Journal of Theoretical and Applied Information Technology

30th November 2023. Vol.101. No 22
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE SUPERIORITY OF YOLOV4 MODEL FOR ENHANCING PALM OIL FRUIT DETECTION

VINCENT FANDITAMA WIJAYA[1], PATRICIA CITRANEGARA KUSUMA[2], BENFANO SOEWITO[3]

[1,2,3] Computer Science Department, Master of Computer Science, BINUS Graduate Program, Jakarta, Indonesia

E-mail: [1]vincent.wijaya007@binus.ac.id, [2]patricia.kusuma@binus.ac.id, [3]bsoewito@binus.edu

ABSTRACT

This paper addresses challenges by investigating the integration of color and texture information to enhance object detection. We conducted a comparative analysis of various YOLOv4 models, including YOLOv4, YOLOv4 Tiny, YOLOv4 Tiny_3l, and YOLOv4_csp. Our study primarily employs the YOLOv4 model for image detection and focuses on experiments using the oil palm fruit bunch dataset. This dataset is driven by the potential to leverage texture and color analysis to assess the maturity level of oil palm fruits. Our research objectives center on evaluating how color and texture impact object recognition and exploring the capabilities of YOLOv4 models in this regard. The dataset consists of 4156 images categorized into 6 classes: Overripe, Ripe, Raw, Underripe, Abnormal, and Empty. Our experimental findings reveal that the YOLOv4 model excels in accurately identifying the color and texture attributes of oil palm fruit bunches. Notable performance metrics include an average IoU of 89.95%, mAP at 50 of 99.85%, recall of 0.996, F1-scores of 0.987, and precision of 0.979. These results underscore the superior capabilities of the YOLOv4 model in object recognition within this context. The significance of our results lies in advancing our understanding of how the integration of color and texture information can augment object detection. Furthermore, our findings highlight the efficacy of YOLOv4 models in this specific application, emphasizing their potential for broader applications in the field of object detection.

Keywords: *Object detection, YOLOv4, Color, Texture, Palm Fruit*

1. INTRODUCTION

Object detection technology has revolutionized the automation of tasks that were once performed manually. By using object detection algorithms, workflows are streamlined, and the need for human intervention is significantly reduced. This technology finds application in a wide range of fields, including autonomous driving [1], robotics [2], surveillance [3], and medical imaging [4]. One of the primary objectives of object detection is to enable automation. This level of automation not only improves operational efficiency but also enhances safety and reduces the occurrence of human error, particularly in critical situations.

However, these object detection approaches rely on manual feature engineering and encounter challenges when processing complex images with occlusions, scale changes, location variations, and varying lighting conditions. There are various traditional techniques for object detection recognition, among which are Support Vector Machines (SVM) and Random Forests [5]. In contrast, Convolutional Neural Networks (CNNs) serve as the supporting element for object detection, enabling them to learn discriminative features directly from the input and revolutionizing the field of object identification. By analyzing extensive amounts of labeled data, CNNs can automatically classify and locate objects in photos and videos, resulting in highly accurate and scalable object identification systems capable of handling diverse visual scenarios [6].

The latest advancements in deep learning techniques have significantly enhanced object detection e.g., SSD, R-CNN, and YOLO, making it a popular and extensively studied area in the field of Computer Vision [7]. Researchers have proposed various object detection algorithms over the years, but YOLOv4 has emerged as one of the most accurate and efficient systems with faster interference time compared to SSD. Another

advantage of YOLOv4 is in high accuracy in detecting small objects. However, the question remains: Can we further improve the performance of YOLOv4 object detection based on color and texture factors?

Performing successful identification and localization tasks heavily relies on acquiring relevant information from the objects under consideration. In this regard, color and texture emerge as vital and valuable features that significantly contribute to object detection and enable the characterization of various object attributes [8]. Texture, defined as patterns on an object's surface or the arrangement of pixels within an object or image area, plays a crucial role in providing essential information related to an object's smoothness, granularity, and hardness [9]. Analyzing and differentiating textures entails the extraction of distinct features such as patterns, angles, corners, shapes, and edges. Color, another critical feature, plays a pivotal role in object detection, aiding in the differentiation of objects based on features like brightness and saturation. By leveraging both texture and color information, objects sharing similar shapes but exhibiting diverse textures and colors can be accurately distinguished [10]. In the context of this research paper, the authors examine the efficacy of incorporating color and texture attributes into the YOLOv4 object identification model. Specifically, the study focuses on investigating how the integration of color and texture information can enhance the precision and accuracy of object detection when employing the YOLOv4 model. YOLOv4 deep learning model builds on the successes of its predecessors, introducing key advancements that elevate its significance [11]. YOLOv4 employs a one-stage detection, enabling it to predict bounding boxes and class probabilities directly in a single pass, outperforming traditional two-stage detectors in speed and efficiency. YOLOv4 utilizes a powerful backbone network CSPDarknet53 [12], extracts rich informative features from input images, complemented by the contemporary Mish activation function, which imparts soft, non-monotonic properties to neural activations. The architecture of the neck uses Spatial Pyramid Pooling (SPP) and Path aggregation network (PAN), also YOLOv3 as a head. YOLOv4 can effectively handle objects of varying sizes within an image, enhancing its versatility. Improved post-processing techniques, such as Non-Maximum Suppression and anchor box clustering, refine its predictions. Remarkably, YOLOv4 balances deep architecture with speed.

Our research conducted on object detection tasks using the YOLOv4 model revealed significant improvements when incorporating color and texture information. The findings indicated that the model exhibited enhanced accuracy, recall, and overall performance, as evidenced by higher F1 scores [13]. Importantly, the inclusion of color and texture features did not adversely affect the processing speed of the model, demonstrating that the improved performance was achieved without compromising efficiency. This highlights the potential benefits of integrating color and texture attributes in the YOLOv4 model for more accurate and reliable object detection. The integration of color and texture information into object detection especially in YOLOv4 model can provide benefits, including:

1. Increased Accuracy: By incorporating color and texture information, object detection models can make more informed decisions when classifying objects. This can lead to higher accuracy in identifying and categorizing objects within an image.
2. Better Handling of Complex Scenes: In complex scenes where objects may be partially occluded, having color and texture information can help the model better piece together the presence of an object, even if only a portion of it is visible.
3. Enhanced Robustness: Texture and color features can make object detection models more robust to variations in lighting conditions, backgrounds, and object appearances. This can lead to better performance in real-world scenarios.
4. Enhanced Contextual Understanding: Texture and color provide context and detail about the appearance of objects. This additional information can aid in understanding the object's characteristics and surroundings, leading to more precise recognition.

However, utilizing color and texture features for object detection can pose certain challenges. Extracting texture and color features from an image itself can be complex due to the various methods available, each with its advantages and disadvantages. To determine the optimal method for a specific object identification task, it may be necessary to experiment with multiple methodologies and combinations of features. Additionally, lighting conditions can pose challenges as they can significantly impact the appearance of texture and color in an image. For

instance, changes in lighting can result in glare, overexposure, color distortion, or shadows, which can introduce false textural details or color variations. Overcoming these challenges often involves employing image processing techniques to compensate for variations in illumination or capturing images under controlled lighting conditions. Despite these difficulties, when used carefully and appropriately, the detection of texture and color can be effective in locating and identifying objects in images. Furthermore, with advancements in machine learning and computer vision, these strategies are becoming easier to implement and more effective for a wide range of applications.

To gain a comprehensive understanding of the YOLOv4 model, our research paper aims to compare and analyze multiple iterations of the model. Specifically, the performance of YOLOv4, YOLOv4 tiny, YOLOv4 tiny_3l, and YOLOv4_csp will be examined. These models have undergone rigorous testing to evaluate their object detection capabilities, employing various criteria for performance assessment. The paper will present the results of these studies, with a particular focus on examining the influence of color and texture on object recognition. To ensure a thorough analysis, multiple datasets with varying levels of color and texture information were carefully selected for experimentation. By investigating the impact of color and texture on object recognition across these different YOLOv4 model iterations, our work contributes to an enhanced understanding of their capabilities and potential applications.

In this research paper, the authors conducted a series of comprehensive tests using diverse datasets to evaluate our hypothesis. We aimed to examine the impact of integrating color and texture information into YOLOv4 models, comparing the results against the YOLOv4 reference data. The outcomes of our experiments demonstrate a significant improvement in detection accuracy with the inclusion of color and texture information in YOLOv4 models, particularly in challenging scenarios characterized by intricate backgrounds and varying lighting conditions. These findings underscore the substantial benefits of incorporating color and texture attributes for enhancing object detection capabilities within YOLOv4 models.

In this paper, our contributions are:

1. Experiment with a sample of palm oil bunch dataset
2. Setting up the dataset preprocessing to eliminate color interference.
3. Configure the hyperparameter of the YOLO model.
4. Classify objects with similar colors and textures.
5. Compares the evaluation metrics with some pre-trained YOLOv4 model.

This article focuses on evaluating the performance of the YOLOv4 algorithm specifically for object detection in a sample dataset of palm oil bunches. To ensure improved accuracy in categorizing the objects, a dataset preprocessing step was implemented to eliminate any potential color interference. The YOLOv4 algorithm was then fine-tuned by adjusting its hyperparameters to optimize its performance on the given dataset. Notably, this model successfully tackled the challenging task of accurately classifying objects with similar color and texture, a feat that posed difficulties for previous computer vision systems. Comparing the evaluation metrics of our YOLOv4 model with those of other pre-trained YOLOv4 models, we observed that our model outperformed the others across various metrics. Our findings highlight the exceptional performance of the YOLOv4 algorithm for object classification tasks, particularly when dealing with complex and real-world datasets such as those encountered in the agriculture sector.

2. RELATED WORKS

The key issue in computer vision is object detection, which includes locating and classifying things in an image or video. Due to their capacity to extract distinguishing characteristics from data, convolutional neural networks (CNNs) have become an effective tool for object detection [14]. One of the most widely used strategies for object detection in real-time applications is the You Only Look Once (YOLO) method. A grid of cells is created from a picture via YOLO, and each cell's bounding bounds and class probabilities are predicted. When compared to conventional object identification techniques, this method has demonstrated exceptional performance in both accuracy and speed. The theoretical foundation of YOLO and associated research will be covered in this article. The input picture is divided into a grid of cells by the YOLO technique, and each cell forecasts the class probabilities and bounding box coordinates for items inside the cell [15]. This method is substantially quicker than conventional object detection techniques since it can detect several items in a single run across the network. Moreover, YOLO enhances accuracy and simplifies the system by

using a single neural network for both object detection and categorization.

The foundation of the YOLOv4 method is the object detection framework, which allows a neural network to detect objects in a single forward pass. Figure 1 shows what YOLOv4 architecture looks like. A backbone network and a detecting head are the two primary parts of the method. The backbone network oversees taking features out of the input picture, while the detection head forecasts the object class, position, and confidence level. The CSPDarknet53 backbone network used by the YOLOv4 algorithm is an adaptation of the Darknet53 design that employs a cross-stage partial network to enhance information flow between layers. It has been demonstrated that the CSPDarknet53 backbone can lower processing expenses while increasing object detection accuracy.

To enhance its effectiveness, YOLOv4 makes use of several cutting-edge strategies. For instance, it makes use of a modified Darknet architecture that includes SPP and cross-stage partial connections (CSP) [16]. SPP is used to capture objects of multiple sizes, whereas CSP is used to improve feature reuse and cut down on the number of network parameters.

The usage of anchor boxes is one of YOLO's main characteristics. The technique employs anchor boxes, which are pre-defined boxes of various sizes and forms, to forecast the bounding boxes for objects. By decreasing the number of false positives and enhancing object localization, anchor boxes increase algorithmic accuracy. YOLOv4 variant excels in achieving high accuracy and processing speed through its adoption of advanced techniques such as the CSPDarknet53 backbone network and anchor boxes. These innovations enable it to capture objects of various sizes effectively while minimizing computational costs. Consequently, YOLOv4 stands as a robust choice for applications requiring real-time object detection, where both speed and precision are paramount.

A more compact and swifter variation of YOLOv4 is called YOLOv4 tiny, and it is made for low-power gadgets. To increase performance, it makes use of a condensed backbone and head network, as well as SPP and CSP. Figure 2 which shows what YOLOv4_tiny architecture looks like.

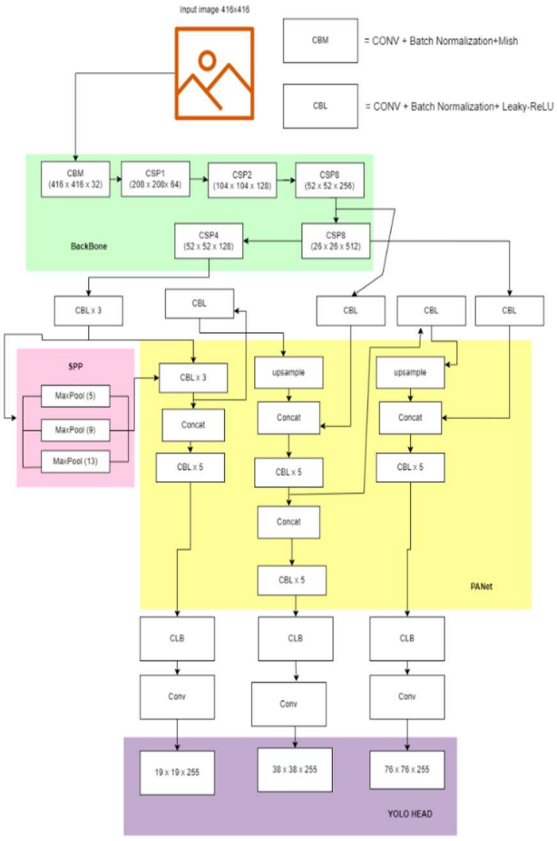

Figure 1: YOLOv4 Architecture

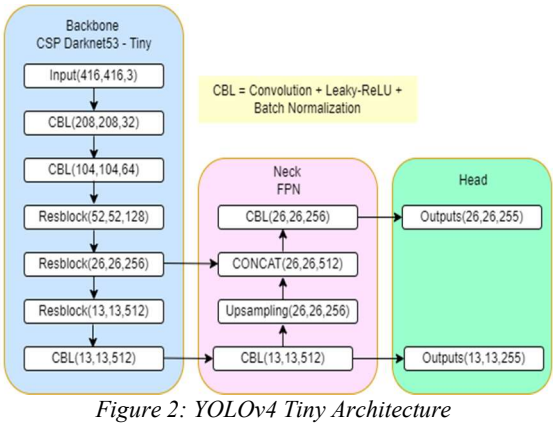

Figure 2: YOLOv4 Tiny Architecture

The YOLOv4 tiny_3l version, which utilizes only three convolutional layers as the backbone network, is an improved version of the original YOLOv4 tiny. This keeps excellent accuracy while lowering the number of parameters and the complexity of the computations. Figure 3 which shows what YOLOv4 tiny_3l architecture looks like.

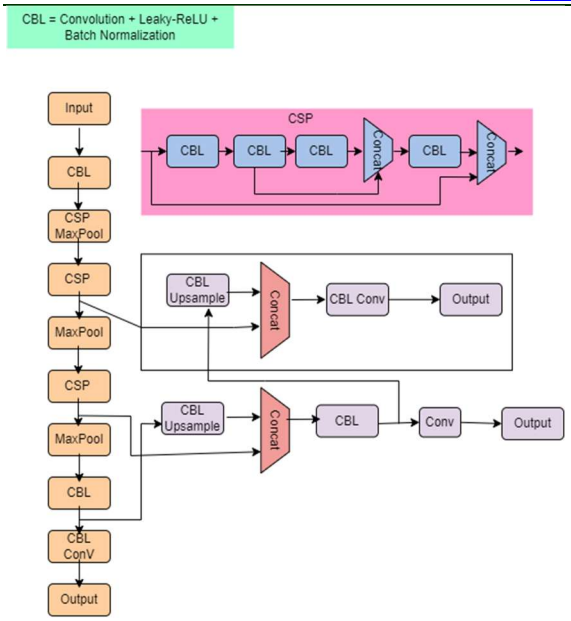

Figure 3: YOLOv4 Tiny_3l Architecture

Another YOLOv4 variant that has been optimized for CSP use is called YOLOv4_csp, and it is used in both the head and backbone networks. For tiny objects, this enhances the performance of YOLOv4. YOLOv4_csp variant proves invaluable in scenarios where small object detection is crucial. Below is Figure 4 which shows what the YOLOv4_csp architecture looks like.

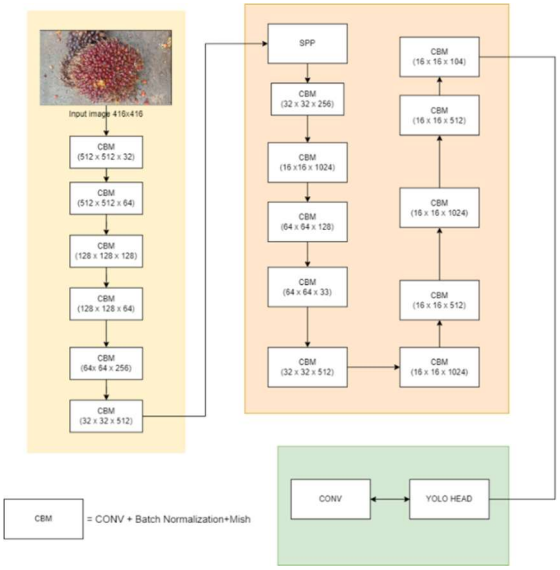

Figure 4: YOLOv4_csp Architecture

Modern object identification algorithms like EfficientDet and DETR are outperformed by YOLOv4 and its variations, according to a recent study. On the COCO dataset, for instance, YOLOv4 achieves a mean average accuracy (mAP) of 43.5%, which is greater than the mAP of 42.3% attained by EfficientDet-D7. Additionally, YOLOv4 outperforms EfficientDet-D7, which only manages 44 frames per second (fps), by achieving a speed of 65 fps on a single GPU [11].

Since YOLOv4 has been used extensively in previous studies, a comparison of multiple YOLOv4 models (YOLOv4-CSP-512, YOLOv4-CSP-608, YOLOv4-512, YOLOv4-608, YOLOv4-Tiny-512, and YOLOv4-Tiny-608) was done to determine the amount of maturity of oil palm fruit in trees. Evaluation of the various models was done by comparing recall, precision, average IoU, F1 score, and mAP values after 1000 iterations. Based on this study, it was determined that the YOLOv4 model is superior to the YOLOv4-CSP and YOLOv4-Tiny models [17]. Thirty-nine images from the dataset, taken while the fruit was still on the oil palm tree, were used in our work with a ratio of 9:1 for training and testing. The research conducted [18] uses YOLOv4 to detect three oil palm fractions (fraction 1 has a red color and only one bunch that falls, fraction 2 is shiny bright red and with two to five bunches that fall, and fraction 3 is orange colored with six to ten bunches that fall). The study shows that the accuracy of AP and mAP at checkpoint 6000 has the maximum value when comparing the values of TP, FN, FP, recall, precision, average IoU, F1 score, and mAP @ 0.5 of each weight arising from each iteration [18]. In prior research conducted [19], which is real-time detection of the ripeness of bananas using a camera and YOLOv4 on a single GPU was performed; the resulting mAP value was 87.6%, with the best iteration occurring on the 5000th iteration [19].

The other research done using YOLOv5 with an epoch of 100 has also been used in earlier investigations in addition to YOLOv4 to identify objects in inclement weather. For automobile detection with YOLOv5, the overall accuracy is 72.3%, while for truck detection it is 53.7% [20]. When compared to YOLOv3 Algorithms, the object detection method employing YOLOv5 has been proven to be quicker and more accurate. The YOLOv5 model's primary benefit is the classification and positioning of objects in a single network run. By providing extremely quick frame-by-frame processing, it enables real-time video processing.

The present research paper conducted a comprehensive evaluation of various object detection models, including YOLOv3, YOLOv4Tiny, YOLOv4, SSD, and RetinaNet, as undertaken [21]. The modifications were made to

YOLOv4, involving manual adjustment of Anchor box values to increase the dimensions of target bounding boxes, and setting the NMSThreshold based on categories. The results revealed that YOLOv4 outperformed the other models in terms of its effectiveness for optical remote sensing images. It achieved an impressive mean average precision (mAP) of 75.15% and demonstrated the highest average precision (AP) value for each category, surpassing the performance of the other models under evaluation [21]. These findings highlight the exceptional performance of YOLOv4 in the context of object detection for optical remote sensing imagery, solidifying its superiority over the compared models. Another study of modified YOLOv4 was previously used to create an underwater garbage-cleaning robot. It was modified by converting images that were originally 13×13, 26×26, and 52×52. into 13×13, 26×26, 52×52, and 104×104, and the evaluation method was then used to compare the mAP of the modified model and from research conducted [22]. Additionally, it is revealed that YOLOv4 is an enhanced version of YOLOv3 and that the Darknet53 model performs better than Resnet in terms of speed. This means that the YOLOv4 model is good. However, the updated YOLOv4 model may produce a 4.15% higher mAP yield than the original YOLOv4 model [22]. Then, while assessing picture test data, the modified YOLO approach was also used and obtained a high confidence level of more than 90% utilizing an intriguing dataset [23]. Multiple image processing techniques (extracted feature segmentation, grey threshold segmentation, edge segmentation, region segmentation, edge segmentation, light-colored impurities segmentation, dark impurity segmentation technique, and combine dark and light impurities) have been used to modify YOLOv4 so that it can perform recognition and classification by combining the segmentation multi-channel fusion technique and the YOLOv4 model. The dirt recognition rate on cotton increased by 5.6% because of this study, which was utilized to identify filthy cotton so that dirt could be detected [24]. In other research that utilized a modified version of YOLOv4 for drone detection, three convolutional layers were incorporated into the YOLOv4 framework to enable differentiation between birds and four distinct types of drones. [25], conducted a comprehensive analysis, comparing various performance metrics such as the confusion matrix, F1-Score, accuracy, precision, recall, and mean average precision (mAP), for both the modified YOLOv4 and the original YOLOv4. The study findings indicated that the modified YOLOv4 exhibited an accuracy that was 4% higher than the original YOLOv4 [25].

3. PROPOSED METHODS

3.1 Steps in the Research Approach

Figure 5 depicts the steps of research, modeling, and evaluation. First, the data was gathered at the palm field's locations. Second, using Roboflow preprocessing tools, apply the preprocessing data like resize to 416x416, auto orient, and auto adjust contrast also classified the oil palm fruit into 6 ripeness classes (Overripe bunch, Ripe bunch, Raw bunch, Underripe bunch, Abnormal bunch, and Empty bunch). Third, the dataset was split into 7:2:1 training, validation, and testing pictures by *Roboflow*. Fourth, set YOLOv4's hyperparameter based on the description in Table 1 below. Fifth, apply hyperparameters into several YOLO models like YOLOv4, YOLOv4 tiny, YOLOv4 tiny_3l, and YOLOv4_csp. Sixth, import pre-trained weight into the selected YOLO model and train the algorithm. Seventh, calculate the evaluation metrics with aid from the confusion matrix and equation formula. And the last step is analyzing the results from several YOLOv4 models.

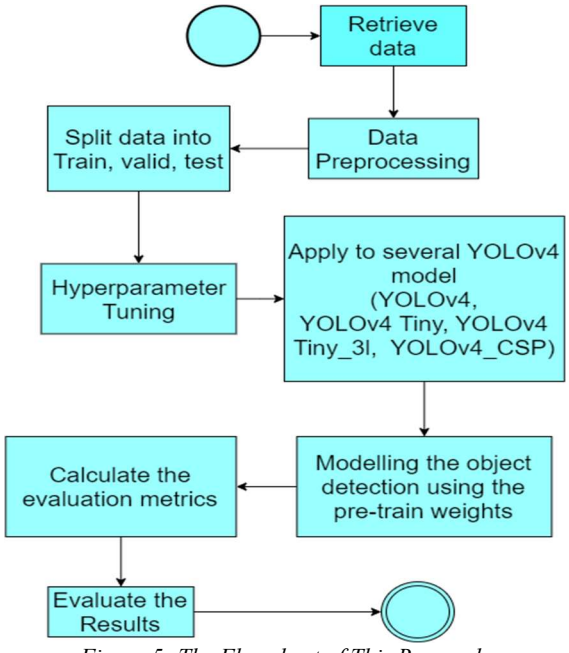

Figure 5: The Flowchart of This Research

3.2 YOLO Hyperparameter Tuning

On Google Colab, a classification model for different oil palm ripeness levels was built using YOLOv4 models. The research was statistically conducted using Python programming and the TensorFlow package. Utilizing the darknet, features

were extracted. This study assessed the effectiveness of the YOLOv4 that was previously taught using several YOLOv4 models. According to the object detection metrics (Precision, mAP, recall, and IoU) employed in this study, the YOLOv4 model was the best. To categorize oil palm fruit, the best map and the least overall loss were utilized. The fundamental cause is that the YOLOv4 mAP model's outputs have the greatest mean average precision of 83% and may generate high performance, i.e., 14 FPS. It was determined that, when compared to the previous YOLO model, YOLOv4 could create the best model. To train the detection of a unique object, the YOLOv4 model architecture must have its hyperparameters tuned. Fresh palm fruit bunches serve as the subject of this essay. Table 1 displays the hyperparameter tuning. There are additional adjustments to the YOLO head architecture where the filter was changed according to the (class × 5) + 3 formula.

Table 1: Hyperparameter Tuning of YOLO Model

Parameter	Value
Width	416
Height	416
Epochs	6000
Learning rate	0.01
Batch Size	64
Subdivision	8

3.3 Pre-Train Model

Pre-trained weights for YOLO models are typically derived from extensive training on large-scale image datasets, with the COCO (Common Objects in Context) dataset being a commonly chosen source. COCO is highly renowned in the field of object detection, boasting a diverse image collection spanning 80 distinct object categories, each meticulously annotated. This dataset's comprehensiveness positions it as an ideal fit for general object detection tasks.

The influence of pre-training on COCO significantly shapes YOLOv4 in multiple dimensions. Firstly, it facilitates transfer learning, enabling the model to harness the knowledge gleaned from the expansive COCO dataset, encompassing object recognition, shape understanding, and contextual awareness. This initial training phase imparts a foundational understanding of the model.

Moreover, the pre-trained weights encapsulate critical hierarchical features, including edges, shapes, and textures shared among a multitude of objects, serving as a pivotal starting point for subsequent fine-tuning endeavors. For instance, when fine-tuning for specific tasks like palm oil fruit detection, this pre-existing knowledge expedites training convergence, sparing the model from relearning these fundamental features from scratch.

Furthermore, pre-training on a diverse dataset like COCO bolsters the model's capacity to generalize effectively across a spectrum of object detection tasks. However, the subsequent fine-tuning of task-specific datasets, such as those tailored for palm oil fruit detection, further elevates the model's prowess in recognizing objects within that distinct domain.

3.4 Evaluation Metrics

Typically, object detection is assessed using the best metrics for each issue. The bounding box for each object discovered in the data image or video made it easy to assess the detection performance. The formula below may be used to measure detection performance using generic metrics such as precision Equation (1), mAP Equation (2), recall Equation (3), and F1 score Equation (4) [26].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (1)$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (2)$$

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

$$\text{f1 Score} = 2 \times \frac{\text{recall} \times \text{precison}}{\text{recall} + \text{precison}} \quad (4)$$

Precision plays an important role in assessing and evaluating our object detection. In this work, precision is defined as the ratio of real positive values to those designated as positive values (TP + FP). When a model has high precision, it may demonstrate that the model has high accuracy by defining the precision as the accuracy of the detection. The proportion of True Positive (TP) values among those with positive (TP + FN) values is referred to as the recall. The mean Average Precision (mAP) is the average Precision (AP) of each class. While the weight can reflect an improvement in recall from the prior criteria, the average accuracy can be calculated by averaging the precisions at each threshold. The threshold that can be established, and which must be stated, is what the AP metric depends on. The region of overlap between the predicted box of the actual item in the picture and the ground truth is known as the intersection over union or IoU. Figure 6 shows the calculation of IoU. The higher value of IoU can show that there is a good agreement between the

anticipated bounding box coordinates and the ground truth box coordinates.

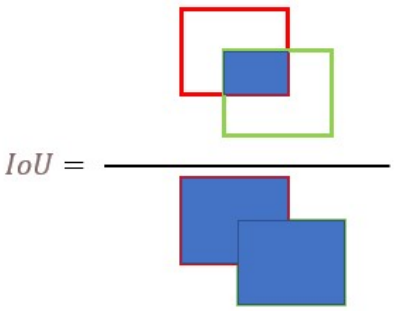

Figure 6: IoU Illustration

If a confusion matrix is present, the assessment metrics of recall and accuracy may be determined. The components of the confusion matrix are True Positive (TP), False Positive (FP), and False Negative (FN). True Positive when the event object detection value is accurate. False Positive occurs when the event object detection value is off. False Negative occurs when the event object detection value is mistakenly zero. The true negative (TN) category for object detection is not applied since an endless number of bounding boxes aren't identified in an image [27].

4. EXPERIMENTS

4.1 Overview

The Experiment compares 4 YOLO models (YOLOv4, YOLOv4 tiny, YOLOv4 tiny_3l, and YOLOv4_csp) to achieve the best model that can detect objects based on the color and texture of the object in each class. In this case, the research was performed with palm oil fruit images. The reason palm oil fruit images are used is because in terms of texture analysis the palm oil fruit can be used to determine palm oil fruit maturity. The texture of palm oil fruit varies as it ripens, from hard and smooth to softer and wrinkled. The maturity level of palm oil fruit can also be determined by analyzing the color of the fruit. The ripe palm oil fruit color often has a reddish-orange color, whereas the unripe immature fruit is purplish black. The experiment used a data preprocessing method to convert video data collection to be image frame. Every YOLO model used in this experiment has a different layer and activation function. To perform the YOLO object detection, we do the same parameter value of hyperparameter tunning so we can see which model results in the best score. The Experiment result shows the comparison result between the model to show the precision, F1 Score, mAP, and recall score.

4.2 Dataset

The data for this study were obtained by investigating Indonesia's Riau Province, Sumatra Island oil palm fields. The data was recorded through the camera on a smartphone with a resolution of 1280x720 pixels and saved as a video file in mp4 format. The video collection has 43 fresh fruit bunch videos that have been categorized according to maturity degree and 58 mixed maturity degree fresh fruit bunch videos. Utilizing VLC Player tools with a recording split ratio of 10 to preprocess video collection by turning the video into an image. The image was collected per 0.16 or 10/60 seconds for the video with 60 frames per second (FPS), or about 16 images may be generated if the video length is 10 seconds.

After processing the video data into images format in .jpg, the collection image data was uploaded to open source annotate tools for manual annotation. After the annotation process, the result of the annotated images was retrieved as JSON files (annotation files) and images. Consistency in annotations across all images was meticulously ensured through a manual process conducted by author. This comprehensive approach involved the development of detailed annotation guidelines, regular quality control checks, periodic annotation review meetings, open communication channels, comprehensive documentation, and an iterative approach for continuous improvement, collectively ensuring uniform and reliable annotations for the object detection dataset. The dataset comprises 4156 images classified into six classes (Overripe, Ripe, Raw, Underripe, Abnormal, and Empty Bunch). Examples of image data can be seen in Figures 7-10. Figure 7 shows an abnormal palm fruit bunch, Figure 8 contains 2 Raw palm fruit bunches, Figure 9 contains 2 Empty palm fruit bunches, as well as Figure 10, contains multiclass palm oil fruit bunch which 1 Underripe, 1 Ripe, and 1 Overripe. Once the picture and annotation files had been obtained, the researcher uploaded the file to *Roboflow* to partition the data into training (70%), testing (20%), and validation (10%) sets, which contain 2900 images for training, 839 images for testing, and 417 images for validation.

Figure 7: Abnormal Bunch

Figure 8: Raw Bunch

Figure 9: Empty Bunch

Figure 10: Multiclass Bunch (Underripe, Ripe, Overripe)

4.3 Experimental Setup

In our work, we attempted to create an effective deep-learning model for YOLOv4 architecture to classify the maturity of palm oil fruit. We used the Google Colab GPU for model training and assessment to do this. The palm oil fruit photos in the dataset for this investigation were taken from a nearby palm oil plantation and represented various stages of ripeness. The dataset was split into six classes: Overripe bunch, Ripe bunch, Raw bunch, Underripe bunch, Abnormal bunch, and Empty bunch, with about equal numbers of photos in each class. The photos were resized to 416x416 pixels and enhanced with random horizontal flips and rotations as part of the dataset's preprocessing. We utilized the Darknet framework and the TensorFlow package to implement the YOLOv4 model in Python. The model was trained using 64-batch training with a learning rate of 0.01 across 100 epochs. For model training, we utilized the Adam optimizer and the cross-entropy loss function. Following model training, we assessed the model's performance using several measures, such as accuracy, recall, and F1 score. To evaluate the model's precision and generalizability, we also conducted a visual evaluation of its predictions on a set of test photographs. We also deployed and assessed the performance of 3 other YOLOv4 models, YOLOv4 tiny, YOLOv4 Tiny_3l, and YOLOv4_csp, to compare the performance of our YOLOv4 model with that of other cutting-edge YOLOv4 models. To guarantee a fair comparison, we utilized the same dataset and training settings for both models. Finally, we examined and contrasted the computational needs of our YOLOv4 model with those of other models.

4.4 Main Result and Discussion

The YOLOv4 baseline of the YOLOv4 model for the detection COCO dataset has been performed [11]. with the AP50 score of 62.8% [11] The experiment results using test datasets that compared 4 YOLO models are shown in Table 2. From Table 2 YOLOv4 model has the best result with mAP@50 99.85%, Average IoU 89.95%, Precision 0.979, F1-Score 0.987, and Recall 0.996. According to a prior study, YOLOv4 performs better results than other YOLO models in terms of speed performance, accuracy [22], and mAP [21]. From Table 2 we can also see that the YOLOv4_csp has the lowest result with mAP@50 60.05%, Average IoU 39.36%, Precision 0.493, F1-Score 0.568, and Recall 0.670. Every model of YOLO has a different architecture of layers so the various values for each model are impacted by the activation functions, the existence of residual connections, network resolution, as well as the neck and backbone layer [28]. The most frequent assessment matrix for a classification model includes precision, recall, and F1-Score. The IoU score can be used for indicating that the predicted bounding box coordinates of detection are very close to the genuine box coordinates. The IoU value itself is influenced by the value of TP, FP, and FN. From this experiment, it's shown that YOLOv4 has the best Average IoU value.

© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Experimental Results

	YOLOv4	YOLOv4 Tiny	YOLOv4 Tiny _ 3l	YOLOv4 _csp
TP	**3005**	2988	2996	2069
FP	**63**	241	247	2122
FN	**11**	37	30	1019
Precision	**0.979**	0.925	0.923	0.493
F1 Score	**0.987**	0.955	0.955	0.568
mAP@50	**99.85%**	99.40%	99.65%	60.05%
Recall	**0.996**	0.987	0.990	0.670
Average IoU	**89.95%**	80.15%	82.15%	39.36%

Furthermore, the results graph of training and validation data may be utilized to determine the best model shown in Figure 11-12. From the figure below we can see that each image describes 2 models, so each figure includes four graphs (each model has two graphs, loss, and mAP graph). The Training graph of each model shows the loss convergence except for the Figure 12 YOLOv4_csp (brown-purple) graph. In object detection and identification, loss convergence defines a movement toward a condition in which purple) graph learned to correctly react to a set of training patterns within a certain range of error. The loss convergence will show when the value grows closer to a specified value as the iteration goes on. Figures 11 and 12 show the YOLOv4 loss value (Figure 11 Blue) began to converge on the iteration of 1800th, YOLOv4_Tiny (Figure 12 Light green) on the iteration of 1200th, and YOLOv4Tiny_3l (Figure 12 Blue) on the iteration of 1800th iteration. On the other hand, YOLOv4_csp exhibits the loss graph but not the convergent loss graph in Figure 12. The non-convergent loss indicates that the model is unstable, whereas the convergent loss indicates that the model is stable for image classification. YOLOv4_csp was observed to converge the slowest due to the computational difficulty of CSP use and the use of Mish as the activation. The learning rate and model activation alone has an impact on the loss graph in Figure 11-12. The mAP value is a metric used for evaluating the object detection model. From Figure 11-12 we can observe the mAP value of YOLOv4 (Figure 11 red) is the fastest (on the iteration of 1800th) and the most stable reaching a mAP value of 99-100% at a certain iteration. Besides that, the other model also gets 99-100% mAP during the training and validation. But from the test results shown in Table 2, the YOLOv4-CSP model did not show good mAP results. From the model of YOLOv4, the example of the image detection result is shown in Figure 13 below.

Based on the experimental results, this experiment has a classification of 6 classes (Overripe, Ripe, Raw, Underripe, Abnormal, and Empty) and outperforms some previous experimental results. For example, experiment [17] only classifies and detects fresh fruit bunch based on 1 class (ripe). In addition, in [18], the class division is only divided into 3 fractions. Each of the other experiments used YOLOv4 as the Object Detection model, lighting factors, and image capture angle as the determining factors for success.

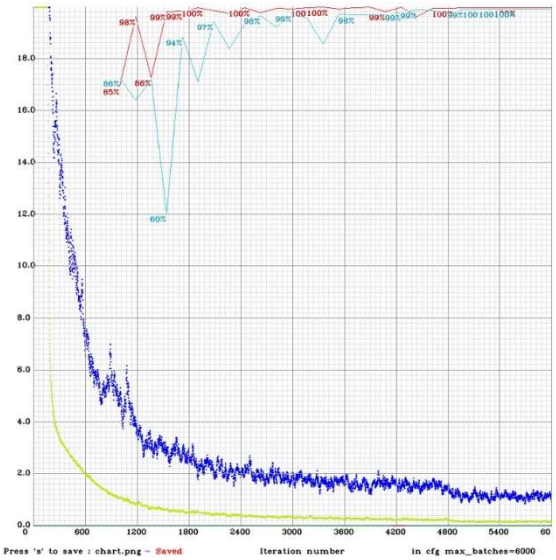

Figure 11: YOLOv4 (Red-Blue) vs YOLOv4 Tiny (Cyan-Light Green) Training Graph

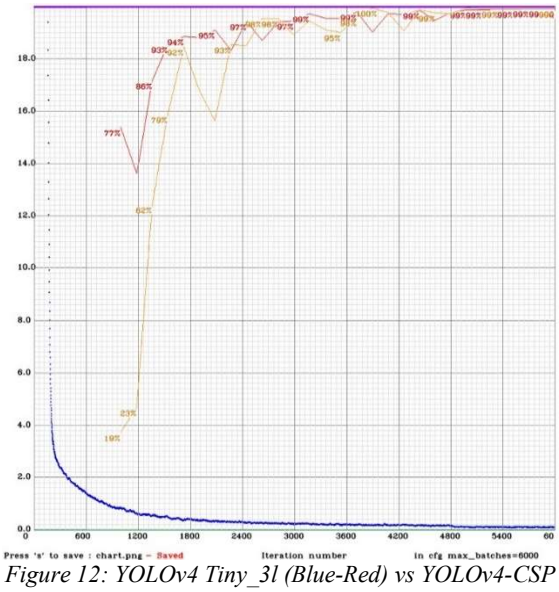

Figure 12: YOLOv4 Tiny_3l (Blue-Red) vs YOLOv4-CSP (Brown-Purple) Training Graph

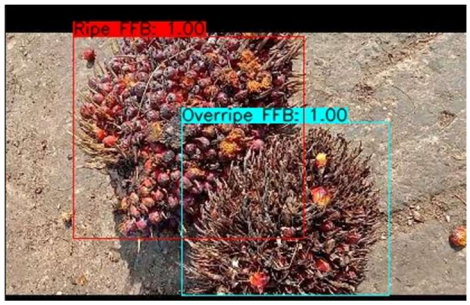

Figure 13: Detection Result

5. CONCLUSIONS AND FUTURE WORKS

In conclusion, this experiment aimed to employ the YOLO model to detect objects based on color and texture features, using palm oil fruit as the subject due to the distinguishable maturity indicators within its texture and color. Despite encountering various challenges such as differing lighting conditions, object variations, background diversity, and potential image interference, our focus remained on evaluating the effectiveness of different YOLO models for classifying and detecting oil palm fruit maturity stages. The comparison encompassed YOLOv4, YOLOv4 Tiny, YOLOv4 Tiny_3l, and YOLOv4_csp models. The comprehensive analysis revealed that the YOLOv4 model outperformed others, exhibiting exceptional precision with recall, mAP, precision, F1 scores, and Average IoU values of 0.996, 99.85%, 0.979, 0.987, and 89.95%, respectively. This underscores the YOLOv4 model's remarkable proficiency in object detection, especially when reliant on color and texture features, as exemplified in the case of oil palm fruit. The results reinforce the YOLOv4 model's superiority in leveraging these attributes for precise and reliable object identification, while concurrently highlighting the diminished effectiveness of the YOLOv4_csp model in oil palm fruit detection. In essence, our findings substantiate the significance of the YOLOv4 model for tasks where color and texture play a crucial role in object identification.

In the future, YOLOv4 may be improved and applied to sort palm oil fruit into several categories based on its color and texture, particularly for conveyor sorting and mobile applications. Developing a mobile application for real-time palm oil fruit classification using YOLOv4 presents certain challenges. These challenges could include optimizing the model for mobile hardware, ensuring efficient real-time processing, and managing resource constraints like memory and processing power. This might involve techniques like model optimization, compression, and efficient coding practices to make the application run smoothly on mobile devices. Refining the model's architecture and training procedure to attain even greater levels of fruit categorization accuracy is one possible direction for future investigation. A different option is to investigate the use of transfer learning, in which the model is pre-trained on a big dataset of comparable fruits, to enhance its performance on the particular job of classifying fruits that contain palm oil. The creation of a smartphone application that uses YOLOv4 for real-time fruit categorization might also be researched. This could be very helpful for farmers and other stakeholders in the palm oil business. Overall, there are a lot of enhancing possibilities for further exploring and developing YOLOv4 in the context of classifying palm oil fruits. The experiment resulting the comparison of YOLO model performance based on precision, mAP, recall, and F1 scores.

REFERENCES

[1] D. Feng, A. Harakeh, S. L. Waslander and K. Dietmayer, "A Review and Comparative Study on Probabilistic Object Detection in Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems,,* vol. 23, no. 8, pp. 9961-9980., 2021.

[2] E. Maiettini, G. Pasquale, L. Rosasco and L. Natale , "On-line object detection: a robotics challenge," *Autonomous Robots,* vol. 44, p. 739–757, 2020.

[3] S. Jha, C. Seo, E. Yang and G. P. Joshi, "Real time object detection and trackingsystem for video surveillance system," *Multimedia Tools and Applications,* vol. 80, p. 3981–3996, 2021.

[4] R. Yang and Y. Yu, "Artificial Convolutional Neural Network in Object Detection and Semantic Segmentation for Medical Imaging Analysis," *Frontiers in oncology,* vol. 11, p. 638182, 2021.

[5] M. N. Khan and M. M. Ahmed, "Snow Detection using In-Vehicle Video Camera with Texture-Based Image Features Utilizing K-Nearest Neighbor, Support Vector Machine, and Random Forest," *Transportation research record,* vol. 2673, no. 8, pp. 221-232, 2019.

[6] D. G. León, J. Gröli, S. R. Yeduri, D. Rossier, R. Mosqueron, O. J. Pandey and L. R. Cenkeramaddi, "Video Hand Gestures

Recognition Using Depth Camera and Lightweight CNN," *IEEE Sensors Journal,* vol. 22, no. 14, pp. 14610-14619, 2022.

[7] R. L. A, A. K. S, K. B. E, A. N. D and K. K. V, "A Survey on Object Detection Methods in Deep Learning," in *Proc. of 2021 Second Int. Conf. on Electronics and Sustainable Communication Systems (ICESC).*, Coimbatore, 2021.

[8] Q. Zhang, J. Lin, Y. Tao, W. Li and Y. Shi, "Salient object detection via color and texture cues," *Neurocomputing,* vol. 243, pp. 35-48, 2017.

[9] B. A. Varghese, S. Y. Cen, D. H. Hwang and V. A. Duddalwar, "Texture Analysis Of Imaging: What Radiologists Need To Know," *American Journal of Roentgenology,* vol. 212, no. 3, pp. 520-528, 2019.

[10] S. M. Islam and F. T. Pinki, "Colour, Texture, and Shape Features based Object Recognition Using Distance Measures," *Int. J. Eng. Manuf,* vol. 4, pp. 42-50, 2021.

[11] A. Bochkovskiy, C.-Y. Wang and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv,* vol. 2004.10934, pp. 1-17, 2020.

[12] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh and I.-H. Yeh, "CSPNet: A New Backbone That Can Enhance Learning Capability of CNN," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, Seattle, 2020.

[13] L. Zhu and P. Spachos, "Support vector machine and YOLO for a mobile food grading system," *Internet of Things,* vol. 13, no. 8, p. 100359, 2021.

[14] A. Dhillon and G. K. Verma , "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence,* vol. 9, no. 2, p. 85–112, 2020.

[15] S. Suhartini, N. Hidayat, N. A. Rohma, R. Paul, M. B. Pangestuti, R. N. Utami, I. Nurika and L. Melville, "Sustainable Strategies for Anaerobic Digestion of Oil Palm Empty Fruit Bunches in Indonesia: A Review," *Intenational Journal of Sustainable Energy,* vol. 41, no. 11, pp. 2044-2096, 2022.

[16] C.-Y. Wang, A. Bochkovskiy and H.-Y. M. Liao, "Scaled-YOLOv4: Scaling Cross Stage Partial Network," in *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, Nashville, 2021.

[17] J. W. Lai, H. R. Ramli, L. I. Ismail and W. Z. W. Hasan, "Real-Time Detection of Ripe Oil Palm Fresh Fruit," *IEEE Access,* vol. 10, pp. 95763-95770, 2022.

[18] A. Sopian, K. B. Seminar and Sudradjat, "System Detection Ripeness of Fresh Fruits Bunch Palm Oil," in *The International Conference on Industrial Engineering and Operations Management*, Istanbul, 2022.

[19] W. Widyawati and R. Febriani, "Real-time Detection of Fruit Ripeness Using the YOLOv4 Algorithm," *Teknika: Jurnal Sains dan Teknologi,* vol. 17, no. 2, pp. 205-210, 2021.

[20] T. Sharma, B. Debaque, N. Duclos, A. Chehri, B. Kinder and P. Fortier, "Deep Learning-Based Object Detection and Scene Perception," *Electronics,* vol. 11, no. 4, p. 563, 2022.

[21] Zakria, J. Deng, R. Kumar, M. S. Khokhar, J. Cai and J. Kumar, "Multiscale and Direction Target Detecting in Remote Sensing Images via Modified YOLO-v4," *IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing,* vol. 15, pp. 1039-1048, 2022.

[22] M. Tian, X. LI, S. Kong, L. Wu and J. Yu, "A Modified YOLOv4 Detection Method for a Vision-Based Underwater Garbage Cleaning Robot," *Frontiers of Information Technology & Electronic Engineering,* vol. 23, no. 8, pp. 1217-1228, 2022.

[23] Z. Cheng and F. Zhang, "Flower End-to-End Detection Based on YOLOv4 Using a Mobile Device," *Hindawi: Wireless Communications and Mobile Computing,* vol. 2020, pp. 1-9, 2020.

[24] C. Zhang, T. Li and W. Zhang, "The Detection of Impurity Content in Machine Picked Seed," *Agronomy,* vol. 12, no. 1, p. 66, 2022.

[25] F. D. Javan, F. Samadzadegan, M. Gholamshahi and F. A. Mahini, "A Modified YOLOv4 Deep Learning Network for Vision Based," *Drones,* vol. 6, no. 7, p. 160, 2022.

[26] A. Malta, M. Mendes and T. Farinha, "Augmented Reality Maintenance Assistant Using YOLOv5," *Applied Sciences. 2021,* vol. 11, no. 11, p. 4758, 2021.

[27] R. Padilla, S. L. Netto and E. A. B. da Silva, "A Survey on Performance Metrics for Object-

© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Detection Algorithms. 2020 International Conference on Systems," *Signals and Image Processing (IWSSIP),* pp. 237-242, 2020.

[28] A. I. B. Parico and T. Ahamed, "Real Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT," *Sensors,* vol. 21, no. 4803, pp. 1-32, 2021.