ILKOM Jurnal Ilmiah Vol. 17, No. 3, December 2025, pp.213-226
Accredited 2nd by KEMDIKTISAINTEK No. 10/C/C3/DT.05.00/2025; E-ISSN 2548-7779 | P-ISSN 2087-1716

Evaluation of Deep Learning Model for Detection of Banana Consumption Feasibility Using Yolov8 Method

Guntur Eka Saputra [a,1,*]; Revanza Raditya Putra Yanni [a,2]
[a] *Universitas Gunadarma, Jl. Margonda Raya 100, Depok, Indonesia*
[1] *guntur@staff.gunadarma.ac.id* ; [2] *revanza.rpy@gmail.com*
** Corresponding author*

Article history: Received April 21, 2025; Revised May 01, 2025; Accepted September 12, 2025; Available online October 29, 2025

Abstract

This study aims to improve the accuracy of banana edibility detection using the YOLOv8 deep learning model. A total of 346 banana images were captured using a smartphone camera and split into training (303), validation (29), and testing (14) subsets. The research framework consisted of four main stages: data collection, preprocessing, model training, and performance evaluation. Preprocessing was conducted using the Roboflow platform and included several techniques such as image annotation, resizing, automatic orientation correction, contrast adjustment, and data augmentation through rotation, mosaic, and noise addition to enrich data variation and model robustness. The YOLOv8 model was trained for 60 epochs, achieving optimal convergence in 0.173 hours. Random search was utilized for hyperparameter optimization to achieve the best model configuration. The evaluation demonstrated remarkable results with a precision of 99.7%, recall of 100%, and mean Average Precision (mAP) of 99.5%. Visualization metrics, including the Precision-Confidence, Recall-Confidence, and F1-Confidence curves, each reached 100%, and the normalized confusion matrix demonstrated flawless classification performance. Testing on unseen data further confirmed the model's ability to accurately detect and classify bananas into Good Quality and Bad Quality classes with high confidence scores. These findings highlight the capability of YOLOv8 as a robust and reliable model for automated fruit quality assessment. The implementation of this approach offers a non-destructive, fast, and consistent method for evaluating banana edibility, reducing dependency on manual inspection and human error. In addition, this study contributes to the advancement of smart agriculture and post-harvest management by demonstrating the potential of deep learning and computer vision to support real-time quality control and decision-making in the agricultural industry.

Keywords: Deep Learning, Banana, Consumption Feasibility, Detection, YOLOv8

Introduction

Bananas are one of the most important tropical fruits in the global economy, particularly in developing countries. Nearly 90% of global banana production comes from regions in Asia, Latin America, and Africa [1], [2] . Known for their high nutritional value and versatility in processing into various food products, bananas are widely consumed worldwide [3], [4]. In Indonesia, bananas are particularly popular, with an average consumption of 24.71 grams per capita per day, making them the most consumed fruit [5], [6]. According to the Coordinating Ministry for Economic Affairs, bananas are the most consumed fruit, with an average of 24.71 grams/capita/day [7].

This growing demand for bananas has led to a greater focus on production, which heavily depends on the post-harvest stage. The ripening process, a crucial factor in determining the nutritional quality of bananas, directly influences organoleptic characteristics such as taste, texture, and freshness, affecting both consumer satisfaction and market value [8]. Achieving optimal ripeness is essential for ensuring bananas are delivered in their best condition to consumers.

Traditionally, banana ripeness detection has been done manually by farmers, retailers, and producers, who rely on experience, visual assessments, and destructive methods such as peeling or cutting the fruit [1], [9]. While these methods can be accurate, they are costly, time-consuming, and subjective, leading to inconsistencies in product quality. This becomes particularly inefficient on an industrial scale, where the risk of overripe or underripe bananas entering the market is high [8], [10]. As such, there is a clear need for more accurate, efficient, and non-destructive techniques for ripeness detection. Therefore, a more accurate, efficient, and reliable approach is needed to detect the

level of ripeness of bananas, such as non-destructive techniques. In this approach, testing is carried out without damaging the fruit by utilizing the analysis of physical and spectral properties, such as color, texture, and chemical composition using deep learning methods [1], [11], [12], [13].

Recent developments in image processing and deep learning have provided new solutions to this problem. Techniques such as thermal imaging and color space transformations (HSV and HSI) have been explored, but they are limited by environmental factors and lighting conditions, which affect the accuracy of detection [14], [15], [16]. To address these limitations, artificial intelligence (AI) and computer vision, specifically deep learning models such as neural networks, are increasingly being utilized for fruit ripeness detection [17], [18], [19], [20], [21]. Various studies have applied machine learning and deep learning approaches to detect or classify the ripeness of bananas. The Support Vector Machine (SVM) model achieves 98.89% accuracy with 1242 banana images [22] K-Nearest Neighbor (KNN) showed an accuracy of 89.30% with 194 images [23] while the combination of HSV and KNN techniques produced 95.52% accuracy [24].

Among these models, the Convolutional Neural Network (CNN) has shown great promise for detecting banana ripeness with accuracies up to 96.14% [25], [26]. CNNs are widely used in computer vision tasks such as image classification, object detection, and image recognition [27] Meanwhile, the You Only Look Once (YOLO) algorithm, particularly in its latest iteration YOLOv8, offers real-time object detection with high precision, making it a strong candidate for industrial applications requiring rapid and reliable ripeness detection [28], [29]. YOLOv8 has consistently demonstrated accuracy levels above 90%, showing its potential to improve efficiency in the banana ripeness detection process [30]. However, most existing studies are still limited to controlled laboratory conditions with small or less diverse datasets, and often focus only on accuracy without considering real-world challenges such as varying lighting conditions, fruit occlusions, and scalability for large-scale deployment. These limitations indicate a research gap in developing and evaluating robust detection models that can perform reliably in dynamic agricultural environments, thus motivating further exploration of deep learning approaches for practical and scalable banana ripeness detection systems.

This study aims to explore the use of YOLOv8 in automatically detecting the ripeness of bananas, reducing reliance on manual judgment and human error, while ensuring product quality and improving operational efficiency throughout the banana supply chain.

This study introduces a more effective and accurate model for banana ripeness classification by integrating the latest YOLOv8 object detection algorithm with various image processing techniques, such as image annotation, auto-orienting, resizing, contrast auto-adjust, rotation augmentation, noise augmentation, and mosaic augmentation. By optimizing the YOLOv8 parameters, the model achieves a mean Average Precision (mAP) of 99.5%, significantly enhancing classification accuracy. Additionally, a custom dataset has been developed, processed with these techniques to ensure high quality and suitability for training the model. The findings are expected to contribute to advancements in agricultural technology, particularly in the post-harvest management of bananas.

Method

Figure 1. Flow Diagram of Study

The Method of this study is using SEMMA method. The aim of this study is to evaluate of deep learning model for detection of bananas consumption feasibility using the YOLOv8 architecture. Dataset of banana images is needed to train the model. Specifically, images of bananas which are classified feasible and unfeasible. The dataset used is the primary dataset. The data is preprocessed using augmentation in roboflow to expand the dataset and divide it into validation, training and testing. YOLOv8 models are created using preprocessed dataset. **Figure 1** shows the flow diagram of study.

A. Sample

The dataset in this study was obtained by capturing images of banana using a smartphone. The total amount of data collected was 346 images. The dataset was split to partition the data set into training, validation, and testing subsets. [31]. The training subset is employed to fine-tune the parameter of model. The validation subset is used to configure the hyperparameter of model. The testing subset is used to assess the performance of model against the training data. The dataset in this study is divided into 88% training subset, 8% validation subset, and 4% testing subset. **Figure 2** shows a training data subset of 303 images, a validation data subset of 29 images, and a testing data subset of 14 images

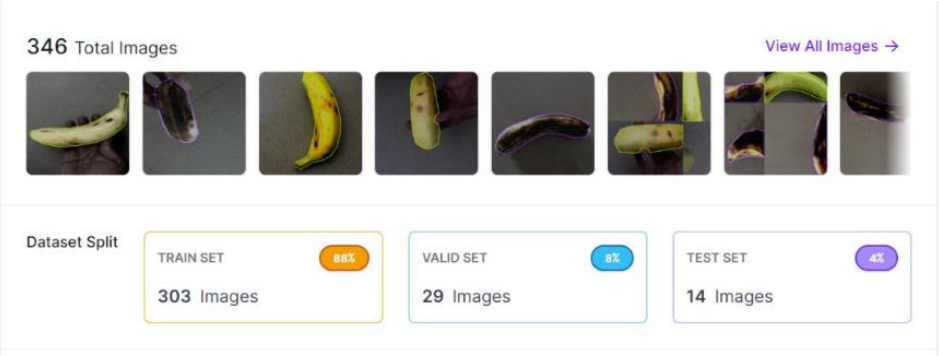


Figure 2. Dataset

B. Explore

Quality The dataset further divided in two classes such as good quality banana and bad quality banana. The images were in .jpg format and 1600 × 900 as its image size. Feasible bananas usually have a bright yellow skin color with a slight hint of small brown spots. This ripeness can also be measured by a decrease in chlorophyll levels and an increase in yellow pigment in banana peels [32], [33].

Overripe or rotten bananas tend to have skin with many large brown spots or even blacken. Overripe or rotten bananas are usually too mushy, and the flavor may have started to deteriorate. Bananas with mostly black skin are considered unfit for consumption, especially if the texture of the flesh has also been damaged [33]. The following **Table 1** is an example of good quality banana and bad quality banana.

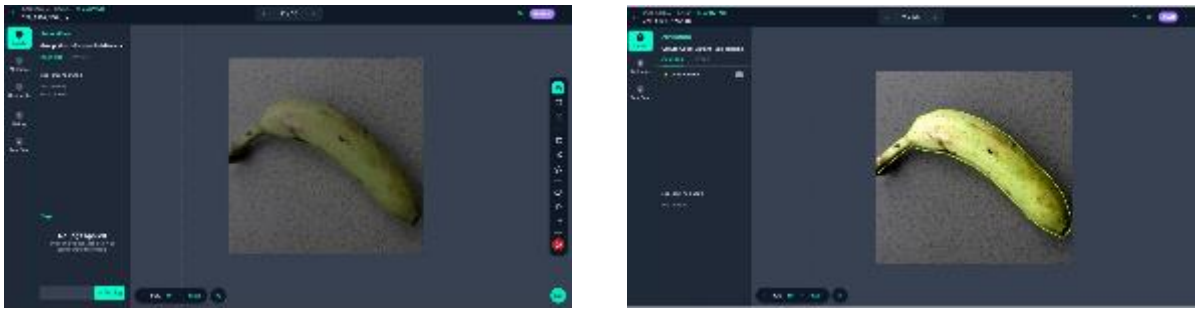
Table 1. Example of Dataset

Data Type	Good Quality Banana	Bad Banana
Primary data (Smartphone)		

C. Modify

Modify data or data Preprocessing was carried out using the Roboflow platform. Several preprocessing stages are carried out such as annotation, split dataset, auto-orient, resize, auto-adjust contrast, rotate augmentation, noise augmentation, and mosaic augmentation.

The first stage of data preprocessing is Annotation. The annotation stage was the implementation of labeling on the image using annotation bounding boxes to provide information on the position and type of image [34]. The annotation process is used to produce a dataset used for training with high image quality to support object detection. **Figure 3a** shows that the banana image before annotation has poor image quality, then the annotation is applied resulting in an image with better quality as seen in **Figure 3b**.



(a) Before Annotation (b) After Annotation

Figure 3. Annotations

The second stage of data preprocessing is Auto-Orient. The auto-orient stage is used to overcome differences in object orientation in images in the dataset [35]. The auto-orient feature aims to automatically adjust the orientation of objects in the dataset to produce precise, consistent, and accurate object placement. **Figure 4** shows the auto-orient stages of various object orientations.

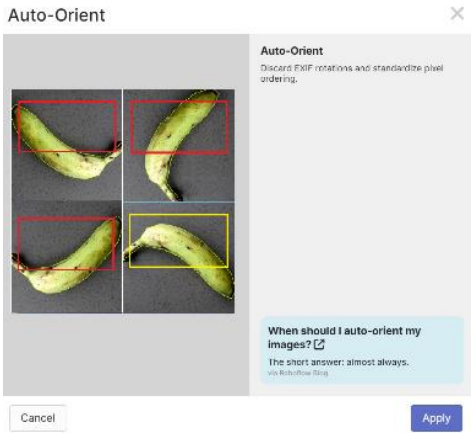


Figure 4. Auto-orient

The third stage of data pre-processing is resizing. The resize stage on the Roboflow platform is used to transform the image size. The image size used is 640×640 pixels because important details in the image can be maintained and can reduce computation. The resize stage aims to change the size of the image so that the model training process is more efficient. **Figure 5** shows the resizing stages to change the size of the original image to a 640×640 pixels of images.

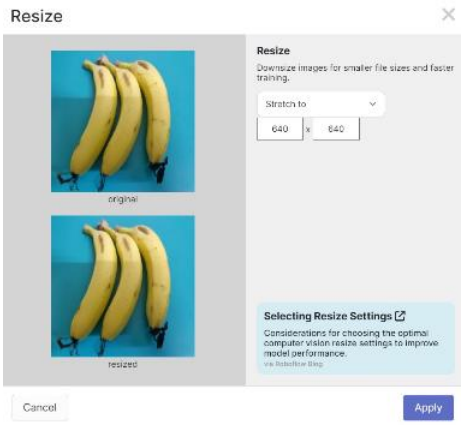


Figure 5. Resize

The fourth stage of data pre-processing is Auto-Adjust Contrast. The auto-adjust contrast stage on the RoboFlow platform is used to adjust the image contrast so that the image contrast can be increased according to the image histogram. Auto-adjust contrast also helps in normalizing edge detection on objects in various lighting conditions. The

auto-adjust contrast technique used is Contrast-Stretching. **Figure 6** shows the adjusted image and the differences between the pixels are very clearly visible so that the image details are easier to see.

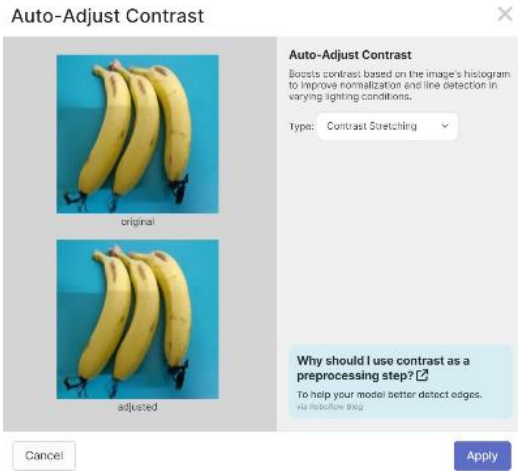
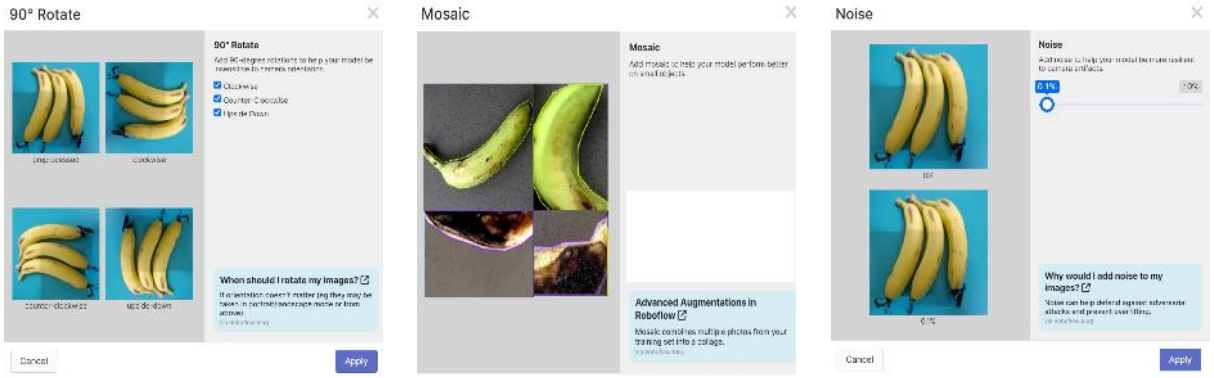


Figure 6. Auto-Adjust Contrast

The fifth stage of pre-processing is Augmentation. The augmentation techniques used are 90˚ rotate, noise, and mosaic. The 90˚ rotate augmentation stage is used to rotate the image by a 90˚ angle clockwise or counter clockwise. The 90˚ rotate augmentation technique also aims to increase image variations in machine learning model training. The large number of variations in image rotation causes the model to be more robust to variations in object orientation.

The noise augmentation stage is used to add noise to the image. Noise augmentation aims to enable machine learning models to improve their ability to deal with unclean image data, so that the model can recognize objects from images that contain noise. The mosaic augmentation stage is used to combine several sub-parts of the image into one image. The original image is divided into several image subsections arranged in a grid. Mosaic augmentation aims to increase data variation by increasing the number of objects in one image, so that machine learning can recognize objects in various contexts. The augmentation is seen in **Figure 7**. The 90˚ rotate augmentation in **Figure 7a** uses clockwise, counter-clockwise, and upside down techniques. The noise augmentation in **Figure 7b** uses noise reaching 0.1%. The mosaic augmentation in **Figure 7c** divides one image into several image sub-sections.



(a) 90˚ rotate (b) Mosaic (c) Noise

Figure 7. Augmentation

D. Model

Data modelling is used for training and testing data which aims to depict data mathematically [36]. Data modelling can determine predictions based on identifying patterns, trends, or relationships between data in a dataset [37]. Architecture model of banana consumption feasibility detection shows in **Figure 8**.

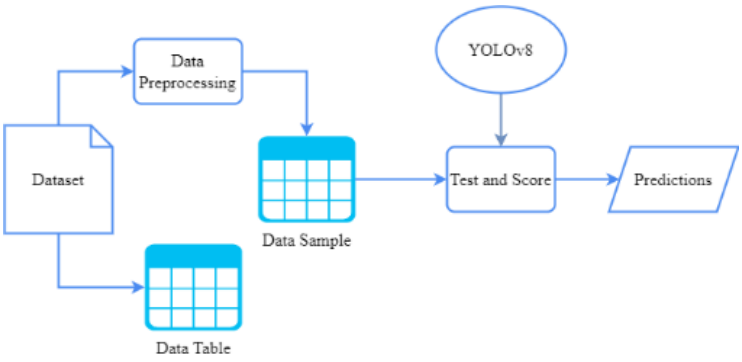


Figure 8. Architecture Model

Figure 8 shows the architecture model with YOLOv8 algorithm. Widget dataset provided as input file [36]. Data preprocessing widget is used to enhance data quality using annotation, auto-orient, resize, auto-adjust contrast, rotate augmentation, noise augmentation, and mosaic augmentation. The data sample widget is data after preprocessing [38]. The deep learning method is applied using YOLOv8. The efficacy of YOLOv8 is assessed using the test and score widget. The forecast for the testing data is produced by the prediction widget.

YOLOv8 was released on January 10, 2023 by Ultralytics, a community that developed YOLOv5 [39]. YOLOv8 introduces architectural advancements that enhance speed, accuracy, and versatility, making it suitable for various domains, including agriculture, autonomous vehicles, and surveillance [13], [39]. This version aims to balance inference speed and detection accuracy, crucial for real-world tasks like detecting flying objects or fruits in agricultural environments [40].

The core architecture of YOLOv8 consists of two primary components: the backbone and the head. YOLOv8 employs a modified CSPDarknet53 (Cross Stage Partial Darknet) backbone, composed of 53 convolutional layers. This backbone extracts essential features from input images using the C2f module, which enhances feature extraction by concatenating outputs from multiple bottleneck layers. This approach is particularly effective for detecting smaller objects. Additionally, the Spatial Pyramid Pooling Faster (SPPF) module processes features at various scales, improving the model's ability to detect objects of different sizes [41]. YOLOv8's head is designed to be detachable, allowing for the separation of object scoring, classification, and bounding box regression tasks, which ultimately enhances precision and overall accuracy [40].

One of the key innovations in YOLOv8 is its anchor-free detection mechanism, a significant departure from previous YOLO models. Instead of relying on predefined anchor boxes, YOLOv8 predicts the centers of objects directly, improving speed by reducing the number of bounding box predictions. This streamlined approach enhances post-processing tasks such as Non-Maximum Suppression (NMS) and makes YOLOv8 especially suitable for real-time applications like agricultural fruit detection, where speed and accuracy are essential. Additionally, YOLOv8 introduces a new convolutional layer, the C2f layer, which replaces the C3 layer from YOLOv5, optimizing feature reuse and improving detection accuracy for small and complex objects [39].

In terms of performance, YOLOv8 is versatile, supporting training across a wide range of hardware, from low-end systems to cloud-based platforms. This flexibility makes it suitable for various applications, including image classification, object detection, and segmentation. For example, in tasks like fruit detection, YOLOv8 has demonstrated near-perfect precision and high recall rates, minimized false positives and made it an ideal solution for automated harvesting and similar tasks. The model also performs exceptionally well in detecting fast-moving and small objects, such as flying drones or aircraft, even under challenging conditions like occlusion and small object size, as demonstrated by Reis [41].

Real-world applications further highlight YOLOv8's effectiveness. In agriculture, YOLOv8 has been applied to optimize fruit detection and classification in real-time. Its ability to handle varying object sizes and adapt to different environmental conditions makes it valuable for precision farming. Studies have shown that YOLOv8 can accurately detect fruits under challenging conditions, such as varying lighting and object occlusion, making it a highly adaptable tool in agricultural automation. Similarly, in flying object detection, YOLOv8 achieved a remarkable 50 frames per second (fps) with a mean Average Precision (mAP50) of 99.1%, demonstrating its exceptional speed and accuracy in identifying fast-moving objects like drones and airplanes [40]. In this study, random search was adopted for hyperparameter optimization because it allows efficient exploration of parameter spaces with minimal computational

cost. This makes it particularly suitable for smaller datasets, where quick and reliable performance tuning is essential [42].

E. Yolov8 Model for Banana Classification

When inputting a 640x640 image of a banana into the YOLOv8 network for the purpose of classifying the banana as either feasible or unfeasible to consume, the network follows a structured process that begins with preprocessing at the input layer and culminates in the final classification at the output layer. Given that the input image already matches the network's expected dimensions, it bypasses any resizing operations, thus preserving the integrity of important features such as color, texture, and visual markers of decay or damage, which are crucial for determining whether the banana is fit for consumption. The image is then normalized by scaling the pixel values, typically between 0 and 1, to stabilize the network's performance. If handling multiple images, they are batched and converted into tensors that are used as inputs to the network.

The backbone of YOLOv8 is responsible for extracting the key features needed to classify the banana. Based on the CSPDarknet53 architecture, this backbone effectively captures features at multiple scales. As the image is passed through the initial convolutional layers, the network detects basic visual features such as edges, shapes, and color gradients. These layers help to identify critical cues, such as the overall color and surface condition of the banana, focusing on whether the banana appears healthy and fresh (feasible to consume) or shows signs of spoilage or rot (unfeasible to consume). As the image progresses deeper into the network, subsequent layers capture more abstract and complex features, such as variations in texture, the appearance of mold or black spots, and bruising, which are often indicators of a banana that is no longer suitable for consumption.

Once the feature extraction is complete, the neck of the YOLOv8 network aggregates these multi-scale features through the Spatial Pyramid Pooling Fast (SPPF) module and Path Aggregation Network (PAN). These components combine information from different scales, allowing the model to analyze both the broader characteristics of the banana (such as its overall shape and color distribution) and finer details, like small blemishes or areas of rot. The PAN module ensures that crucial contextual information is preserved, allowing the model to make a well-informed classification decision based on subtle differences in appearance that help determine if the banana is fit for consumption or not. This multi-scale feature aggregation is essential for the robustness and accuracy of the classification process.

In the final stage of processing, the head of the YOLOv8 network performs object detection and classification. While the head predicts bounding boxes that localize the banana within the image, the primary objective in this case is classification. Based on the features extracted by the earlier layers, the network outputs a prediction indicating whether the banana is "feasible" or "unfeasible" to consume. This prediction is accompanied by a confidence score, which reflects the network's certainty in its classification. This confidence score is critical for evaluating the reliability of the model's decision, especially when determining the edibility of the banana. If multiple predictions are made, post-processing techniques like Non-Maximum Suppression (NMS) can be applied to refine the results, though in a scenario where a single banana is present, this step may not be necessary.

By inputting a 640x640 image of a banana into the YOLOv8 network, we leverage the network's sophisticated architecture to accurately classify whether the banana is feasible or unfeasible to consume. YOLOv8's backbone, neck, and head work together to extract, aggregate, and analyze multi-scale features, ultimately producing a reliable classification based on the visual condition of the banana. The final output includes a class label—either "feasible" or "unfeasible"—along with a confidence score that provides insight into the network's certainty in its classification. This process demonstrates YOLOv8's effectiveness in making informed and efficient classifications, particularly in scenarios where the accurate determination of food edibility is crucial.

Results and Discussion

A. Result of Sample

The dataset used in this study was collected using a smartphone camera, and all images were manually labelled to ensure annotation accuracy. In total, 346 images were obtained. The dataset was divided into 88% training data, 8% validation data, and 4% test data, corresponding to 303 training images, 29 validation images, and 14 test images. The proportion of training data was intentionally made larger to allow the model to learn as many variations as possible from the limited dataset. A higher percentage of training samples is particularly important in this case, given the relatively small number of images, as it helps the model capture more representative features and improves its generalization capability. Meanwhile, smaller portions of validation and test data were retained to monitor performance

and evaluate the model without discarding important training information. The splitting data of the dataset into training, validation, and test subsets is illustrated in **Figure 9**.

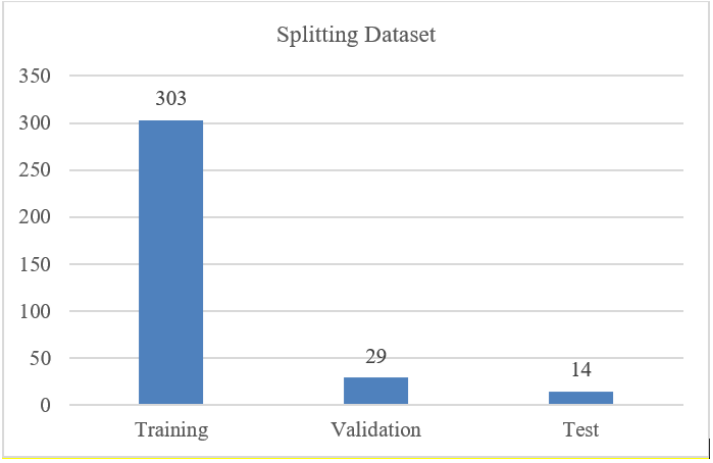


Figure 9. Splitting Dataset

B. Assess

This study was performed on a computer with Apple M3 @ 4.06 GHz Processor and 8 GB RAM. The software used is Google Colaboratory and Roboflow.

Evaluation was carried out to measure the performance model in detecting the feasibility of banana consumption using the YOLOv8 method. This evaluation aims to determine how efficient the model is in identifying the feasibility of banana consumption in new images. The evaluation model is carried out on training results and testing results.

C. Evaluation of Training Results

Prior to the main training stage, hyperparameter optimization was performed using the Random Search method, considering the relatively small size of the dataset. The optimization process consisted of 5 iterations, with each iteration including 10 training epochs. The results of the best hyperparameter optimization using YOLOv8 are summarized in **Table 2**.

Table 2. Best Hyperparameter Optimization using Random Search

Hyperparameter	Value	Description
lr0	0.0010	initial learning rate at start of training
lrf	0.0010	final learning rate factor applied at the end of training
momentum	0.9370	momentum parameter to accelerate gradient descent convergence
weight_decay	0.0001	regularization term to reduce overfitting by penalizing large weights
hsv_h	0.0050	hue augmentation (color shift).
hsv_s	0.6000	saturation augmentation (color intensity)
hsv_v	0.3000	value augmentation (image brightness)
translate	0.0500	translation augmentation (shifting object position)
scale	0.4000	scaling augmentation (zoom in / out)
shear	0.1000	shear augmentation (geometric distortion)
perspective	0.0005	perspective transformation augmentation
flipud	0.0000	probability of flipping the image vertically (up-down)
fliplr	0.3000	probability of flipping the image horizontally (left-right)
mosaic	1.0000	mosaic augmentation (combining multiple images)
mixup	0.1000	mix-up augmentation (blending two images)

Model training was carried out using epochs 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100. The image size is 640 × 640 pixels, and the batch size is 16. The training results based on the number of epochs are in **Table 3**.

Table 3. Training Based on Epoch

Epoch	Time (Hour)	Precision	Recall	MAP
1	0.009	0.00551	0.955	0.313
10	0.032	0.921	0.978	0.987
20	0.066	0.960	1	0.995
30	0.093	0.951	1	0.995
40	0.119	0.992	1	0.995
50	0.148	0.992	1	0.995
60	0.173	0.997	1	0.995
70	0.220	0.946	1	0.995
80	0.279	0.943	0.992	0.995
90	0.346	0.965	1	0.995
100	0.432	0.945	1	0.995

The training model results in **Table 3** show indicate that the number of epochs influences both the training time and the model's performance metrics, including precision, recall, and MAP. As the number of epochs increases, the training duration becomes longer and the performance metrics vary accordingly. The best performance was observed at the 60 epochs, where the model achieved a precision of 0.997, a recall of 1.0, and a MAP of 0.995. Beyond this point, additional training did not provide further improvement and even caused slight fluctuations in precision, while recall and MAP remained relatively stable. These findings suggest that in the 60 epochs represents an optimal stopping point, balancing efficiency and accuracy. Therefore, the 60 epochs can be regarded as the optimal stopping point (early stopping), ensuring computational efficiency while maintaining stable performance. The stability of the model's performance at this stage is further illustrated in the performance curves and normalized confusion matrix, with the recall–confidence curve shown in **Figure 10** reinforcing the robustness of the model.

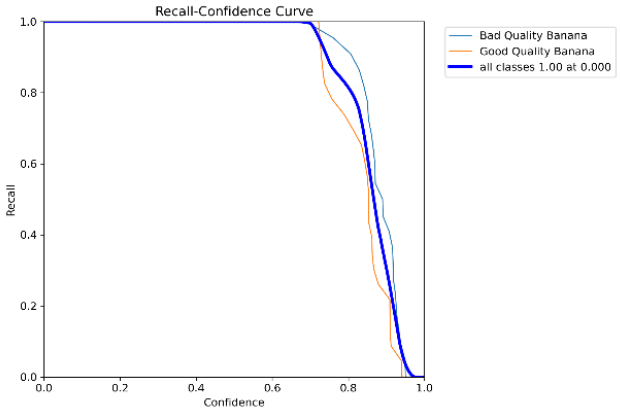


Figure 10. Recall-Confidence Curve

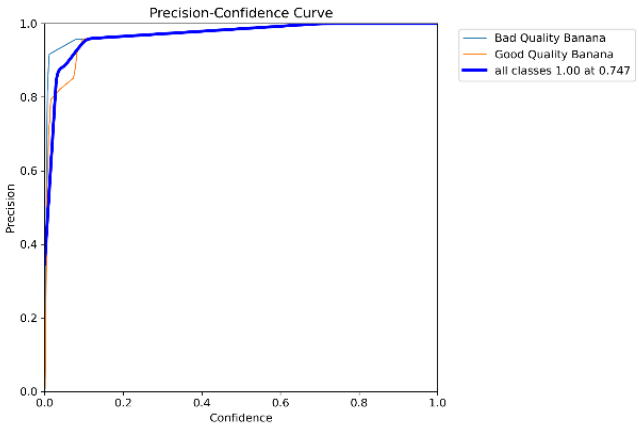


Figure 11. Precision-Confidence Curve

The recall-confidence curve, it visualizes the connection between the level of confidence and the corresponding recall values. The goal of the recall-confidence curve is to assess the model's performance in correctly identifying all true positive samples across different confidence thresholds [40]. It can be seen in Figure 9 that the recall-confidence value reaches 100%. The precision-confidence curve is shows in **Figure 11**.

The precision-confidence curve illustrates the relationship between precision and confidence values. Its objective is to evaluate the model's performance in accurately predicting all true positive samples at various confidence levels [40]. As shown in Figure 10, the precision-confidence value reaches 100%. The F1-confidence curve is displayed in **Figure 12**.

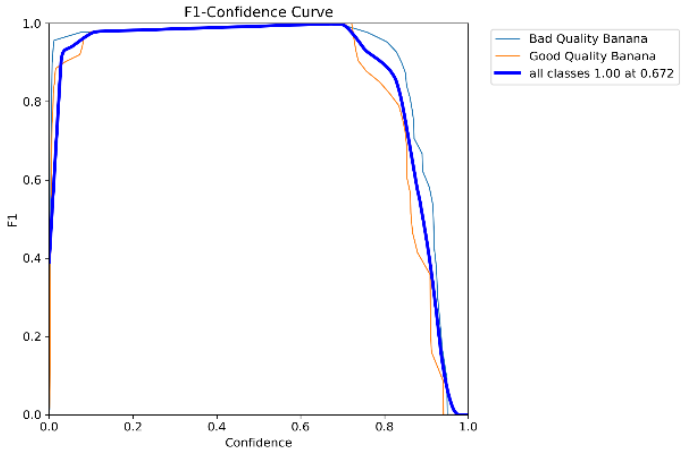


Figure 12. F1-Confidence Curve

The F1-confidence curve demonstrates the relationship between the F1 score and confidence. The F1 score represents the average comparison between precision and recall values [43]. The purpose of the F1-confidence curve is to evaluate model performance in balancing precision and recall at various levels of confidence [40]. It can be seen in **Figure 11** that the F1-confidence value reaches 100%. Normalized Confusion Matrix is shows in **Figure 13**.

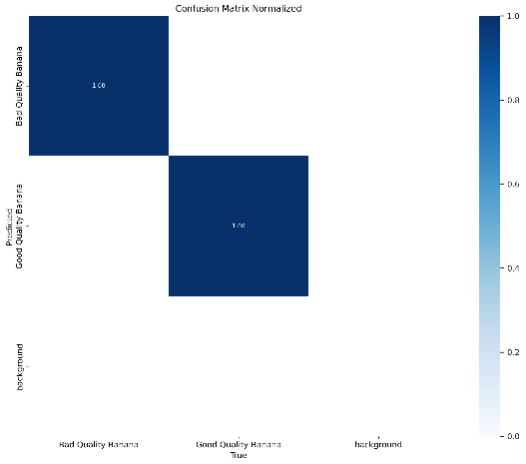


Figure 13. Normalized Confusion Matrix

The confusion matrix is a metric for assessing model performance, used to gauge the accuracy of the capability model to categorize data correctly [44]. The confusion matrix can be normalized in percentage form which provides a representation of model performance. It can be seen in **Figure 12** that the Normalized Confusion Matrix value reaches 100% in the Good Quality Banana and Bad Quality Banana classes.

D. Evaluation of Testing Results

Model testing was carried out using banana image data that was not used for training. This test was conducted to evaluate the model's performance in detecting the feasibility of banana consumption in the Good Quality Banana and Bad Quality Banana categories using the Confidence Score value. Testing was carried out on a model that had been trained using 60 epochs. The test results are shown in **Table 4**.

Table 4. Test Results for Detection of Banana Consumption Feasiblity

No	Object Detection	Confidence Score
1		Bad Quality Banana (0.87)
2		Good Quality Banana (0.83)
3		Good Quality Banana (0.78) Good Quality Banana (0.77)
4		Bad Quality Banana (0.90) Bad Quality Banana (0.83) Bad Quality Banana (0.85) Bad Quality Banana (0.94)

Table 3 shows the banana image used for testing with the confidence score and bounding box on all parts of the banana object which shows the banana classification. The results of image testing on numbers 1 and 4 show the classification of bananas in the "Bad Quality Banana" category with a confidence score reaching 0.94. The results of image testing in numbers 2 and 3 show the banana classification in the "Good Quality Banana" category with a confidence score reaching 0.87.

Conclusion

This research demonstrates that the YOLOv8 model effectively detects banana quality, categorizing them into 'Good Quality Banana' and 'Bad Quality Banana' classes. The model was trained using hyperparameter random search with 60 epochs, image size of 640 × 640 pixels and a batch size of 16. Model performance was evaluated using key metrics: a recall value of 100%, precision of 99.7%, and mAP of 99.5%. Additionally, visual representations of the model's performance were provided through Recall-Confidence, Precision-Confidence, and F1-Confidence curves, all reaching 100%, along with a normalized Confusion Matrix that shows 100% accuracy in both classes. Testing with

unseen images further confirmed the model's ability to detect bananas accurately, as indicated by the confidence score on the bounding boxes. However, it is important to note that comparisons with prior research methods are needed to clearly highlight the improvements and unique contributions of this study.

References

[1] P. Baglat, A. Hayat, F. Mendonça, A. Gupta, S. S. Mostafa, and F. Morgado-Dias, "Non-Destructive Banana Ripeness Detection Using Shallow and Deep Learning: A Systematic Review," *Sensors*, vol. 23, no. 2, pp. 1–20, 2023, doi: 10.3390/s23020738.

[2] P. P. Sarma, N. Gurumayum, A. K. Verma, and R. Devi, "A pharmacological perspective of banana: Implications relating to therapeutic benefits and molecular docking," *Food Funct*, vol. 12, no. 11, pp. 4749–4767, 2021, doi: 10.1039/d1fo00477h.

[3] N. Saranya, K. Srinivasan, and S. K. P. Kumar, "Banana ripeness stage identification: a deep learning approach," *J Ambient Intell Humaniz Comput*, vol. 13, no. 8, pp. 4033–4039, 2022, doi: 10.1007/s12652-021-03267-w.

[4] S. Sharma and S. Kaur, "Studies on banana nutrition: Function and processing kinetics," *Plant Cell Biotechnol Mol Biol*, vol. 21, no. 63–64, pp. 130–139, 2020.

[5] A. A. Nugraha, D. Darsono, and S. Marwanti, "Analysis of indonesian banana export performance in major export destination countries," *SVU-International Journal of Agricultural Sciences*, vol. 5, no. 2, pp. 127–136, 2023, doi: 10.21608/svuijas.2023.219439.1294.

[6] E. Rumapea, W. Roessali, and E. Prasetyo, "Analysis Of Consumer Attitudes And Preferences To Banana Purchase Decisions In The Traditional Market Of Semarang City," *Agrisocionomics: Jurnal Sosial Ekonomi Pertanian*, vol. 6, no. 1, pp. 94–104, 2021, doi: 10.14710/agrisocionomics.v6i1.8364.

[7] J. L. Loupatty *et al.*, "Effect of calcium carbide on the rate of ripening of Ambon Bananas (Musa paradisiaca Var. Sapientum (L). Kunt)," *BIOEDUPAT: Pattimura Journal of Biology and Learning*, vol. 3, no. 2, pp. 118–124, 2023.

[8] M. Utami, J. Andika, and S. Attamimi, "Artificial Intelligence For Banana's Ripeness Detection Using Conventional Neural Network Algorithm," *Jurnal Teknologi Elektro*, vol. 12, no. 2, p. 73, 2021, doi: 10.22441/jte.2021.v12i2.005.

[9] J. Guo, J. Duan, J. Li, and Z. Yang, "Mechanized technology research and equipment application of banana post-harvesting: A review," *Agronomy*, vol. 10, no. 3, pp. 1–15, 2020, doi: 10.3390/agronomy10030374.

[10] W. Widyawati and R. Febriani, "Real-time detection of fruit ripeness using the YOLOv4 algorithm," *Teknika: Jurnal Sains dan Teknologi*, vol. 17, no. 2, p. 205, 2021, doi: 10.36055/tjst.v17i2.12254.

[11] R. E. Saragih and A. W. R. Emanuel, "Banana Ripeness Classification Based on Deep Learning using Convolutional Neural Network," *3rd 2021 East Indonesia Conference on Computer and Information Technology, EIConCIT 2021*, pp. 85–89, 2021, doi: 10.1109/EIConCIT50028.2021.9431928.

[12] C. Wang, H. Wang, Q. Han, Z. Zhang, D. Kong, and X. Zou, "Strawberry Detection and Ripeness Classification Using YOLOv8+ Model and Image Processing Method," *Agriculture (Switzerland)*, vol. 14, no. 5, 2024, doi: 10.3390/agriculture14050751.

[13] X. Gao and Y. Zhang, "Detection of Fruit using YOLOv8-based Single Stage Detectors," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 12, pp. 83–91, 2023, doi: 10.14569/IJACSA.2023.0141208.

[14] Q. Wang, C. Yang, S. Duan, and S. Wei, "Research on Fire Prediction Algorithm Based on Thermal Infrared Image," in *ACM International Conference Proceeding Series*, 2021, pp. 216–221. doi: 10.1145/3467707.3467739.

[15] R. Arunkumar, C. V Dharani, N. S. Kavisri, and K. A. Dhanusha, "Enhanced Wildfire Detection in Forest Areas via OPENCV with Deep Convolutional Neural Networks for Smoke and Flame Segmentation," in *2025 International Conference on Emerging Smart Computing and Informatics, ESCI 2025*, 2025. doi: 10.1109/ESCI63694.2025.10988026.

[16] D. Sheng, W. Jin, X. Wang, and L. Li, "Thermal image visualization using multi-discriminator CycleGAN with unpaired thermal–visible image training set," *Infrared Phys Technol*, vol. 140, 2024, doi: 10.1016/j.infrared.2024.105352.

[17] B. Xiao, M. Nguyen, and W. Q. Yan, "Apple Ripeness Identification Using Deep Learning," in *Communications in Computer and Information Science*, 2021, pp. 53–67. doi: 10.1007/978-3-030-72073-5_5.

[18] A. Rendón-Vargas, A. Luna-Álvarez, D. Mújica-Vargas, M. Castro-Bello, and I. Marianito-Cuahuitic, "Application of Convolutional Neural Networks for the Classification and Evaluation of Fruit Ripeness," in *Communications in Computer and Information Science*, 2024, pp. 150–163. doi: 10.1007/978-3-031-77290-0_10.

[19] R. Ye, G. Shao, H. Zhang, T. Li, and Q. Gao, "Strawberry Maturity Detection Algorithm Based on Improved YOLOv9," in *Proceedings - 2024 International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2024*, 2024, pp. 759–766. doi: 10.1109/AEECA62331.2024.00133.

[20] R. Hamza and M. Chtourou, "Comparative Study on Deep Learning Methods for Apple Ripeness Estimation on Tree," in *Lecture Notes in Networks and Systems*, 2022, pp. 1325–1340. doi: 10.1007/978-3-030-96308-8_123.

[21] S. R. N. Appe, G. Arulselvi, and G. N. Balaji, "Tomato Ripeness Detection and Classification using VGG based CNN Models," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 1, pp. 296–302, 2023

[22] J. Enríquez and M. Macas, "Artificial Intelligence-Based Banana Ripeness Detection," in *Communications in Computer and Information Science*, 2023, pp. 197–211. doi: 10.1007/978-3-031-24985-3_15.

[23] C. H. Mendigoria, H. Aquino, R. Concepcion, O. J. Alajas, E. Dadios, and E. Sybingco, "Vision-based Postharvest Analysis of Musa Acuminata Using Feature-based Machine Learning and Deep Transfer Networks," in *IEEE Region 10 Humanitarian Technology Conference, R10-HTC*, 2021. doi: 10.1109/R10-HTC53172.2021.9641575.

[24] L. Ningsih, A. M. Priyatno, and A. Yusmar, "A New Triple-Weighted K-Nearest Neighbor Algorithm for Tomato Maturity Classification," *Jurnal RESTI*, vol. 9, no. 4, pp. 955–962, 2025, doi: 10.29207/resti.v9i4.6441.

[25] M. Utami, J. Andika, and S. Attamimi, "Artificial Intelligence For Banana's Ripeness Detection Using Conventional Neural Network Algorithm," *Jurnal Teknologi Elektro*, vol. 12, no. 2, p. 73, 2021, doi: 10.22441/jte.2021.v12i2.005.

[26] J. Guo, J. Duan, J. Li, and Z. Yang, "Mechanized technology research and equipment application of banana post-harvesting: A review," *Agronomy*, vol. 10, no. 3, pp. 1–15, 2020, doi: 10.3390/agronomy10030374.

[27] W. Hua, C. Li, and X. Wang, "Review of Convolutional Neural Network Models and Image Classification," *Academic Journal of Science and Technology*, vol. 10, no. 3, pp. 178–184, 2024, doi: 10.54097/644jqv20.

[28] F. M. A. Mazen and A. A. Nashat, "Ripeness Classification of Bananas Using an Artificial Neural Network," *Arab J Sci Eng*, vol. 44, no. 8, pp. 6901–6910, 2019, doi: 10.1007/s13369-018-03695-5.

[29] V. J. Sinanoglou *et al.*, "Quality Assessment of Banana Ripening Stages by Combining Analytical Methods and Image Analysis," 2023. doi: 10.3390/app13063533.

[30] C. Wang, H. Wang, Q. Han, Z. Zhang, D. Kong, and X. Zou, "Strawberry Detection and Ripeness Classification Using YOLOv8+ Model and Image Processing Method," *Agriculture (Switzerland)*, vol. 14, no. 5, 2024, doi: 10.3390/agriculture14050751.

[31] K. M. Kahloot and P. Ekler, "Algorithmic Splitting: A Method for Dataset Preparation," *IEEE Access*, vol. 9, pp. 125229–125237, 2021, doi: 10.1109/ACCESS.2021.3110745.

[32] F. M. A. Mazen and A. A. Nashat, "Ripeness Classification of Bananas Using an Artificial Neural Network," *Arab J Sci Eng*, vol. 44, no. 8, pp. 6901–6910, 2019, doi: 10.1007/s13369-018-03695-5.

[33] V. J. Sinanoglou *et al.*, "Quality Assessment of Banana Ripening Stages by Combining Analytical Methods and Image Analysis," 2023. doi: 10.3390/app13063533.

[34] K. Pugdeethosapol, M. Bishop, D. Bowen, and Q. Qiu, "Automatic Image Labeling with Click Supervision on Aerial Images," in *Proceedings of the International Joint Conference on Neural Networks*, 2020. doi: 10.1109/IJCNN48605.2020.9207363.

[35] F. Z. Cetin and M. F. Amasyali, "Reversed Image Detection with Convolutional Neural Networks," in *Proceedings - 2020 Innovations in Intelligent Systems and Applications Conference, ASYU 2020*, 2020. doi: 10.1109/ASYU50717.2020.9259899.

[36] R. J. Suhatril, R. D. Syah, M. Hermita, and B. Gunawan, "Evaluation of Machine Learning Models for Predicting Cardiovascular Disease Based on Framingham Heart Study Data," *Ilkom Jurnal Ilmiah*, vol. 16, no. 1, pp. 68–75, 2024, doi: 10.33096/ilkom.v16i1.1952.68-75.

[37] S. Gocheva-Ilieva, "Statistical Data Modeling and Machine Learning with Applications," *Mathematics*, vol. 9, no. 23, 2021, doi: 10.3390/math9232997.

[38] D. E. Birba, "A Comparative study of data splitting algorithms for machine learning model selection," KTH, School of Electrical Engineering and Computer Science (EECS), 2020.

[39] M. Sohan, T. Sai Ram, and Ch. V. Rami Reddy, "A Review on YOLOv8 and Its Advancements," in *International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*, I. J. Jacob, S. Piramuthu, and P. Falkowski-Gilski, Eds., Singapore: Springer Nature Singapore, 2024, pp. 529–545. doi: 10.1007/978-981-99-7962-2_39.

[40] L. YU and S. LIU, "A Single-Stage Deep Learning-based Approach for Real-Time License Plate Recognition in Smart Parking System," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, pp. 1142–1150, Jan. 2023, doi: 10.14569/IJACSA.2023.01409119.

[41] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," 2023.

[42] Y. A. Ali, E. M. Awwad, M. Al-Razgan, and A. Maarouf, "Hyperparameter Search for Machine Learning Algorithms for Optimizing the Computational Complexity," *Processes*, vol. 11, no. 2, 2023, doi: 10.3390/pr11020349.

[43] K. L. Kohsasih and Z. Situmorang, "Comparative Analysis of C4.5 and Naïve Bayes Algorithms in Predicting Cerebrovascular Disease," *Jurnal Informatika*, vol. 9, no. 1, pp. 13–17, 2022.

[44] D. Krstinić, A. K. Skelin, I. Slapničar, and M. Braović, "Multi-Label Confusion Tensor," *IEEE Access*, vol. 12, pp. 9860–9870, 2024, doi: 10.1109/ACCESS.2024.3353050.