ABE

Agricultural and Biological Engineering

https://ph04.tci-thaijo.org/index.php/abe/index

Published by the Faculty of Engineering, Khon Kaen University, Thailand

Potential utilisation of Convolutional Neural Network (CNN)-based banana bunch ripeness classification to effectuate banana harvesting process

Eka Riskawati* and Taufik Djatna

Department of Agroindustrial Technology, Faculty of Agricultural Technology, IPB University, Bogor, 16680, Indonesia

*Corresponding author.
Email address: ekariskawatieka@apps.ipb.ac.id
doi: 10.14456/abe.2026.3

Received 18 November 2025
Revised 1 January 2026
Accepted 7 January 2026

Abstract

The classification of banana bunch maturity represents a vital preliminary phase for maintaining fruit quality. However, prior studies related to non-destructive maturity classification have predominantly focused on ready-to-sell finger bananas despite the application of industrial-scale banana harvesting, which is done by bunches. This research aimed to categorize banana fruit bunches' ripeness status before the harvesting process. The classification process distinguishes between two maturity levels (unripe and ripe) utilizing the model comparison between Convolutional Neural Network (CNN), Visual Geometry Group (VGG) 16, and EfficientNet methodology. The dataset comprises 500 banana bunch images for labeling purposes. The data was partitioned in a 4:1 ratio for training and testing. The developed model utilizes CNN architecture that includes convolutional (Conv2D), pooling (MaxPooling2D), and fully connected layers. Evaluation outcomes indicate that the model effectively classifies the maturity of banana bunches, demonstrating high accuracy, precision, and recall. The conventional basic CNN resulted in the most optimal model among VGG16 and EfficientNet with precision up to 91.11%. This CNN-based classification system is anticipated to be integrated into the banana industry, aiming to maintain the harvested banana bunches. By employing CNN for classifying the maturity of banana bunches, the harvesting process can be made more efficient with less time needed. Furthermore, the system enhances automation and consistency in product quality while decreasing dependence on manual labor. Additionally, the classification outcomes can be directed towards appropriate processing pathways, thereby facilitating the implementation of smart technology-driven postharvest systems over time.

Keywords: Agriculture technology, Deep learning, Fruit quality, Harvest

1. Introduction

1.1 Background

The fruit harvesting process represents a pivotal phase in the fresh banana sector [1]. As bananas are climacteric fruits with short shelf lives, application of appropriate harvesting methods is essential to preserve the quality of bananas intended for sale [2]. These methods encompass the effective assessment of the readiness of banana bunches for harvesting. The ripeness stage of bananas, including the plantain and fresh banana type, influences the fruit's nutritional profile [3-4]. When the fruit is harvested during the pre-ripening stage or is still in an unripe condition, the fruit contains high starch content and low sugar content. As the banana ripens, the starch content will significantly decrease and the sugar content will conversely increase. Specifically, as the fruit ripens, there is also an increase in the levels of ascorbic acid, phenolics, flavonoids, and beta carotene. While suitable post-harvest practices analysis can sustain the physical integrity and texture of the fruit, it is imperative that the harvesting phase is executed with a non-destructive method that has accuracy, taking into account the ripeness level of the fruit [5-6]. This practice could potentially minimise waste generated during the post-harvest phase.

Conversely, the timing and justification behind fruit harvesting conducted by both small-scale and industrial-scale farmers continue to follow traditional methods based on physical characteristics, coloration, and the age of the fruit since its blossoms [2, 7-8]. It is estimated that the fruit is ready to be harvested around 12-13th months since plantation and 10-13 after flower emergence [9-10]. Since this process depends on the skills of the harvesting personnel, it is susceptible to inaccuracies and requires more labour [8, 11]. On the other hand, there is also a conventional harvesting process that includes calculating soluble solid content (SSC), pH, and firmness (FM) in determining fruit quality, which is less efficient and cannot be applied to meet the demands of fast-paced market needs [12]. Consequently, there is a need for an alternative technology that can classify the ripeness of banana bunches with precision, accuracy, and time efficiency.

Image processing is one of the potential technological innovations in improving the efficiency of the precise ripeness classification. The image processing techniques used digital images to classify the ripeness of banana bunches based on color, size, and texture. Methods for this approach include MATLAB, convolutional neural networks, and machine learning algorithms [7, 13-16]. The image processing approach uses a color segmentation process and calculation of the average intensity of colors in RGB values to classify the maturity range [17-18]. The segmentation results will be used to build a model that will then be applied to image data from real objects. The image processing approach is preferable to use because it can process the data with high accuracy, efficiency, and non-destructiveness [11, 19].

Convolutional neural networks (CNNs), utilized as image processing techniques, facilitate the automatic extraction of the most pertinent features, which proves advantageous in classification systems that necessitate the identification of subtle variations in ripeness stages [20-22]. The CNN approach possesses the benefit of being robust against data variations, owing to its augmentation capabilities, which can enhance its capacity to generalize across diverse datasets and conditions [23-24]. In comparison to alternative methods, such as those that hybridize with Random Forest, CNN demonstrates a balanced level of accuracy, efficiency, and user-friendliness [25-26].

Several studies have been carried out regarding the application of CNN models in the classification systems for banana fruit [13, 23, 26-29]. The CNN methodology is utilized either as an independent technique or in conjunction with Support Vector Machine (SVM), Random Forest, or optimization algorithms like the Adam optimizer. The modelling predominantly relies on training and testing datasets comprising images of comb bananas, with a particular emphasis on post-harvest applications, including storage. In contrast, actual industrial practices involve harvesting bananas in bunches. Therefore, it is essential to implement the CNN image processing technique on datasets consisting of banana bunch images. The expected results of this study are intended to aid in the creation of a classification model for banana bunch ripeness, which can proficiently tackle practical issues within the industry, especially in evaluating the harvest condition of banana bunches.

1.2 Related works

Image processing is a widely used and developed method in determining the classification of an object. This method involves manipulating and analyzing images to extract important information and categorize them into predefined classes. The image classification process is divided into training and testing stages. This stage consists of image processing, feature extraction, and application of machine learning algorithms [30-31]. Several types of image processing approaches have been used in the classification of banana ripeness classes (Table 1).

Table 1 Image processing applications in banana ripeness classification

Reference	Research objectives	Dataset	Category	Method	Accuracy
Widodo, et al. [11]	Analyze the banana maturity level using thermal images and its correlation with physical and chemical quality	5 banana hands into 160 banana fingers	2 levels (ripe and unripe)	Image processing – thermal imaging	R square
Mutrofin, et al. [23]	Analyze the application of simple CNN-based automatic classification and comparison with DenseNet20 and VGG16.	Banana finger	4 levels (unripe, half-ripe, ripe, overripe)	Shallow CNN + (RMSprop)	99%
Ramadhan, et al. [13]	Develop a Cavendish banana ripeness level identification system based on skin color	Banana Cavendish finger	4 levels (unripe, half-ripe, ripe, overripe)	General CNN with two optimizer treatments (Adam and SGD)	Optimizer Adam: 93.25% Optimizer SGD: 94.12%
Yashu, et al. [26]	Develop a banana ripeness detection using a combination of CNN and Random Forest	-	-	CNN + Random Forest	Total accuracy (86.97%), Presisi (81.97-91.37%), F1-score (87)
Sandra Prayogi, et al. [18]	Designing a color sensor-based banana ripeness prediction tool	Banana Barlin fingers	4 ripeness level (RL): 1,3,5, dan 7	rule-based RGB method	RL 1: 100% RL 3: 80% RL 5: 100% RL 7: 60%
Maity, et al. [16]	Analyze and classify banana ripeness based on image with MATLAB.	Banana fruit	-	MATLAB color features and statistics	-
Mohamedon, et al. [32]	Designing a mobile application for banana ripeness identification	Banana fruit (mixed)	3 levels (ripe, unripe, overripe)	CNN EfficientNet-Lite	98.25%
Han, et al. [33]	Improving the accuracy of banana ripeness using a combination of features	Hand banana	4 levels (unripe, underripe, ripe, overripe)	CNN VGG16 + XgBoost	CNN only: 85% CNN +XgBoost: 91.25%
Arunima, et al. [29]	Develop a CNN model for Nendran banana classification	Banana Nendran fingers	4 levels (unripe, medium ripe, ripe, overripe)	CNN	95%
Raghavenra, et al. [34]	Develop CNN sorting model with dual-channel CNN system	Banana Rashtali fingers	4 levels (unripe, semi-ripe, ripe, overripe)	CNN dual channel	97.65%

According to the identification of relevant studies presented in Table 1, it is evident that the CNN model ranks among the most commonly utilized methods for classifying banana ripeness. The model's capability for automatic feature extraction is a significant advantage [35-36]. This feature enables classification to occur without the inaccuracies that may arise from human involvement [37]. Nevertheless, in practical applications, it is essential to tailor the model to the specific type of object and the intended purpose of its development.

Several research efforts have concentrated on categorizing the ripeness of bananas, predominantly employing banana fingers and hands, with overarching goals directed towards prolonging the storage life of post-harvested fruits. Conversely, one particular study utilizing banana hands seeks to assess the enhancement in the accuracy of the developed model through the incorporation of boosters. Consequently, for the development of a classification model for banana bunch ripeness and its practical applications to support harvesting efficiency, additional research is essential. The model's development must take into account the pre-processing phase, the implementation of boosters, and the assessment of model performance.

2. Materials and methods

Research commenced with a literature review to gather information related to the banana harvesting process and identify specific constraints in an industrial-scale production. Interviews with field staff confirmed the identified challenges and informed the existing practice of the banana harvesting process. The research stages were organized to facilitate a solution to the confirmed challenges related to the ripeness assessment of banana fruit bunches (Figure 1).

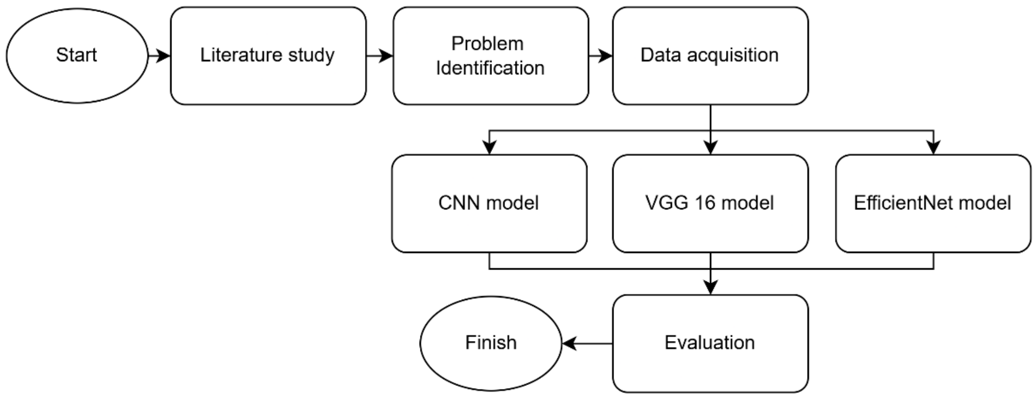

Figure 1 Flow of the research stages

In the data acquisition stage, the dataset was sourced from Kaggle, uploaded by Krishna Kishor Kammaje under the title "Counting and Weighing Bananas”. The Kaggle dataset is comprised of 713 picture frames. Data were selected to minimize duplication. Following the principles of supervised learning, the data were categorized into two labels as the maturity stage of the banana bunches: ripe and unripe. The image data underwent a preprocessing stage to standardize data attributes and remove the noise. Total image utilized was consisted of 2 ripeness category with 250 images corresponding to each label category.

Data possessing uniform attributes is subsequently processed during the construction of three model; CNN, Visual Geometry Group (VGG)16, and EfficientNet. For CNN model as the conventional structure, TensorFlow and Keras are utilized for the development of the CNN architecture. The data undergoes processing using NumPy, Pandas, and PIL, enabling visualization through matplotlib.pyplot and seaborn functionalities. The models and their corresponding labels are assessed using sklearn. The outcomes of the evaluation are presented in terms of accuracy, precision, recall, and F-1 score. These metrics will indicate the performance level of the resultant model and its appropriateness in addressing the requirements of the problem.

The VGG16 model is implemented by adopting the architecture of CNN with deeper application. Referring to the study by Gupta et al. [38], the use of CNN is intended as a provider of basic structure, with in-depth views and complex patterns using the VGG model. This approach is motivated by the question of how to improve the effectiveness of CNN performance on large image datasets [39]. The VGG16 architecture consists of 16 weight layers organized like a CNN structure: an input layer, convolutional blocks 1-5, fully connected layers, and pooling. For comparison, to determine the most suitable model for optimizing banana bunch maturity classification, the EfficientNet model was also constructed. The architecture of this model adopts the basic architecture of CNN with efficiency lying in the balance of scale and parameters.

2.1 Dataset Description

The dataset utilized in this research, sourced from Kaggle, comprises 713 images that were filtered to eliminate duplicates and excessive noise, resulting in 250 images for each label category. The specific variety or cultivar represented in the dataset was not specified by the original sources. Due to this condition, the label category was decided based on visual characteristics observed. The label categories consisted of two ripeness stages; ripe fruit bunches and unripe fruit bunches. The original image data measured 920 x 1280 pixels and featured a non-uniform background, which was then standardized during the image pre-processing phase.

2.2 Image Pre-processing

The dataset's image data undergoes processing to remove the background and noise, resulting in a clear depiction of the banana bunch object (Figure 2). The image is then resized to a consistent dimension of 480 x 640 pixels. The complete set of image data is then labelled across the two label categories.

Figure 2 Result of background and noise removal from banana bunch image data

The image data obtained has been labeled with mature and immature categories. The data is first segmented with a ratio of 4:1 for training data (200 files) and testing data (50 files). Each layer is built linearly with Sequential and Keras tools and libraries. The features are extracted from the image data by the convolution layer with the Conv2D tool. The feature dimensions are reduced to retain important features with the MaxPooling2D tool. The feature results are then deformed from 2D pooled data to 1D vectors with the Flatten tool, and the results are ready to enter the fully connected screen (Figure 4).

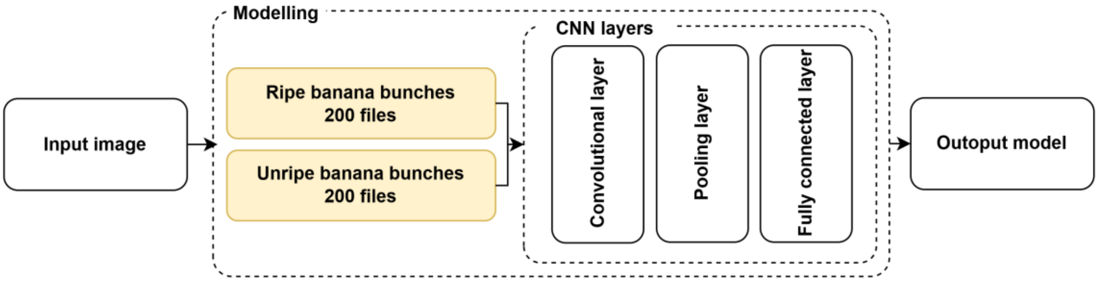

Figure 4 CNN modelling process

The trained model was then analyzed through cutting stopping point testing to determine the level of fitting and avoid underfitting or overfitting conditions. The test results yielded the most optimal model from the training process as well as the optimal number of epochs for that model. The result of the fully connected layer is the classification of banana bunch ripeness, which consists of “ripe” and “unripe.” Categorizing maturity into two types is based on the expected final goal of model implementation, which is to support harvesting decisions. Therefore, simplifying the options provided will support accelerating harvesting decision-making. The test image data that has been previously prepared is used to test the model through the model evaluation stage. The test results will determine the classification accuracy of the resulting model.

2.3 Model evaluation

Model evaluation is carried out using the confusion matrix method, which is a table that compares the label recognition of the categorization results with the actual category label [39]. The results of model testing with a confusion matrix will be shown in several types of metrics, namely accuracy, precision, recall, and F1 score [40] as shown in Equation 1-4. The following equation is used to determine the effectiveness of the CNN model with these metrics:

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

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

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1\ Score = 2\,x \frac{\frac{TP}{TP+FP}\ x\ \frac{TP}{TP+FN}}{\frac{TP}{TP+FP}+\frac{TP}{TP+FN}} = 2\,x \frac{Precisionx\ Recall}{Precision+Recall} \quad (4)$$

The Matplotlib.pyplot tool is used to plot a generalized graph representing the parameters assessed during the evaluation. is utilized indirectly to feed the evaluation results into the visualization. Heatmaps are employed to offer a visual summary of the metrics for each class.

3. Results and discussion

The developed CNN, VGG16, and EfficientNet model had detail structure showed in Table 2. The CNN model was built using the Keras Sequential API with a total of 44,397,124 parameters, all of which are trainable. The architecture begins with three consecutive convolutional blocks: Conv2D (32, 3x3) with ReLU activation for basic feature extraction, followed by Conv2D (64, 3x3) for more complex pattern detection, and Conv2D (128, 3x3) for high-level feature representation. Each convolutional block is followed by MaxPooling2D (2x2), which functionally to reduces the spatial dimensions to improve computational efficiency and translational invariance. The Flatten layer converts a 3D tensor (height × width × channels) into a 1D vector with 86,528 features, creating a significant bottleneck. A fully-connected layer Dense (512) with ReLU activation then processes this vector, resulting in 44,270,592 parameters (99.7% of the total model parameters), indicating massive parameter bloat. Dropout (0.5) is applied for regularization to prevent overfitting by randomly deactivating 50% of neurons during training. The output layer uses Dense (2) with softmax activation for binary classification (ripe or unripe), producing a probability distribution over both classes. The model is compiled using the Adam optimizer, categorical_crossentropy loss function, and accuracy metrics to monitor training performance.

Model VGG16 utilizes transfer learning with a base model pre-trained on the ImageNet dataset, having a total of 134,268,738 parameters with 119,564,050 trainable parameters (89%) and 14,714,688 frozen parameters. Implementation begins by loading the VGG16 base using keras.applications. VGG16 with detailed layers consisted of "weights='imagenet'", parameter "include_top=False", and input shape dimention (224, 224, 3) with extracts 13 convolutional layers with a tiered convolutional block architecture subsquently; 64, 128, 256, 512, and 512 filters. The parameter "include_top=False" removes the original VGG16 fully-connected layers, allowing for a custom classification head for the specific task of banana ripeness. The base model is unfrozen to retain the learned features from ImageNet, resulting in output feature maps of size 7×7×512 (25,088 features). The custom classification head starts with layer "Flatten()", which converts the feature maps into a one-dimensional vector, followed by "Dense(4096)" with ReLU activation, yielding 102,764,544 parameters, and then "Dropout(0.5)" for regularization. The second Dense(4096) layer with ReLU activation and Dropout(0.5) adds 16,781,312 parameters, creating a massive accumulation of parameters in the fully-connected layers (89% of the total). The output layer Dense(2) with softmax activation produces classification probabilities, with a total of 119,545,856 parameters in the FC layers, indicating an extreme architectural bottleneck in the transfer learning implementation using compile with the Adam optimizer and callbacks such as EarlyStopping and ModelCheckpoint utilized for monitoring training progress.

Furthermore,l the EfficientNet model employs transfer learning with a state-of-the-art, efficient architecture comprising a total of 4,214,309 parameters, of which only 164,482 (3.9%) are trainable, while 4,049,827 parameters remain frozen from the base model. It is implemented using Keras applications and utilizes the compound scaling method to achieve an optimal balance in scaling depth, width, and resolution. The base model contains 239 layers featuring Mobile Inverted Bottleneck Convolution (MBConv) blocks, which incorporate depthwise separable convolutions for parameter efficiency and squeeze-and-excitation blocks for channel-wise feature recalibration. The fine-tuning strategy involves unfreezing the last 50 layers via a for loop to adapt high-level features specifically to the banana domain. The custom classification head begins with GlobalAveragePooling2D(), which transforms the feature maps (7×7×1280) into a 1280-dimensional vector without introducing additional parameters, thereby avoiding the parameter inflation associated with the Flatten-Dense architecture. This is followed by a Dense(128) layer with an L2 kernel regularizer, BatchNormalization() for training stabilization and accelerated convergence, and Dropout(0.3) for regularization. A second Dense(128) layer, also accompanied by BatchNormalization() and Dropout(0.3), enhances the model's representational capacity, culminating in an output Dense(2) layer with softmax activation. The model is compiled using the Adam optimizer and preprocessing function to normalize pixel values based on ImageNet statistics. Additionally, the ReduceLROnPlateau callback is employed to adaptively adjust the learning rate during training plateaus as shown in Table 2.

Table 2 Model architecture and structures analysis

Specification	CNN	VGG16	EfficientNet
Total Parameters	44,397,124	134,268,738	4,214,309
Trainable Parameters	44,397,122	119,564,050	164,482
Non-trainable Parameters	2	14,714,688	4,049,827
% Trainable	100.0%	89.0%	3.9%
Model Size (MB)	169.36	512.18	16.08
Base Model	None	VGG16 (ImageNet)	EfficientNetB0
Parameter Distribution	99.7% in Dense layer	89% in FC layers	Optimal (GAP used)
Main Bottleneck	Flatten, Dense(512)	Dense(4096) x2	Too few trainable
Uses Batch Norm	No	No	Yes
Uses Dropout	Yes	Yes	Yes

Based on the evaluation results of the model, it is evident that the built basic CNN model exhibits high values across all evaluation variables (Table 3). The overall evaluation scores were derived from the best model and best epoch which resulted in 26 epochs. Out of a total of 50 evaluations within the ripe category, true positive results were identified in 46 images, while false positives were recorded in 4 images. As for the unripe category, 41 images were correctly identified and 9 were identified as the opposite. The evaluation results are also shown by the precision value of 91.11% for the unripe category and 83.63% for the ripe category. The recall value of 82% in the immature category and 92% in the mature category shows that the model has the ability to remember and detect positive data as a whole. Then the F1-score results, which show the balance between the level of precision and recall, resulted in 87.61% for the ripe category and 86.31% for the unripe category.

Table 3 Basic CNN Model evaluation result

Category	Precision	Recall	F1-Score
Unripe	0.911111	0.82	0.939591
Ripe	0.836364	0.92	0.876190
Accuracy	0.870000	0.87	0.870000
Macro average	0.873737	0.87	0.869674
Weighted average	0.873737	0.87	0.869674

A comparison analysis of the three models showed that the CNN achieved the best performance with an accuracy of 87.00% and an F1-score of 86.97%, surpassing VGG16 (84.00%) and EfficientNet, which failed completely (50.00%). Analysis per model revealed that the CNN model had an excellent balance between the Unripe (85.71% F1-score) and Ripe (88.46% F1-score) classes, while EfficientNet showed extreme bias with an inability to detect the Unripe class (0% for all metrics). Model generalization shows Custom CNN is superior with minimal overfitting of 0.58%, in contrast to VGG16, which experienced severe overfitting of 12.81% and training collapse from a peak of 87.5% down to 81.25%.

Parameter efficiency proves conventional CNN achieves 1.96% accuracy per million parameters with 44.4M fully trainable parameters, while VGG16 only achieves 0.63% with 119.6M parameters, and EfficientNet failed with 164K trainable parameters (3.9%), which was insufficient. The size-performance trade-off shows that the Custom CNN, optimized with 169 MB, achieves 0.51% accuracy per MB, compared to the VGG16's catastrophic 512 MB (0.16% per MB) and the EfficientNet's 16 MB, which is non-functional. Training stability becomes a crucial differentiator, where the custom CNN converges stably after 26 epochs, VGG16 collapses after the 3rd epoch, and EfficientNet fails to learn, indicating fundamental implementation issues. The empirical paradox is confirmed that sophisticated architecture does not guarantee superior performance without proper implementation, conversely validating the principle that "simple but properly implemented" outperforms "complex but poorly tuned." The findings recommend conventional CNN model as a baseline with 95% confidence, supported by excellent generalization, training stability, and proven reliability despite architectural inefficiency. To be more detail, the CNN model evaluation result showed in Table 4.

Table 4 Model efficiency evaluation result

Efficiency Metric	CNN	VGG16	EficientNet
Accuracy per M Parameters	1.96%	0.63%	11.88%
Accuracy per MB Size	0.51%	0.16%	3.13%
Training Time	1.0x (baseline)	1.5x	0.8x
Inference Time (CPU)	~50ms	~250ms	~60ms
Memory Footprint (RAM)	~850MB	~2.5GB	~300MB

Overall, the precision metric of the model's performance which ranging from 83.63% to 91.11% effectively reflects the model's readiness to classify the banana ripeness in the bunch shape image, which is related to the existing practice of banana harvesting. The output form, reliability, and classification quality are positively affected by the resulting precision level. The precision metric illustrates the degree of closeness among multiple measurements, which is crucial for elucidating random errors and their statistical variability [41]. According to Rainio et al. [42], certain evaluation metric needed to assess the performance of supervised learning models. For the purpose of this CNN model application, a high precision value indicates the accuracy of the classification so that the output has minimal error. Consequently, it leads to an increase in efficiency, allowing classification tasks to be executed swiftly. The implementation of this model is anticipated to considerably reduce the classification time required during banana bunch harvesting as well as transform the manual and age-based determination.

Compared to the prior related works, the images of the bunches-shaped dataset used in this study already represent the real practices in the industry. This condition supports the possibility of CNN model utilization as a non-destructive ripeness determination supporting system. Therefore, this study classifies banana bunches into two primary ripeness stages: "ripe" and "unripe," while the previous study listed in Table 1 aims for a more granular classification of four or more stages (unripe, half-ripe, ripe, and overripe). This binary classification is strategically aligned with the final aim to immediately determine the fruit that is ready to harvest. For this specific application, a clear and simple category is often sufficient for the decision process, particularly for field operators.

A significant constraint of this research is the lack of precise information regarding the banana variety utilized and the environmental factors considered in the training dataset. As different banana varieties (e.g., Cavendish, Plantain, Nendran, and Rashtali) exhibit distinct morphological features and ripening characteristics, including unique color progressions, size, and skin textures, a CNN model trained on an unidentified variety may not perform optimally or accurately when applied to images of other banana types. Same as the environment-related factors that could bias the banana ripening characteristic in the captured image. Consequently, the generalizability of the resulting CNN model to diverse banana cultivars remains untested, and its direct applicability is currently confined to the specific, albeit unidentified, variety present in the used dataset.

This lack of specific varietal data also complicates direct comparative analysis with existing literature, where specific banana varieties are often declared (Table 1). It is complicated to definitively ascertain if observed statistical performances are exclusively attributable to model architecture and training or if they are influenced by intrinsic visual and maturation attributes unique to the unspecified cultivar within the utilized dataset. For future applied industrial implementation, a ripeness classification system necessitates validation and dedicated training for established commercial banana cultivars.

4. Conclusions

This research has been conducted to fill the gap related to the development of a CNN model that can classify the maturity of banana fruit in the form of bunches, adjusting with its long-term goal to support the efficiency of the harvesting process in the fresh banana industry. The model achieves an accuracy rate of 87.37%, with the precision for each class category ranging from 83.63% to 91.11%. It demonstrates readiness for practical application in classifying the ripeness of banana bunches in real-world scenarios. This study can be improved by conducting research in a similar scope with the addition of factors that better reflect environmental conditions in the field, such as lighting variables and shooting angles. However, a limitation of this research is the restricted dataset, indicating that further studies are necessary, involving scientific collaboration with banana industry stakeholders and integration with the company's secondary data to develop a more robust model.

To enhance the robustness and practical applicability of the developed model, future work should prioritize the collection of meticulously documented datasets. This includes specifying the exact banana variety, systematically recording image acquisition parameters (e.g., lighting, camera settings), and correlating visual ripeness cues with objective physico-chemical metrics (e.g., Brix, firmness) across various maturity stages. Such controlled data collection will allow for the development of more generalizable and industry-specific ripeness classification models.

5. Acknowledgements

This study would like to thank the Agroindustrial Production System Research Assistants for supporting with methodology deepen understanding and providing consideration of method determination.

6. References

[1] Diezma B, Franco S, Lleó L, Presečki T, Roger JM. Grading banana by VNIR hyperspectral imaging spectroscopy. Acta Hortic. 2018;1194:1283-9. https://doi.org/10.17660/ActaHortic.2018.1194.181.

[2] Triardianto D, Bintoro N. The effect of different time durations of ozone treatment and storage temperatures on postharvest quality of banana (Musa acuminata). IOP Conf. Ser. Earth Environ. Sci., vol. 759, IOP Publishing Ltd; 2021. https://doi.org/10.1088/1755-1315/759/1/012012.

[3] Garcés-Moncayo MF, Guevara-Viejó F, Valenzuela-Cobos JD, Galindo-Villardón P, Vicente-Galindo P. Modeling of the Physicochemical and Nutritional Composition of Musa paradisiaca (Williams Variety) at Different Ripening Stages in Ecuador. Agric. 2025;15. https://doi.org/10.3390/agriculture15101025.

[4] Huang P-H, Cheng Y-T, Lu W-C, Chiang P-Y, Yeh J-L, Wang C-C, et al. Changes in Nutrient Content and Physicochemical Properties of Cavendish Bananas var. Pei Chiao during Ripening. Horticulturae. 2024;10. https://doi.org/10.3390/horticulturae10040384.

[5] Chillet M, P. Castelan F, Abadie C, Hubert O, De Lapeyre De Bellaire L. Necrotic leaf removal, a key component of integrated management of Mycospaerella leaf spot diseases to improve the quality of banana: The case of Sigatoka disease. Fruits. 2013; 68:271-7.

[6] Moscetti R, Zambelli S, Taormina E, Bandiera A, Benelli A, Riccardo M. Optimization on Banana Maturation Classification for Logistics Efficiency Using Computer Vision. Lect Notes Civ Eng. 2025;586 LNCE:505-12. https://doi.org/10.1007/978-3-031-84212-2_62.

[7] Malabag BA, Santiago CS, Cahapin EL, Reyes JL, Legaspi GS. Fuzzy Logic-Based Size and Ripeness Classification of Banana using Image Processing Technique. Int J Emerg Technol Adv Eng. 2022;12:11–8. https://doi.org/10.46338/ijetae1022_02.

[8] Mo S, Dong T, Zhao X, Kan J. Discriminant model of banana fruit maturity based on genetic algorithm and SVM. J Fruit Sci. 2022;39:2418-27. https://doi.org/10.13925/j.cnki.gsxb.20210586.

[9] Guo J, Fu H, Yang Z, Li J, Jiang Y, Jiang T, et al. Research on the physical characteristic parameters of banana bunches for the design and development of postharvesting machinery and equipment. Agric. 2021;11. https://doi.org/10.3390/agriculture11040362.

[10] Kumara A, Kanchana K, Senerath A, Thiruchchelvan N. Use of maturity traits to identify optimal harvestable maturity of banana Musa AAB cv. "embul" in dry zone of Sri Lanka. Open Agric. 2021;6:143–51. https://doi.org/10.1515/opag-2021-0015.

[11] Widodo SE, Waluyo S, Latansya R. Detection of fruit maturity of "Cavendish" banana using thermal image processing. In: null R, Y.F. C, E.D. H, P. P, D. E, null A, editors. AIP Conf. Proc., vol. 2616, Department of Agronomy and Horticulture, Jl. Prof. Dr. Sumantri Brojonegoro No. 1, Bandar Lampung, 35145, Indonesia: American Institute of Physics Inc.; 2023. https://doi.org/10.1063/5.0135795.

[12] Wang M, Wang B, Zhang R, Wu Z, Xiao X. Flexible Vis/NIR wireless sensing system for banana monitoring. Food Qual Saf. 2023;7. https://doi.org/10.1093/fqsafe/fyad025.

[13] Ramadhan YA, Djamal EC, Kasyidi F, Bon AT. Identification of cavendish banana maturity using convolutional neural networks. Proc. Int. Conf. Ind. Eng. Oper. Manag., IEOM Society; 2020.

[14] Wang J-J. Recognition system for fruit classification based on 8-layer convolutional neural network. EAI Endorsed Trans e-Learning. 2022;7:173455. https://doi.org/10.4108/eai.17-2-2022.173455.

[15] Mazen FMA, Nashat AA. Ripeness Classification of Bananas Using an Artificial Neural Network. Arab J Sci Eng. 2019;44:6901-10. https://doi.org/10.1007/s13369-018-03695-5.

[16] Maity I, Samanta S. Utilization of Image Processing Tools for a Comparative Study on RGB and HSV Color Space: Targeting Feature Extraction of Banana Fruit Image During Different Ripeness Stages. 2024 IEEE Silchar Subsect. Conf. SILCON 2024, Institute of Electrical and Electronics Engineers Inc.; 2024. https://doi.org/10.1109/SILCON63976.2024.10910858.

[17] Prabha DS, Kumar JS. Assessment of banana fruit maturity by image processing technique. J Food Sci Technol. 2015;52:1316–27. https://doi.org/10.1007/s13197-013-1188-3.

[18] Sandra, Prayogi IY, Damayanti R, Djoyowasito G. Design to prediction tools for banana maturity based on image processing. In: S. S, W.B. S, H.Y. S, N.M. S, P. S, M. N, et al., editors. IOP Conf. Ser. Earth Environ. Sci., vol. 475, Institute of Physics Publishing; 2020. https://doi.org/10.1088/1755-1315/475/1/012010.
[19] Chuquimarca LE, Vintimilla BX, Velastin SA. A review of external quality inspection for fruit grading using CNN models. Artif Intell Agric. 2024;14:1–20. https://doi.org/10.1016/j.aiia.2024.10.002.
[20] Fang W, Zhang F, Sheng VS, Ding Y. A method for improving CNN-based image recognition using DCGAN. Comput Mater Contin. 2018;57:167–78. https://doi.org/10.32604/cmc.2018.02356.
[21] Prakash AJ, Prakasam P. An intelligent fruits classification in precision agriculture using bilinear pooling convolutional neural networks. Vis Comput. 2023;39:1765–81. https://doi.org/10.1007/s00371-022-02443-z.
[22] Dai D. An Introduction of CNN: Models and Training on Neural Network Models. Proc. - 2021 Int. Conf. Big Data, Artif. Intell. Risk Manag. ICBAR 2021, Institute of Electrical and Electronics Engineers Inc.; 2021, p. 135-38. https://doi.org/10.1109/ICBAR55169.2021.00037.
[23] Mutrofin S, Setiawan E, Fatichah C, Yuniarti H. Convolutional Neural Networks Performance Investigation in Banana Ripeness Classification: Impact of Model, Padding, and Optimizer. 2024 9th Int. Conf. Informatics Comput. ICIC 2024, Institute of Electrical and Electronics Engineers Inc.; 2024. https://doi.org/10.1109/ICIC64337.2024.10956746.
[24] Huong PT, Hien LT, Son NM, Tuan HC, Nguyen TQ. Enhancing deep convolutional neural network models for orange quality classification using MobileNetV2 and data augmentation techniques. J Algorithms Comput Technol. 2025;19. https://doi.org/10.1177/17483026241309070.
[25] Nafi'udin F, Pratiwi H, Zukhronah E. Efficiency and Accuracy of Convolutional and Fourier Transform Layers in Neural Networks for Medical Image Classification. Barekeng. 2024;18:2387-96. https://doi.org/10.30598/barekengvol18iss4pp2387-2396.
[26] Yashu, Kukreja V, Srivastava P, Garg A. Fruitful Fusion: CNN-Random Forest Synergy in Banana Ripeness Detection. 2024 IEEE Int. Conf. Inf. Technol. Electron. Intell. Commun. Syst. ICITEICS 2024, Institute of Electrical and Electronics Engineers Inc.; 2024. https://doi.org/10.1109/ICITEICS61368.2024.10625429.
[27] Nafi'Iyah N, Wardhani R, Prakasa E. Identification of Banana Ripeness using Convolutional Neural Network Approaches. Proc - 2023 10th Int Conf Comput Control Informatics Its Appl Explor Power Data Leveraging Inf to Drive Digit Innov IC3INA. 2023 2023:330–5. https://doi.org/10.1109/IC3INA60834.2023.10285749.
[28] Saranya N, Srinivasan K, Kumar SKP. Banana ripeness stage identification: a deep learning approach. J Ambient Intell Humaniz Comput. 2022;13:4033–9. https://doi.org/10.1007/s12652-021-03267-w.
[29] Arunima PL, Gopinath PP, Geetha Lekshmi PR, Esakkimuthu M. Digital assessment of post-harvest Nendran banana for faster grading: CNN-based ripeness classification model. Postharvest Biol Technol. 2024;214:112972. https://doi.org/10.1016/j.postharvbio.2024.112972.
[30] Kumar PV, George KM, Nair AK, Sangeetha M. Palm vein image classification using neural network. Int J Recent Technol Eng. 2018;7:122-4.
[31] Bi Y, Xue B, Zhang M. Introduction. Adapt Learn Optim. 2021;24:1-10. https://doi.org/10.1007/978-3-030-65927-1_1.
[32] Mohamedon MF, Abd Rahman F, Mohamad SY, Omran Khalifa O. Banana Ripeness Classification Using Computer Vision-based Mobile Application. Proc 8th Int Conf Comput Commun Eng ICCCE; 2021 2021;2019:335–8. https://doi.org/10.1109/ICCCE50029.2021.9467225.
[33] Han X, Zhang L, Zhao Y, Wang C. Banana ripeness determination based on CNN and XgBoost. Food Mach. 2024;40. https://doi.org/10.13652/j.spjx.1003.5788.2024.60015.
[34] Raghavendra S, Ganguli S, Selvan PT, Nayak MM, Chaudhurry S, Espina RU, et al. Deep Learning Based Dual Channel Banana Grading System Using Convolution Neural Network. J Food Qual. 2022. https://doi.org/https://doi.org/10.1155/2022/6050284.
[35] Gour A, Bhanodia PK, Sethi KK, Rajput S. Novel Framework for Image Classification Based on Patch-Based CNN Model. Lect Notes Networks Syst. 2024;786:317-37. https://doi.org/10.1007/978-981-99-6547-2_25.
[36] Zhou L, Li Q, Huo G, Zhou Y. Image classification using biomimetic pattern recognition with convolutional neural networks features. Comput Intell Neurosci. 2017;2017. https://doi.org/10.1155/2017/3792805.
[37] Rababaah AR. Deep learning of human posture image classification using convolutional neural networks. Int J Comput Sci Math. 2022;15:273-88. https://doi.org/10.1504/IJCSM.2022.10049409.
[38] Gupta S, Tripathi AK, Lewis N. Pre-trained noise based unsupervised GAN for fruit disease classification in imbalanced datasets. Pattern Anal Appl. 2025;28. https://doi.org/10.1007/s10044-025-01418-9.
[39] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition Vol 12 Issue 08. SSRN Electron J. 2012;12:301–7.
[40] Erbani J, Portier P-É, Egyed-Zsigmond E, Nurbakova D. Confusion Matrices: A Unified Theory. IEEE Access. 2024. https://doi.org/10.1109/ACCESS.2024.3507199.
[41] Kounev S, Lange K-D, von Kistowski J. Statistical Measurements. Syst. Benchmarking Sci. Eng. 2nd Ed., Springer Nature; 2025, p. 71–100. https://doi.org/10.1007/978-3-031-85634-1_4.
[42] Rainio O, Teuho J, Klén R. Evaluation metrics and statistical tests for machine learning. Sci Rep. 2024;14:1–14. https://doi.org/10.1038/s41598-024-56706-x.