Journal of Engineering and Sustainable Development
Vol. 30, No. 04, July 2026
https://jeasd.uomustansiriyah.edu.iq/index.php/jeasd
ISSN 2520-0917
Research Article
https://doi.org/10.31272/jeasd.3790

A Comparative Study for Fruit Classification Using Different CNN Machines

Abdulqader F. Abdulqader[1], Yahya T. Qassim[2*]

[1]College of Pharmacy, Alnoor University, Mosul, Iraq.
[2] Department of Medical Devices Engineering Techniques, Alnoor University, Mosul, Iraq
[2]School of Engineering and Built Environment-Electrical and Electronic Engineering, Griffith University, Nathan Campus, Brisbane, Australia

*Email: yahya.taher@alnoor.edu.iq

Article Info

Received 18/11/2025
Revised 12/05/2026
Accepted 12/05/2026

Abstract

The automation of fruit categorization and recognition is gaining popularity, yet it is becoming increasingly challenging due to low contrast and ambiguous features. Nonetheless, autonomous fruit categorization is a complex process reliant on the locations, shapes, colors, and sizes of the objects. In this paper, five Convolutional Neural Network (CNN) models for fruit classification are evaluated to determine the best-performing model. These models are: AlexNet, VGG16, VGG19, ResNet50, and GoogleNet. The architecture layers of the deep learning methods are presented, including convolutional, pooling, and fully connected layers. To evaluate different measures of the model's performance, the confusion matrix is applied. The performance of the CNN models is evaluated on a dataset of 1000 images, with 200 images per category. The results show that VGG16 achieves the best performance among the models used in these investigations. The VGG16 model has achieved an accuracy of 0.901, an error rate of 0.099, and 0.9 for all of (precision, recall, and F-measure, respectively).

Keywords: AlexNet, CNN, Fruit classification, Machine learning, VGG16.

1. Introduction

Most people in the world depend on fruit for survival. Manual human classification has some limitations, such as the effects of human psychological states, inconsistencies in fruit classification, and human visual limitations [1]. Therefore, it is difficult for humans to classify fruits manually [2]. Image classification is popular in many fields, including trade (export and import of fruit) and the enormous fruit industry. Image classification has supported several applications, including image classification, video analysis, and facial recognition. Researchers are using machine learning techniques to help them overcome those difficulties. Machine learning helps build smart agriculture. In this work, five kinds of CNN models are used to classify fruit images. The goal is to calculate the accuracy of each CNN model for classification, i.e., to identify the best model. With the development of 4G communication services and the widespread use of various mobile devices, people have generated a tremendous number of images, videos, audio, and other information, and image classification technology has gradually matured. Image fruit classification has attracted researchers' attention due to its excellent performance and low-cost devices [3]. Every fruit has its own special features, like color, shape, and texture. These features are used to identify every fruit. Sorting fruits at supermarkets manually is time-consuming and costly. The fruit may spoil due to time constraints; misclassification of fruit occurs due to human fatigue. The proposed method is to test these models using the same dataset to select the most efficient deep learning classification model. This research will address an image classification problem in which the number of training samples for each class is small. The dataset contains images of bananas, apples, mangos, grapes, and strawberries. Fruit packing for retail automation could be a good application for the classification results of this paper.

Experimental results explain the performance of every model and investigate which model can improve its performance in the future. This paper is organized as follows: Section two examines the related work. In Sections three and four, the methodology related to the proposed work is discussed. Sections five and six present the performance and investigation of CNN models in fruit image classification. The results and discussion are in Section seven, whereas Section eight concludes this paper.

2. Related Work

Many earlier researchers have researched image processing with fruit images using various methods and techniques. Mahajan and Patil [4] proposed a combined CASVM method for classification; they selected 8 types of fruit and achieved an accuracy of 96.77%. However, nothing was mentioned about evaluation metrics of precision, recall, or F1 measurement in the work. Ghazal et al. [5] proposed an SVM method that uses deep features extracted from the fully connected layer of a CNN. The achieved accuracy was 99-100%; reporting such high accuracy probably indicates overfitting, especially without cross-dataset validation or external test sets [6]. Rojas-Aranda [7] proposed a fruit image classification method using a lightweight CNN to facilitate the checkout process in markets. The classification accuracy was 95% for plastic-bag-free fruits and 93% for fruits in plastic bags. However, the study only considered 3 classes of fruit. Amol et al. [8] developed a system that uses CNNs and image segmentation techniques. Their work achieved an accuracy of 74% because their model exhibited noticeable overfitting due to the small size of the training data; however, this was rectified by expanding the training data, which significantly increased the test accuracy to more than 90%. Hemalatha et al. [9] proposed a combined method of KNN, SVM, and CNN. The achieved accuracy was 95.83%. Ukwuoma [10] conducted a literature survey to help beginners working on fruit classification and detection with the basic concepts of deep learning. Lastly, Rahman et al. [11] used four models: Inception_V3, VGG19, MobileNet, and ResNet50. They achieved the highest accuracy of 99%. However, due to the seasonality of fruit availability, their work included only a limited number of local fruit varieties. Our work, a comparative study of 5 CNN models, seeks to identify the best-performing classification model.

3. Method of the Research

The classification methods are used to classify the input dataset into five classes. There are two steps in each method: training, where the networks are trained from scratch, followed by network testing. Machine learning is mainly used for image classification into correct classes, such as banana, apple, mango, grape, and strawberry. Five approaches have been used in the work investigated: AlexNet, VGG16, VGG19, ResNet50, and GoogleNet.

3.1. Alexnet

The architecture of AlexNet comprises five convolutional layers, with kernel sizes varying by layer, and ReLU as the activation function. The five convolutional layers represent feature extraction for the input image. The output of these layers is fed into the dense Fully Connected (FC) layer. The architecture of AlexNet is available in [12].

3.2. VGG16 Architecture

It is an abbreviation for Visual Geometry Group-16, a machine learning model designed for image classification and recognition. The architecture of VGG16 comprises five convolutional layers, with kernel sizes varying by layer, and ReLU as the activation function. The five convolutional layers are treated as feature-extraction stages. The output of these layers is fed into a dense FC layer. The architecture of VGG16 is depicted in [13].

3.3. VGG19 Architecture

It is an abbreviation for the Visual Geometry Group-19. It is composed of 19 convolutional layers in a deep neural network, has more weight, and is fully connected with max pooling. The architecture of VGG19 comprises five convolutional layers, with kernel sizes varying by layer, and ReLU is its activation function. The output of these layers is fed into dense FC layers. The architecture of VGG19 is available in [14].

3.4. ResNet 50 Architecture

The architecture of ResNet50 comprises five stages, after the zero-padding stage. The first stage consists of one convolutional layer followed by batch normalization, ReLU, and max pool. Each of the remaining layers consists of a convolution block and an ID Block; at the end, there is an Avg Pool, Flattening, and FC, then the output. The output of these stages is fed into a dense FC layer. The architecture of ResNet50 is available in [15].

3.5. GoogleNet Architecture

The architecture of GoogleNet comprises thirteen convolutional layers, with kernel sizes varying across layers and ReLU as the activation function. The thirteen convolutional layers serve as feature-extraction stages. The output of these layers is fed into a dense FC layer. The architecture of GoogleNet is illustrated in [16]. The comparison between the 5 CNN architectures is listed in Table 1.

4. Layers of CNN Models

In this section, the layers of the deep learning methods will be explained. The layers are: convolution, Nonlinear activation (ReLU), pooling, and FC.

4.1. Convolution Layer

In this layer, the kernel matrix is applied to the input image to produce a feature map for the next layer. Thus, the mathematical convolution operation will be executed on the input array. At each location, element-wise multiplication is performed and summed, and the result is stored in the feature map. Convolution can be applied along one axis. If there is a two-dimensional image (I) and a two-dimensional kernel filter (k), one calculates the convolution image as in (1) [19].

Table 1. The five CNN architecture models.

Alexnet [17]	• Number of parameters is 60 million. • Consists of 5 conv. layers, 3 FC layers, and 3 max pooling operations. • Input image size is 227x227. • Developed in 2012.
VGG16 [18].	• Consists of 16 layers and 3 fully connected layers. • Number of parameters is 138 million. • The architecture is simple and easy to understand. • Developed in 2014.
VGG19[18].	• Consists of 19 layers and 3 fully connected layers. • Number of parameters is 143 million. • If compared to VGG16, it is improved; it has improved accuracy. • Developed in 2014
ResNet50[17]	• Consists of 25.6 million parameters. • Consists of 50 convolutional layers with residual connections. • ResNet50: reliable architecture for identifying a wide range of classes. • Developed in 2015
GoogleNet [17]	• Number of parameters is 7 million. • Inception modules. • It is wider and deeper than AlexNet, but it has a much smaller number of network parameters. • Developed in 2014

$$\boldsymbol{S(i,j) = \sum_m \sum_n I(m,n)k(i-m,j-n)} \quad (1)$$

4.2. Non-linear Activation Function (ReLU)

An activation function is a node that is located after the convolution layer. It is a non-linear transformation that is applied to the input signal. ReLU is a piecewise linear function that makes the output similar to the input; if it is positive, it will be zero [19].

4.3. Pooling Layer

It is the process of reducing an image's size. This will fasten the feature map process. So, it solves the problem of the large feature map size by using downsampling in the convolution layer [19].

4.4. Fully Connected (FC) Layer

At the end of a convolutional neural network, the output of the last pooling layer is represented as an input to the FC layer. There may be one or three layers. Abbreviate FC in each node in the first layer is connected to each node in the second layer, as in [19].

5. Performance Evaluation of Deep Learning Models

To evaluate the performance of the proposed model, several metrics can be used. These metrics help assess the model's effectiveness for predicting the correct outcomes. A confusion matrix will be a common tool for visualizing the performance of the proposed model.

A confusion matrix is used to evaluate various measures of a model's performance, such as accuracy, recall, F1 score, and precision. These measures provide information on the model's capability to correctly classify images from both classes, and their individual performance values indicate the model's ability to classify the input sample correctly. The information from these measures enables the person to understand the performance of any classification model by calculating evaluation scores, such as true positive (TP), false positive (FP), true negative (TN), and false negative (FN). The confusion matrix is available in [20]:

- True positive (TP): These are the positive cases, and the model correctly predicted them as positive cases
- False positive (FP): These are the cases that are not positive, but the model predicted them as positive.
- True negative (TN): These are the negative cases, and the model correctly predicted them as negative cases.
- False negative (FN): These are the cases that are actually positive, but the model has incorrectly predicted them as negative cases.

Some performance metrics can be calculated from the confusion matrix [21] as follows:

- Accuracy: is the ratio of the correctly predicted observations to the total observations. This metric is useful if datasets are symmetric, with false negatives and false positives. It can be calculated by (2):

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

- Precision: the ratio between correctly predicted positive observations and total predicted positive observations, and the false positive rate is represented in (3).

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

- Recall (Sensitivity): The ratio of correctly predicted positive observations to the total observations in the actual class. It can be calculated by (4).

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

- F1-score: is equivalent to the weighted average of precision and recall. It is more useful than the accuracy if the class distribution is unequal, or if false negatives and false positives have different costs. It can be calculated by (5).

$$\text{F1\_score} = 2 / ((1 / \text{precision}) + (1/\text{Recall})) \quad (5)$$

6. Investigated Methods

In this work, an approach is investigated for selecting the most suitable machine learning technique to classify fruit. The block diagram of the classifier is shown in Fig. 1, which consists of an input stage, a machine learning classifier model stage, and an output stage. Five classes of fruit (banana, mango, grape, apple, and strawberry) are used. The specifications of the classification models used are shown in Table 2, where network depth is the maximum number of sequential convolutions. The inputs to all networks are RGB images.

The evaluation of each model is conducted, and a comparison among them is presented. The diagram shown in Fig. 1 below has been used to test these models.

Table 2. CNN models specification [16].

Network	Depth	Image input size
Alexnet	8	227-by-227
VGG16	16	224-by-224
VGG19	19	224-by-224
Resnet50	50	224-by-224
GoogleNet	22	224-by-224

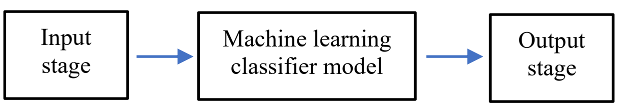

Figure 1. Block diagram of the investigated methods.

The dataset archive contains 1000 fruit images. It is downloaded from www.kaggle.com. They are divided into five classes: banana, grape, mango, apple, and strawberry. Each class has 200 images with different dimensions; thus, image resizing is essential prior to training a machine learning classifier. The existing images in the database have different dimensions, implying the need to unify image size at the input stage before entry into the classifier model. At the classification stage, the dataset is divided into training, testing, and validation subsets in a 60% training, 20% testing, and 20% validation split. Despite using the same dataset for classification, each classifier model has different classification capabilities compared to the others. Following the training, the testing phase begins. The learning rate is varied for each model to study its effect on time and accuracy. Image class prediction is initiated by the model and then sent to the output stage. At the final stage, if the prediction is correct, the model's accuracy will increase, and vice versa. The confusion matrix is constructed for each model, and several evaluation metrics are computed, such as accuracy, precision, F1 score, and recall.

7. Results and Discussion

As the testing dataset contains 200 images, each model's performance is assessed. The parameters used for the five models are: learning rate is 0.00001, epoch is 6, number of iterations is 360, and min batch size is 10. The results were obtained on a PC with a Core i5 processor, 16 GB of RAM, and MATLAB version 2024. After performing the measurements (accuracy, error rate, precision, recall, and F-measure), the results show that VGG16 achieved the highest performance among the methods, whereas AlexNet performed the worst (Table 3).

Table 3. Machine learning models and their performance.

CNN model	accuracy	Error rate	Precision	Recall	F1 measurement
Alexnet	0.7952	0.204	0.79	0.79	0.79
VGG16	0.901	0.099	0.9	0.9	0.9
VGG19	0.8069	0.1931	0.8	0.8	0.8
Resnet50	0.8515	0.1485	0.85	0.85	0.85
GoogleNet	0.8465	0.1535	0.84	0.84	0.84

Table 3 shows that VGG16 achieves the best performance compared to other network models. Table 4 compares this work with other studies in the literature.

Table 4. Comparison with the literature.

Refs.	Model	Highest Accu-racy %	#Dataset images	#classes
[11]	Inception_v3, VGG19, MobileNet and ResNet_50	99.21	1960	8
[10]	Masked_RCNN	95	81226	120
[9]	KNN, SVM, CNN	95.83	90483	131
[8]	CNN modified	90	90483	131
[5]	Using an SVM based on Deep feature	96	23848	40
[4]	GASVM	96.77	178	3
This work	Alexnet, VGG16, VGG19, Resnet50, Googlent	90.1	1000	5

7.1. Studying the Effect of Learning Rate on System Performance

To study the effect of learning rate on the accuracy, the learning rate is varied from 0.1 to 0.00001. The accuracy is calculated and recorded for each step. From the results, one can select the value that achieved the highest accuracy. These steps are repeated with all models. Table 5 shows the effect of varying the learning rate on accuracy for the five methods.

Table 5. The relationship between accuracy and learning rate for five methods.

	Accuracy				
Learning rate	Alexnet	Vgg16	Vgg19	Resnet50	GoogleNet
0.1	0.1931	0.198	0.198	0.2376	0.2030
0.01	0.2228	0.2822	0.198	0.5149	0.2079
0.001	0.4802	0.198	0.5743	0.8123	0.8211
0.0001	0.8218	0.8564	0.8861	0.8464	0.8119
0.00001	0.7525	0.7426	0.8267	0.5099	0.6931

The learning rate value of (0.0001) has achieved high accuracy. Therefore, it is suitable to utilize with all classification models.

7.2. Studying the Effect of Measuring Elapsed Time for each Model

After a suitable learning rate is chosen, the elapsed time is calculated for each model, as shown in Table 6, which indicates that the AlexNet model is the slowest, whereas the GoogleNet is the fastest.

Table 6. Elapsed time for each machine learning method.

Method	Elapsed processing time (hours: minutes: seconds)
Alexnet	3:44:00
Vgg16	2:22:58
Vgg19	2:53:32
Resnet50	1:18:23
GoogleNet	00:23:25

7.3. Performance of Classification Models: Accuracy and Losses Versus Iteration

The plots show the effect of varying the number of iterations on loss and accuracy. The plot starts from the zero-iteration value. The number of iterations increases until the accuracy and loss are stable, and the model completes its classification. This step is repeated for each model. Fig. 2 shows the relationship between iteration and accuracy and loss for AlexNet.

From the figure above, it can be seen that accuracy is low and loss is high at low iteration counts; however, as the number of iterations increases, the models' performance improves in terms of accuracy and loss until they stabilize.

7.4. Construction of Confusion Matrix

The confusion matrix for each model is constructed and shown in Figs. 3 to 7 for the five methods.

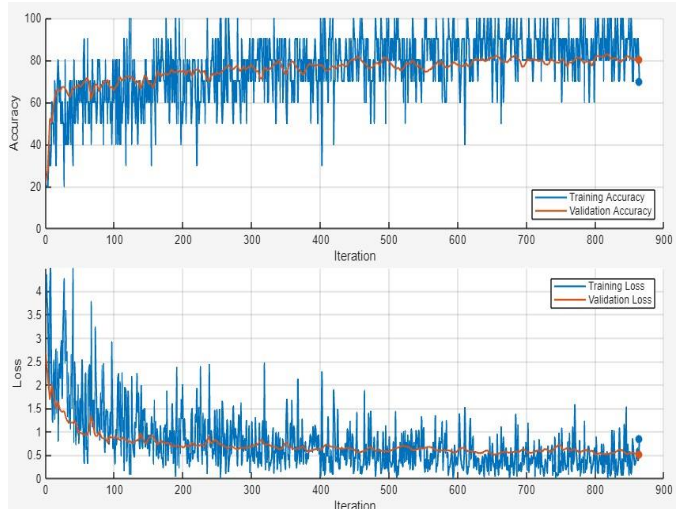

Figure 2. Accuracy and No. of iterations (upper). Loss and No. of iterations (lower) (AlexNet).

Figure 3. Confusion matrix (AlexNet).

For AlexNet, the parameters of the confusion matrix for the apple class are: TP = 34, FP = 16, FN = 9, TN = 151. For banana class TP= 28, FP= 8, FN= 12, TN=111. For the grape class, TP = 33, FP = 6, FN = 9, TN = 78. For the mango class, TP = 35, FP = 9, FN = 9, TN = 36. For the strawberry class, TP = 36, FP = 5, FN = 5, TN = 0.

7.5. Model Testing Phase

In this phase, four samples are utilized. The main goal is to test each model's ability in probabilistic classification. As shown in Fig. 8, the image labeled mango is incorrectly classified. It is FP because its color closely matches the features of the mango.

As shown in Fig. 9, the image labeled "apple" is incorrectly classified. It is a false positive (FP) because the features' shape and color are similar to those of an apple.

Again, as shown in Fig. 10, the image in the banana class is incorrectly classified. It is a false positive (FP) because the shape and color resemble a banana.

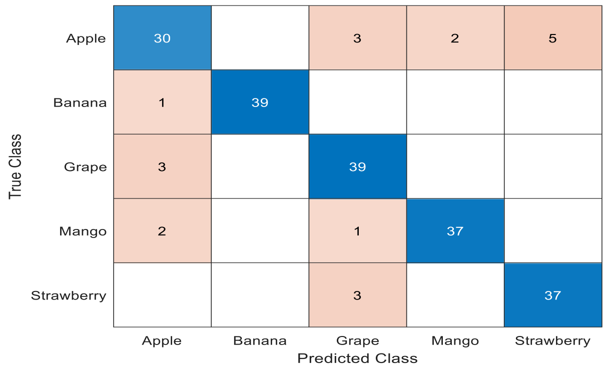

Figure 4. Confusion matrix (VGG16).

For VGG16, the parameters of the confusion matrix are: for the apple class, TP = 30, FP = 6, FN = 10, TN = 152. For the banana class, TP = 39, FP = 0, FN = 1, TN = 117. For the grape class, TP = 39, FP = 7, FN = 3, TN = 74. For the mango class, TP = 37, FP = 2, FN = 3, TN = 37. For the strawberry class, TP = 37, FP = 5, FN = 3, TN = 0.

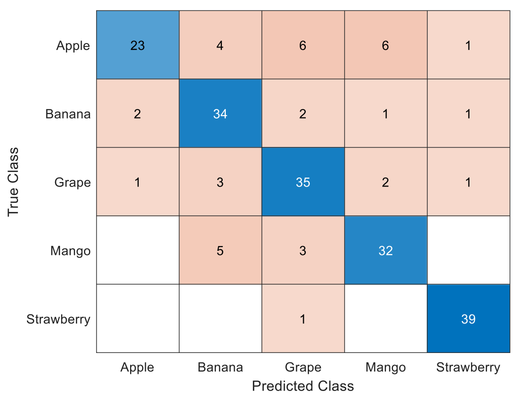

Figure 5. Confusion matrix (VGG19).

For VGG19, the parameters of the confusion matrix for the apple class are: TP = 23, FP = 3, FN = 17, TN = 159. For the banana class, TP = 34, FP = 12, FN = 6, TN = 113. For the grape class, TP = 35, FP = 12, FN = 7, TN = 71. For mango class TP= 32, FP= 9, FN=8, TN= 39. For the strawberry class, TP = 39, FP = 3, FN = 1, TN = 0.

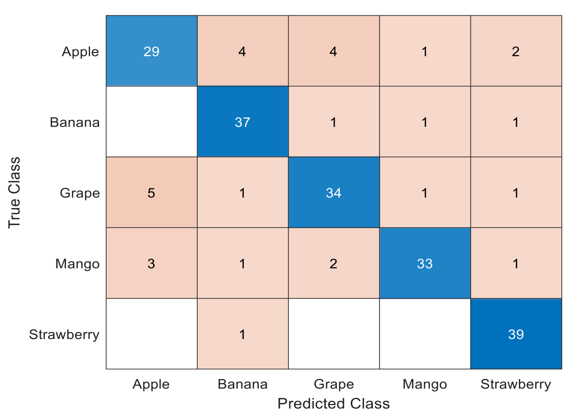

Figure 6. Confusion matrix (Resnet50).

For Resnet50, the confusion matrix parameters for the apple class are: TP = 29, FP = 8, FN = 11, TN = 154. For the banana class, TP = 37, FP = 7, FN = 3, TN = 111. For the grape class, TP = 34, FP = 7, FN = 8, TN = 73. For the mango class, TP = 33, FP = 3, FN = 7, TN = 39. For the strawberry class, TP = 39, FP = 5, FN = 1, TN = 0.

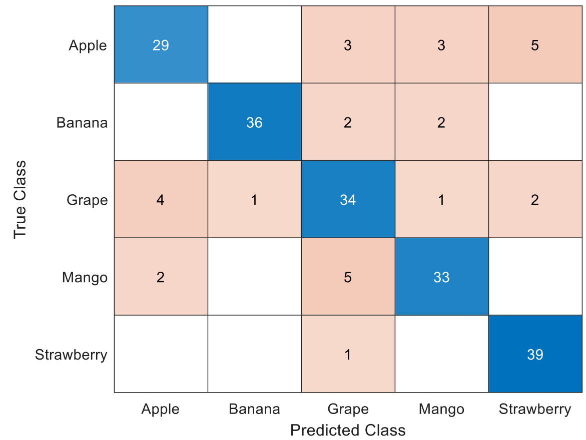

Figure 7. Confusion matrix (GoogleNet).

For GoogleNet, the parameters of the confusion matrix are related to the apple class: TP = 29, FP= 6, FN= 11, TN= 156. For the banana class, TP = 36, FP = 1, FN = 4, TN = 115. For the grape class, TP = 34, FP = 11, FN = 8, TN = 72. For the mango class, TP = 33, FP = 6, FN = 7, TN = 39. For the strawberry class, TP = 39, FP = 7, FN = 1, TN = 0.

Figure 8. Prediction of the AlexNet approach with fruit classification.

Figure 9. Prediction of the VGG16 approach with fruit classification.

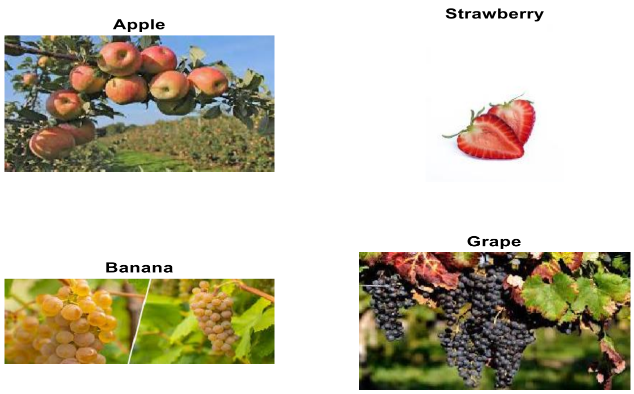

Figure 10. Prediction of the VGG19 approach with fruit classification.

As shown in Fig. 11, the image under the class grape is incorrectly classified. It is a false positive (FP) because its features (shape) are close to the shape of a cluster of grapes.

Figure 11. Prediction of the ResNet50 approach with fruit classification.

As shown in Fig. 12, the image labeled mango is incorrectly classified. It is a false positive (FP); it contains many classes. In the designed system, if any image does not belong to the four categories used in the research, it will be considered as one of the closest to these categories.

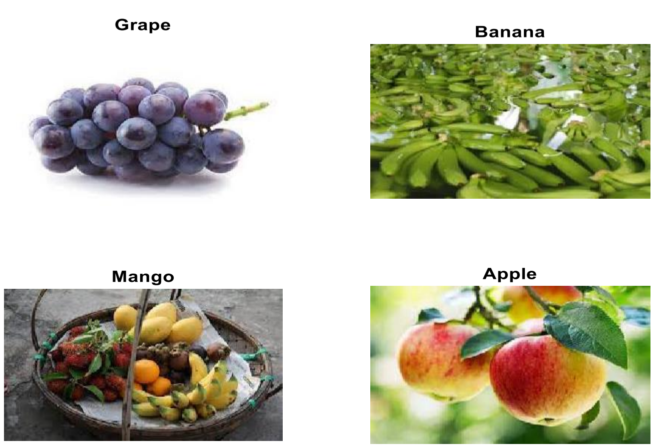

Figure 12. Prediction of GoogleNet approach with fruit classification.

8. Conclusions

Machine learning techniques for fruit classification are presented in this research. Building a reliable system for object classification and determining the optimal classification network model in terms of performance were the primary goals of this work. The main dataset contained 1000 images collected from Kaggle. VGG16 has achieved the highest accuracy and the lowest error rate; however, it incurs an elapsed processing time, whereas the AlexNet approach has achieved lower accuracy and a higher error rate, and it also incurs an elapsed time. The GoogleNet approach is the shortest in terms of elapsed time, but its accuracy is lower than that of VGG16.

From this work, we conclude that VGG16 outperforms the other methods used in the manuscript. In addition, we found that a learning rate value of 0.0001 is more suitable for all classification network models. Another conclusion is that the GoogleNet model is the fastest, whereas AlexNet is the slowest. Furthermore, if the image does not belong to one of the five categories used in the research, the system will classify it to the nearest category. As future work, it can be seen that VGG16 has successfully classified different fruit classes compared to the other models used in this paper. It is recommended to employ VGG16 for image classification and enhance its accuracy by providing an objective analysis and model complexity assessment through transfer learning. As another future work, applying a Gabor filter for edge detection and feature extraction at the preprocessing stage.

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Author Contribution

Abdulqader F. Abdulqade constructed the conceptualization, methodology, and data curation. He also conducted the formal analysis, writing-review, editing, and visualization.

Both Authors were responsible for the work related to using the software, validation, and resources.

Acknowledgment

The authors are grateful to the researchers supporting (ANUI/2024/ENG23) for their kind support. Alnoor University, Mosul, Iraq.

References

[1] D. M. Asriny and R. Jayadi, "Transfer Learning VGG16 for Classification of Orange Fruit Images," *Journal of System and Management Sciences*, vol. 13, no. 1, pp. 206-217, Feb. 2023, doi: https://www.aasmr.org/jsms/Vol13/No.1/Vol.13.No.1.12.pdf

[2] K. B Mirra and R. Rajakumari, "Classification of Fruits Using Deep Learning Algorithms," *SSRN Electronic Journal*, 2022, doi: https://doi.org/10.2139/ssrn.4068366

[3] G. Xue, S. Liu, and Y. Ma, "A Hybrid Deep learning-based Fruit Classification Using Attention Model and Convolution Autoencoder," *Complex & Intelligent Systems*, vol. 9, pp. 2209-2219, Oct. 2020, doi: https://doi.org/10.1007/s40747-020-00192-x.

[4] M. S. Mahajan and S. T. Patil, "Optimization and Classification of Fruit Using Machine Learning Algorithm," *International Journal for Innovative*

Research in Science & Technology, vol. 3, no. 1, Jul. 2016, Accessed: Aug. 30, 2025. [Online]. Available: https://www.academia.edu/download/47640418/IJIRSTV3I1139.pdf

[5] S. Ghazal et al., "Analysis of visual features and classifiers for Fruit classification problem". *Computers and Electronics in Agriculture*, vol. 187, 106267, August 2021, doi: https://doi.org/10.1016/j.compag.2021.106267

[6] J. Naranjo-Torres, et al., "A review of convolutional neural network applied to fruit image processing". *Applied Sciences*, vol. 10, no. 10, p. 3443, 2020, doi: https://doi.org/10.3390/app10103443

[7] J. L. Rojas-Aranda, J. I. Nunez-Varela, J. C. Cuevas-Tello, and G. Rangel-Ramirez, "Fruit Classification for Retail Stores Using Deep Learning," *In Pattern Recognition: 12th Mexican Conference,* MCPR 2020 Morelia, Mexico, June 24-27, Proceedings 12, pp.3-13. Springer International Publishing (2020), doi: https://doi.org/10.1007/978-3-030-49076-8_1.

[8] A. Amol, S. Smriti, B. Rishikesh, G. Roy, and S. Varma, "Fruit Classification and Calories Measurement Using Machine Learning and Deep Learning," *International Research Journal of Engineering and Technology (IRJET),* vol. 08, no. 05, pp. 3476-3481, 2021. https://www.semanticscholar.org/paper/Fruit-Classification-and-Calories-Measurement-using-Amol-Smriti/

[9] N. Hemalatha, P. Sukhetha, and R. Sukumar, "Classification of Fruits and Vegetables Using Machine and Deep Learning Approach," *2022 International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT),* pp. 1–4, Oct. 2022, doi: https://doi.org/10.1109/TQCEBT54229.2022.10041654

[10] C. C. Ukwuoma, Q. Zhiguang, M. B. Bin Heyat, L. Ali, Z. Almaspoor, and H. N. Monday, "Recent Advancements in Fruit Detection and Classification Using Deep Learning Techniques," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–29, Jan. 2022, doi: https://doi.org/10.1155/2022/9210947.

[11] Md. M. Rahman, Md. A. Basar, T. S. Shinti, Md. S. I. Khan, H. Md. H. Babu, and K. M. M. Uddin, "A Deep CNN Approach to Detect and Classify Local Fruits through a Web Interface," *Smart Agricultural Technology,* vol. 5, p. 100321, Oct. 2023, doi: https://doi.org/10.1016/j.atech.2023.100321.

[12] Y. Liu et al., "Machine learning-based modeling for estimation of drug solubility in supercritical fluid by adjusting important parameters," Chemometrics *and Intelligent Laboratory Systems*, vol. 254, no. 105241, pp. 105241–105241, Oct. 2024, doi: https://doi.org/10.1016/j.chemolab.2024.105241.

[13] G. S. Nugraha, M. I. Darmawan, and R. Dwiyansaputra, "Comparison of CNN's Architecture GoogleNet, AlexNet, VGG-16, LeNet-5, ResNet-50 in Arabic Handwriting Pattern Recognition," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control,* vol. 8, no. 2, May 2023, doi: https://doi.org/10.22219/kinetik.v8i2.1667.

[14] V. Sudha and T. R. Ganeshbabu, "A Convolutional Neural Network Classifier VGG-19 Architecture for Lesion Detection and Grading in Diabetic Retinopathy Based on Deep Learning". *Computers, Materials & Continua*, vol. 66, no. 1, 2021, doi: https://doi.org/10.32604/cmc.2020.012008

[15] S. H. Khaleefah et al., "Applying Deep Learning Models to Breast Ultrasound Images for Automating Breast Cancer Diagnosis," *JOIV* International *Journal on Informatics Visualization*, vol. 8, no. 3–2, pp. 1779–1779, Nov. 2024, doi: https:doi.org/10.62527/joiv.8.3-2.1912.

[16] R. H. Sahib, D. H. M. Jawad, A. Z. Sameen, and A. I. Ahmed, "Leveraging Machine Learning and low-cost Hardware for Economical wind-driven Water Level Prediction," *Service Oriented Computing and Applications,* vol. 18, no. 4, pp. 311–320, Mar. 2024, doi: https://doi.org/10.1007/s11761-024-00390-2.

[17] K. Kamal and H. EZ-ZAHRAOUY, "A Comparison between the VGG16, VGG19 and ResNet50 Architecture Frameworks for Classification of Normal and CLAHE Processed Medical Images," preprint, Apr. 2023, doi: https://doi.org/10.21203/rs.3.rs-2863523/v1.

[18] S. Deepa, J. Loveline Zeema, and S. Gokila, "Exploratory Architectures Analysis of Various Pre-trained Image Classification Models for Deep Learning," *Journal of Advances in Information Technology*, vol. 15, no. 1, pp. 66–78, Jan. 2024, doi: https://doi.org/10.12720/jait.15.1.66-78.

[19] S. Tammina, "Transfer Learning Using VGG-16 with Deep Convolutional Neural Network for Classifying Images," *International Journal of Scientific and Research Publications (IJSRP),* vol. 9, no. 10, pp. 9420, Oct. 2019, doi: https://doi.org/10.29322/ijsrp.9.10.2019.p9420.

[20] E. Hossain, Machine Learning Crash Course for Engineers. Springer Nature, Switzerland. 2024. https://link.springer.com/book/10.1007/978-3-031-46990-9

[21] P. Gayathri, A. Dhavileswarapu, S. Ibrahim, R. Paul, and R. Gupta, "Exploring the Potential of VGG-16 Architecture for Accurate Brain Tumor Detection Using Deep Learning*," Journal of Computers, Mechanical and Management,* vol. 2, no. 2, pp. 13–22, Jun. 2023, doi: https://doi.org/10.57159/gadl.jcmm.2.2.23056.