Comparative Analysis of Banana Detection Models: Deep Learning and Darknet Algorithm

Original Scientific Paper

Abdul Haris Rangkuti
Bina Nusantara University
School of Computer Science, Computer Science
Jl Paskal, Bandung, Indonesia
rangku2000@binus.ac.id

Sian Lun Lau
Sunway University
School of Engineering and Technology
Kuala Lumpur, Malaysia
sianlunl@sunway.edu.my

Varyl Hasbi Athala
Bina Nusantara University
School of Computer Science, Computer Science
Jl Paskal, Bandung, Indonesia
varyl.athala@binus.ac.id

Rudi Aryanto
Bina Nusantara University
School of Computer Science, Computer Science
Jl Paskal, Bandung, Indonesia
rudiaryanto@binus.ac.id

Abstract *– This study aims to compare and evaluate the performance of banana detection models utilizing deep learning techniques and the Darknet algorithm. The objective is to identify the most effective approach for accurately detecting bananas in various real-world scenarios. The analysis involves training and testing multiple models using different datasets and evaluating their performance based on precision, recall, and overall accuracy. The results provide valuable insights into the strengths and weaknesses of each approach, enabling researchers and practitioners to make informed decisions when implementing banana detection systems. To detect banana objects, several convolutional neural network (CNN) models were used, including MobileNetV2, YOLOv3-Nano, YOLO Fastest 1.1, YOLOv3-tiny-PRN, YOLOv4-tiny, YOLOv7, and DenseNet121-YOLOv3. The training process utilizes the Darknet algorithm to facilitate the identification of banana types/classes captured by a camera, resulting in an MP4 film file. In this research, various experiments were carried out using different CNN models. However, these six models achieve optimal accuracy above 80%. Among them, the YOLOv7 model shows the highest average accuracy (MAP) at 100%, followed by the small model YOLOv4 at 92%. Meanwhile, for performance measurements, the accuracy of the YOLOv4-tiny model was 87%, followed by the YOLOv7 model at 84%. In the banana fruit experiment, several models showed very good performance, such as recognition of the Ambon, Kepok, and Emas banana classes up to 100% using the YOLOv7 and YOLOv4-tiny models. The YOLOv7 model itself can recognize other banana classes up to 100% in the Barangan, Rjbulu, Uli, and Tanduk classes. Furthermore, theYOLOv4-tiny model can recognize other banana classes, up to 90% of the Barangan, Rjbulu, Rjsereh, and Uli banana types. Thus, this experiment provides very good average accuracy results on 2 CNN models, namely YOLOv7 and YOLOv4-tiny. Future research will involve grouping pictures of bananas, which produces different image shapes, so it requires a different way to recognize them. It is hoped that this research can become a basis for further research in this field.*

Keywords*:* *information security, information system, security awareness, user behavior*

Received: November 16, 2023; Received in revised form: February 26, 2024; Accepted: February 26, 2024

1. INTRODUCTION

Bananas are the most popular type of fruit, in demand by all levels of society worldwide. Additionally, they offer numerous benefits as they are rich in essential nutrients for the human body. In general, bananas stand out as one of the most renowned fruits in Indonesia. The characteristic sweet taste of bananas makes them highly favored among Indonesian citizens, despite the fruit having various types with distinct shapes and colors. This support enables the automation of diverse tasks in agriculture and plantations with maximum accuracy, making smart farming concepts a reality. An overview of computer vision, coupled with the acquisition of high-quality images using remote cameras or drones, facilitates efficient non-contact and technology-based solutions in agriculture, plantations, and forestry [1].

An intelligent fruit ranking system is imperative due to the slow, labor-intensive, error-prone, and tedious nature of grading and sorting performed by humans in fruit categorization. Deep learning has proven to be highly effective in this area, with significant advancements in accurately and efficiently categorizing fruits. This plays a crucial role in agriculture, quality control, and automated fruit sorting systems, impacting fruit quality evaluation and the export market for producers [2-4]. A broad review of fruits vand vegetables among various horticultural products in agriculture was carried out. The review covered specific models, data pre-processing, data analysis methods, and the overall value of performance accuracy using clear performance metrics [5]. Another research informs a convolutional neural network-based predictive model in identifying bruised apples based on shape information (in the form of three-dimensional [3D] surface meshes) obtained from a 3D infrared imaging system. Often there are irregularities on the surface of bruised apples, which can be used to distinguish bruised from unbruised apples [6]. The absence of an automatic system for classifying dates poses a challenge in the fruit industry, leading to the reliance on manual expertise, which is labor-intensive, costly, and prone to bias [7]. Based on the Machine Learning techniques development, it is possible to create automation in the agricultural sector, including especially fruit farming. An automatic system based on ML would effectively handle the process of classifying and sorting fruit which is often carried out by human experts [8]. This research is based on performance analysis of several deep learning models including the impact of various parameters on accuracy and efficiency in fruit categorization systems. However, Convolutional Neural Networks are used through different approaches in evaluating effectiveness [9].

This research identifies various types of bananas found in Indonesia, which is known as the producer of the most banana varieties in the world. The research is focused on detecting and classifying 9 types of bananas that are widely consumed by Indonesian people. The research aims to analyze the performance of deep learning models and the influence of various parameters on the accuracy and efficiency of fruit categorization systems. By understanding these factors, researchers can further improve the automation of fruit classification in industry.

2. RELATED STUDY

Bananas are widely known in Indonesia as a delicious fruit that can be enjoyed in various ways. This means that manual expertise is required, which involves hard work, costs, and potential bias [10]. The agricultural sector, especially fruit farming, has greatly benefited from the development of ML techniques. This progress has resulted in the emergence of automated systems that utilize ML to effectively carry out fruit classification and sorting tasks, which were previously reliant on human expertise [11]. The objective of this study is to analyse the performance of deep learning models and assess the impact of various parameters on the accuracy and efficiency of fruit categorization systems. Specifically, Convolutional Neural Networks with different approaches will be employed for evaluation to support the image recognition [12].

This application of AI in the food industry, specifically in maximizing resource utilization and reducing human error. By leveraging artificial intelligence and data science, the quality of restaurants, cafes, online food delivery aims, hotels, and food outlets can be improved through increased production using different pairing algorithms for sales prediction [13]. The important aspect of research is the training and processing part. A study found that by using a GPU as the main processing power, they achieved a 177x acceleration on training data and a 175x acceleration on test data [14]. To ensure the reliability of the dataset used, the researchers utilized the fruis-360 dataset, which had already been successfully used in previous research [15]. This dataset includes photos of 30 different fruit classes. The researchers employed prominent deep learning architectures such as VGG16 and ResNet50 to build their classification system. The models achieved 86% and 85% accuracy on the public dataset and 99% and 98% accuracy on their custom dataset [16]. Another study, a wider variety of fruit types was used. The experiment involved 24 classes of fruit, consisting of 3924 pictures. The authors pre-processed the data by applying augmentation techniques and trained a convolutional neural network (CNN) with a batch size of 16 and 100 epochs. The model achieved a test accuracy of 95.5% [17]. The Study used a similar fruit dataset and focused on various types of apples. The average accuracy values for training and test datasets were 100% and 73% respectively. There would be a difference between the normal image and the kernel image. The Kernel Process achieved the desired effects such as various blurring or sharpening effects [18].

In terms of image equalization, CLAHE (Contrast Limited Adaptive Histogram Equalization) differs from ordinary adaptive histogram equalization by using clipping histograms to equalize the image [19]. When it comes to computer vision-based technology for fruit identification and classification, one study focused on bananas. They aimed to build a computer vision-based model identified bananas images, to determine the variety of bananas, and assess the quality of bananas [20]. In the fruit detection for automatic harvesting, deep learning had been widely used due to its ability to extract high-dimensional features from fruit images. DL is employed to detect fruits in scenarios [21].

This research aims to determine the performance of detecting optimal banana image objects. Examples of banana images can be seen in Fig. 1.

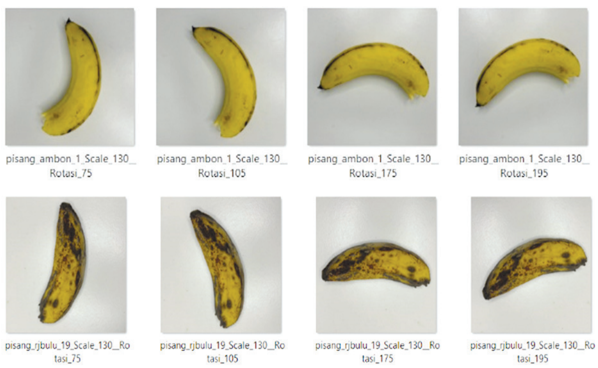

Fig. 1. The banana dataset consists of Ambon, Rjbulu and Rjsereh bananas

Fig. 1 provides examples of several images of several types of bananas that have undergone changes in rotation and scale from the original image. The rotations are 75, 105, 175 and 195 degrees and the scales are 130, 133, 137 and 140 percent of the original image. In this experiment there were 9 classes of banana datasets, each of which had different characteristics and orientation. From this dataset, 7504 images of bananas were obtained, consisting of 4506 training datasets, 1122 validation datasets and 1876 testing datasets. Thus, the system can detect 9 classes of banana objects with image conditions of different scales and rotations. An example of banana object detection can be seen in Fig. 2.

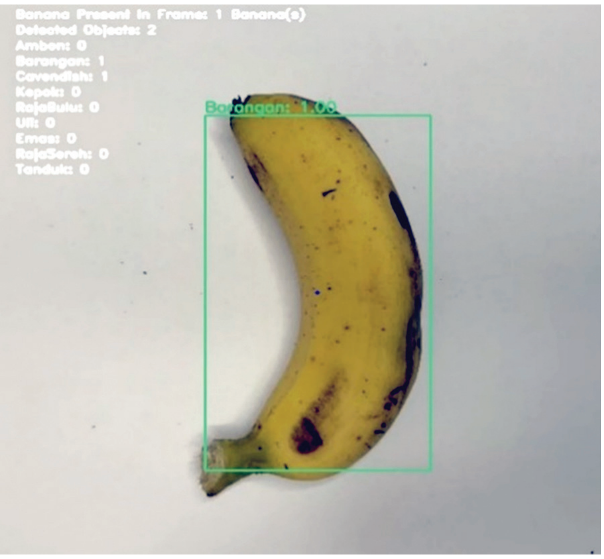

Fig. 2. Examples of Banana image object detection in Banana datasets

In Fig. 2 present object detection involves identifying different objects within an image and outlining them with a boundary to provide localization coordinates. Researchers in the field have taken an interest in object detection specifically in banana images, as aerial views from cameras mounted on drones offer stereo perspectives [22]. The purpose of this paper is to provide accurate banana image detection using deep learning-based object detection methods, particularly on Banana datasets. It aims to serve as a comprehensive resource for current advancements in deep learning-based Banana object detection in Banana datasets.

Our research focuses on identifying suitable convolutional network models for detecting banana fruit image objects. The banana dataset can be classified into nine different classes. The goal of this research is to determine the accuracy of object classification based on the characteristics depicted in the fruit dataset. We aim to determine the extent to which the average accuracy improves in recognizing detected objects. The main objectives of this research are as follows:

1. Evaluate the performance of different convolutional network models in detecting banana fruit image objects.
2. Determine the accuracy of object classification based on the characteristics depicted in the fruit dataset.
3. Assess the improvement in average accuracy in recognizing detected objects.

The detection system described the stages of banana object detection using deep learning using several CNN models. Where the dataset was collected using a digital camera positioned correctly between 30 - 40 cm from the banana object.

In conducting research on detecting banana objects using a camera positioned between 30 - 40 cm from the banana object, the data collection process for several banana classes was carried out. The objective of this process was to facilitate the detection of image objects using deep learning algorithms. Two main factors need to be considered in this detection process: the accuracy of object detection and the calculation of banana similarity across different classes. The detection results for two types of bananas using one of the CNN models can be seen in Fig. 2 and Fig. 3.

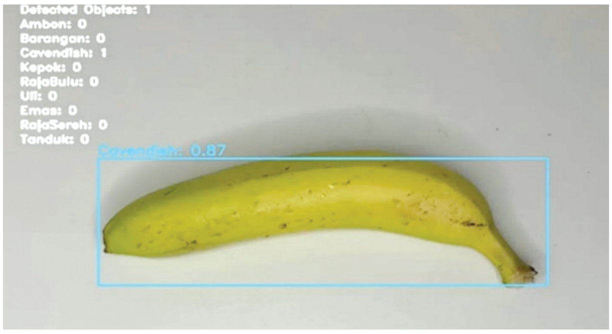

Fig. 3. The result of Banana Detection using MobilenetV2 - YoloV3-Nano

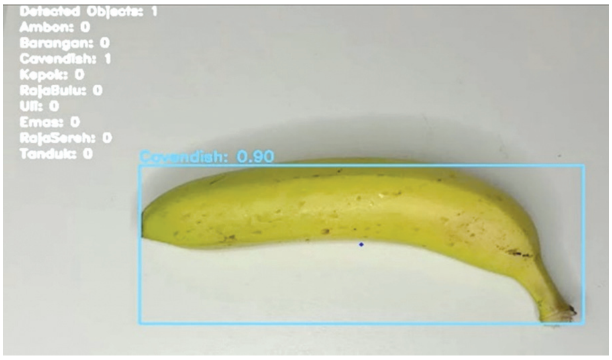

Fig. 4. The result of Banana Detection using Yolo V7

Fig. 3 and Fig. 4 illustrate the results of the accuracy in detecting Banana image class objects from processing the Banana Dataset. The class of Banana objects is determined using two different CNN models. The training process utilizes the Darknet algorithm, which characterizes the class of Banana objects detected within a bounding box. The concept of object class detection from Banana Datasets in video form is a novel approach that holds promise for further development in MAP.

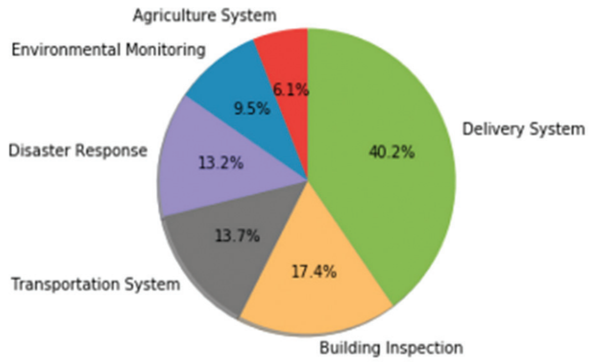

Fig. 5. Several examples of the use of object detection in various fields can be seen in Figure 4.0. [22]

Fig. 5 clearly informed the surge in research publications in recent years, driven by the advent of deep learning-based object detection. However, the accuracy achievable in detecting bananas or other fruits directly from images remains limited. While the object detection domain is vast, encompassing various object types and development stages, the focus remains on algorithms that perform well on images of objects captured directly from a predetermined distance [23]. The literature on object detection in images can be broadly categorized into two approaches: classical and modern.

3. PROPOSED METHODS

This research on detecting banana objects in several classes is divided into three main parts: the preparation stage, the training stage, and the testing stage. The preparation stage involves collecting or creating banana video datasets and analyzing banana class datasets. In the training stage, all collected banana data is processed, and the weight of each data is calculated to enable recognition in the testing process During the testing stage, the banana test data is recognized based on the training data that has been collected. For a more thorough explanation refer to the figure below:

A. Collecting Materials and Preparation

The initial process carried out in detecting the 9 classes of bananas that must be carried out is the preprocessing stage or also referred to as the preparation stage where the steps carried out can be seen in Fig. 6.

Fig. 6 provides information about the preparation stage. Description of the fields of image object detection can be broadly categorized into two approaches: classical and modern. Classical categorization includes conventional techniques consisting of vision-based approaches based on machine fruit classifiers. Modern categorization refers to deep learning-based algorithms, which is the focus in this research. Classical approaches to object detection include all major advances in the field of imaging using human-generated feature-based machine learning approaches [24]. The collected data will be used for training and testing. Therefore, video obtained from video dataset needs to be processed first to make it understandable for machines. Using FFMPEG, each banana video sample is processed to extract video screenshots at regular intervals. This process is done to prepare the images for the image labeling process. Image labeling is essential for the machine to understand and identify the objects it needs to detect. Fig. 6 shows an example of the drone video being converted into a new image, along with a text file for each image containing object data for the machine to learn from. The labeled data then undergoes a training process using pretrained convolutional weights to determine which model produces the highest accuracy.

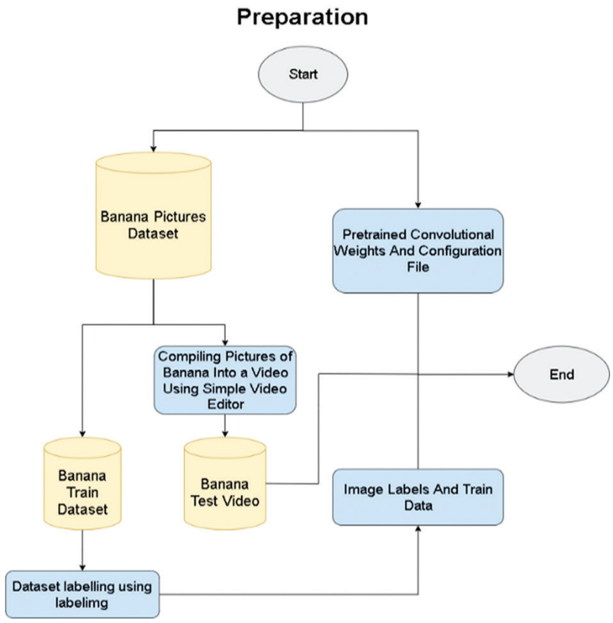

Fig. 6. The preparation stages in Banana fruit image object detection

After converting and labeling the images, another file called "trainer. Data" needs to be prepared. This file contains labeled images of banana fruits, the total number of classes, and the training output folder. Additionally, at this stage, pre-trained convolutional weights and the configuration of the CNN model are obtained from the internet. The configuration file must be adjusted based on the total number of classes. These configurations are known as training hyperparameters.

B. Training Dataset

The next stage is the training process which can be seen in Fig. 7.

Fig. 7 After completing the initial processing of the required banana image data, the next step is to begin the training process. During this training stage, an iterative process will be performed until either the iteration modulus is complete or the 1000th iteration module is completed. The objective of this stage is to train the banana image data, along with the supporting files, using the Darknet algorithm at each iteration. The outcome of the training stage will be weight files for each banana image. In general, the choice of weights is based on the object space of the target image, which heavily relies on the properties of the objects in the training set and the predicted properties. To ensure the most accurate data, the training process will be conducted using the same training set and image data set. The resulting weight file will strike a balance between being neither too ambiguous nor too clear. This object detection process was executed to support experiments involving various classes of banana fruit images. The Darknet framework is leveraged solely to assist in the training process carried out by several pretrained CNN models.

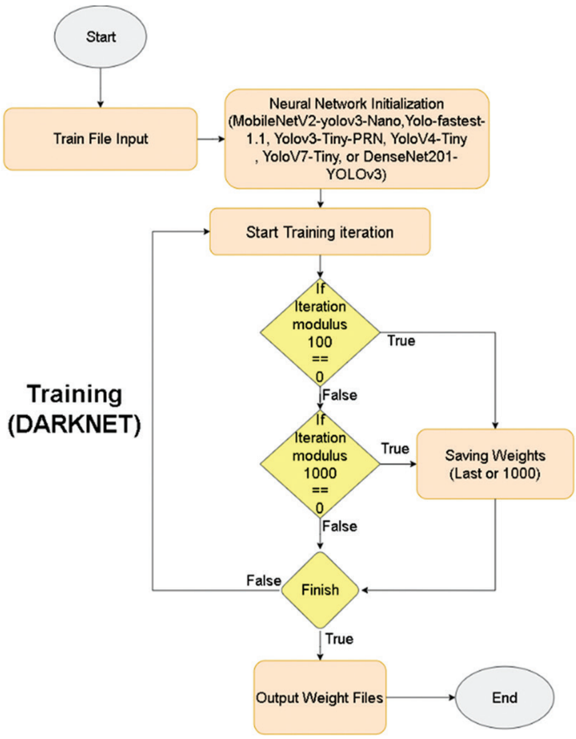

Fig. 7. The training stage in Banana fruit image object detection

C. *Testing Process*

After completing the training process, the testing process will be carried out on the banana fruit object data taken from the Banana Dataset. The goal is to detect all test banana data optimally based on the training data. In this experiment, 6 CNN models were used to determine which model provided optimal performance results for detecting banana fruit objects. The stages of the testing process in Fig. 8.

Fig. 8 explains and describes the process after the training stage is completed. The next step is to begin the testing stage. During this stage, the first step is to import the necessary files for testing, which include the trained weight files, the configuration files for training results, and the data files for the trainer. The testing process starts by extracting frames from the video dataset of banana fruit images. Then, the prediction is calculated using the non-max Suppression (NMS) function. The subsequent step involves drawing a bounding box around the banana object and determining its type/class.

This also includes calculating the prediction accuracy of the banana object class as the target object. The bounding box also includes other crucial information such as class and coordinates, which play a vital role in detecting banana class objects. The system also draws a line to indicate the direction of the object if it has moved within the frame. Finally, the banana image bounding box must respond to the accuracy percentage as per the specified binding.

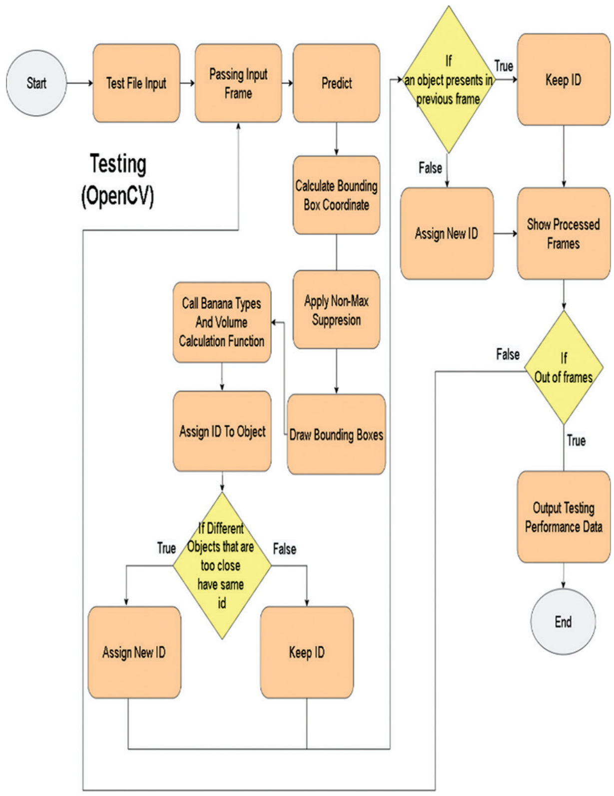

Fig. 8. The testing stage in Banana fruit image object detection

During the testing stage, the output produced is in the form of accuracy, precision, and F1 Score. These metrics are used to measure the performance of the Convolutional Neural Network (CNN) models. These metrics help determine which approach is effective for detecting objects in several classes of bananas based on the banana fruit dataset in video form. The testing process for validation data for the stages can be seen in Fig. 9.

Fig. 9 explains the validation stages using the darknet algorithm. The input data uses three parameters: the output weight, the configuration model used, and the training data. All files input with these three parameters will be processed with the validation function. The resulting accuracy of the validation banana fruit data will be predicted. The performance of each model is assessed using standard evaluation metrics, including precision (the ratio of accurately detected bananas to all detected objects), recall and overall accuracy. Additionally, computational efficiency and speed are compared to determine the trade-offs between accuracy and real-time performance.

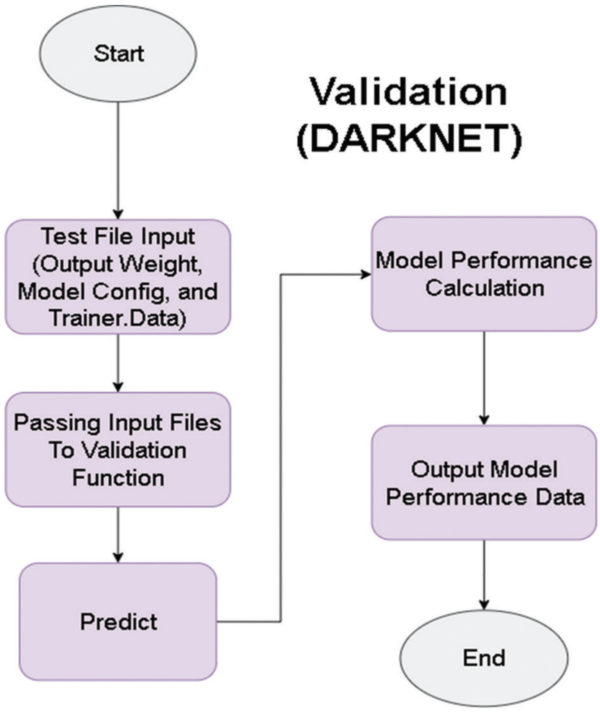

Fig. 9. The Validation stage in Banana image object detection

1) Precision and Recall

To determine the performance in detecting banana objects, the process of calculating the precision value must be supported by 2 important parameters, namely the first to get the right number of samples which are classified correctly by the model, while the second parameter must get the total samples which are classified correctly, regardless of whether the sample can be classified correctly or not. Precision calculations are between 0 and 1, with the reference being that 0 is the lowest value and 1 is the highest value. Conversely, recall is calculated by dividing the number of positive samples correctly classified by the model by the total number of positive samples. Both precision and recall formulas are illustrated below:

$$Precision = \frac{true\ positive}{true\ positi\ ve + false\ positive} \quad (1)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

Where:

1. True Positive: number of correct examples after a correct classification process.
2. False Positive: the number of negative samples from the wrong model but the correct samples.
3. False Negative: the number of positive samples but cannot be classified by the model.

2) F1-Score and mAP (mean Average Precision)

The F1 score is a measure of the harmonic mean of precision and recall. It is commonly used as an evaluation metric in binary and multi-class classification. The F1 score integrates precision and recall into a single metric to gain a better understanding of model performance [25]. The resulting F1-score value ranges from 0 to 1, where 1 represents the highest accuracy value.

$$F1\ Score = \frac{2\ (Precision\ x\ Recall)}{(Precision + Recall)} \quad (3)$$

In addition to the F1-score, which encapsulates the two-preceding metrics, the mean average precision (mAP) is a metric that represents the average value of average precision across the detection process for all previously identified classes. An object detection performance accuracy test was conducted on the Banana image dataset. The mean average precision (mAP) was employed as the performance metric [26], with the formula as follows:

$$\mathrm{MAP} = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \quad (4)$$

In the formula, the variable Q is the total query in a data set, and the parameter AveP(q) describes the average precision (AP) value in a particular query process for parameter q. Next, for each query, q, the corresponding AP is calculated, then averaging all the scores from the AP parameters to produce a number called MAP, which is a measure of the effectiveness of the model's query processing capabilities. This performance metric can essentially be one of the metrics used to determine which model shows the best overall performance.

4. EXPERIMENTAL RESULTS

A. *Training Results*

In the training process on the banana dataset, PC technology is used which is equipped with the latest Central Processing Unit and Graphic Processing Unit specifications or as needed for banana image processing. An approach with the right PC technology will provide advantages in speeding up the duration of image training with CNN models supported by the Darknet framework in making GPU acceleration during the training phase process. The Specification of Hardware needed in the experiment in Table 1.

Table 1. Hardware Specification on the Experiment

No.	Description	Specification
1	CPU	Intel R7 RTX 4060 B760
2	RAM	32 GB DDR4
3	VGA	NVIDIA RTX 3 060
4	VRAM	6 GB

Precise and larger model architectures result in slower training times. The size of the CNN model also reaches the size of the model output.

All tested models produced good average loss results, which can be visualized in the form of a CNN model training graph, as shown in Figure below.

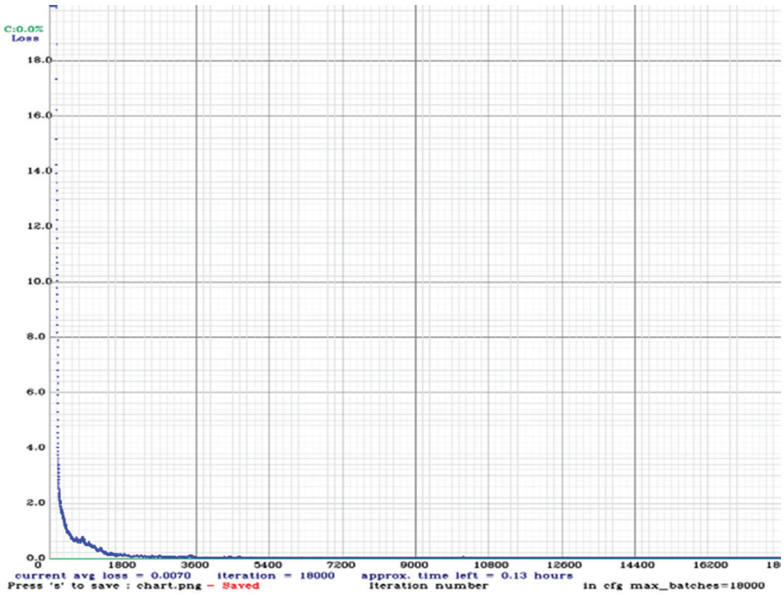

Fig. 10. Banana image Training using DenseNet201-YOLOv3

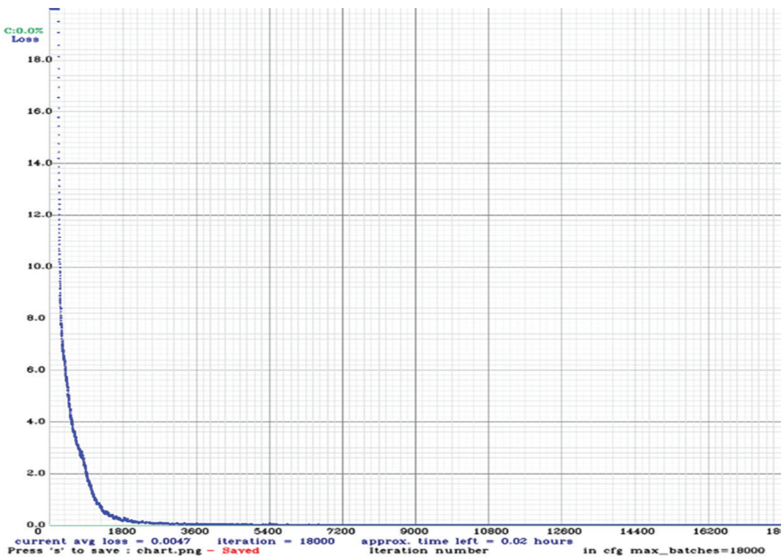

Fig. 11. Banana image Training using MobilenetV2-YoloV3-Nano

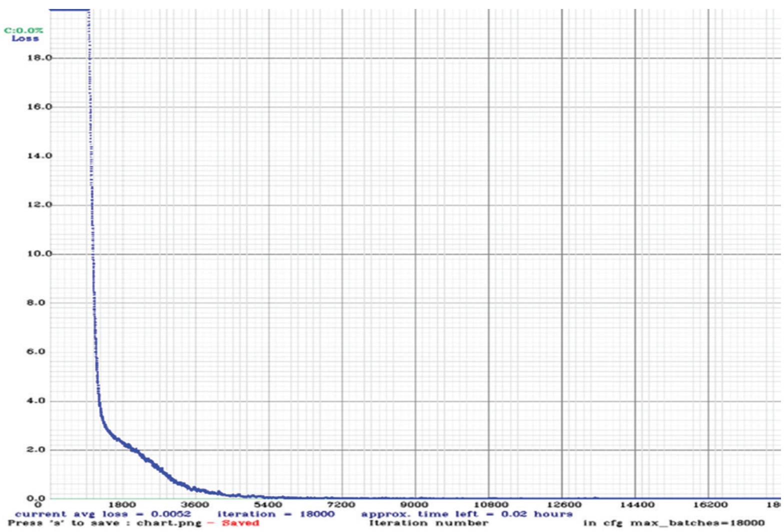

Fig. 12. Banana image Training using Yolo-Fastest 1.1

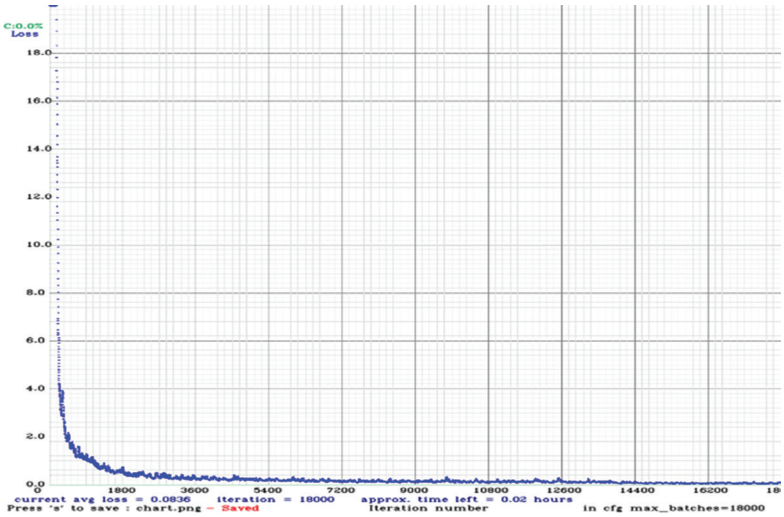

Fig. 13. Banana image Training using YOLOv3-tiny-prn

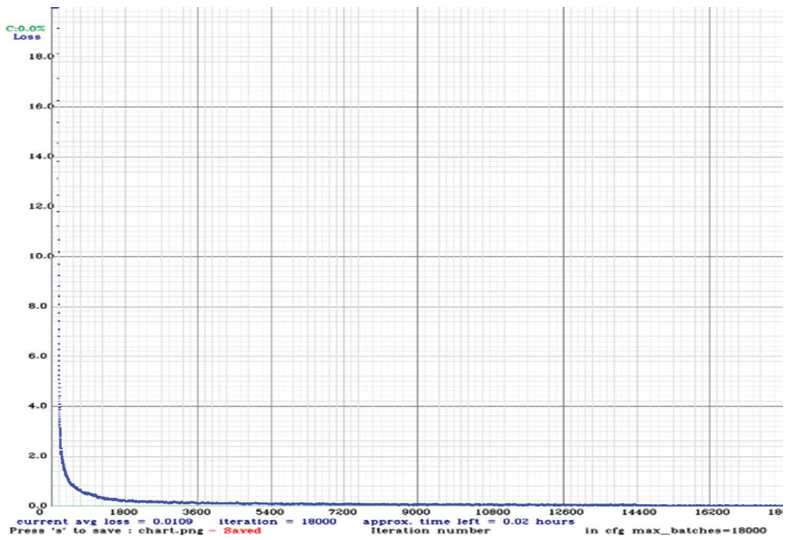

Fig. 14. Banana image Training using Yolov4-tiny

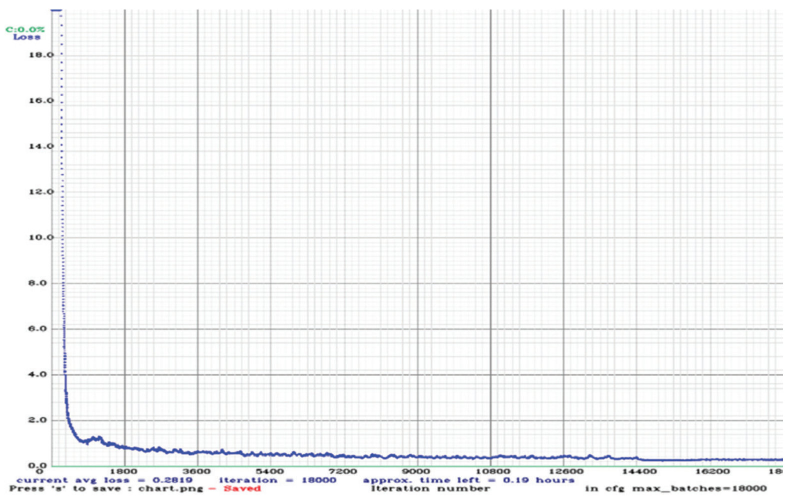

Fig. 15. Banana image Training using Yolov7

Figs. 10, 11, 12, 13, 14, 15 depicts the curve graph of the training stage results for the banana image, which can vary depending on the CNN model. In contrast, YoloV4-tiny and YOLOv7 exhibit faster degradation than the other models. However, the training process is not affected if the loss drops to a certain level. After a sharp initial decline, the losses stabilize into a gentle curve. This indicates that the model is beginning to understand the dataset and is adapting to it gradually. The graph illustrates that the loss remains stable until the end of the iteration, which is 18,000 iterations for all models. This result implies that the trained model has successfully learned the specific banana image dataset without issues. In training result in Table 2.

Table 2. Training result of detection Banana Image using CNN model or Yolo Methods

Model	Average Loss(%)	Approx time
YOLOv3-tiny-prn	0.0836	0.02 hours
YOLO-fastest 1.1	0.0052	0.02 hours
YOLOv7	0.2819	0.19 hours
DenseNet201-YOLOv3	0.0070	0.13 hours
MobilenetV2-YoloV3-Nano	0.0047	0.02 hours
Yolov4-tiny	0.0109	0.02 hours

In Table 2, the average loss of the CNN model in determining performance in detecting bananas is provided. Table 2 shows that all models have an average loss below 0.1, which is considered quite low and acceptable for experiments in detecting fruit images. The MobilenetV2-YoloV3-Nano model has the lowest average loss of 0.0047, followed by YOLO-fastest 1.1 with 0.0052, and DenseNet201-YOLOv3 with 0.0070. When detecting banana objects, the CNN YOLOv3-tiny-prn model has the second highest average loss of 0.0836 with an estimated training time of 5.1 hours. The final model tested, YOLOv7, had the highest average loss of 0.2819 with an estimated training time of 5.2 hours. However, the difference in training time between the two models is only 0.1 hour. A solely based on the average loss value. An overview of the training configuration can be seen in Table 3.

Table 3. Configuration information parameter's

CNN Model Name	Batch	Subdivision	MaxBatch	Classes
Densent201-YoloV3	28	24	18000	9
MobilenetV2-YoloV3-nano	32	24	18000	9
Yolo-Fastest 1.1	32	24	18000	9
Yolo3-tiny-pm	32	24	18000	9
Yolov4-tiny	32	24	18000	9
Yolov7	32	24	18000	9

In Table 3 describes the configuration information from the processing results in detecting banana images. In general, this process describes the batch starting from 32 to a maximum of 18,000 to recognize 9 classes of banana images in each CNN model.

B. Simulation and Results

Before testing the weights trained on the system, the banana object needs to import the necessary supporting files. This file includes training labels for banana images, image positions, model configurations, and a .data file called trainer. Data. The trainer. data file is necessary to run the banana data testing process using the OpenCV library for inference. Once the supporting files are imported, the self-service of accuracy and banana image class detection will automatically be initiated. Then, an experiment of banana image object detection ensures the processed the trained object detection works properly. An overview detect banana fruit class objects in the banana dataset can be seen in Fig. 16.

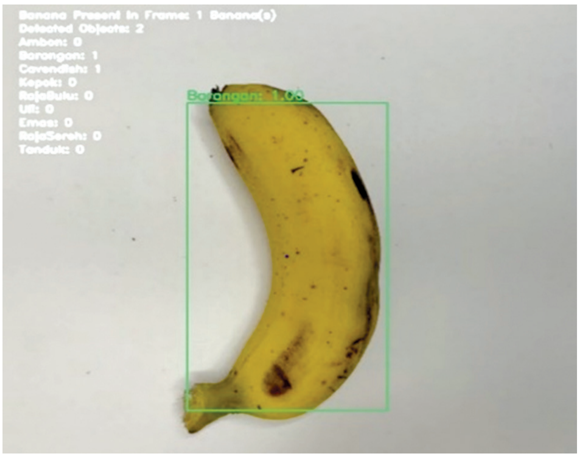

Fig.16. Banana fruit image class object detection simulation on Banana fruit dataset

Fig. 16 showcases the system's capability to accurately detect multiple classes of bananas in banana videos, while maintaining acceptable performance and excellent response times. However, achieving real-time banana object detection requires the implementation of appropriate methods and concepts to ensure seamless processing between frames and accurate interactions within the boundary box.

This approach reduces the processing load, which can potentially increase processing time. The results of the bounding box correctness calculation are shown in Fig. 17.

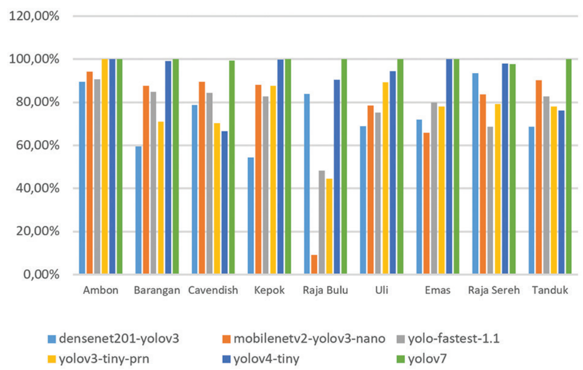

Fig.17. Banana image of average precesion after optimization CNN Model Comparison

Fig.17 shows the average precision results for banana image detection for 9 classes using 6 CNN models. Based on the experiments carried out, the Yolov 7 model was able to detect all banana classes very well with an average precision of 98%, followed by Yolo v4 - tiny that delivers an average accuracy of 89%. Based on experiment this model only the banana Tanduk class has a detection precision of less than 78%. Another model whose precision is quite good is Pisang Rjbulu 43% only. the Yolo v3-tiny-pm whose precision reaches 70.5%, while there is 1 class of bananas whose precision is low namely Rjbulu.

The results of this experiment can be used as a guide for further research. The mean of Procession (mAP) and time processing for each Banana object is shown in Fig. 18.

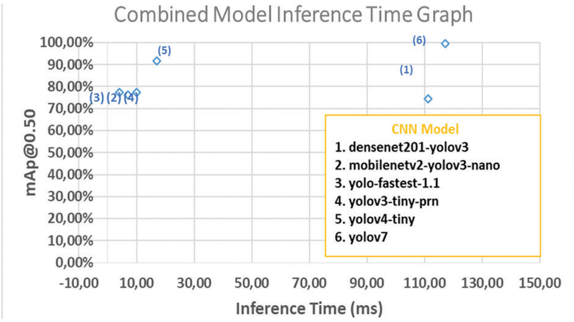

Fig. 18. Combined inference time model graph before and after optimization

Fig. 18 illustrates the inference time of 6 CNN models in relation to their mAP@0.50 percentage or their accuracy in detecting the banana class. Inference time measures the processing time between capturing an image and obtaining the detection results in the form of predictions. Based on this figure, the CNN model exhibits an additional 1% accuracy with approximately half the original inference time. In Banana object detection informed the average inference time significantly higher than the average accuracy of the images predicted by the system. For example, the YOLOv7 model achieves 100% accuracy, but its inference time is less than 50%.

Instead, the model's inference time is over 100%, reaching 110 ms. Other models, such as YOLOv4-Tiny, demonstrate results above 90%. The YOLOv4-Tiny model achieves an accuracy of 92.2%, while its inference time exceeds 50%, reaching 14.2 ms. The YOLO-fastest 1.1 model exhibits an average accuracy of 80.3%, but its inference time is also above 50%, reaching 8.2 ms. MobileNetV2-YOLOv3-Nano and YOLOv3-Tiny have inference times of 9-10 ms and average accuracies of 78% and 80%, respectively. DenseNet201-YOLOv3 possesses the highest inference time at 90.8 ms but also has the lowest mAP@0.50 percentage at 72.3%. Conversely, YOLO-fastest 1.1 exhibits a low inference time (8.2 ms) while maintaining good compatibility (80.3% mAP@0.50). The results are shown in Fig. 19.

Fig. 19 presents an experiment conducted using a Banana dataset to detect nine object classes. The experiment reveals that the CNN Yolo4-Tiny model achieves the highest accuracy, with precision, recall, and F1 Score reaching 84%. The YOLOv7 model follows closely behind with an average accuracy of 80%. On the other hand, the YOLOv3-Tiny-prn and Yolov4-Tiny models exhibit the lowest accuracy, with an average value of 53-54%. The minimum average accuracy exceeds 50%, indicating that this approach has performed well and can detect objects almost perfectly. To illustrate the combined average Precision for every CNN model can be in Fig. 20.

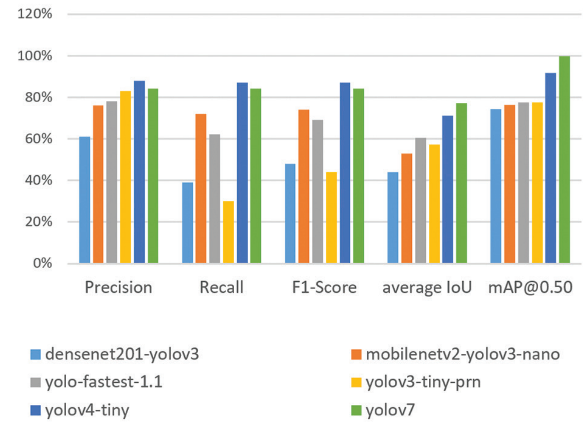

Fig. 19. The information of Precisions, recall, F1-score, average IoU and mAP@0.50 of each model CNN

Fig. 20. Banana Image object detection performance using confusion matric

Fig. 20 presents the Yolov4-tiny can recognize more banana images than other models, specifically 413 images, which have true positive. However, there are 61 Banana images that are falsely identified as negative. The Yolov7 model recognizes 396 Banana images as true positive, but there are 78 images that are falsely identified as negative, which is more than the previous model. Another model that recognizes a significant number of banana images is MobileNetV2-Yolov3-Nano, with 342 images identified correctly. However, there are 132 images that cannot be recognized or are falsely identified as negative. On the other hand, the Yolo-fastest-1.1 model can recognize 295 banana images, but it fails to identify 179 images. this figure can effectively recognize banana images, with a total of 400 images recognized accurately.

In this figure, it can be concluded that only two models, yolov4-tiny and yolov7, effectively recognize banana images. In the next following tables are the accuracy results for detecting each class of banana using each CNN model.

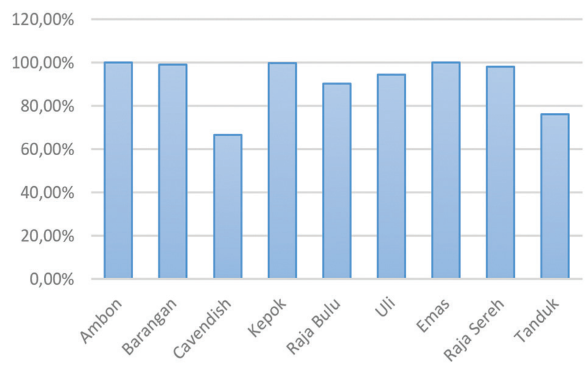

Fig. 21. Accuracy results in detecting banana images using the Yolov4- tiny model

Fig. 21 presents comparison of accuracy results for each Banana Class Detected Using the YOLOv4-Tiny Model Based on the experimental results obtained using the YOLOv4-tiny model, the highest accuracy was achieved for the Ambon, Barangan, Kepok, and Emas banana classes, with 100% accuracy. The Rjsereh banana class followed with an accuracy of 98%, while Uli and Rjbulu achieved accuracies of 96% and 94%, respectively. Additionally, the model demonstrates good performance in recognizing other banana classes.

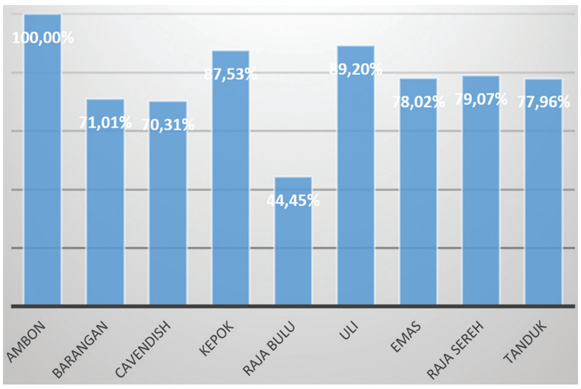

Fig. 22. Accuracy results in detecting banana images using the Yolov3- tiny-prn model

Fig. 22 shows a comparison of the accuracy results for each class of banana objects detected using the yolov3-tiny-prn model. Based on the results of experiments using this model, the highest accuracy results were found in the Ambon banana classes only, namely 100%. But the others have accuracy lest than 80% such as Uli, Emas, Rjsereh and Tanduk Banana Class. All the class banana has accuracy used this model less than 80% but more than 70%. However, based on the experiment this model has a good enough to detect and recognize the 9 class of Banana.

Fig. 23 presents comparison of the accuracy results for each class of banana objects detected using the YOLO-fastest-1.1model. Based on the results of experiments using this model, the highest accuracy results were found in the Ambon, Barangan, Cavendish Kepok, Emas, Tanduk classes is more than 80%. followed by the RjSereh banana class with an accuracy of less than 70% and Rjbulu less than 50%. . Based on this image, the accuracy produced by this model can detect several classes of bananas perfectly. However, for other banana classes, it is also good at recognizing the detected banana class.

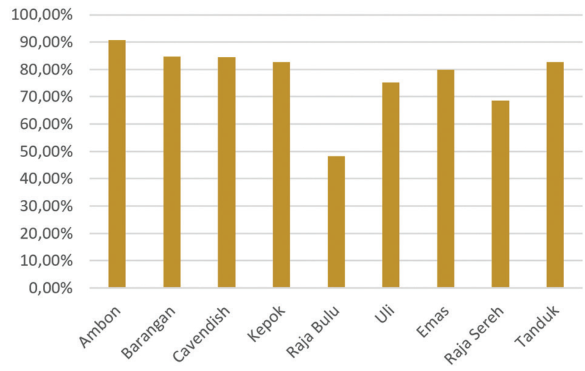

Fig. 23. Accuracy results in detecting banana images using the Yolov4- fastest – 1.1 model

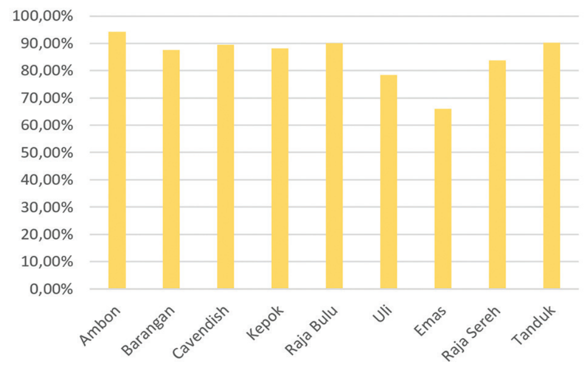

Fig. 24. Accuracy results in detecting banana images using the mobilenetV2-yolov3-nano

Fig. 24 presents comparison of the accuracy results for each class of banana objects detected using the yolov3-nano model. Based on the results of experiments using this model, the highest accuracy results were found in the Ambon, Cavendish, Rjbulu and Tanduk banana classes, namely 90% or more, followed by the Barangan, Kepok banana class with an accuracy of less than 90%. But Uli and Emas have accuracy at 78% and 63%. Based on this experiment, the accuracy produced lower than 90%. However, from other banana classes, it is also good at recognizing the detected banana class.

Fig. 25 presents a comparison of the accuracy results for each class of banana objects detected using the YOLO-Fastest-1.1 model. Based on the experimental results, the Ambon class achieved the highest accuracy of 90%, followed by the Barangan, Cavendish, Kepok, and Tanduk classes with accuracies above 80%. However, the Uli and Rjsereh classes had accuracies below 80%, at 73% and 68%, respectively. The Rjbulu class had the lowest accuracy of 48%. These results suggest that the YOLO-Fastest-1.1 model can effectively detect several banana classes.

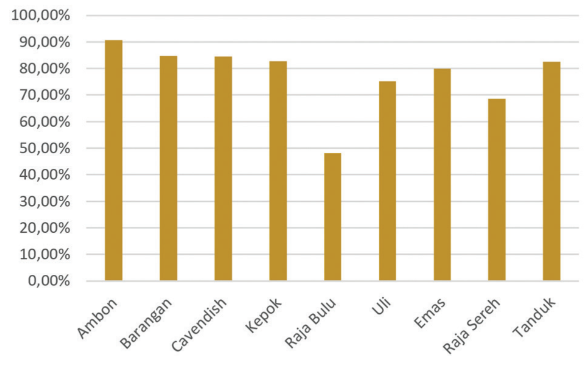

Fig. 25. Accuracy results in detecting banana images using the Yolo-fastest- 1.1

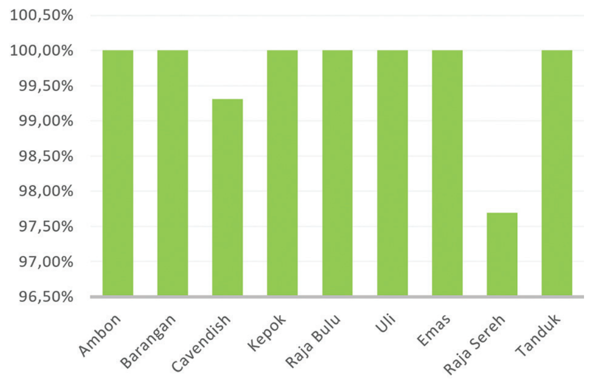

Fig. 26. Accuracy results in detecting banana images using the Yolov7

Fig. 26 presents a comparison of the accuracy results for each class of banana objects detected using the yolov4-tiny model. Based on the results of experiments using this model, the highest accuracy results were found in the Ambon, Barangan, Kepok, and Emas banana classes, namely 100%, followed by the Rjsereh banana class with an accuracy of 98%, Pisang Uli at 96%, and Raja Uli at 94%. Based on this figure can be seen that the accuracy produced by this model can detect several classes of bananas perfectly.

5. EVALUATION

In the experiment conducted to detect banana fruit classes, six CNN models were used. These models had been tested in object detection technology but had not been previously tested on the banana fruit dataset. While most of the CNN models used can detect the Ambon banana class with an average accuracy of over 90%, the average accuracy for Kepok, Barangan, and Emas bananas reaches over 80%. The experiments revealed significant variations in the number of detections for each banana class across the different CNN models. One of the challenges encountered is the issue of camera movement and the size of the objects being detected. These factors make it challenging for the system to accurately track the targeted objects. Despite these challenges, the number of objects detected can serve as an indicator of the model's performance.

6. CONCLUSION

1. Based on the experimental results on the banana dataset, there are 2 CNN models that have optimal accuracy. The YOLOv4-Tiny model produces an mAP@50 performance percentage of 92%, an average precision of 88%, while recall and F1 Score reach 86% and 84%. Meanwhile, the YOLOv7 model produces an mAP@50 performance percentage of 10%, while precision, recall and F1 Score produce the same performance of 84%. However, the two models have different Intersection over Union (IoU) accuracy, where YOLOv7 is greater at 77% compared to YOLOv4-Tiny at 71%.
2. Furthermore, the YOLOv4 Tiny model has the best potential when using the Banana fruit dataset because of its low inference time, only 13 ms, and a high mAP@0.50 accuracy percentage, namely 90.2%. The YOLOv7 model has an inference time of 96 ms and mAP@0.50 accuracy percentage reaching 100%.
3. Based on the low inference time of 12 MS and high accuracy reaching 90%, YOLOv4-Tiny has the potential to be the best option to be applied in the case of real-time banana object detection using video streams of various bananas on processor devices with power limitations. Another option is to use YOLOv7 because of its high accuracy capability reaching 100%, but the inference time is also very high, reaching 110 ms. However, due to its high inference time, it is only recommended when using high-power devices.
4. In further research recommended to using more complex and clearer datasets and better object tracking systems. These improvements have the potential to enhance the performance of real-time object detection systems.

7. REFERENCES

[1] S. Elayabalan, S. Subramaniam, V. Shobana, K. A. Kumar, "An overview on phytochemical Composition of Banana (Musa spp.)", Indian Journal of Natural Sciences, Vol. 7, No. 42, 2017, pp. 12408-12419.

[2] A. Koirala, K. B. Walsh, Z. Wang, C. McCarthy, "Deep learning–Method overview and review of use for fruit detection and yield estimation", Computers and Electronics in Agriculture, Vol. 162, 2019, pp. 219-234.

[3] S. Anatya, V. C. Mawardi, J. Hendryli, "Fruit maturity classification using convolutional neural networks method", IOP Conference Series: Materials Science and Engineering, Vol. 1007, No. 1, 2020, p. 012149.

[4] F. M. A. Mazen, A. A. Nashat, "Ripeness classification of bananas using an artificial neural network", Arabian Journal for Science and Engineering, Vol. 44, No. 8, 2019, pp. 6901- 6910.

[5] R. A. D. Putri, A. N. F. Rahman, A. Syarifuddin, "Proximate composition and sensory evaluation of Indonesia traditional food", AIP Conference Proceedings, Vol. 2596, No. 1, 2023.

[6] J. Wäldchen, M. Rzanny, M. Seeland, P. Mäder, "Automated plant species identification—Trends and future directions", PLoS Computational Biology, Vol. 14, No. 4, 2018, p. e1005993.

[7] K. Albarrak, Y. Gulzar, Y. Hamid, A. Mehmood, A. B. Soomro, "A deep learning-based model for date fruit classification", Sustainability, Vol. 14, No. 10, 2022, p. 6339.

[8] L. Zhou, C. Zhang, F. Liu, Z. Qiu, Y. He, "Application of deep learning in food: a review", Comprehensive Reviews in Food Science and Food Safety, Vol. 18, No. 6, 2019, pp. 1793-1811.

[9] A. Gupta, R. Kushwaha, K. Patidar, "A Survey: Fruit Classification Using Machine Learning", International Journal of Engineering Technology and Applied Science, Vol. 5, No. 7, 2019.

[10] S. Naik, B. Patel, "Machine vision-based fruit classification and grading a review", International Journal of Computer Applications, Vol. 170, No. 9, 2017, pp. 22-34.

[11] A. Bhargava, A. Bansal, "Fruits and vegetables quality evaluation using computer vision: A review", Journal of King Saud University-Computer and Information Sciences, Vol. 33, No. 3, 2021, pp. 243-257.

[12] B. Dhiman, Y. Kumar, M. Kumar, "Fruit quality evaluation using machine learning techniques: review, motivation and future perspectives", Multimedia Tools and Applications, Vol. 81, 2022, pp. 16255-16277.

[13] J. Ye, Z. Shen, P. Behrani, F. Ding, Q. Shi, "Detecting USM image sharpening by using CNN", Signal Processing: Image Communication, Vol. 68, 2018, pp. 258-264.

[14] A. H. Rangkuti, A. Harjoko, A. E. Putra, "Improvement of Accuracy in Batik Image Retrieval Due to Scale and Rotation Changes Using M2ECS-LBP Algorithm", Journal of Theoretical and Applied Information Technology, Vol. 97, No. 14, 2019, pp. 97-14.

[15] A. H. Rangkuti, V. H. Athala, N. F. Luthfi, S. V. Aditama, J. M. Kerta, "Reliable of traditional cloth pattern Retrieval Using Convolutional Neural Network", Proceedings of the 2nd International Conference on Artificial Intelligence and Data Sciences, 2021, pp. 1-6.

[16] Y. Hamid, S. Wani, A. B. Soomro, A. A. Alwan, Y. Gulzar, "Smart Seed Classification System based on MobileNetV2 Architecture", Proceedings of the 2nd International Conference on Computing and Information Technology, Tabuk, Saudi Arabia, 25-27 January 2022, pp. 217-222.

[17] H. Liu, R. Wang, S. Shan, X. Chen, "Deep supervised hashing for fast image retrieval", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27-30 June 2016, pp. 2064-2072.

[18] D. C. Lepcha, B. Goyal, A. Dogra, S. H. Wang, J. S. Chohan, "Medical image enhancement strategy based on morphologically processing of residuals using a special kernel", Expert Systems, 2022, p. e13207.

[19] A. Bilal, G. Sun, S. Mazhar, "Finger-vein recognition using a novel enhancement method with convolutional neural network", Journal of the Chinese Institute of Engineers, Vol. 44, No. 5, 2020, pp. 407-417.

[20] N. Darapaneni, A. Tanndalam, M. Gupta, N. Taneja, P. Purushothaman, S. Eswar, T. Arichandrapandian, "Banana Sub-Family Classification and Quality Prediction using Computer Vision", arXiv:2204.02581, 2020.

[21] F. Xiao, H. Wang, Y. Xu, R. Zhang, "Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review", Agronomy, Vol. 13, No. 6, 2023, p. 1625.

[22] P. Mittal, R. Singh, A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey", Image and Vision computing, Vol. 104, 2020, p. 104046.

[23] Y. Amit, P. Felzenszwalb, R. Girshick, "Object detection", Computer Vision: A Reference Guide, Springer International Publishing, 2021, pp. 875-883.

[24] Z. Li, W. Shi, P. Lu, L. Yan, Q. Wang, Z. Miao, "Landslide mapping from aerial photographs using change detection-based Markov random field", Remote Sensing of Environment, Vol. 187, 2016, p. 76-90.

[25] Esri, "How the Compute Accuracy For Object Detection tool works," ArcGIS Pro, https://pro.arcgis.com/en/pro-app/latest/tool-reference/image-analyst/how-compute-accuracy-for-object-detection-works.htm (accessed: 2023)

[26] A. F. Gad, "Accuracy, Precision, and Recall in Deep Learning", https://blog.paperspace.com/deep-learning-metrics-precision-recall-accuracy/ (accessed: 2023)