sindhi	Ripe
4.class	R_Green	Riyali	Green
5.class	R_MatureGreen	Riyali	Mature Green
6.class	R_Ripe	Riyali	Ripe
7.class	T_Green	Thadhrami	Green
8.class	T_MatureGreen	Thadhrami	Mature Green
9.class	T_Ripe	Thadhrami	Ripe

2.2. The general structure of convolutional neural networks

Convolutional neural networks generally consist of a convolution layer, activation layer, pooling layer, flatten layer, and full-connected layer. The input image is first transmitted to the convolution layer. The convolution layer uses different convolution kernels to obtain different feature maps of the input images. Each kernel generates feature maps by convolving the input data in local regions. The output of this layer is a tensor of these feature maps. Furthermore, each convolution layer is followed by an activation function. These functions help the network to detect non-linear features. Rectified Linear Unit (ReLu), Hyperbolic Tangent Function (Tanh), Softmax and Sigmoid function are some of the commonly used activation functions. The pooling layer aims to achieve translation invariance by reducing the resolution of feature maps. It is usually placed between two convolution layers and summarizes the feature maps. Maximum pooling, average pooling, global average pooling, minimum pooling are some of the pooling layers used. Convolution and pooling layers can form deep architectures to incrementally extract higher level feature representations. This allows the network to extract more abstract and high-level information. With the flattening layer, all features obtained from the deep network architecture are flattened and given as input to the fully connected layer. Finally, one or more fully connected layers form the final results of the network. The last layer of the CNN is the output layer, which produces the final predictions for a specific task (Raiaan et al., 2024). The structure of a traditional convolutional network is visually represented in Figure 2.

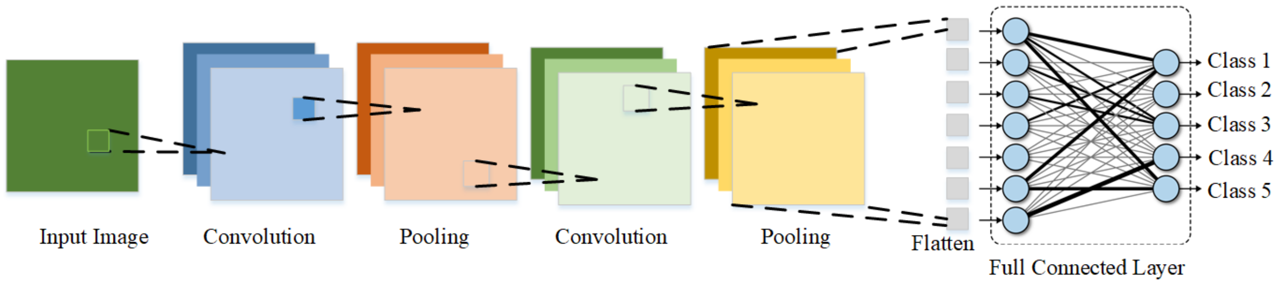


Figure 2. Traditional Convolutional Neural Networks

2.3. Pre-trained convolutional neural networks models

In the agricultural sector, creating images of data is often laborious. When the number of images is small, deep learning approaches cannot train the network sufficiently. To overcome this problem, pre-trained network architectures with different datasets are used. This method is known as transfer learning (TL). With transfer learning, the training time is reduced and features are not extracted repeatedly. In this paper, pre-trained network models are used using millions of images from the ImageNet dataset. These network models directly take input images of Guava fruit and perform classification by extracting high-level features. Here, the 1000 class of the network in the ImageNet architecture is set to 9, considering the 9 species of the Guava fruit.

The pre-trained CNN models used in this study include GoogLeNet, Visual geometry group - 19 Layers (Vgg19), residual network - 50 Layers (ResNet50), and densely connected convolutional networks (DenseNet). GoogLeNet is a CNN architecture created by Google in 2014. This architecture used the "Inception" structure, which is different from the traditional CNN architecture. Inception extracts features from input with convolutional kernels of different sizes and combines these features to pass them to the next layer of the network. Thus, more features are extracted and higher accuracy is achieved. A total of 9 Inception modules are used in the network architecture. The input image size of the network is 224x224x3 pixels. GoogLeNet architecture has proven its success in different classification problems (Chen et al., 2023; Assari et al., 2022). The Vgg19 architecture is a traditional convolutional neural network model proposed in 2014 by (Simonyan et al., 2014). The architecture, which has 19 layers in total, uses the first 16 levels to extract features and the final three layers to do classification. A pooling layer is utilized after each of the five blocks made up of the sixteen layers used for feature extraction. The ImageNet dataset is used to pre-train the architecture. The size of the final layer is set to 1000 in order to categorize the samples in this dataset into 1000 distinct groups. 224x224x3 pixels make up the input picture, and the kernel size is 3x3. A 50-layer design called the ResNet50 architecture was suggested in 2016 (He et al., 2016) ResNet design offers an answer to the difficult task of training very deep networks, which gets more difficult with depth increases. The challenge of training very deep networks—which gets more difficult as depth increases and vanishing gradients arises—is addressed by the ResNet design. This design has a connection type known as a "skip connection," which combines the block's input and output at the conclusion of each convolutional block. The input image size for the network is 224x224x3 pixels. The DenseNet201 architecture is a CNN network first introduced in 2017 by (Huang et al., 2017). The architecture consists of 5 layers and the layers contain Dense Block. These Dense Blocks serve the purpose of merging the feature maps of consecutive layers. As you progress between these blocks, the filters change. This increases the architecture of the deep learning network. The input image size of the DenseNet201 architecture is 224x224x3 pixels. The general structure of these pre-trained CNN architectures is presented in Table 2.

Table 2. Architectural Structure of the Used CNN Networks

GoogLeNet	**Vgg19**	**ResNet50**	**DenseNet201**
Convolution	Convolution	Zero Padding	Convolution
Pooling	Convolution	Convolution	Pooling
Convolution	Pooling	Batch Normalization	Dense Block1
Pooling	Convolution	ReLU	Convolution
Inception (3a)	Convolution	Max pool	Pooling
Inception (3b)	Pooling	Conv Block	Dense Block2
Pooling	Convolution	ID Block x2	Convolution
Inception (4a)	Convolution	Conv Block	Pooling
Inception (4b)	Convolution	ID Block x3	Dense Block3
Inception (4c)	Convolution	Conv Block	Convolution
Inception (4d)	Pooling	ID Block x4	Pooling
Inception (4e)	Convolution	Conv Block	Dense Block4
Pooling	Convolution	ID Block x5	Global Avg. Pool
Inception (5a)	Convolution	ReLU	Softmax
Inception (5b)	Convolution	Avg. pool	
Avg. pool	Pooling	Flattening	
Dropout (40%)	Convolution	Fc	
Linear	Convolution	Softmax	
Softmax	Convolution		
	Convolution		
	Pool		
	Fc6		
	ReLU		
	Fc7		
	ReLU		
	Fc8		
	Softmax		

2.4. Proposed methods

In the last time, CNN architectures have achieved remarkable success both in agriculture and other fields. This has led to the organization of various object recognition, classification, and segmentation competitions and the creation of extensive datasets. In this paper, instead of training the model from scratch, it involves using pre-trained weights of deep architectures in various competitions and extensive datasets. In the proposed method, a transfer learning approach is adopted for the classification of Guava fruit species. First, the dataset was trained on four different pre-trained architectures using k-fold-stratified. Then, the data set was split in a fixed ratio of 80:20 and the performance result of each model was compared. Figure 3 shows the flowchart of the proposed model.

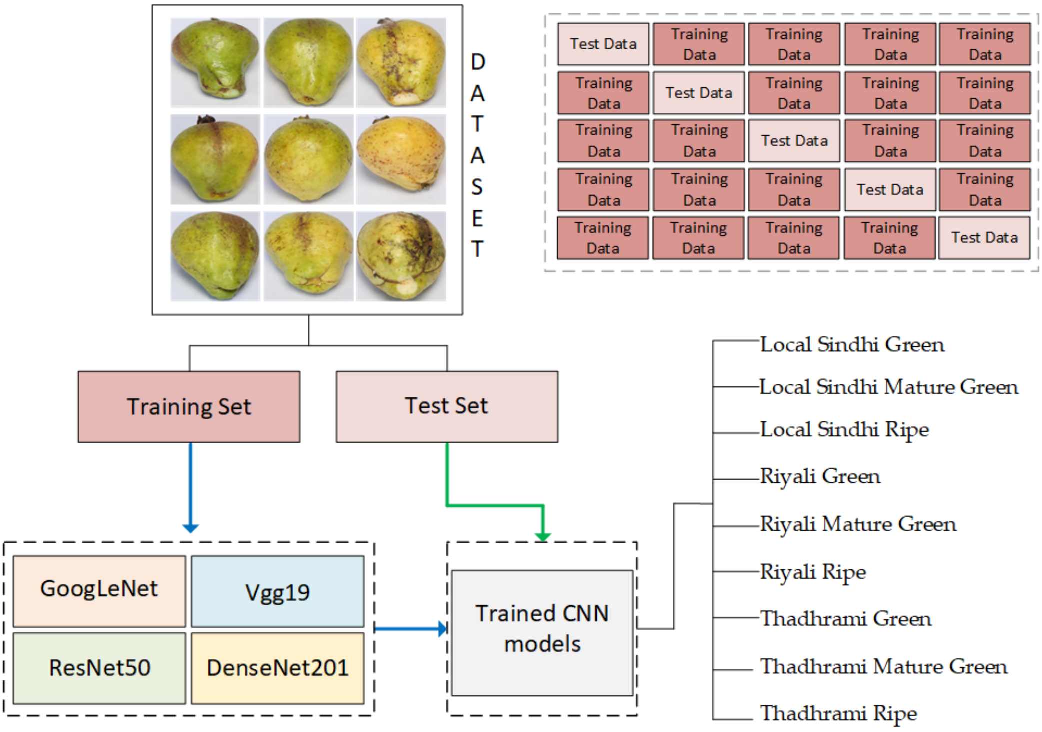


Figure 3. The proposed system's flowchart.

3. Results

This section is divided into three phases. First, the performance metrics used are introduced. Then, the application environment and the parameters used in the experiments are described. In the third phase, the experiments are described and the results are presented.

3.1. Performance metrics

One of the steps in creating a reliable and comparable classification is to evaluate the effectiveness of the proposed classification system. This requires the use of acceptable performance measures (Luque et al., 2019). Accuracy, sensitivity, specificity, F1-score, mcc and kappa measures were used in this study. Table 3 shows the performance metrics used and their mathematical equivalent. The mathematical expressions of the performance metrics are generated from the confusion matrix. A visualization of the confusion matrix is given in Figure 4.

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Figure 4. Confusion matrix

Table 3. Used performance metrics

Metrics	Definition
Accuracy	$Acc = \frac{(TP+TN)}{(TP+TN+FP+FN)}$
Sensitivity	$Sn = \frac{TP}{(TP+FN)}$
Specificity	$Sp = \frac{TN}{(TN+FP)}$
F1-score	$F1 - score = 2.\left(\frac{(Pr \times Sn)}{(Pr+Sn)}\right)$ $Pr = \frac{TP}{(TP+FP)}$
Matthews Correlation Coefficient (MCC)	$MCC = \frac{(TN \times TP)-(FN \times FP)}{\sqrt{(TP+FP)(TP+FN)(TN \times FP)(TN \times FN)}}$
Kappa	$K = \frac{P_{agree} - P_{chance}}{1 - P_{chance}}$ P_{agree} = Proportion of trial in which judges agree P_{chance} = Proportion of trial in which agreement would be expected due to chance

3.2. Application environment and experimental setup

Experimental studies were conducted using the MATLAB environment. A computer with Windows 10 64-bit, powered by an AMD Ryzen 3 CPU (3.10 GHz) and equipped with 32 GB of RAM, was used for all applications.

Table 4 shows the hyper-parameters of all CNN networks used in transfer learning. These parameters were applied throughout the experiment. Furthermore, training and test data were determined using 5-fold- stratified cross validation.

Table 4. Hyper-parameters of Pre-Trained Networks.

Input size	Optimization method	Initial learning rate	Max epochs	Learning rate drop factor	Mini batch size	Shuffle
224x224x3	SGDM	0.001	20	0.9	32	Every-epoch

3.2. Experimental results

The experimental studies were carried out on four different pre-trained convolutional networks. The convolutional networks used are GoogLeNet, Vgg19, ResNet50, and DenseNet201 architectures. First, Guava fruit data was trained and tested on these networks using 5-fold- stratified. The fastest running architecture was ResNet50 and the slowest running architecture was DenseNet201. The working speed of the architectures for each fold value is presented in Table 5. The confusion matrix obtained from each fold architecture is presented in Figure 5. When the dataset is divided using k-fold stratified, each fold value ensures that there is at least one sample representing each class. This approach helps prevent the issue of failure in imbalanced classes, which can occur in the classical k-fold method. When interpreting the confusion matrix in multiple classes, each data is interpreted according to its predicted class (corresponding to these columns) and its actual class (corresponding to these rows). In such an interpretation, the diagonal values represent correctly classified data. Data outside the diagonal represents mislabeled data. The magnitude of the diagonal values is related to how successful the classifier is. For instance, in Figure 5, during the k-1-fold loop, the second class was correctly classified with 100% accuracy by ResNet50 and DenseNet201, while GoogLeNet assigned one label to class three and Vgg19 assigned two labels to class three. Upon detailed examination of Figure 5, it was observed that all models achieved the best performance for classes one and seven in every fold.

The accuracy result of the architectures at each fold value is given in Figure 6. The performance results of the cross-validation are presented in Table 6. According to this table, DenseNet201 has the highest accuracy rate of 96.58%. The other models performed competitively.

Table 5. In each fold, the training speed of the architectures

	GoogLeNet	**Vgg19**	**ResNet50**	**DenseNet201**
K1	7 min. 2 sec.	6 min. 57 sec.	6 min. 56 sec.	80 min. 23 sec.
K2	7 min. 14 sec.	7 min. 16 sec.	7 min. 17 sec.	86 min. 17 sec.
K3	7 min. 13 sec.	7 min. 16 sec.	7 min. 19 sec.	71 min. 15 sec.
K4	7 min. 14 sec.	17 min. 15 sec.	7 min. 17 sec.	66 min. 22 sec.
K5	7 min. 14 sec.	17 min. 59 sec.	7 min. 20 sec.	65 min. 35 sec.

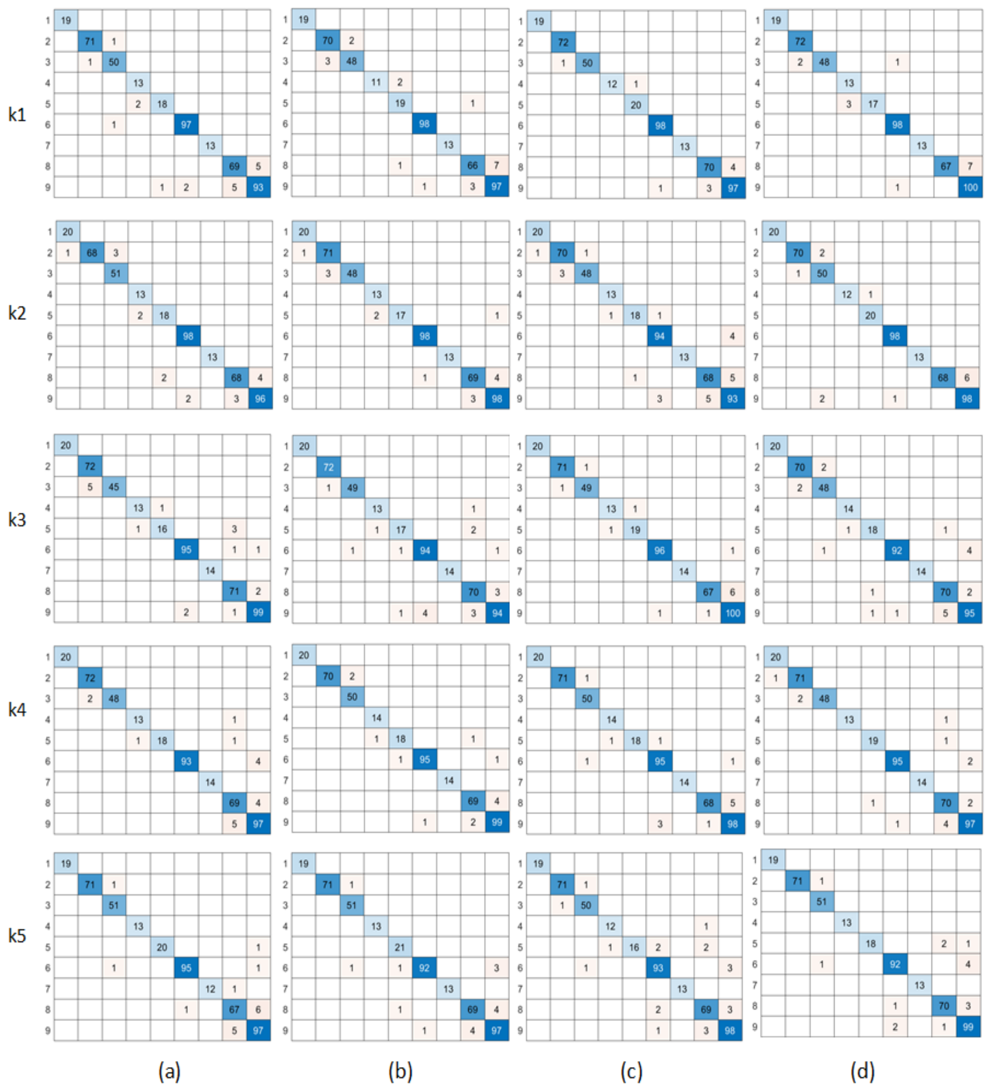


Figure 5. A confusion matrix for each fold. Vertical column true class, horizontal column predicted (a) GoogLeNet (b) Vgg19 (c) ResNet50 and (d) DenseNet201

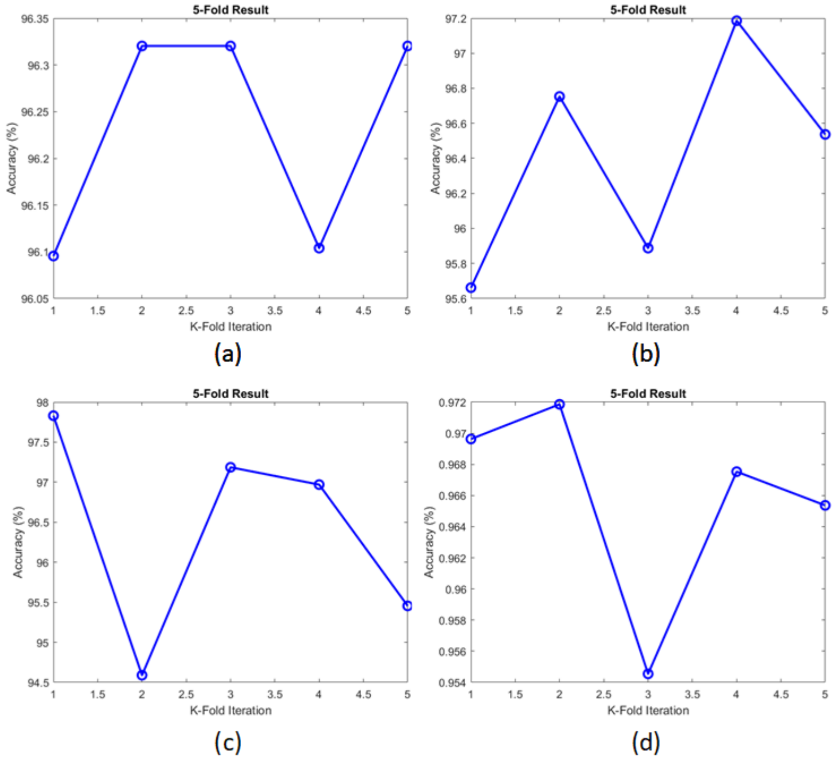


Figure 6. 5-fold results (a) GoogLeNet (b) Vgg19 (c) ResNet50 and (d) DenseNet201

Table 6. 5-fold average results

Pre-Trained CNN	Accuracy	Sensitivity	Specificity	F1-score	MCC	Kappa
GoogLeNet	0.9623	0.9626	0.9950	0.9628	0.9582	0.8092
Vgg19	0.9641	0.9652	0.9952	0.9646	0.9602	0.8180
ResNet50	0.9641	0.9648	0.9951	0.9666	0.9621	0.8181
DenseNet201	0.9658	0.9677	0.9954	0.9681	0.9640	0.8268

Figure 7 shows the confusion matrix we obtained when we split the training data as 80% training and 20% testing instead of k-fold. Here we aimed to measure the response of each model on the same data. Therefore, we randomly split our training and test data into 80% training and 20% test. We then tested this data on each model without changing it at all. Looking at Figure 6, it's evident that the models produced different results when distinguishing classes 6, 8, and 9. In this experiment, the runtimes for the architectures are as follows, in order: GoogLeNet (7 min 19 sec), Vgg19 (8 min 30 sec), ResNet50 (7 min 5 sec), and DenseNet201 (143 min 56 sec). Keeping all conditions including data sets equal, the fastest architecture was the ResNet50 architecture, while the slowest architecture was the DenseNet201 architecture. The speed difference here is related to the depth of the architectures. The ResNet50 architecture has 50 layers while the DenseNet201 architecture has 201 layers.

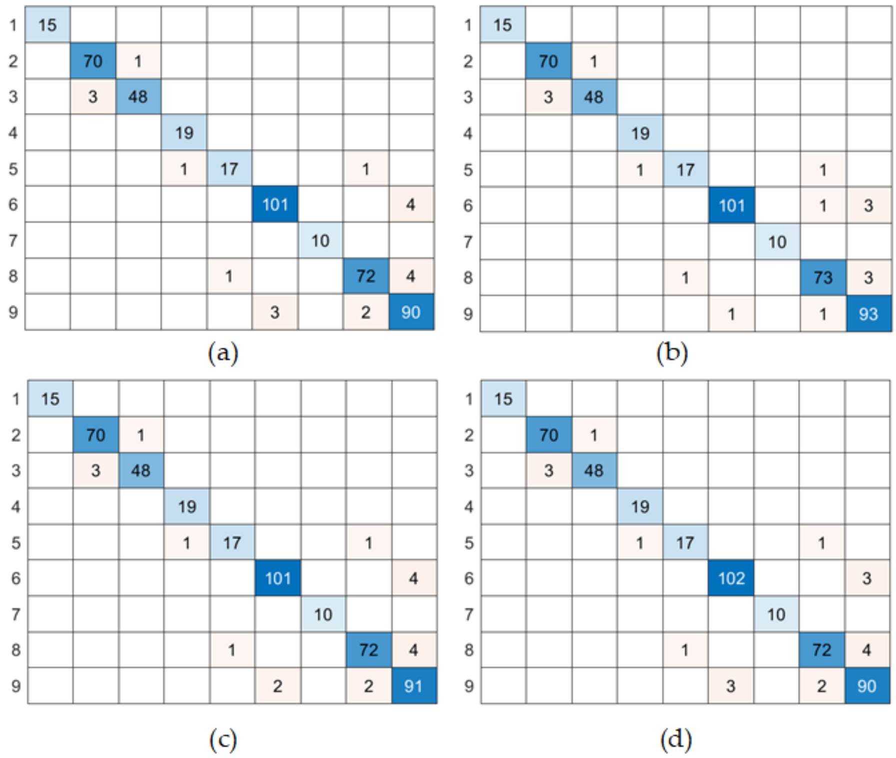


Figure 7. Confusion matrix results in a fixed split dataset. Vertical column true class, horizontal column predicted (a) GoogLeNet (b) Vgg19 (c) ResNet50 and (d) DenseNet201

Table 7. 80 training 20 test average results

Pre-Trained CNN	Accuracy	Sensitivity	Specificity	F1-score	MCC	Kappa
GoogLeNet	0.9567	0.9629	0.9941	0.9635	0.9578	0.781
Vgg19	0.9653	0.9678	0.9953	0.9682	0.9638	0.825
ResNet50	0.9588	0.9640	0.9944	0.9646	0.9593	0.792
DenseNet201	0.9588	0.9639	0.9944	0.9646	0.9592	0.792

4. Discussion and conclusions

This study involves the classification of guava fruit species based on the transfer learning method. There are three different types of guava fruit and each type has three different maturity levels. The dataset of the study contains a total of 2309 data. Pre-trained CNN architectures were used in the classification process. These architectures are GoogLeNet, Vgg19, ResNet50 and DenseNet201. The preferred architectures are those that have proven successful in other classification studies (Islam et al., 2023; Şahin et al., 2023; Toğaçar et al., 2020). The dataset was split by k-fold-stratified method to train and test the network. Here, the value of k is preferred to be 5. Finally, the results of the network architectures were evaluated with six different performance metrics (accuracy, sensitivity, specificity, F1-score MCC and kappa). The highest performance in all metrics was achieved with the DenseNet201 architecture. The performance results for the DenseNet201 architecture in terms of accuracy, sensitivity, specificity, F1-score, MCC, and kappa are as follows: accuracy - 0.9658, sensitivity - 0.9677, specificity - 0.9954, F1-score - 0.9681, MCC - 0.9640, and kappa - 0.8268. These results indicate that the model has a strong performance on the dataset. The MCC metric shows the overall quality of classification success. The obtained rate of - 0.9640 indicates that the model has strong and balanced classification performance. The kappa metric generally represents a measure of the model's consistency and reliability. The obtained rate of 0.8268 is evidence of the model's reliability. Although the GoogLeNet architecture had the lowest performance, it still competed well with the other architectures.

The dataset has been re-split into 80% for training and 20% for testing, and training has been conducted on four models. Here, the dataset was kept constant to ensure fair evaluation of the models, meaning that both training and testing data, as well as training parameters, were the same across experiments. Additionally, all experiments were conducted under equal conditions on a device with the same performance capabilities. The highest accuracy rates, 95.88%, were achieved by the DenseNet201 and ResNet50 architectures. Although the

ResNet50 architecture reached the same level of performance accuracy as DenseNet201, it required less time to train. These results indicate that classification success is not always linearly proportional to the network's architecture. As we delve deeper, more detailed insights are obtained, yet it becomes apparent that not all models consistently perform the same. Additionally, when evaluating the performance of deep learning models, not only the accuracy rate is important, but also factors such as the training duration and applicability of the model.

Despite the strengths of the study, several limitations must be acknowledged. Firstly, the unique dataset used in this study has not been previously utilized by other researchers, which precluded a performance comparison with existing studies. This limitation might affect the generalizability of our findings across different datasets. However, this study's methodology and findings can serve as a benchmark for future research using this dataset. Secondly, while the study presents robust initial results, it did not fully address the issue of class imbalance within the dataset. To improve upon this, there are plans to develop an application that specifically targets this issue within the framework of the proposed method. This development will aim to enhance the performance of the classification system, making it more effective for researchers interested in classifying guava fruit types and potentially other similar applications. By acknowledging and addressing these limitations, we aim to refine our approach and provide a more reliable foundation for future research in this are.

Acknowledgement

None

Author contribution

Buket Toptaş: Conceptualization, Methodology, Software, Writing. Sara ALTUN GÜVEN: Writing – review & editing

Declaration of ethical code

The authors of this article declare that the materials and methods used in this study do not require ethics committee approval and/or legal-special permission

Conflicts of interest

The authors declare that there is no conflict of interest

References

Adige, S., Kurban, R., Durmuş, A., & Karaköse, E. (2023). Classification of apple images using support vector machines and deep residual networks. Neural Computing and Applications, 35(16), 12073-12087.

Alsirhani, A., Siddiqi, M. H., Mostafa, A. M., Ezz, M., & Mahmoud, A. A. (2023). A novel classification model of date fruit dataset using deep transfer learning. *Electronics*, 12(3), 665.

Assari, Z., Mahloojifar, A., & Ahmadinejad, N. (2022). A bimodal BI-RADS-guided GoogLeNet-based CAD system for solid breast masses discrimination using transfer learning. *Computers in Biology and Medicine*, 142, 105160.

Chen, J., Chen, J., Zhang, D., Sun, Y., & Nanehkaran, Y. A. (2020). Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture*, 173, 105393.

Chen, S. H., Wu, Y. L., Pan, C. Y., Lian, L. Y., & Su, Q. C. (2023). Breast ultrasound image classification and physiological assessment based on GoogLeNet. *Journal of Radiation Research and Applied Sciences*, 16(3), 100628.

Dogan, M., Taspinar, Y. S., Cinar, I., Kursun, R., Ozkan, I. A., & Koklu, M. (2023). Dry bean cultivars classification using deep cnn features and salp swarm algorithm based extreme learning machine. *Computers and Electronics in Agriculture*, 204, 107575.

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*, 770-778.

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

Huang, Z., Wang, R., Cao, Y., Zheng, S., Teng, Y., Wang, F., ... & Du, J. (2022). Deep learning-based soybean seed classification. *Computers and Electronics in Agriculture*, 202, 107393.

Islam, M. M., Barua, P., Rahman, M., Ahammed, T., Akter, L., & Uddin, J. (2023). Transfer learning architectures with fine-tuning for brain tumor classification using magnetic resonance imaging. *Healthcare Analytics*, 100270.

Jamieson, S., Wallace, C. E., Das, N., Bhattacharyya, P., & Bishayee, A. (2022). Guava (Psidium guajava L.): a glorious plant with cancer preventive and therapeutic potential. *Critical reviews in food science and nutrition*, 63(2), 192-223.)

Kapila, G., Vandana, B., Khaitan, A., Francis Avinash, A., and Ajay Kumar, C. H., (2021). Apple fruit classification and damage detection using pre-trained deep neural network as feature extractor. *In Innovations in Electronics and Communication Engineering: Proceedings of the 9th ICIECE*, 235-243. Springer Singapore.

Khan, M. A., Alqahtani, A., Khan, A., Alsubai, S., Binbusayyis, A., Ch, M. M. I., ... & Cha, J. (2022). Cucumber leaf diseases recognition using multi-level deep entropy-ELM feature selection. *Applied Sciences*, 12(2), 593.

Koklu, M., Unlersen, M. F., Ozkan, I. A., Aslan, M. F., & Sabanci, K. (2022). A CNN-SVM study based on selected deep features for grapevine leaves classification. *Measurement*, 188, 110425.

Luque, A., Carrasco, A., Martín, A., & de Las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, 91, 216-231.

Loddo, A., Loddo, M., & Di Ruberto, C. (2021). A novel deep learning-based approach for seed image classification and retrieval. *Computers and Electronics in Agriculture*, 187, 106269.

Maitlo, A. K., Aziz, A., Raza, H., & Abbas, N. (2023). A novel dataset of guava fruit for grading and classification. *Data in Brief*, 49.

Raiaan, M. A. K., Sakib, S., Fahad, N. M., Al Mamun, A., Rahman, M. A., Shatabda, S., & Mukta, M. S. H. (2024). A systematic review of hyperparameter optimization techniques in convolutional neural networks. *Decision Analytics Journal*, 100470.

Pinto, C., Furukawa, J., Fukai, H., & Tamura, S. (2017). Classification of green coffee bean images based on defect types using convolutional neural network (CNN*). In 2017 International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA)* 1-5.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv*:1409.1556.

Singh, D., Taspinar, Y. S., Kursun, R., Cinar, I., Koklu, M., Ozkan, I. A., & Lee, H. N. (2022). Classification and analysis of pistachio species with pre-trained deep learning models. *Electronics*, 11(7), 981.

Şahin, N., Alpaslan, N., İlçin, M., & Hanbay, D. (2023). Evrişimsel sinir ağı mimarileri ve öğrenim aktarma ile bitki zararlısı çekirge türlerinin sınıflandırılması. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 35(1), 321-331.

Toğaçar, M., Ergen, B., & Cömert, Z. (2020). Classification of white blood cells using deep features obtained from convolutional neural network models based on the combination of feature selection methods. *Applied Soft Computing*, 97, 106810.

Türkoğlu, M., Hanbay, K., Sivrikaya, I. S., & Hanbay, D. (2020). Classification of apricot diseases by using deep convolution neural network. *BEU Journal of Science*, 9, 334-345.