Science in Information Technology Letters
Vol. 5 ., No. 1, May 2024, pp. 1-15
ISSN 2722-4139
http://pubs2.ascee.org/index.php/sitech

Classification of plants by their fruits and leaves using convolutional neural networks

Martins E. Irhebhude [a,1,*], Adeola O. Kolawole [a,2], Chat Chinyio [a,3]

[a] Department of Computer Science, Nigerian Defence Academy, Kaduna, Nigeria
[1] mirhebhude@nda.edu.ng; [2] adeolakolawole@nda.edu.ng; [3] chateechinyio@gmail.com
* Corresponding Author

ARTICLE INFO

Article history
Received March 24, 2024
Revised April 07, 2024
Accepted April 15, 2024

Keywords
Fruit classification
Leave classification
Plants classification
Sequential cnn
Computer vision

ABSTRACT

The population growth of the world is exponential, this makes it imperative that we have an increase in food production. In this light, farmers, industries and researchers are struggling with identifying and classifying food plants. Over the years, there have been challenges that come with identifying fruits manually. It is time-consuming, labour intensive and requires experts to identify fruits because of the similarity in fruit's leaves (citrus family), shapes, sizes and colour. A computerized detection technique is needed for the classification of fruits. Existing solutions to fruits classifications are majorly based on fruit or leave used as input. A new model using Convolutional Neural Network (CNN) is proposed for fruits classification. A dataset of 5 classes of fruits and fresh & dry leaves plants (Mango, African almond, Guava, Avocado and Cashew) comprising of 1000 images each. The proposed model hyperparameters were: Conv2D layer, activation layer, dense layer, a learning and dropout rates of 0.001 and 0.5 respectively were used for the experiment. Various performances for accuracies of 91%, 97%, 78% and 97% were obtained for proposed model on local dataset, proposed model on benchmark dataset, benchmark model on local dataset and benchmark model on benchmark dataset. The proposed model is robust on both local and benchmark datasets and can be used for effective classification of plants.

1. Introduction

Over the years, intelligent systems based on computer vision and machine learning have been developed for fruit defect detection, ripeness grading, and categorization [1], [2]. The diversity of different fruits plays a very important role in various areas such as medical science, foodstuff, environment protection, and industrial growth [3]. Automatic fruit classification is a fascinating subject in the fruit growing and retailing industry, since it can assist fruit producers and supermarkets in identifying different fruits and their condition from stock or containers, hence increasing production efficiency and profit [1]. Plants can be identified by their parts, such as its seeds, flowers, fruits, leaves, and stems. When compared to other plant parts, leaves in particular play a significant role because they are easily accessible, abundant, and distinguishable in all seasons [4]. One of the trusted authorities that can give appropriate categorization of various plants are the botanists, and if employed could increase consultation costs for industries and waste of time to classify these plants from person to person and in varying locations [5].

Botanists and computer scientists have conducted extensive research on the identification of plants [6]. Building numerous intelligent sub-systems for the identification of herbs has been encouraged by the emergence of domain areas like IoT, computer vision, and machine learning, which is still a difficult task [7]. Fruits on the other hand, are the clearest, the most distinctive feature in identifying plants, but they are seasonal, in the seasons when plants have not begun to produce fruits, the trees are predominantly occupied by their leaves and this can also be another component for identifying plants, but it can lead to some sort of misclassification of these plants as some leaves from different plants have similar texture, colour and shape like the citrus family, but this is not sufficient when it comes to reliability of the plant classification model [8]. The process of identification, classification, and grading of fruits if not done precisely can be slow, labour intensive and tedious due to human perception subjectivity [9]. The most popular technique for identifying leaves is human visual inspection, this method requires time and efforts, and its margin of error depends on the visual ability to distinguish various components of the plant, such as the leaves and fruits, as well as the possibility of making mistakes exists [10], [11] highlighted the need to implement an automation system for the fruit sector. Machine learning approaches combined with appropriate image processing principles have a lot of potential for providing intelligence for developing an automation system that can differentiate fruit; types, varieties, maturity, and intactness [12]. Different image processing and computer vision techniques have been used for the automatic detection, identification, classification and segmentation of fruits from their colours, shapes and textures [2], [13]. The classification of plants using machine learning and deep learning approaches has been looked into by various authors using a single part of the plant, such as the leaves, flowers and fruits [2], [14]–[17]. Because various plants frequently have identical organs, it is not adequately dependable to identify a plant using only one component. This study focuses on the classification of selected plants by their fruits and leaves using Convolutional Neural Network.

Reference [2] ripe, unripe and defective mangoes and oranges were classified and predicted using a recognition system. SVM and decision tree algorithm (DTA) were used to classify the fruit images into defective, ripe, and unripe for local datasets, while public datasets were classified into ripe and unripe. In addition to being scaled and having background distortion removed, images also had their color and texture components retrieved. Histogram and Haralick texture features from each pre-processed image were recovered as feature vectors and used as transformation inputs. Additionally, locality preserving projection (LoPP) was used to compute the extracted features and used for classification. From the experiments using the LoPP dimension reduction algorithm on the extracted local features, resulted to 93.4% and 92.2% on mangoes and oranges respectively. [18] examined the classification of different plant species using CNN compact models (N1, N2, N3) and AlexNet. The models were trained and evaluated using the PlantVillage and Flavia datasets separately. The initial accuracies were 86.58%, 92.09%, 89.61% and 98.53% for N1, N2, N3 and AlexNet respectively. To improve the results, the data were augmentation resulting in improved accuracies of 99.45%, 99.65%, 99.55%, and 99.73% respectively for N1, N2, N3 models and AlexNet for PlantVillage dataset classified into nine plant species. Flavia dataset on the other hand was used to classify plants species into 32 classes as well as diseases in tomato plant leaves with accuracies of 99.17%, 99.59% 99.36% and 99.87% for N1, N2, N3 and AlexNet model respectively. Reference [15] proposed the identification of plant species using a combination of CNN layer embedded with the LSTM. Experiment was conducted on three datasets (Plant task, ICL leaf, LeafSnap dataset) using the proposed CNN-LSTM technique. Results obtained outperformed all other competent state-of-the-art methods achieving 95.06% accuracy. The research showed that the proposed system can be expanded to support an integrated plant species identification system to operate in real ecosystem services. Shahi, et al. [1] studied the classification of fruit images using attention-convolution module based MobileNetV2. Experiment was conducted using three publicly available datasets; D1, D2 and D3 and achieved 95.75%, 96.74%, and 96.23% accuracies respectively. D2 reported highest accuracy due to its light weight nature, and has substantial potential to be used by farms and industries that process and grow fruits for classification and identification of fruits. Koc and Vatandas [19] classified fruits according to their size and color features using image processing algorithm and classification techniques; Decision Tree, K Nearest Neighbour (KNN), MLP, Naive Bayes, and Random Forest (RF). A total of 300 samples of apple fruits (50 fruit samples from the Valencia Midnight and Washington Navel orange varieties, the Ekmek and Esme quince varieties, and the Starkrimson Delicious and Golden Delicious apple varieties) were used for the experiment. After training the classifiers, the result reported 93.6% for KNN, while DT, Naives Bayes, MLP and RF recorded 90.3%, 88.3%, 92.6%, and 94.3% respectively. With the technique used, the authors concluded that fruits can be classified both online and offline method. [20], proposed automatic fruit recognition system based on their texture, colour and shape. These features were blended into three categories; fruits colour and shape feature (FCSF), fruits colour and texture feature (FCTF) and fruits colour, shape and texture feature (FCSTF) for classification purposes. The fruit 360 dataset consisting of 2400 images of fruits with 24 classes of fruits was used for the experiment with five classifiers; Naive Bayes, KNN, linear discriminant analysis (LDA), DT and Error correcting output code (ECOC). The result showed K-NN recorded the best accuracy of 100% each on FCSF and FCTF while FCSTF reported 98.95%. [5] proposed a CNN based classification method for the classification of fruits to detect early crop disease damage. Experiments were conducted with a dataset of 200 images of fruits, containing 50 images of apples fruit, 50 images of mangoes, 50 images of oranges, and 50 images of grapes with MATLAB 2018b. Back Propagation Neural Network (BPNN), Support Vector Machine (SVM), and CNN classifiers were employed in the experiment. Among these, CNN demonstrated the highest accuracy at 90%, surpassing BPNN and SVM. The accuracy, specificity, and sensitivity values for different fruit types were used, where CNN achieved specific accuracies of 90% for grapes, 91% for mangoes, 88% for apples, and 91% for oranges, resulting in an overall accuracy of 90%. [21], proposed a model that combined CNN models to classify 5 fruits; apple, grapes, guava, mango and orange into leaves, fruits and unknown. The CNN model further classified the images of fruits or leaves as either healthy or disease and identifies the particular disease. Accuracy of above 90% was achieved on all the five classes of fruits and leaves, authors recommended addition of more images to give higher accuracy.

Reference [22], developed a method for diagnosing diseases and its symptoms, multilayer convolutional neural network (MCNN) model was used to classify mango leaves infected by Anthracnose fungal. Self-captured real-time dataset of healthy and unhealthy leaves consisting of 1070 images of Mango leaves and 1130 images from plantVillage dataset were used. The proposed method achieved 97.13% accuracy. Further comparison was done with other state of art approach; Particle Swarm Optimization (PSO) gave 88.39%, SVM resulted in 92.75%, and Radial Basis Function Neural Network (RBFNN) yielded 94.20% accuracies. To improve the classification rate for RGB and thermal image fruit datasets, [23] utilized color and texture-based features. Fruits were classified into eleven categories, Color (Color Moments and Color Coherence Vector) and texture (GLCM) feature extraction approaches were used. In order to improve the classification accuracy of various fruit images, features such as color and texture were combined to identify the images using RF and KNN classifiers. For RGB images, color features gave 100% accuracy with RF and KNN classifiers, however for thermal images, fusion of GLCM, color moments, and color coherence vector gave 93.4% accuracy with RF classifier. [24], proposed the use of image processing technique to grade tobacco leaf based on color and quality. They assessed the quality of the leaves by size, and position of leaf defects by the appearance of holes on the surface of the leaves. The proposed method was able to detect a leaf defect and categorize tobacco leaves into K (yellow), M (brown), and B (green) color groups with 91.667% accuracy. The categorization approach determines the local color category in small areas of the tobacco leaf image using a series of thresholds. Each patch's majority vote was used to define the global color category. [25], proposed a technique for leaf recognition, where refined color picture, vein image, xy-projection histogram, handcrafted texture features, and Fourier descriptors were features extracted from leaf images. These properties were then translated into a better representation by neural network-based encoders before being classified using SVM model. On the Flavia leaf dataset, the proposed algorithm achieved 99.58% accuracy on test sets under random 10-fold cross-validation, surpassing earlier methods. [26], proposed image processing techniques for the identification of 140 images of 70 apples and 70 oranges. The region of interest was segmented using Otsu thresholding method, then wavelet transformation was applied to extract statistical texture features on segmented images. SVM was used to classify images resulting in 100% accuracy. [27], proposed a shared dataset for medicinal plants and a CNN model based on Deep Learning named AyurLeaf to categorize medicinal plants using leaf attributes such as; shape, size, color, and texture. AyurLeaf dataset containing leaf samples from 40 medicinal plants was used for the experiment to efficiently extract features from the dataset, a deep neural network modeled after AlexNet was used with SVM classifiers to achieve 96.76% recognition accuracy.

By creating an accurate plant recognition model that categorizes particular plants according to both their fruits and leaves, this study aims to fill a significant gap in the existing literature. This research used a convolutional neural network (CNN) that was trained from scratch, contrary to other studies that primarily focused on particular plant components for classification. The model's accuracy and reliability will be significantly improved by incorporating datasets of both fruits and leaves, addressing the drawbacks of previous approaches. This research advances the science of plant classification by offering a more thorough and precise method that incorporates various plant components and makes applications in botany, agriculture, and other fields easier.

2. Method

The diagrammatic flow of the proposed methodology is shown in Fig. 1 indicating each block and the various stages involved, these steps are briefly described in the next subsection.

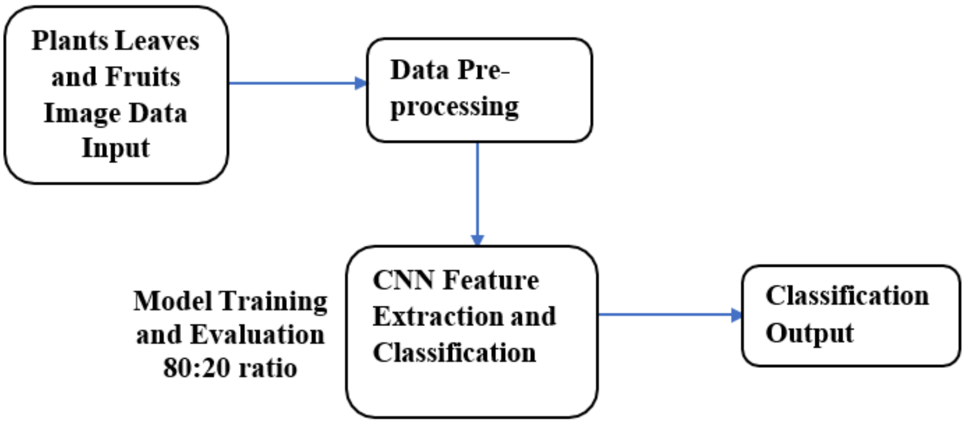

Fig. 1. Proposed Methodology

2.1. Plant Leaves and Fruits Data Input

Images of fruits and leaves were captured locally using a mobile phone camera and separated into different folders based on the classes. The dataset includes 5 classes of selected plant fruits and leaves namely; Mango, African almond, Guava, Avocado and Cashew. Each class comprise image of 300 fruits at the ripened stage and 700 leaves (500 fresh and 200 dry leaves) making 1000 images per class and total of 5000 images in the dataset respectively. Table 1, shows sample images of the fruits and leaves from the dataset.

2.2. Data preprocessing

To improve the quality of the image before applying any computational processes to it, image preprocessing is required [28]. The fully connected layers of convolutional neural networks required that all the images be in arrays of the same size. In this study, the dataset's input images were of varying sizes and shrunk to 224x224 pixels having input shape of (224, 224, 3) by resizing them after normalization to [-1, 1].

Table 1. Sample Images from Local Dataset

Class	Fruit (300)	Fresh Leave (500)	Dried Leave (200)
Cashew			
Avocado			
Guava			
Mango			
African almond			

2.3. Convolution Neural Network

One of the most common types of neural networks for image recognition and classification is the CNN, it works by developing a hierarchy of features that can be used for classification, rather than by manually creating features [29]. Building components like convolution layers, pooling layers, and fully connected layers are all part of the CNN architecture. The recurrence of a stack of numerous convolution layers and a pooling layer, followed by one or more fully connected layers, makes up a common design [30].

2.3.1. Convolution layer

In any CNN architecture, the convolutional layer is the most crucial element. It has a number of convolutional kernels (also known as filters), which are convolved with the input image's N- dimensional metrics to produce an output feature map [31]. A specific kind of linear operation called convolution is used to extract features. It applies a tiny array of numbers, known as a kernel, over the input, which is an array of numbers known as a tensor. The output value in the corresponding place of the output tensor is obtained by computing an element-wise product between each element of the kernel and the input tensor at each point of the tensor and summing it [30]. Various kernels can be thought of as different feature extractors, and this process is repeated, applying different kernels to produce an infinite number of feature maps that represent different properties of the input tensors. The two essential hyperparameters that characterize the convolution process are the size and the number of kernels. The size of the kernels is typically 3 × 3, but it can be extended to 5 × 5 or 7 × 7 to further deepen the depth of the output feature maps.

2.3.2. Pooling Layer

The feature maps generated after convolution operations are sub-sampled using the pooling layers, i.e., they are shrunk from their original bigger size to one of a smaller size. It always keeps the most important features (or information) in each pool step when the feature maps are shrunk [30]. Like the convolution operation, the pooling operation is carried out by defining the size of the pooled region and the operation stride. Various pooling methods, including max pooling, min pooling, average pooling, gated pooling, tree pooling, and others, are employed in pooling layer. The most well-liked and frequently employed pooling method is Max Pooling, this pooling layer enables CNN to determine if a particular feature is present in the input image or not without worrying about the feature's precise location [31].

2.3.3. Fully Connected Layer

The final convolution or pooling layer's output feature maps are typically flattened, or converted into a one-dimensional (1D) array of numbers (or vector), and then connected to one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a trainable weight. A subset of fully connected layers then maps these characteristics to the network's final outputs, such as the probabilities for each class in classification tasks, after they have been extracted by the convolution layers and down sampled by the pooling layers. Regularly, the number of output nodes in the final fully linked layer is equal to the number of classes. A nonlinear function, such as ReLU, follows each completely linked layer [30]. The final fully connected layer typically receives a different activation function than the others. Each task requires a different activation function, which must be chosen accordingly.

2.4. Feature Extraction and Classification Using CNN

The classification of plants by their fruits and leaves is actualized by using a developed CNN model. Two components make up a CNN classifier; the feature extraction part, which consists of a few convolution layers followed by pooling layers and an activation function and the classification part, which consists of a few fully connected layers and a loss function. The developed technique is named sequential CNN model. The sequential model was developed utilizing the parameters and the hyper tuned parameter as shown in Table 2.

Table 2. Hyper Parameter Tuning

Hyper-parameter	No. of layer in the model
Conv2D layer	4
Activation layer	7
Batch Normalization layer	6
Max Pooling2D layer	4
Dropout layer	6
Flatten layer	1
Dense layer	3
Learning rate	0.001
Epochs	90
Dropout rate	0.5
Activation Function	Elu

Table 2 depicts the complete sequential model built with various layers, as well as the trainable and non-trainable parameters at each layer. Fig. 2 depicts the snapshots of sequential model built with various layers, as well as the trainable and non-trainable parameters at each layer.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 224, 224, 16)      160

 activation (Activation)     (None, 224, 224, 16)      0

 batch_normalization (BatchN  (None, 224, 224, 16)     64
 ormalization)

 max_pooling2d (MaxPooling2D  (None, 112, 112, 16)     0
 )

 dropout (Dropout)           (None, 112, 112, 16)      0

 conv2d_1 (Conv2D)           (None, 112, 112, 32)      4640

 activation_1 (Activation)   (None, 112, 112, 32)      0

 batch_normalization_1 (Batc  (None, 112, 112, 32)     128
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 56, 56, 32)       0
 2D)

 dropout_1 (Dropout)         (None, 56, 56, 32)        0

 conv2d_2 (Conv2D)           (None, 56, 56, 64)        18496

 activation_2 (Activation)   (None, 56, 56, 64)        0

 batch_normalization_2 (Batc  (None, 56, 56, 64)       256
 hNormalization)

 max_pooling2d_2 (MaxPooling  (None, 28, 28, 64)       0
 2D)

 dropout_2 (Dropout)         (None, 28, 28, 64)        0
 conv2d_3 (Conv2D)           (None, 28, 28, 128)       73856

 activation_3 (Activation)   (None, 28, 28, 128)       0

 batch_normalization_3 (Batc  (None, 28, 28, 128)      512
 hNormalization)

 max_pooling2d_3 (MaxPooling  (None, 14, 14, 128)      0
 2D)

 dropout_3 (Dropout)         (None, 14, 14, 128)       0

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 256)               6422784

 activation_4 (Activation)   (None, 256)               0

 batch normalization 4 (Batc  (None, 256)              1024
 dense_1 (Dense)             (None, 256)               65792

 activation_5 (Activation)   (None, 256)               0

 batch_normalization_5 (Batc  (None, 256)              1024
 hNormalization)

 dropout_5 (Dropout)         (None, 256)               0

 dense_2 (Dense)             (None, 5)                 1285

 activation_6 (Activation)   (None, 5)                 0

=================================================================
Total params: 6,590,021
Trainable params: 6,588,517
Non-trainable params: 1,504
_________________________________________________________________
Epoch 1/90
```

Fig. 2. CNN Hyperparameter Sequential of the Proposed Model

2.5. Performance Metrics

The model's performance is validated using the accuracy, recall, precision, and F1 scores. All of the performance measures used in this study are described.

TP: True Positive: The actual value was positive and the model predicted a positive value, FP: False Positive: The prediction is positive, and actual is negative,

FN: False Negative: The prediction is negative, and actual is positive,

TN: True Negative: The actual value is negative and the model predicted a negative value as reported in [32].

Recall is the proportion of true positives to the total of true positives and false negatives, as shown in equation 1:

$$Recall = \frac{TP}{(TP+FN)} \tag{1}$$

Precision is the ratio of true positives to the total of true positives and false positives as shown in equation 2:

$$Precision = \frac{TP}{(TP+FP)} \tag{2}$$

According to equation (3), accuracy is a system's degree of veracity.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \tag{3}$$

A metric that combines precision and memory is the harmonic means of accuracy and recall, sometimes referred to as the balanced F1-score or the traditional F1-measure. The Recall and Precision scores are equally weighted by the F1-score. The F1-score is an important metric to look at when comparing two class issues, and it typically performs better.

3. Results and Discussion

This section delves into the findings derived from the research conducted, presenting outcomes from a series of four distinct experiments. The focus was on assessing a developed CNN model, specifically examining its performance using a dataset that was sourced locally. This initial test aimed to validate the model's effectiveness and reliability in interpreting and analyzing data that it was specifically trained on.

To extend the scope of evaluation and test the model's adaptability and broad applicability, a second experiment was conducted using a well-recognized benchmark dataset [1]. This step was crucial in determining how well the developed CNN model could generalize its learning to new, unseen data, thereby assessing its potential for wider application in similar tasks outside the confines of the initial training set.

The third experiment took an alternative approach by applying a benchmark model [1] previously established and validated in the field to the locally sourced dataset. This inversion aimed to observe how well-established models could interpret and analyze the locally sourced data, providing insights into the compatibility and performance of external models on this specific dataset.

Lastly, the research also involved a critical evaluation of the benchmark model using the dataset originally developed for this study. This part of the investigation highlighted differences in system configurations between the benchmark model and the newly developed CNN model. By comparing the performance of the benchmark model on a new dataset with a different configuration, this experiment offered valuable insights into the flexibility and adaptability of existing models when applied to new research contexts and datasets.

3.1. Evaluation of the Developed Model on the Locally Sourced Dataset

Fig. 3 presents the outcomes from the experiment in which the newly developed model underwent evaluation using a dataset sourced locally. Fig. 4 and Fig. 5 provide visual representations of the confusion matrix, the comparison of training versus validation accuracy, and the comparison of training versus validation error, respectively.

	precision	recall	f1-score	support
Class 0 (African_Almond)	0.97	0.98	0.97	200
Class 1 (Avocado)	0.80	0.96	0.87	200
Class 2 (Cashew)	0.94	0.88	0.91	200
Class 3 (Guava)	0.97	0.78	0.86	200
Class 4 (Mango)	0.90	0.94	0.92	200
accuracy			0.91	1000
macro avg	0.92	0.91	0.91	1000
weighted avg	0.92	0.91	0.91	1000

Fig. 3. Classification Report of the developed model on Locally Sourced Data

Fig. 3. shows a comprehensive classification report of the model's performance across various fruit classes. The analysis of the report examines the precision, recall, and F1-score metrics for each class, as well as the overall performance measures.

The African Almonds class demonstrates remarkable precision, recall, and F1-score, all above 0.97. This indicates the model's exceptional ability to identify African Almonds accurately. Such high scores underscore the robustness of the model in distinguishing this particular class from others in the dataset. Moving on to the Avocado class, it obtained a precision of 0.80 and a high recall of 0.96. While the recall suggests that the model correctly identified most Avocado instances, the precision could be improved to minimize false positives. Nevertheless, the model shows promise in accurately categorizing Avocados. In Cashew, the model demonstrated commendable precision and recall, both above 0.90. This indicated the model's effectiveness in correctly classifying Cashews, striking a balance between precision and recall as reflected in the F1-score of 0.91. Guava exhibited high precision but comparatively lower recall. While the model correctly identified most Guavas, there is room for improvement in recall to capture more instances of this class accurately. Enhancing recall could further refine the model's performance for Guava classification. Finally, Mango showcased balanced performance with both precision and recall exceeding 0.90. The model effectively identified Mangoes with few misclassifications, leading to a high F1-score of 0.92. Upon evaluating the overall performance, the study achieved an impressive accuracy of 91%.

Fig.4 visualize the confusion matrix for the evaluation of fruit leaves using CNN. The model accurately classified 196 instances belonging to African Almond. Avocado instances, with a total of 192 correctly classified instances. For Cashew, the model demonstrated its capability by correctly classifying 176 instances. Despite potential challenges or complexities associated with Guava leaves, the model accurately classified 155 instances of this class. Finally, the model excelled in classifying Mango instances, achieving a total of 189 correctly classified instances.

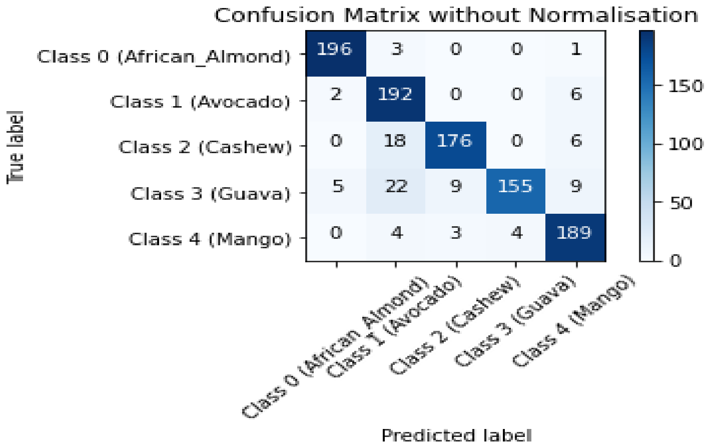

Fig. 4. Confusion Matrix of the proposed Model on Locally Sourced Dataset

However, the misclassified instances across various classes where in African Almond, the model misclassified 3 instances as Avocado and 1 instance as Mango. Similarly, Avocado exhibited misclassifications, with 2 instances incorrectly classified as African Almond and 6 instances misclassified as Mango. In the case of Cashew, a significant number of instances (18) were misclassified as Avocado, along with 6 instances misclassified as Mango. Guava exhibited multiple misclassifications, including 5 instances misclassified as African Almond, 22 instances misclassified as Avocado, 9 instances misclassified as Cashew, and 9 instances misclassified as Mango. Lastly, Mango experienced misclassifications, with 4 instances incorrectly classified as Avocado, 3 instances misclassified as Cashew, and 4 instances misclassified as Guava.

3.2. Model Training and Validation Accuracy

The model was trained using a percentage split of 80:20 for training and validation. Fig. 5 illustrates the training versus validation accuracy and loss. Analyzing the accuracies per epoch reveals a consistent growth in training, indicating the model's progressive learning from the training data. Initially, the training accuracy and loss stood at 45.82% and 1.6149 in the first epoch and steadily climbs to approximately 97.03% and 0.0855. Similarly, the validation displayed a trajectory initially, reaching approximately 56.80% and 0.4270 in the first epoch.

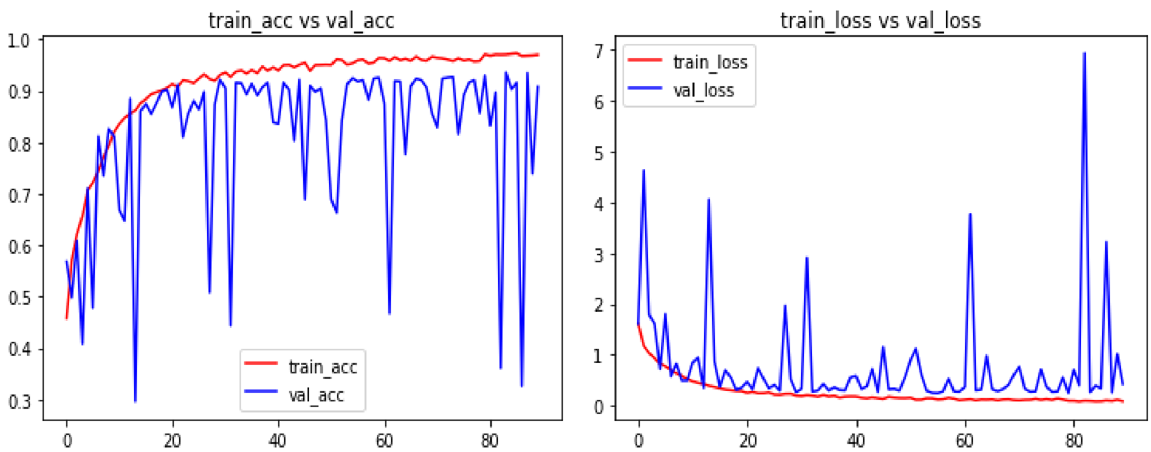

Fig. 5. Train vs Validation Accuracy and Loss of the proposed Model on Local Dataset

However, it fluctuates throughout the validation, suggesting random prediction performance on the unseen data. The accuracies and errors per epoch provide valuable insights into the training dynamics of the CNN model for fruit leaves classification. While the model achieves impressive training accuracy and effectively minimizes its training loss, the observed plateauing of the validation accuracy and fluctuation of the validation loss highlight the need for further regularization techniques to mitigate overfitting and enhance the model's generalization capabilities. Monitoring these metrics over epochs is crucial for ensuring the development of robust and reliable machine learning models.

3.3. Evaluation of the Developed Model on the Benchmark Dataset

Fig. 6 and Fig. 7 offer visual depictions of the confusion matrix, alongside comparisons of training versus validation accuracy and error.

Fig. 6 reveals that the model achieved high accuracy in correctly classifying instances across various fruit categories. Specifically, it correctly identified all the fruits in classes 4, 10 and 13 indicating a robust capability in distinguishing between nuanced classifications of apple quality.

Despite the high rates of correct classifications, the model did encounter instances of misclassification, highlighting areas for potential improvement. Among these, 4 instances of good apples were misclassified as good guavas, and 4 as good oranges, with an additional 6 instances being misclassified within the apple category itself as Pomegranate_Bad. Similarly, good guavas saw 11 instances misclassified as good apples, with sporadic misclassifications involving other fruit categories and sub classifications within apples. Lime and orange categories also faced misclassifications, notably with 3 instances of bad limes being incorrectly identified as good apples and 10 instances of bad oranges as good apples, amongst other errors. These findings underscore the model's proficient ability to accurately classify a vast majority of instances across a spectrum of fruit types and conditions. However, the instances of misclassification provide critical insights into specific areas where the model might benefit from further refinement or training to enhance its overall accuracy and reduce the likelihood of misidentification, especially in distinguishing closely related categories and subcategories within the dataset.

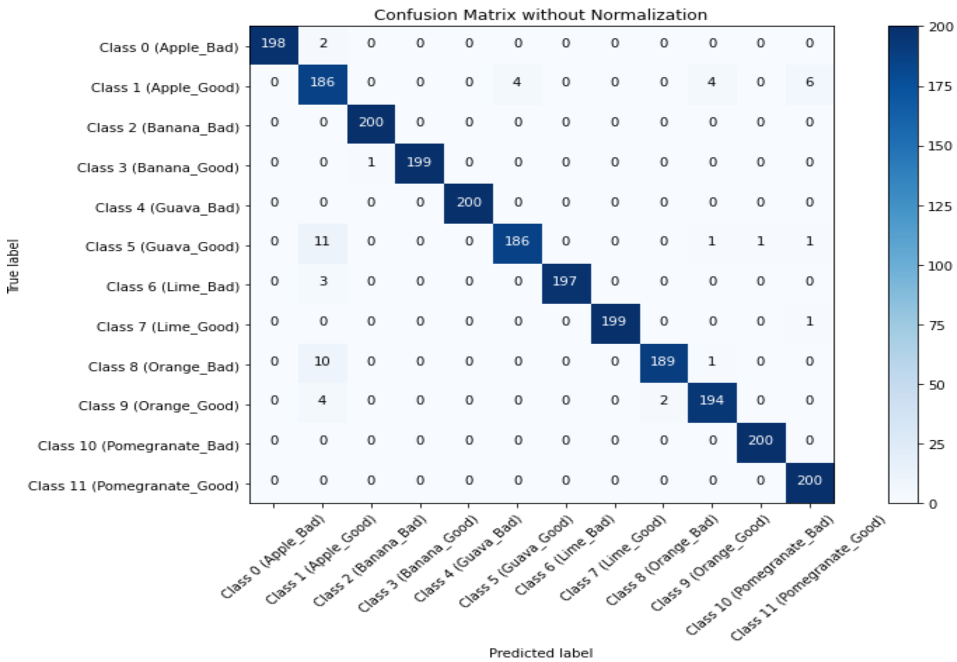

Fig. 6. Confusion matrix of the proposed Model on Benchmark Dataset

Fig. 7 the accuracy of the model on the training set steadily increases over epochs, reaching a peak of approximately 98.75% accuracy by the end of training. On the other hand, the accuracy on the validation set exhibits fluctuations, reaching a peak of approximately 98.67% but generally hovering around the 98% mark. The validation accuracy occasionally dips, indicating instances where the model might be overfitting to the training data. Similarly, the loss on both the training and validation sets shows a decreasing trend over epochs, which is expected as the model learns to minimize its error. However, the training loss decreases more steadily compared to the validation loss. The validation loss exhibits fluctuations, occasionally spiking, which suggests that the model's performance on unseen data is not as consistent as its performance on the training data.

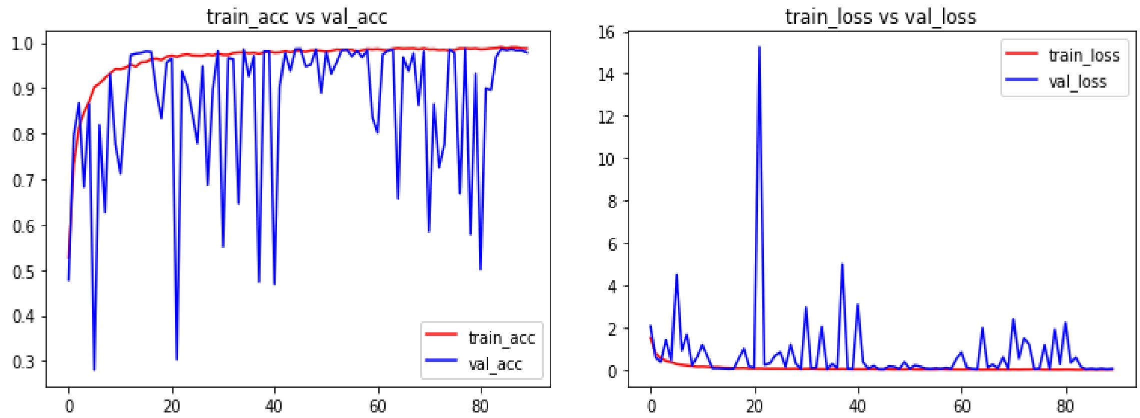

Fig. 7. Training vs. Validation Accuracy and Loss of the proposed Model on Benchmark Dataset

3.4. Comparison with Benchmark Model

In the classification of plants, various authors worked extensively and obtained varying performances. [1] used MobileNetV2, on publicly available benchmark dataset for the classification of fruits, the performance of the proposed model was compared with the results obtained by the authors and are displayed in Table 3.

Table 3. Analysis of Results with State-of-Art Model [1]

Training order	Validation Loss	Validation Accuracy
Benchmark model on benchmark Dataset		97%
Benchmark model on Local Dataset		78%
Proposed model on benchmark Dataset		**97%**
Proposed model on Local Dataset		**91%**

The proposed CNN model was trained from scratch on 90 epochs and it achieved validation accuracy of 91% on local dataset. When compared with [1] model, performances of 97%, 78% and 97% were obtained for various experiments as shown in Table 3. Results clearly shows the robustness of the proposed model over the state-of-the-art model for plants classification.

4. Conclusion

The classification of selected plants by their leaves and fruits using CNN model was the aim of this study and the objectives are to get local image dataset and train the CNN model from scratch with the image datasets. This study adopted the use of CNN model to train images of selected plants'fruits and leaves into their corresponding classes. The training of the model was done using 90 epochs and batches of 32 images. The model achieved accuracies of 91% and 97% on local and benchmark datasets respectively. The model was also compared with a benchmark model recording excellent accuracies. In the future model classes of plants will be considered for classification through their plants and leaves.

Acknowledgment

Authors would like to thank Nigerian Defence Academy for giving us the platform to carryout this study.

Declarations

Author contribution. Martins E. Irhebhude supervised this study from proposal, to experiments and review process to completion. Adeola O. Kolawole did the review of literature, revise the initial submission to completion. Chat Chinyio took the responsibility of gathering data and submitting the draft article.

Funding statement. The authors declare no funding was provided for this study.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper

Data and Software Availability Statements

Data and Software availability statements provide a statement about where data and software supporting the results reported in a published article can be found, including hyperlinks to publicly archived datasets and software analyzed and generated during the study/experiments.

References

[1] T. B. Shahi, C. Sitaula, A. Neupane, and W. Guo, "Fruit classification using attention-based MobileNetV2 for industrial applications," *PLoS One*, vol. 17, no. 2, p. e0264586, Feb. 2022, doi: 10.1371/journal.pone.0264586.

[2] M. E. Irhebhude, A. O. Kolawole, and F. B. Bugaje, "Recognition of Mangoes and Oranges Colour and Texture Features and Locality Preserving Projection," *Int. J. Comput. Digit. Syst.*, vol. 11, no. 1, pp. 963–975, Mar. 2022, doi: 10.12785/ijcds/110179.

[3] I. Kumar, J. Rawat, N. Mohd, and S. Husain, "Opportunities of Artificial Intelligence and Machine Learning in the Food Industry," *J. Food Qual.*, vol. 2021, pp. 1–10, Jul. 2021, doi: 10.1155/2021/4535567.

[4] R. Shailendra *et al.*, "An IoT and Machine Learning Based Intelligent System for the Classification of Therapeutic Plants," *Neural Process. Lett.*, vol. 54, no. 5, pp. 4465–4493, Oct. 2022, doi: 10.1007/s11063-022-10818-5.

[5] R. Yamparala, R. Challa, V. Kantharao, and P. S. R. Krishna, "Computerized Classification of Fruits using Convolution Neural Network," in *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, Jul. 2020, pp. 1–4, doi: 10.1109/ICSSS49621.2020.9202305.

[6] G. Agarwal *et al.*, "First steps toward an electronic field guide for plants," *Taxon*, vol. 55, no. 3, pp. 597–610, Aug. 2006, doi: 10.2307/25065637.

[7] N. Kumar *et al.*, "Leafsnap: A Computer Vision System for Automatic Plant Species Identification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7573 LNCS, no. PART 2, Springer, Berlin, Heidelberg, 2012, pp. 502–516, doi: 10.1007/978-3-642-33709-3_36.

[8] Anfeng He and Xinmei Tian, “Multi-organ plant identification with multi-column deep convolutional neural networks,” in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2016, pp. 002020–002025, doi: 10.1109/SMC.2016.7844537.

[9] S. Chopde, M. Patil, A. Shaikh, B. Chavhan, and M. Deshmukh, “Developments in computer vision system, focusing on its applications in quality inspection of fruits and vegetables-A review,” *Agric. Rev.*, vol. 38, no. 02, Jun. 2017, doi: 10.18805/ag.v38i02.7940.

[10] M. Hussein and A. H. Abbas, “Plant Leaf Disease Detection Using Support Vector Machine,” *Al-Mustansiriyah J. Sci.*, vol. 30, no. 1, pp. 105–110, Aug. 2019, doi: 10.23851/mjs.v30i1.487.

[11] J. S. Cope, D. Corney, J. Y. Clark, P. Remagnino, and P. Wilkin, “Plant species identification using digital morphometrics: A review,” *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7562–7573, Jun. 2012, doi: 10.1016/j.eswa.2012.01.073.

[12] S. K. Behera, A. K. Rath, A. Mahapatra, and P. K. Sethy, “Identification, classification & grading of fruits using machine learning & computer intelligence: a review,” *J. Ambient Intell. Humaniz. Comput.*, pp. 1–11, Mar. 2020, doi: 10.1007/s12652-020-01865-8.

[13] J. K. Patil, R. Kumar, B. Vidyapeeth, C. O. E. Kolhapur, and B. Vidyapeeth, “Advances In Image Processing For Detection Of Plant Diseases,” *J. Adv. Bioinforma. Appl. Res.*, vol. 2, pp. 135–141, 2011. [Online]. Available at: http://www.bipublication.com.

[14] A. Aakif and M. F. Khan, “Automatic classification of plants based on their leaves,” *Biosyst. Eng.*, vol. 139, pp. 66–75, Nov. 2015, doi: 10.1016/j.biosystemseng.2015.08.003.

[15] J. Banzi and T. Abayo, “Plant Species Identification from Leaf Images Using Deep Learning Models (CNN-LSTM Architecture),” *Tanzania J. For. Nat. Conserv.*, vol. 90, no. 3, pp. 93–103, Nov. 2021. [Online]. Available at: https://www.ajol.info/index.php/tjfnc/article/view/217008.

[16] M. Malik, A. Ikram, S. N. Batool, and W. Aslam, “A Performance Assessment of Rose Plant Classification Using Machine Learning,” in *Communications in Computer and Information Science*, vol. 932, Springer, Singapore, 2019, pp. 745–756, doi: 10.1007/978-981-13-6052-7_64.

[17] H. Yalcin and S. Razavi, “Plant classification using convolutional neural networks,” in *2016 Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, Jul. 2016, pp. 1–5, doi: 10.1109/Agro-Geoinformatics.2016.7577698.

[18] S. A. Wagle, R. Harikrishnan, S. H. M. Ali, and M. Faseehuddin, “Classification of Plant Leaves Using New Compact Convolutional Neural Network Models,” *Plants*, vol. 11, no. 1, p. 24, Dec. 2021, doi: 10.3390/plants11010024.

[19] D. Gerdan Koç and M. Vatandaş, “Classification of Some Fruits using Image Processing and Machine Learning,” *Turkish J. Agric. - Food Sci. Technol.*, vol. 9, no. 12, pp. 2189–2196, Dec. 2021, doi: 10.24925/turjaf.v9i12.2189-2196.4445.

[20] L. Rajasekar and D. Sharmila, “Performance analysis of soft computing techniques for the automatic classification of fruits dataset,” *Soft Comput.*, vol. 23, no. 8, pp. 2773–2788, Apr. 2019, doi: 10.1007/s00500-019-03776-z.

[21] Y. Kumar, N. Hasteer, A. Bhardwaj, and . Yogesh, “Convolutional neural network architecture for detection and classification of diseases in fruits,” *Curr. Sci.*, vol. 122, no. 11, p. 1315, Jun. 2022, doi: 10.18520/cs/v122/i11/1315-1320.

[22] U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, “Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease,” *IEEE Access*, vol. 7, pp. 43721–43729, 2019, doi: 10.1109/ACCESS.2019.2907383.

[23] V. Bhole, A. Kumar, and D. Bhatnagar, “Fusion of color-texture features based classification of fruits using digital and thermal images: A step towards improvement.,” *Grenze Int. J. Eng. Technol.*, vol. 6, no. 1, pp. 133–141, 2020. [Online]. Available at: https://web.p.ebscohost.com/abstract?site=ehost&scope=site&jrnl=23955287&AN=141184365&h=j%2B8C3mf4Pk.

[24] A. Harjoko, A. Prahara, T. W. Supardi, I. Candradewi, R. Pulungan, and S. Hartati, "Image processing approach for grading tobacco leaf based on color and quality," *Int. J. Smart Sens. Intell. Syst.*, vol. 12, no. 1, pp. 1–10, Jan. 2019, doi: 10.21307/ijssis-2019-010.

[25] B. M. Quach, V. C. Dinh, N. Pham, D. Huynh, and B. T. Nguyen, "Leaf recognition using convolutional neural networks based features," *Multimed. Tools Appl.*, vol. 82, no. 1, pp. 777–801, Jan. 2023, doi: 10.1007/s11042-022-13199-y.

[26] P. Chithra and M. Henila, "Fruits Classification Using Image Processing Techniques," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 10, pp. 628–632, 2019. [Online]. Available at: https://www.researchgate.net/publication/332671932_Fruits_Classification_Using_Image_Processing_Techniques.

[27] M. R. Dileep and P. N. Pournami, "AyurLeaf: A Deep Learning Approach for Classification of Medicinal Plants," in *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*, Oct. 2019, vol. 2019-Octob, pp. 321–325, doi: 10.1109/TENCON.2019.8929394.

[28] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.

[29] M. M. Taye, "Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions," *Computation*, vol. 11, no. 3, p. 52, Mar. 2023, doi: 10.3390/computation11030052.

[30] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.

[31] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, "Fundamental Concepts of Convolutional Neural Network," in *Intelligent Systems Reference Library*, vol. 172, Springer, Cham, 2020, pp. 519–567, doi: 10.1007/978-3-030-32644-9_36.

[32] M. E. Irhebhude, A. O. Kolawole, and H. K. Goma, "A Gender Recognition System Using Facial Images with High Dimensional Data," *Malaysian J. Appl. Sci.*, vol. 6, no. 1, pp. 27–45, Apr. 2021, doi: 10.37231/myjas.2021.6.1.275.