

An Application of Machine Learning Methods to Detect Mango Varieties

Abou Bakary Ballo¹, Moustapha Diaby², Adama Coulibaly¹

¹Department of Mathematics and Computer Science, University FÉLIX HOUPHOUËT-BOIGNY, Abidjan, Côte d'Ivoire

²African College of Information and Communication Technologies, Abidjan, Côte d'Ivoire

Email: aboubak2005@yahoo.fr

How to cite this paper: Ballo, A.B., Diaby, M. and Coulibaly, A. (2024) An Application of Machine Learning Methods to Detect Mango Varieties. *Open Journal of Applied Sciences*, 14, 1666-1690.

<https://doi.org/10.4236/ojapps.2024.147109>

Received: May 28, 2024

Accepted: July 14, 2024

Published: July 17, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

The mango, a fruit of immense economic and dietary significance in numerous tropical and subtropical regions, plays a pivotal role in our agricultural landscape. Accurate identification is not just a necessity, but a crucial step for effective classification, sorting, and marketing. This study delves into the potential of machine learning for this task, comparing the performance of four models: MobileNetV2, Xception, VGG16, and ResNet50V2. These models were trained on a dataset of annotated mango images, and their performance was evaluated using precision, accuracy, F1 score, and recall, which are standard metrics for image classification. The Xception model, with its exceptional performance, outshone the other models on all performance indicators. It achieved a staggering accuracy of 99.47%, an F1 score of 99.43%, and a recall of 99.43%, showcasing its remarkable ability to accurately identify mango varieties. MobileNetV2 followed closely with performances of 98.95% accuracy, 98.85% F1 score, and 98.86% recall. ResNet50V2 also delivered satisfactory results with 97.39% accuracy, 97.08% F1 score, and 97.17% recall. VGG16, however, was the least effective, with a precision rate of 83.25%, an F1 score of 83.25%, and a recall of 85.47%. These results confirm the superiority of the Xception model in detecting mango varieties. Its advanced architecture allows it to capture more distinguishing features of mango images, leading to greater precision and reliability. Xception's robustness in identifying true positives is another advantage, minimizing false positives and contributing to more accurate classification. This study highlights the promising potential of machine learning, particularly the Xception model, for accurately identifying mango varieties.

Keywords

Machine Learning, Recognition, Mango Varieties, Algorithms, Feature Extraction, Convolution, Training, Classification

1. Introduction

The mango, an exotic fruit with a delicious taste and many virtues, is the subject of particular attention throughout its supply chain. From the selection of varieties by producers to the choices made by consumers, precise identification is of vital importance. It guarantees taste quality, authenticity, and a fair price for each variety while meeting the individual expectations of mango lovers. Manual recognition of mango varieties, although common, is a laborious and imperfect process, especially when large volumes are involved. Machine learning offers a revolutionary alternative, providing an automated and robust solution to this crucial task. Based on the analysis of digital images of mangoes, machine learning algorithms learn to identify the distinctive characteristics of each variety, such as shape, color, texture, and size. These models, fed by labeled image datasets, become experts in variety recognition, often surpassing the accuracy of the human eye. Adopting machine learning to recognize mango varieties brings a host of benefits, including increased accuracy and tenfold efficiency. In this way, machine learning opens the way to numerous practical applications such as automated sorting and grading, enhanced quality control, variety authentication, and consumer information and marketing. In the face of these challenges, the considerable advantages of recognizing mango varieties using machine learning are motivating research and development efforts. By overcoming the obstacles and harnessing the full potential of this technology, we can expect a major transformation of the mango industry. This transformation will result in a more sustainable, transparent, and fair supply chain while offering consumers an improved shopping experience and superior-quality products.

In this fascinating field of mango variety identification, we conducted an in-depth review of the existing literature. We have explored previous work and unresolved issues. We then turn to the materials and methods section, revealing the tools and techniques that informed our research. We present the carefully selected database, the hardware configuration that powered our analyses, and a detailed explanation of the methodological approach we adopted. We reveal the results of our study and engage in an in-depth discussion of the performance of our system. To conclude this scientific adventure, we bring together the key points of our study and pave the way for future explorations.

2. Related Works

Mango growing, a crucial economic and food activity in many tropical and subtropical regions, has made considerable progress thanks to technological advances. Among these advances, machine learning (ML) is emerging as a promising tool for optimizing farming practices and improving productivity. This study traces the major research efforts devoted to both mango cultivation and machine learning, focusing on the application of the latter in the field of mango cultivation.

Worasawate, D *et al.* developed four common machine learning (ML) classifi-

ers, the k-means, naive Bayes, support vector machine, and feed-forward artificial neural network (FANN), for classifying the maturity stage of mangoes at harvest. The results of their work showed that the FANN classifier performed best, with an average accuracy of 89.6% for the “unripe,” “ripe”, and “overripe” classes [1]. However, the study did not specify the mango varieties used, as ripening characteristics can vary considerably from one variety to another.

A new classification-based artificial intelligence approach for mango leaf recognition has been studied by Maqbool, I *et al.* [2]. They developed an artificial neural network system that extracts specific shape and morphology features from mango leaves based on reducing the dimensionality of the feature space. The resulting experiments achieved a classification rate of 96% to 98%. However, the lack of diversity in terms of mango cultivars, leaf age, and growing conditions can bias the results and affect the performance of the system in real situations.

Ibrahim M *et al.* [3] studied an online system for grading Harumanis mangoes using computer vision. It is a system capable of identifying the irregularity of the mango shape and its estimated mass. Some important features such as length, height, centroid, and parameters were extracted from each image, then the Fourier descriptor and the size and shape parameters were used to describe the shape of the mango. The results of this work gave 94% as the average success rate for mass-based classification. However, this study is limited to the Harumanis variety. The performance of the system may vary for other mango varieties with different shapes and characteristics.

A method based on near-infrared spectroscopy algorithms to detect a variety of mango fruits and vegetables was developed by Tripathi, M. K *et al* [4]. PCA techniques are used for data dimension reduction. The three machine learning techniques, such as ANN, SVM, and KNN, were used to build a detection model for a variety of mangoes. Experimentation shows that the proposed system can contribute significantly to the recognition of various fruits and vegetables. However, the study only used conventional near-infrared spectroscopy algorithms. More recent algorithms, such as deep learning, could achieve better performance.

Tensor Flow Lite was used as a transfer learning tool by Mustaffa, M. R. [5]. This study focuses on six categories, in which four types of mangoes are classified (Harum Manis, Langra, Dasherri, and Sindhri), categories for other types of mangoes, and a “non-mango” category. It is carried out using a mobile application that can be used to distinguish the different types of mangoes based on the category image. However, the study uses a relatively small dataset to train the model. This may limit the system’s ability to generalize to new mango varieties and varied environmental conditions.

Truong Minh Long, N *et al.* [6] proposed a new internal quality assessment based on the external characteristics of the mango as well as its weight. Fruit classification is implemented by four machine learning models: Random Forest (RF), Linear Discriminant Analysis (LDA), Support Vector support (SVM), and

K-nearest neighbors (KNN). The results of their experiments show that the machine learning methods have a high accuracy of over 87.9%. In particular, the RF model achieves an accuracy of 98.1%.

Koirala A. *et al.* [7] studied a method based on the YOLO technique for estimating the number of panicles and their stage of development. By estimating the number of panicles and their stage of development, it is possible to calculate the time required for fruit ripening. Knowing the ripening time is essential for harvest management, as it enables cultivation operations to be planned and fruit quality to be maximized. However, this study does not compare the proposed method with other methods for estimating the number of panicles and their stage of development.

Bhargava *et al.* studied the automatic detection and classification of apples and mangoes using a support vector and multiple-feature classifier [8]. The results of their experiment give an accuracy of 98.48% and 95.72%. However, methodological improvements and more in-depth analyses would be needed to confirm the robustness and generalizability of their approach in realistic scenarios.

Baculo *et al.* study the automated detection of a defect in mango caused by *Cecidomyia* flies, which can affect a significant part of the production yield. Their work also uses modified versions of R-CNN and FR-CNN by replacing the region search algorithms with segmentation-based region extraction [9].

Experimental results show comparable performance between the modified and existing state-of-the-art object detection frames. The results show that the faster R-CNN achieved the highest average accuracy of 0.901 at aP{50} while the modified FR-CNN has the highest average accuracy of 0.723 at aP{50}.

The studies reviewed show a variety of machine learning and Deep Learning approaches for identifying mango fruit.

3. Materials and Methods

In this section, we describe the dataset, algorithms, parameters, methodology, and evaluation measures used in the study.

3.1. Dataset Description

The database used is named Mango Variety. It can be accessed at the following address <https://data.mendeley.com/datasets/tk6d98f87d/2>. This dataset contains 1,661 images of mangoes taken in a controlled environment. The database is composed of 15 mango varieties presented in **Table 1**.

The diversity of mangoes means that they come in different flavors, textures, sizes, shapes and colors. The illustration in **Figure 1** gives a visual overview of these fifteen unique varieties.

Table 1. Table of mango varieties.

| Variety | Number of images |
|----------|------------------|
| Alphonso | 211 |
| Ambika | 100 |

Continued

| | |
|--------------------|-------------|
| Amrapali | 150 |
| Banzanpali | 100 |
| Chausa | 100 |
| Dasheri | 100 |
| Himsagar | 150 |
| Kesar | 100 |
| Vanraj | 100 |
| Langra, | 100 |
| Malgova | 150 |
| Mallika | 100 |
| Neelam | 100 |
| Raspuri | 100 |
| Totapuri | 100 |
| Full images | 1661 |

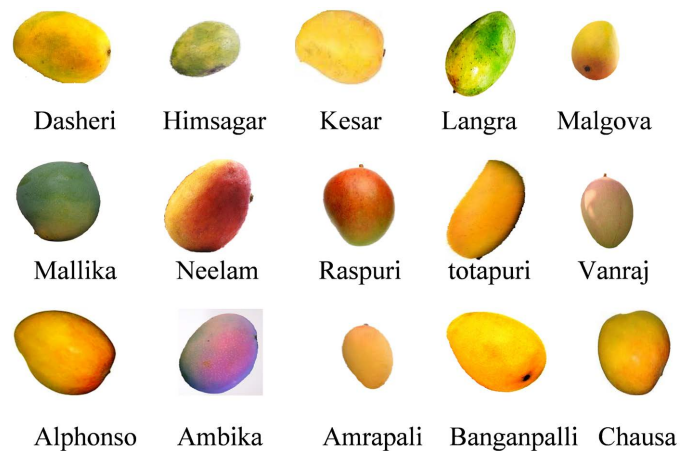


Figure 1. Illustration of mango varieties.

3.2. Learning Algorithms

3.2.1. MobileNetV2

MobileNet V2 is a convolutional neural network (CNN) model [10] developed by Google AI for mobile devices [11] [12]. It is based on the MobileNet architecture but offers enhanced performance thanks to innovations such as the bottleneck block and transfer learning. MobileNet V2 is lighter and more energy-efficient than traditional CNN models while retaining comparable accuracy for image classification, object detection, and facial recognition [13]. In its development, MobileNet V2 uses a “bottleneck” convolution block that reduces the dimensionality of the data while retaining critical information. This reduces the number of model parameters and improves energy efficiency. Transfer learning is used to initialize model weights with knowledge gained from other tasks, improving performance and speeding up convergence when learning on new datasets [14]. Lightweight and energy-efficient, ideal for mobile devices with limited resources, the *MobileNet* algorithm has a performance comparable to traditional CNN models for various computer vision tasks [15]. Its application is in image classification, generally

in the identification of objects, people, and scenes in images, and in object detection by locating and identifying specific objects in images. MobileNet V2 is a powerful and versatile CNN model for mobile devices. Its lightweight and energy efficiency make it an ideal choice for embedded applications where performance and power consumption are critical factors. MobilenetV2 is a CNN designed for high computational efficiency, making it ideal for mobile and embedded applications. It uses “separable convolution blocks” and a “depth separable convolution” architecture to reduce the number of parameters and calculations. MobilenetV2 offers a good compromise between accuracy and speed, making it attractive for classifying mango varieties on mobile devices.

3.2.2. Xception

Xception is a deep neural network architecture derived from Inception V3. It was developed by Google after the release of Inception V3 [16] [17]. Xception’s main innovation lies in the use of separable depth blocks to replace the standard convolution operations of Inception V3. The Inception architecture aims to optimize feature extraction and transfer through a combination of different transformations. These transformations include convolution with kernels of different sizes (1×1 , 3×3 , and 5×5), pooling, and other techniques [18]. Inception’s conceptual structure enables the best combination of transformations to be selected for each case, maximizing model efficiency. Inception’s architecture enables different feature extraction methods to be explored in parallel. This is done by simultaneously applying several transformations (convolution, pooling, etc.) to the model input [19]. The results of these transformations are then concatenated, creating an enriched feature representation. Learning allows the model to determine the optimal combination of transformations for each situation.

Chollet *et al.* developed Xception [20] to tackle channel correlation in convolutional networks. The Xception architecture replaces standard convolution with separable depth blocks. This approach allows the decoupling of spatial and channel transformations [21]. The convolution operation in the original concept-v3 is substituted by the separable solution, which improves Xception’s convergence process and delivers significantly higher accuracy. Xception, based on Google’s “Inception” architecture, stands out for its ability to extract contextual information at different spatial scales. This feature is made possible by its “Inception modules”, which combine convolutions of multiple sizes, thus improving the model’s learning capacity. Xception is particularly well suited to the task of classifying images that vary in size and scale, such as different varieties of mango.

3.2.3. VGG16

VGG16, a revolutionary convolutional neural network (CNN) developed in 2014 by the Visual Geometry Group (VGG) team at the University of Oxford, marked a turning point in image classification thanks to its remarkable depth and performance on the ImageNet dataset. In other words, we exploit the pre-trained weights of VGG16, the fruit of its training on the vast ImageNet dataset [22].

Our network is based on the initial weights of the ImageNet pre-trained model. In effect, we replace the final layers of the original VGG16 architecture with a new fully connected layer (FCL). To preserve the knowledge acquired during training on ImageNet, all layers of the VGG16 network are “frozen”. Thus, only the process of learning the new FCL layer is carried out using the training set specific to our problem. This approach, known as transfer learning, considerably speeds up training time without compromising classifier accuracy. Indeed, the FCL is the only layer to be trained, and it uses a softmax function to generate the final outputs.

Although the VGG16 is a 16-layer convolutional neural network [23] and requires complex training, it is quick to use on a personal computer. Indeed, CNNs rely on parameterized learning: once the network weights have been trained (which is done on a dedicated deep learning server, thus avoiding a significant computational load for the user), the prediction of the class of a new fruit sample is made by a simple multiplication of these stored weights. The aim of this research is, therefore, to develop an accurate, fast-to-run model designed for mobile applications or lightweight software capable of recognizing mango varieties from images. With its simple architecture and large number of convolutional layers, VGG16 is a high-performance convolutional neural network (CNN) for feature extraction. This capability makes it particularly useful for classifying complex images, such as those of different varieties of mango. While VGG16 can be computationally intensive, it offers exceptional accuracy in classifying mango varieties.

3.2.4. ResNet50V2

ResNet-50 is a deep convolutional neural network trained on over a million image databases. The network can classify images into 1,000 objects, including keyboards, mice, pencils, etc. [24]. ResNet50V2 is an enhanced variant of ResNet50 that outperforms ResNet50 and ResNet101 on the ImageNet dataset. In ResNet50V2, the concept of proposing links between blocks has been modified. ResNet50V2 achieves excellent results on the ImageNet dataset. ResNet50V2, an improved variant, outperforms ResNet-50 on the ImageNet dataset. Overall, ResNet-50 and its variants represent significant advances in image classification, demonstrating the potential of Deep Learning to revolutionize computer vision tasks. The network uses “short paths” to link deep and superficial layers, optimizing the propagation of information and boosting model performance. ResNet50V2 is particularly well suited to complex classification tasks, such as distinguishing between subtle varieties of mango.

3.2.5. Comparison of Convolutional Neural Network Parameters

Table 2 details the parameters of the convolutional neural network used in this study. It compares four convolutional neural network models (ResNet50 V2, MobileNetV2, VGG16 and Xception) as a function of several parameters. The trainable parameters are shown for each model.

Table 2. Parameters of the convolutional neural network.

| Model Name | Trainable parameters | Non Trainable parameters | Total Parameters | Size | Epoch |
|-------------|----------------------|--------------------------|------------------|------|-------|
| ResNet50 V2 | 24,389,777 | 45,440 | 24,435,217 | 4 | 10 |
| MobileNetV2 | 2,767,889 | 34,112 | 2,802,001 | 8 | 10 |
| VGG16 | 14,853,969 | 0 | 14,853,969 | 4 | 10 |
| Xception | 21,677,369 | 54,528 | 21,731,897 | 4 | 10 |

These are the model parameters that can be adjusted during the training process. The higher the number, the more flexible the model is likely to be and the more likely it is to learn from the training data.

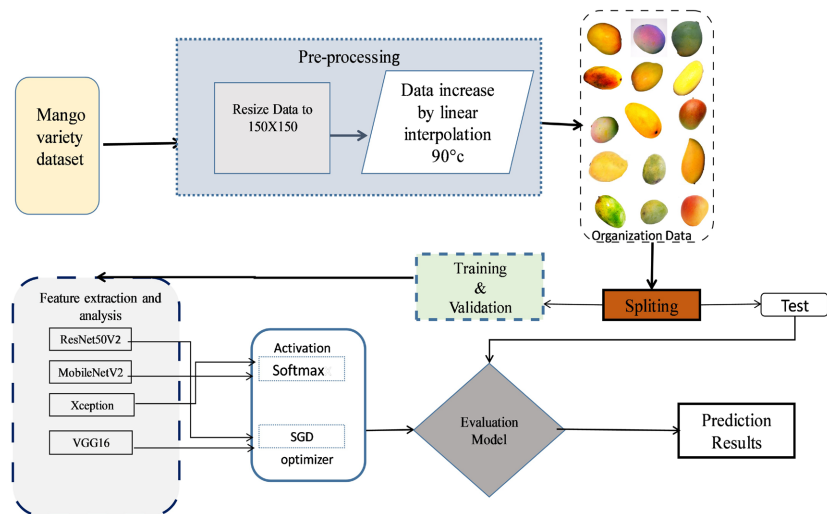
Non-trainable parameters indicate the number of model parameters that remain fixed during training. The total number of parameters is the sum of the trainable and non-trainable parameters, reflecting the overall complexity of the model.

Batch size indicates the number of training samples processed simultaneously by the model during training.

Epochs represent the number of complete runs performed on the training dataset during training.

3.3. Methodology

The methodology used in our study is shown in **Figure 2**.

**Figure 2.** Illustration of our methodology for recognizing mango varieties.

The study's methodology is organized into distinct and clearly defined phases, guaranteeing a rigorous approach to our research. This research explores the potential of four classification algorithms, namely Xception, Random Forest, MobileNetV2, and ResNet50V2, to enable a system to recognize different mango varieties. **Figure 1** shows the various stages involved:

Our methodology is based on a rigorous approach articulated around careful-

ly designed and planned steps:

- Database preparation: In this phase, each image was segmented to identify each fruit in the database precisely. To improve the quality of our data, each fruit image was resized and also subjected to a data augmentation step.
- Data augmentation: Our approach relies on data augmentation to improve the quality and robustness of our results. This step aims to expose the models to a wider variety of cases by randomly inverting the images along random vertical axes, enabling left and right inversions as well as 90-degree clockwise rotations.
- Data division: To ensure rigorous evaluation of our models, we have divided our data into three distinct sets: a training set used for learning the models, a validation set for adjusting the hyper parameters, and a test set for final performance evaluation.
- Use of various algorithms: To tackle our mango variety identification task, we adopted an approach using the Xception, Random Forest, MobileNetV2, and ResNet50V2 algorithms, as well as deep methods based on data analysis with feature extraction. Classification is a supervised learning technique in which a model is trained to predict an example's membership of a predefined class from a set of possible classes [25]. To do this, the model learns to identify patterns and relationships between data features and the corresponding class labels.
- ResNet50V2 is a convolutional neural network (CNN) architecture derived from ResNet50, a widely used model for image recognition. The key element of ResNet50V2 is the use of "residual blocks". Each residual block is made up of one or more "convolutional units. Identity shortcuts" link the outputs of some convolutional units directly to the inputs of their respective residual blocks. These shortcuts maintain a direct flow of information and prevent gradients from disappearing, thus improving model performance. Grouping layers are used to reduce the dimensionality of features and control model complexity. A fully connected output layer is added at the end of the network to map the extracted features to the mango variety classification. The stochastic gradient back-propagation (SGD) algorithm is used to adjust the model parameters to minimize the loss function.

The performance of the model is evaluated on a separate test dataset to measure its accuracy and generalizability.

- At the heart of MobileNetV2 are "separable convolution blocks". These ingenious blocks dismantle standard convolutions into two distinct steps: a depth convolution and a point convolution. This clever strategy drastically reduces the number of parameters and computational operations, giving the model exceptional efficiency. Grouping layers are used to reduce the dimensionality of features and control the complexity of the model. A fully connected output layer is added at the end of the network to map the extracted features to the classification of mango varieties. An optimization algorithm, such as SGD, is used to adjust the model parameters to minimize the loss

function. Its efficient architecture, optimized learning techniques, and rigorous evaluation techniques make it possible to achieve satisfactory performance while limiting the computing resources required.

- VGG16 stands out as a deep convolutional neural network (CNN) architecture, renowned for its simplicity and remarkable performance in various image recognition tasks. Its operation is based on “convolutional blocks”, which combine convolutional, pooling and batch normalization layers. These blocks are used to extract crucial features from input data at different spatial scales. Pooling layers are then integrated to reduce feature dimensionality and control model complexity. VGG16 uses max pooling layers, which select the maximum value in each pooling window. This simplifies the model and improves efficiency. Batch normalization layers are also used to stabilize model learning and speed convergence. These layers normalize the activations of the previous layers, making it easier to learn the model parameters. Finally, fully connected layers are present at the end of the architecture to classify images according to the different varieties of mango. These layers learn to map the features extracted by the convolutional blocks to the corresponding mango variety classes.
- The key element of Xception is the use of “Inception blocks”. These blocks combine convolutions of different kernel sizes (1×1 , 3×3 , 5×5) and max grouping into a single structure, allowing features to be extracted at different spatial scales and reducing the dimensionality of the data. Xception uses a “multi-path data stream” that divides input data into several parallel branches, each passing through a series of Inception blocks. Output from these branches is then combined to generate a richer representation of features. Clustering layers are used to reduce feature dimensionality and control model complexity. Xception uses max and avg grouping layers to extract global and local features from the input data. At the end of the architecture, fully connected layers are used to classify images into different mango varieties. These layers learn to map the features extracted by the Inception blocks to mango variety classes. The performance of the model is evaluated on a separate test dataset to measure its accuracy and generalizability.
- In our work, we used the Softmax classifier and SGD optimizer. The Softmax classifier is used to perform multiclass classification, assigning each input example a probability for each of the possible classes. Deep learning and computer vision have widely adopted the Softmax classifier for classifying vectors into various categories [26]. The matching function F plays a crucial role in the classification process. It takes an input dataset x and transforms it into class labels by performing a scalar product between x and the weight matrix W

$$F(xi, W) = W * xi \quad (1)$$

As for SGD (Stochastic Gradient Descent), it has demonstrated its effectiveness in solving large-scale, sparse machine learning problems typical of text clas-

sification and natural language processing [27]. It is an efficient classifier and is easy to implement, with several possibilities for code tuning. SGD has gained popularity thanks to its state-of-the-art performance in a variety of machine-learning tasks, demonstrating its versatility and efficiency [28]. SGD momentum incorporates the notion of inertia to guide parameter updates in the gradient direction, taking into account previous iterations. At each step (Δw), the next is calculated as a linear combination of the current gradient and the modification. If we denote W_n the coefficients obtained after n iterations, as well as ΔW_n the n -th parameter update:

$$\Delta W_{n+1} = n\Delta Q_i(W) + \Delta W_n \quad (2)$$

$$W_{n+1} = W_n - \Delta W_{n+1} \quad (3)$$

Let's imagine the parameter vector as a particle moving through a vast, often high-dimensional parameter space. Under the influence of the gradient, which acts as a driving force, the particle accelerates towards the optimal solution. Unlike the classical SGD method, this variant tends to continue its trajectory in the same direction, thus minimizing oscillations. The momentum SGD method, which has been used successfully for several decades, is particularly effective in this field.

- **Performance evaluation:** The evaluation of the models was based on their ability to correctly identify mango varieties. This crucial evaluation phase quantified the performance of each approach in terms of precision, recall, and other relevant measures, enabling an objective comparison of their efficiencies.

Our methodology is based on a solid foundation, from meticulous data preparation to in-depth model evaluation, including precise identification and recognition of mango varieties.

3.4. Performance Metric

The effectiveness of the model developed in this study will be assessed using the following indicators.

Accuracy: This performance measure indicates the extent to which the system has classified data into the correct categories or classes.

Precision: This represents the ratio between the number of correctly classified positive images and the total number of positive images.

Recall: This criterion measures a classifier's ability to identify real positive results.

The F1 score: It gives a weighted average of precision and recall, providing a balanced measure of performance.

The Matthews Correlation Coefficient (MCC): Evaluates the quality of classifications in machine learning, taking into account false positives and false negatives.

Mean Squared Error (MSE): measures the mean squared error between estimated and actual values, allowing us to assess the precision of estimates.

Using these measures, we can quantify the performance of the model in our study and assess its ability to accurately classify images of mango varieties.

The equations for the various measurements are specified as follows.

$$Accuracy = (TP + TN) / (TP + FP + TN + FN) \quad (4)$$

$$Precision = TP / (TP + FP) \quad (5)$$

$$Recall = TP / (TP + FN) \quad (6)$$

$$F1Score = 2 * (Precision * Recall) / (Precision + Recall) \quad (7)$$

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

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (9)$$

(5) Where TP, FP, FN, P, R, and F1 represent true positives, false positives, false negatives, precision, recall, and F-1 score, respectively.

The precision in equation (3) is the percentage accuracy of detecting the true positive TP over the total of predicted positives TP + FN.

Recall that equation (4) is based on the detection of true positives out of the total of true positives + FN. The f1 score in equation (5) is intended to balance the results. Equation (6) balances precision and recall in the case of unevenly distributed data sets.

The confusion matrix is a table that summarizes the results of a classification model's predictions. It evaluates the model's predictions against the actual values of the data set, and classifies them into four categories: true positives, true negatives, false positives and false negatives. The confusion matrix is used to compare the overall precision, recall, specificity and accuracy of a model. This choice allows a complete analysis of the model's performance. We can assess its ability to distinguish between positive and negative elements, while avoiding false classifications (false positives and false negatives).

4. Results

The first part of our results will be devoted to evaluating and comparing the performance of the different algorithms we have used. These algorithms are MobileNetV2, Xception, VGG16, and ResNet50V2. Each algorithm was tested and analyzed, enabling us to determine its specific effectiveness in identifying and recognizing mango varieties. We will evaluate the main parameters of each algorithm, which will give us an overview of its ability to meet the needs of the study and this challenge.

Precision measures the proportion of correct predictions among elements identified as positive by the model (true positives). It assesses the model's ability to avoid false positives (negative elements classified as positive). Sensitivity or recall represents the proportion of positive elements correctly identified by the model (true positives). It assesses the model's ability to avoid missing positive

elements (false negatives). Specificity indicates the proportion of negative elements correctly identified by the model (true negatives). It measures the model’s ability to avoid false positives (negative elements classified as positive). F1 Score combines accuracy and sensitivity in a single measure, calculated as the harmonic mean of the two. It provides a balance for assessing the model’s ability to correctly identify positive and negative elements.

This choice allows a complete analysis of the model’s performance. We can assess its ability to distinguish between positive and negative elements, while avoiding false classifications (false positives and false negatives).

Table 3 shows the metric measurements for each model.

Table 3. Parameters of the convolutional neural network.

| Model | Accuracy (%) | Precision (%) | F1-Score | Recall (%) | MCC (%) | MSE | Time (ms) |
|-------------|--------------|---------------|----------|------------|---------|-------|-----------|
| MobileNetV2 | 98.86 | 98.95 | 98.86 | 98.86 | 98.77 | 17.09 | 7255 |
| Xception | 99.43 | 99.47 | 99.43 | 99.48 | 99.38 | 9.11 | 200 |
| ResNet50 V2 | 97.15 | 97.39 | 97.08 | 97.15 | 96.94 | 0.002 | 160 |
| VGG16 | 99.15 | 83.25 | 83.25 | 85.47 | 15.44 | 31.44 | 360 |

Analysis of the above table highlights the performance and trends of the different models. Each model has different characteristics in terms of accuracy, precision, F1 score, recall, Matthews correlation coefficient (MCC) and mean square error (MSE).

Figure 3 shows a broken-line graph representing the relationship between the “learning accuracy” and “validation accuracy” of the MobileNetV2 model.

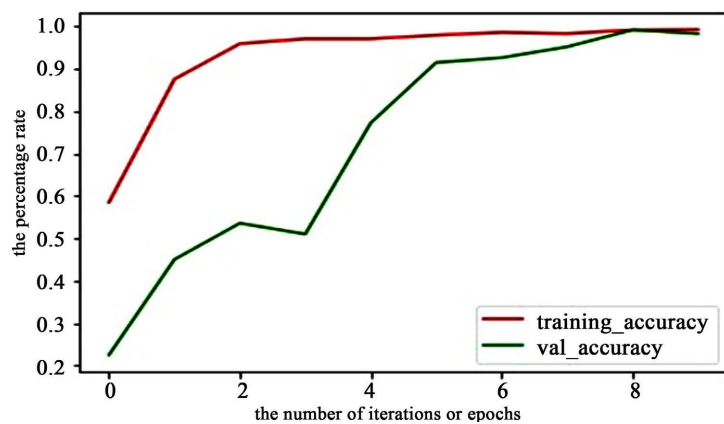


Figure 3. Training accuracy graphs for MobileNetV2.

The two axes are graduated from 0 to 100%. The curve starts at around 60% training accuracy and gradually rises until it reaches a peak of around 99% training accuracy. The curve then falls slightly to stabilize at around 98.95% va-

validation accuracy.

At the beginning of the curve, the training accuracy is relatively low (around 60%). This indicates that the machine learning model is not yet well adapted to the training data. As the model learns, the training accuracy increases, indicating a better ability to correctly predict the labels in the data.

The curve peaks at around 99% training accuracy. This means that the model is getting very accurate results on the training data. After reaching its peak, the curve falls slightly to stabilize at around 98.95% validation accuracy. This suggests that the model generalizes well to the validation data, which is an important indicator of its performance in real-life situations. The slight difference between training accuracy and validation accuracy may be due to a slight over-fitting of the model to the training data.

Figure 4 shows a broken-line graph representing the relationship between “training jobs” and “validation losses” of the MobileNetV2 model. At the start of the curve, validation losses are high, indicating that the machine learning model is failing to correctly predict the labels in the data. As the number of learning tasks increases, validation losses decrease, indicating that the model is getting better at predicting labels. The curve shows a downward trend in validation losses, indicating that the model is converging towards an optimal solution. Convergence implies that the model is learning efficiently and that validation losses no longer fluctuate significantly.

The curve reaches a minimum validation loss value of around 0.2. This means that the model reaches its best performance at this learning point.

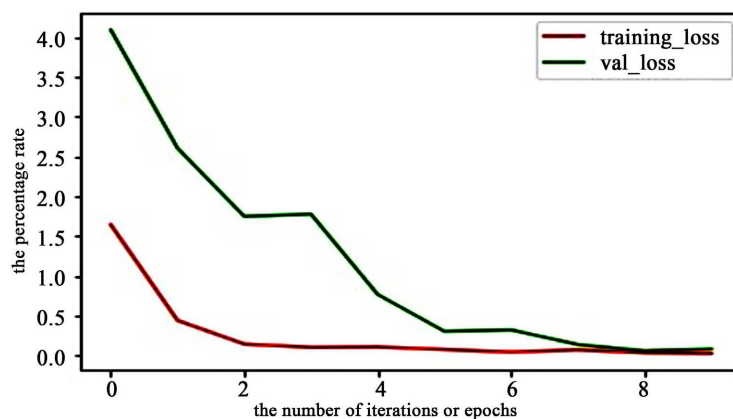


Figure 4. Training loss graphs for MobileNetV2.

The curve shows that the machine learning model converges towards an optimal solution, with validation losses decreasing as the number of training jobs increases.

Figure 5 shows the confusion matrix for our two-class classification model using the MobiNetV2 algorithm. The two classes are represented by the rows and columns of the matrix. The cells of the matrix indicate the number of examples classified by the model in a given class, when they actually belong to another class.

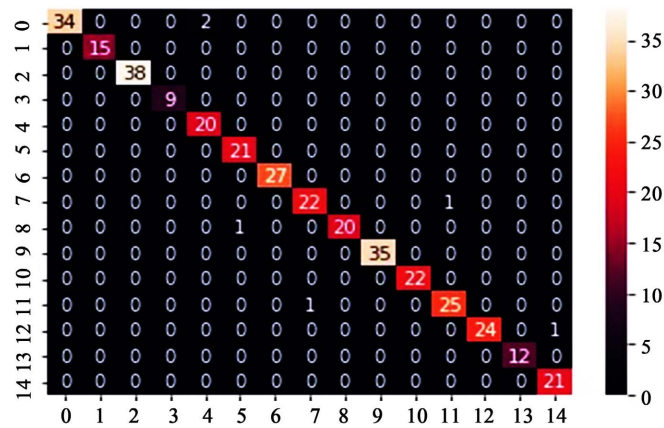


Figure 5. Confusion Matrix for MobileNetV2.

Figure 6 shows the training graph for the Xception model. The results reveal a high-performance model with a precision of 99.47% and an overall accuracy of 99.43%. The F1 score is 99.43%, while recall is 99.48%. Finally, the Matthews correlation coefficient (MCC) is 99.38% for Xception, confirming the model’s excellent performance.

Analysis of the learning curve for the Xception algorithm reveals encouraging results for mango variety detection. The model shows rapid convergence towards an optimal solution, better performance on training data than on validation data, and satisfactory generalization capability.

Figure 6 illustrates the learning curve of the Xception algorithm, highlighting the evolution of learning and validation losses as a function of epochs. The Xception model converges

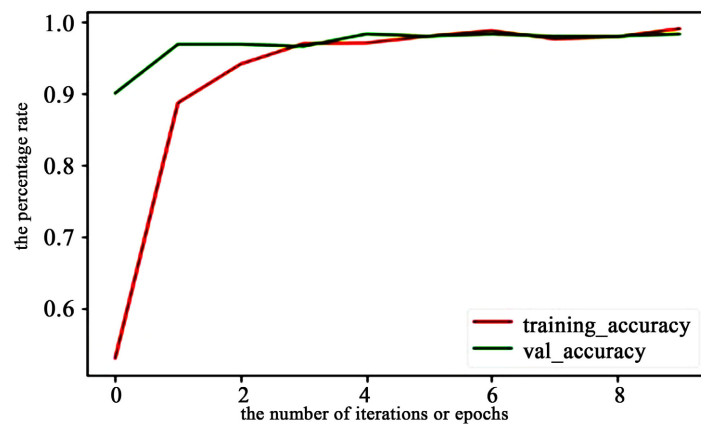


Figure 6. Training accuracy graphs for Xception.

Figure 7 shows the loss curve of the Xception algorithm for an image classification task. The curve represents the relationship between the x-axis, which corresponds to the number of training epochs, and the y-axis, which represents the training loss. Training loss is a measure of the model’s error on training data. A lower training loss indicates that the model predicts image labels better.

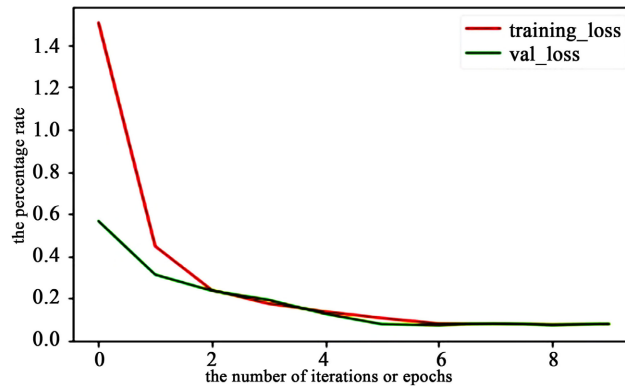


Figure 7. Training loss graphs for Xception.

Figure 8 shows the confusion matrix for the Xception model, used to evaluate its performance in classifying mango varieties. The matrix represents the relationship between the model’s predictions and the actual image labels. The columns represent the actual image classes. Each cell of the matrix contains the number of images belonging to a real class and predicted by the model as belonging to another class. The confusion matrix reveals a high overall classification performance for the Xception model.

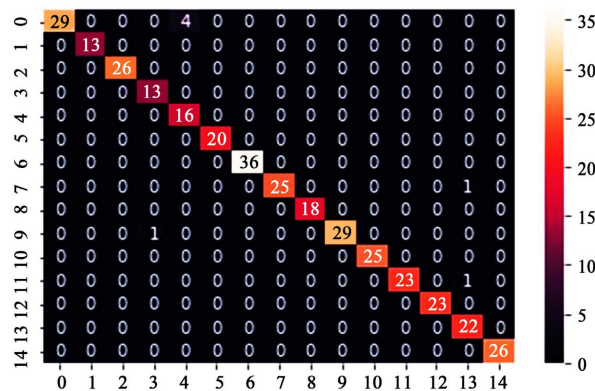


Figure 8. Confusion Matrix for Xception.

The VGG16 model was tested on a separate validation dataset, and the results are presented in Figures 9-11, namely the training curve, the loss curve and the confusion matrix.

The VGG16 model achieved an impressive accuracy score of 99.15%, indicating its ability to correctly classify the majority of images in the dataset. The precision score of 83.25% reinforces this point, underlining the model’s ability to accurately identify positive examples (images belonging to the target class) while minimizing false positives.

The F1-Score, which combines precision and recall, reached 83.25%, demonstrates the model’s balanced performance in both identifying positive examples and avoiding false negatives. The recall score of 85.47% specifically indicates the model’s ability to correctly identify all positive examples.

Figure 9 shows a graph representing the learning and validation accuracy of the VGG16 model for an image classification task. The fact that the validation accuracy curve exceeds the learning accuracy curve at one point suggests that the model may be overlearning the training data.

Figure 10 shows the loss curve of the VGG16 model for an image classification task. Loss is a measure of model error when predicting image classes. A loss curve can be used to evaluate the model’s performance during training, and to detect possible under- or over-learning problems.

The confusion matrix presented in **Figure 11** shows an average classification performance for the VGG16 model. The majority of images are correctly classified, with some misclassification. The main diagonal contains moderate values, with 83% of images correctly classified.

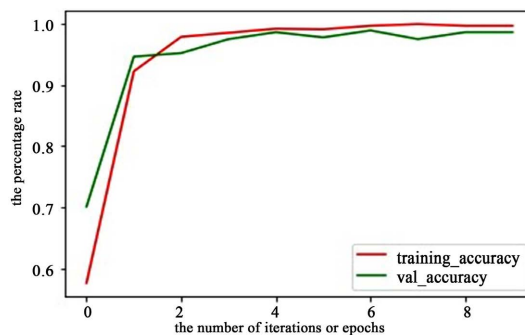


Figure 9. Training accuracy graphs for VGG16.

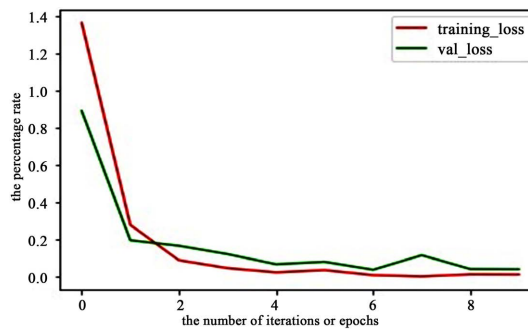


Figure 10. Training loss graphs for VGG16.

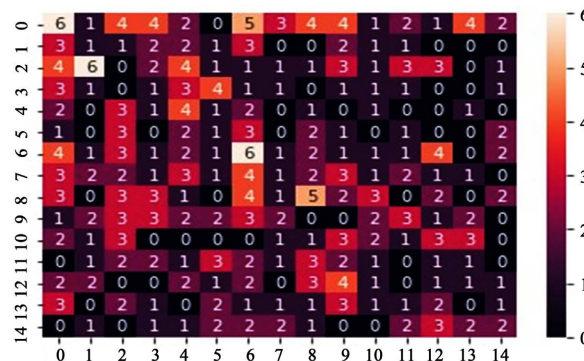


Figure 11. Confusion Matrix for VGG16.

The ResNet50V2 model was evaluated on a dedicated test dataset. The model achieves a precision score of 97.39% and an F1 score of 97.08%, indicating an outstanding ability to correctly identify the classes present in the images. Recall, slightly lower than precision, reflects the proportion of correctly identified images among those actually present in the class.

Accuracy (97.15%) and recall (97.15%) are not the only relevant performance measures. The ResNet50V2 model also scored 96.94% for the Matthews Correlation Coefficient (MCC), which takes into account both accuracy and recall. The MSE (Mean Squared Error) shows a value of 0.0029, indicating a low mean squared error.

In addition to its excellent classification performance, the ResNet50V2 model stands out for its speed of execution. The average execution time over ten epochs is 160 ms, making it ideally suited to applications requiring high reactivity.

Figure 12 of training and validation accuracy of the ResNet50V2 model shows promising learning performance for the image classification task. The model converges quickly to an optimal solution, performs better on training data than on validation data, and shows no signs of overlearning.

Figure 13 shows the loss curve during training of the ResNet50V2 model. The model learns efficiently from the data, and its prediction error gradually decreases during training. The model converges rapidly to an optimal solution, achieves a low training loss, and shows no signs of overlearning.

Figure 14 shows a confusion matrix for the ResNet50V2 model, used to evaluate its performance in classifying mango varieties. The matrix represents the relationship between the model's predictions and the true labels of the mango images.

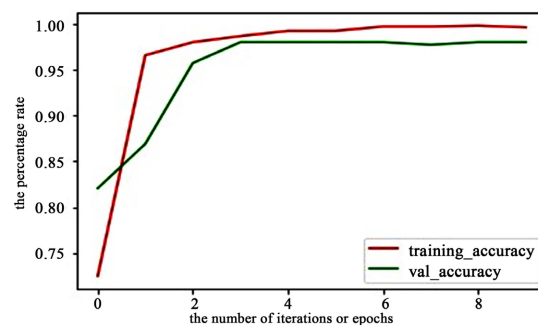


Figure 12. Training accuracy graphs for ResNet50V2.

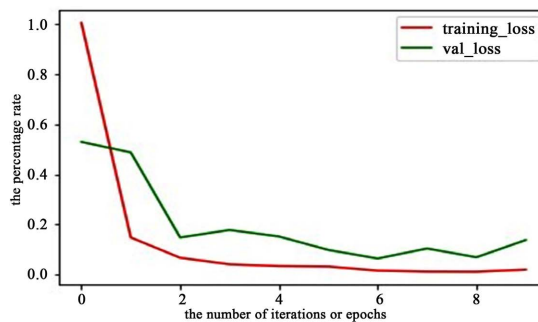


Figure 13. Training loss graphs for ResNet50V2.

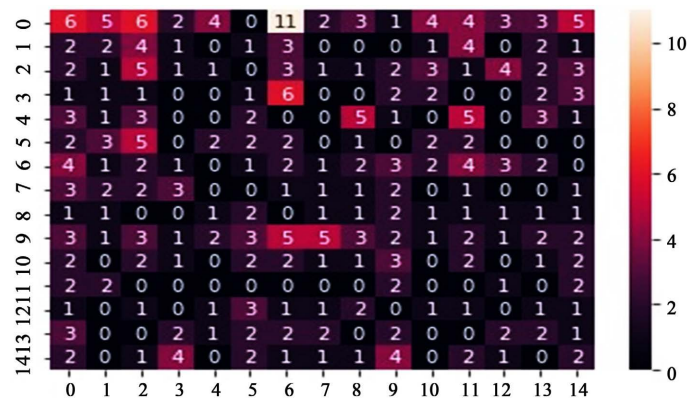


Figure 14. Confusion Matrix for ResNet50V2.

5. Discussion

Analysis of the results highlights key points concerning the performance of the different models evaluated in the mango variety detection task.

In our study, we examined four models for identifying mango varieties. In terms of performance, MobileNetV2 achieved interesting results with an accuracy rate of 98.95%. This indicates that our approach with the MobileNetV2 algorithm is capable of recognizing mango varieties with an accuracy of 98.95%. In addition, its F1 Score and balanced recall at 98.85% and 98.86%, respectively, indicate its ability to identify true positives while maintaining a balance with false negatives.

The Xception machine learning model stands out for its excellent, balanced performance in mango variety recognition. Indeed, it achieves an accuracy rate of 99.47%, an accuracy of 99.43%, an F1-Score of 99.43%, and a recall of 99.43%, clearly outperforming the MobileNetV2 model. What's more, Xception demonstrated notable robustness in detecting true positives, confirming its reliability in this crucial task.

The results of this study clearly demonstrate the superiority of the Xception model over MobileNetV2 for mango variety recognition. Key performance indicators such as precision, F1-Score, and recall testify to Xception's ability to accurately identify different mango varieties. This outstanding performance is due to Xception's advanced architecture, which enables it to capture more discriminating characteristics of mango images. Xception's robustness in detecting true positives is another major asset. This means that the model is able to minimize false positives.

The model based on the VGG16 algorithm has a precision of 83.25%, an F1-Score of 83.25%, and a recall of 85.47%. While these results are not negligible, they remain lower than those of the MobileNetV2 and Xception models. This observation suggests a tendency to generate false positives, *i.e.*, cases where a mango variety is incorrectly identified. Indeed, VGG16 displays the lowest precision of all the models studied, which is corroborated by its low recall and F1-Score, indicating difficulty in correctly identifying all true positives.

The performance of the VGG16 model for mango variety recognition raises questions about its effectiveness and reliability compared with the MobileNetV2 and Xception models. The propensity to generate false positives can be problematic in practical applications, such as mango sorting and grading, as it can lead to costly errors. In addition, the low precision, recall, and F1-Score of the VGG16 model indicate a difficulty in capturing the distinctive characteristics of different mango varieties. This could be due to the architecture of the VGG16 model, which is less suited to the fruit image recognition task than more recent architectures such as MobileNetV2 and Xception.

The study highlights the limitations of the VGG16 model for mango variety recognition. Although the model has some accuracy, it is outperformed by the MobileNetV2 and Xception models in terms of performance and reliability.

The ResNet50V2 model stands out for its remarkable performance in recognizing mango varieties, with an accuracy of 97.39%, an F1-Score of 97.08% and a recall of 97.17%. These results demonstrate the model's ability to correctly identify different mango varieties with high accuracy.

Although the ResNet50V2 model shows promising performance, it is important to point out that it is slightly outperformed by the MobileNetV2 and Xception models in terms of accuracy and F1-Score. Nevertheless, ResNet50V2's performance remains highly satisfactory, confirming its potential for mango variety recognition. Its high efficiency in terms of precision, recall, and F1-Score indicates its ability to capture the distinctive characteristics of different mango varieties with great reliability. The study highlights the promising performance of the ResNet50V2 model for mango variety recognition. Although slightly outperformed by the MobileNetV2 and Xception models, ResNet50V2 remains an effective and reliable tool for this task. **Figure 15** shows the metric histogram of the different models studied for mango variety recognition.

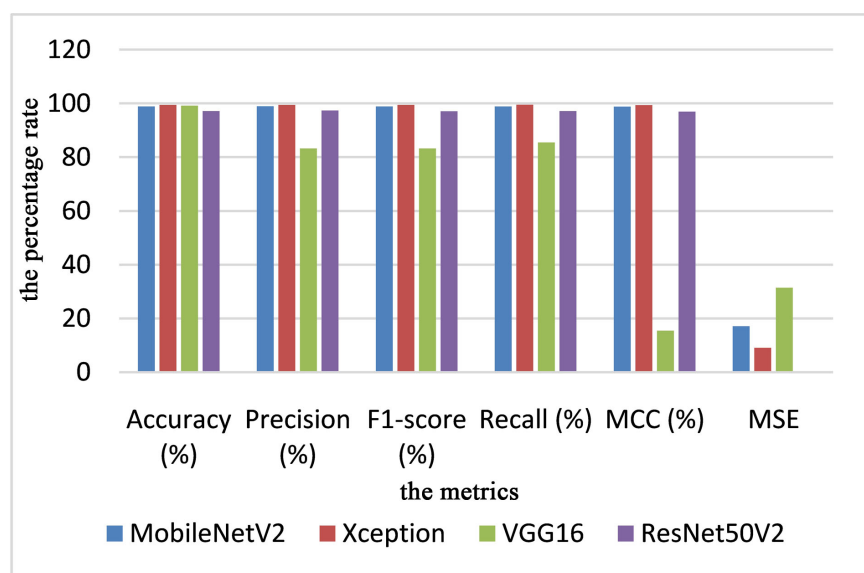


Figure 15. Histogram of the metrics of our different machine learning models.

The Xception model stands out for its exceptional accuracy of 99.47% in detecting mangoes. Its complex architectural structure enables it to extract distinctive features from mango images with great finesse, which explains its excellent accuracy performance. However, this complexity results in an inference time of 200 ms, the second slowest of the models analysed. This means that image processing takes longer with Xception, which can be a drawback for some applications.

MobileNetV2 offers an excellent compromise between accuracy and inference speed. Its accuracy of 98.95% is close to that of Xception, while its inference time of 7255 ms is significantly faster. This compromise is made possible by its lightweight, optimized structure, which enables the model to achieve high accuracy while maintaining relatively fast execution.

ResNet50V2 favours speed of inference over absolute accuracy. Its accuracy of 97.15% is the lowest of the high-performance models, but its inference time of 160 ms is the fastest.

This model offers a good balance between accuracy and inference speed, making it suitable for applications where real-time performance is crucial.

VGG16 has the third fastest inference speed (360 ms), but its accuracy for mangoes is significantly lower than the other models (83.25%). This low accuracy for mangoes suggests a potential over-fitting of the model to the training data, which limits its ability to generalize to new images.

Xception is the best choice, but if speed of inference is also crucial, MobileNetV2 offers an excellent compromise. ResNet50V2 may be an alternative if a slightly tighter balance towards accuracy is required. VGG16, despite its speed, is not recommended for mango detection due to its low accuracy for this category.

6. Comparison between Existing Methods

Our study demonstrates superior performance to previous work in the field of mango variety detection and identification.

Table 4 shows the results obtained when comparing the performance of the different models. An in-depth analysis of the data reveals that our models outperform the models proposed by the other three authors.

Table 4. Comparison table.

| Method | Accuracy (%) |
|------------------------------------|--------------|
| Gururaj, N <i>et al.</i> [29] | 93.23 |
| Bhole, V <i>et al.</i> [30] | 93.33 |
| H. C. Pichhika <i>et al.</i> [31] | 94.4 |
| Our method with ResNet50V2 | 97.39 |
| Our method with MobileNetV2 | 98.95 |
| Our method with Xception | 99.47 |

The results presented in **Table 4** clearly illustrate the superiority of our models in terms of performance. Indeed, our models achieve a significantly higher

level of accuracy than those of the comparative models. This observation confirms the validity of our approach and highlights the effectiveness of our models in the mango variety recognition task.

The study presents an interesting analysis of the use of deep learning for mango variety detection. However, a comparison with traditional methods of variety detection, such as visual analysis by experts or the use of techniques based on predefined rules. Business rules, defined by experts in the field, could be integrated into the models to guide the classification process. This comparison would enable us to better situate the real contribution of deep learning in this field and to determine whether it offers significant advantages over existing approaches.

7. Conclusions

Accurate identification of mango varieties is crucial for efficient classification, sorting, and marketing. This study compares the performance of four machine learning models (MobileNetV2, Xception, VGG16, and ResNet50V2) in the mango variety detection task.

Our machine-learning models were trained on a dataset of annotated mango images. Model performance was evaluated in terms of precision, F1 score, and recall.

The Xception model outperformed the other models in terms of precision, F1 score, and recall, with a precision rate of 99.47%, an accuracy of 99.43%, an F1 score of 99.43%, and a recall of 99.43%, followed respectively by the MobileNetV2 model with performances of 98.95%, 98.85%, and 98.86% and the ResNet50V2 with 97.39%, 97.08% and 97.17% as respective performances. Finally, the VGG16 model has the lowest performance, with a precision rate of 83.25%, an F1 score of 83.25%, and a recall of 85.47%.

Our study opens up promising prospects for mango variety recognition using machine learning. The results obtained, in particular the exceptional performance of the Xception model, demonstrate the potential of this technology to optimize mango classification and sorting processes.

Integrating high-performance machine learning models like Xception into mango sorting and grading systems could revolutionize the industry. It would automate these tedious, human error-prone tasks, leading to increased productivity, reduced costs and improved end-product quality.

In addition, machine learning could facilitate the precise identification of rare or valuable mango varieties, promoting their commercialization and valorization in specialized markets. This technology could also be extended to the recognition of other fruits and agricultural products, helping to improve the efficiency of sorting and grading processes in various agri-food sectors.

In our view, a study could effectively integrate machine learning models into existing sorting and grading systems, and an extension of machine learning to the detection of fruit diseases and defects could provide additional added value.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Worasawate, D., Sakunasinha, P. and Chiangga, S. (2022) Automatic Classification of the Ripeness Stage of Mango Fruit Using a Machine Learning Approach. *AgriEngineering*, **4**, 32-47. <https://doi.org/10.3390/agriengineering4010003>
- [2] Maqbool, I., Qadri, S., Khan, D.M. and Fahad, M. (2015) Identification of Mango Leaves by Using Artificial Intelligence. *International Journal of Natural and Engineering Sciences*, **9**, 45-53.
- [3] Ibrahim, M., Ahmad Sa'ad, F., Zakaria, A. and Md Shakaff, A. (2016) In-Line Sorting of Harumanis Mango Based on External Quality Using Visible Imaging. *Sensors*, **16**, Article 1753. <https://doi.org/10.3390/s16111753>
- [4] Tripathi, M.K., Reddy, P.K., Neelakantappa, M., Andhare, C.V. and Shivendra, S. (2023) Identification of Mango Variety Using near Infrared Spectroscopy. *Indonesian Journal of Electrical Engineering and Computer Science*, **31**, 1776-1783. <https://doi.org/10.11591/ijeecs.v31.i3.pp1776-1783>
- [5] Mustaffa, M.R., Idris, A.A., Abdullah, L.N. and Nasharuddin, N.A. (2023) Deep Learning Mango Fruits Recognition Based on Tensorflow Lite. *International Journal of Advances in Intelligent Informatics*, **9**, 565-576. <https://doi.org/10.26555/ijain.v9i3.1368>
- [6] Truong Minh Long, N. and Truong Thinh, N. (2020) Using Machine Learning to Grade the Mango's Quality Based on External Features Captured by Vision System. *Applied Sciences*, **10**, Article 5775. <https://doi.org/10.3390/app10175775>
- [7] Koirala, A., Walsh, K.B., Wang, Z. and Anderson, N. (2020) Deep Learning for Mango (*Mangifera indica*) Panicle Stage Classification. *Agronomy*, **10**, Article 143. <https://doi.org/10.3390/agronomy10010143>
- [8] Bhargava, A. and Bansal, A. (2021) Machine Learning-Based Detection and Grading of Varieties of Apples and Mangoes. In: Goyal, V., Gupta, M., Trivedi, A. and Kolhe, M.L., Eds., *Proceedings of International Conference on Communication and Artificial Intelligence*, Springer, 455-462. https://doi.org/10.1007/978-981-33-6546-9_43
- [9] Baculo, M.J.C., Ruiz, C. and Aran, O. (2021) Cecid Fly Defect Detection in Mangoes Using Object Detection Frameworks. In: Magnenat-Thalmann, N., *et al.*, Eds., *Advances in Computer Graphics*, Springer, 205-216. https://doi.org/10.1007/978-3-030-89029-2_16
- [10] Xiang, Q., Wang, X., Li, R., Zhang, G., Lai, J. and Hu, Q. (2019) Fruit Image Classification Based on Mobilenetv2 with Transfer Learning Technique. *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, Sanya, 22-24 October 2019, 1-7. <https://doi.org/10.1145/3331453.3361658>
- [11] Gulzar, Y. (2023) Fruit Image Classification Model Based on Mobilenetv2 with Deep Transfer Learning Technique. *Sustainability*, **15**, Article 1906. <https://doi.org/10.3390/su15031906>
- [12] Souid, A., Sakli, N. and Sakli, H. (2021) Classification and Predictions of Lung Diseases from Chest X-Rays Using Mobilenet V2. *Applied Sciences*, **11**, Article 2751. <https://doi.org/10.3390/app11062751>

- [13] Kumar Shukla, R. and Kumar Tiwari, A. (2023) Masked Face Recognition Using Mobilenet V2 with Transfer Learning. *Computer Systems Science and Engineering*, **45**, 293-309. <https://doi.org/10.32604/csse.2023.027986>
- [14] Bouguezzi, S., Fredj, H.B., Belabed, T., Valderrama, C., Faiedh, H. and Souani, C. (2021) An Efficient FPGA-Based Convolutional Neural Network for Classification: Ad-Mobilenet. *Electronics*, **10**, Article 2272. <https://doi.org/10.3390/electronics10182272>
- [15] Chhabra, M. and Kumar, R. (2022) An Advanced VGG16 Architecture-Based Deep Learning Model to Detect Pneumonia from Medical Images. In: Marriwala, N., Tri-pathi, C.C., Jain, S. and Mathapathi, S., Eds., *Emergent Converging Technologies and Biomedical Systems*, Springer, 457-471. https://doi.org/10.1007/978-981-16-8774-7_37
- [16] Lu, X. and Firoozeh Abolhasani Zadeh, Y.A. (2022) Deep Learning-Based Classification for Melanoma Detection Using Xceptionnet. *Journal of Healthcare Engineering*, **2022**, Article ID: 2196096. <https://doi.org/10.1155/2022/2196096>
- [17] Lin, C., Li, L., Luo, W., Wang, K.C.P. and Guo, J. (2018) Transfer Learning Based Traffic Sign Recognition Using Inception-V3 Model. *Periodica Polytechnica Transportation Engineering*, **47**, 242-250. <https://doi.org/10.3311/pptr.11480>
- [18] Shaik, N.S. and Cherukuri, T.K. (2022) Visual Attention Based Composite Dense Neural Network for Facial Expression Recognition. *Journal of Ambient Intelligence and Humanized Computing*, **14**, 16229-16242. <https://doi.org/10.1007/s12652-022-03843-8>
- [19] Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, 21-26 July 2017, 1800-1807. <https://doi.org/10.1109/cvpr.2017.195>
- [20] Müller, K.R., Mika, S., Tsuda, K. and Schölkopf, K. (2018) An Introduction to Kernel-Based Learning Algorithms. In: Müller, K.R., Mika, S., Tsuda, K. and Schölkopf, K., Eds., *Handbook of Neural Network Signal Processing*, CRC Press, 1-4.
- [21] Ghosh, S., Dasgupta, A. and Swetapadma, A. (2019). A Study on Support Vector Machine Based Linear and Non-Linear Pattern Classification. 2019 *International Conference on Intelligent Sustainable Systems (ICISS)*, Palladam, 21-22 February 2019, 24-28. <https://doi.org/10.1109/iss1.2019.8908018>
- [22] Fan, Q., Chen, C.F.R., Kuehne, H., Pistoia, M. and Cox, D. (2019) More Is Less: Learning Efficient Video Representations by Big-Little Network and Depthwise Temporal Aggregation. *Advances in Neural Information Processing Systems*, **32**.
- [23] Zheng, K., Gao, L., Ran, Q., Cui, X., Zhang, B., Liao, W., et al. (2019) Separable-Spectral Convolution and Inception Network for Hyperspectral Image Super-Resolution. *International Journal of Machine Learning and Cybernetics*, **10**, 2593-2607. <https://doi.org/10.1007/s13042-018-00911-4>
- [24] Diarra, M., Jean, A.K., Bakary, B.A. and Medard, K.B. (2021) Study of Deep Learning Methods For Fingerprint Recognition. *International Journal of Recent Technology and Engineering (IJRTE)*, **10**, 192-197. <https://doi.org/10.35940/ijrte.c6478.0910321>
- [25] Duarte, J.M. and Berton, L. (2023) A Review of Semi-Supervised Learning for Text Classification. *Artificial Intelligence Review*, **56**, 9401-9469. <https://doi.org/10.1007/s10462-023-10393-8>
- [26] Bottou, L. (2010) Large-Scale Machine Learning with Stochastic Gradient Descent. *Proceedings of COMPSTAT2010: 19th International Conference on Computational*

- Statistics*, Paris, 22-27 August 2010, 177-186.
https://doi.org/10.1007/978-3-7908-2604-3_16
- [27] Liu, Y., Gao, Y. and Yin, W. (2020) An Improved Analysis of Stochastic Gradient Descent with Momentum. *Advances in Neural Information Processing Systems*, **33**, 18261-18271.
- [28] Mignacco, F. and Urbani, P. (2022) The Effective Noise of Stochastic Gradient Descent. *Journal of Statistical Mechanics: Theory and Experiment*, **2022**, Article ID: 083405. <https://doi.org/10.1088/1742-5468/ac841d>
- [29] Gururaj, N., Vinod, V. and Vijayakumar, K. (2022) Deep Grading of Mangoes Using Convolutional Neural Network and Computer Vision. *Multimedia Tools and Applications*, **82**, 39525-39550. <https://doi.org/10.1007/s11042-021-11616-2>
- [30] Bhole, V. and Kumar, A. (2020) Mango Quality Grading Using Deep Learning Technique: Perspectives from Agriculture and Food Industry. *Proceedings of the 21st Annual Conference on Information Technology Education*, 7-9 October 2020, 180-186. <https://doi.org/10.1145/3368308.3415370>
- [31] Pichhika, H.C. and Subudhi, P. (2023) Detection of Multi-Varieties of On-Tree Mangoes Using MangoYOLO5. 2023 11th *International Symposium on Electronic Systems Devices and Computing (ESDC)*, Sri City, 4-6 May 2023, 1-6. <https://doi.org/10.1109/esdc56251.2023.10149849>