Identification of the Maturity Level of Oil Palm Fruit Using a Combination of the You Only Look Once Version 8 Model and Convolutional Neural Network

Dwi Ratna Sulistyaningrum*, Muchammad Aquila Azahari, Budi Setiyono and Dieky Adzkya

Department of Mathematics, Faculty of Science and Data Analytics, Institut Teknologi Sepuluh Nopember, Jl. Teknik Kimia, 60111 Surabaya, Indonesia
*dwiratna@its.ac.id

Abstract. The palm oil industry plays a vital role in Indonesia's economy. One of the key challenges in maintaining high crude palm oil (CPO) quality is accurately determining the ripeness level of oil palm fruits. Innovative technologies have been established to identify the level of ripening of palm oil fruits. However, existing methods often struggle because of the complex visual characteristics of fruits, including noise, redundant information, and difficulty in extracting relevant features. This study proposes a combination of the You Only Look Once Version 8 (YOLOv8) model and a Convolutional Neural Network (CNN) to determine the ripeness level of oil palm fruits. The dataset used comprised 4,160 images containing 14,559 annotated objects across six categories: raw, underripe, ripe, overripe, abnormal, and empty bunch. Data were collected from an oil palm mill in South Kalimantan, Indonesia. Two training scenarios were conducted: the first evaluated the effect of different backbones (ResNet50, GhostNetP2, GhostNetP6, and DarkNet53), and the second examined the effect of hyperparameter tuning. The best-performing model, YOLOv8 with the DarkNet53 backbone, achieved a mean Average Precision (mAP) of 91.43% at an Intersection over Union (IoU) threshold of 0.50, along with an F1-score of 99.08%. Additionally, the model required only 2.99 hours for training over 163 epochs. These results highlight the effectiveness and efficiency of the proposed approach for automatic ripeness classification in palm-oil quality control.

Keywords: Palm oil fruit, Object detection model, Classification model, You Only Look Once version 8, Convolutional Neural Network.

1 Introduction

The palm oil industry is a vital agricultural sector that contributes significantly to Indonesia's economic growth. Palm oil is widely utilized in various industrial applications because of its oxidative stability, strong coating ability, and capacity to dissolve chemicals. It serves as a primary raw material for cooking oil, industrial oil, and biodiesel fuels [1]. According to the Central Bureau of Statistics, Indonesia's crude palm oil

K. P. Candra et al. (eds.), *Proceedings of the International Conference of Tropical Agrifood Feed and Fuel 2024 (ICTAFF 2024)*, Advances in Biological Sciences Research 47,
https://doi.org/10.2991/978-94-6463-825-7_13

(CPO) production will increase by 3.77% in 2022, reaching 46.82 million tons. This growth had a positive effect on CPO exports, amounting to 25 million tons with a total export value of USD 29.62 billion, representing a 3.56% increase from the previous year. These figures set a new record over the past decade, reinforcing Indonesia's position as the largest producer and exporter of palm oil globally by 2022 [2].

The ripening stage of oil palm fruit can be visually identified by changes in skin color. In its early stages, the fruit appears dark, either deep purple or dark green in color. As it matures, the color transitions to orange or tangerine. A fruit is considered fully ripe when its skin is uniformly orange, its texture softens, and its internal content reaches optimal development [1][3]. Conventionally, the assessment of fruit ripeness in the field relies on human labor, which involves manual observation of color and texture. This method is inherently prone to subjective bias, human error, and inefficiencies, which may affect the profitability of the palm oil industry [4].

To overcome these limitations, recent studies have explored the use of various technologies to automate ripeness identification. These include optical sensors [5], machine vision [1], and machine-learning or deep-learning models [6], [7], [8]. Although existing models have improved performance, they still face difficulties owing to the complex visual characteristics of oil palm fruit, such as irregular color distribution, intra-class variation across fruit varieties, and environmental noise during image capture. Consequently, further research is essential to enhance the accuracy and precision of the ripeness classification tasks.

Convolutional Neural Network (CNN) architecture is a viable method for assessing palm fruit ripeness. CNNs have been widely applied in object detection and classification tasks, particularly in agricultural applications. A typical CNN architecture consists of convolutional layers for feature extraction, pooling layers for spatial downsampling, and fully connected layers for classification. CNN-based object detection models include the Region-Based Convolutional Neural Network (R-CNN), Faster R-CNN, You Only Look Once (YOLO), and Single Shot Multibox Detector (SSD), while popular classification models include AlexNet, VGGNet, GoogLeNet, ResNet, MobileNet, and EfficientNet [9].

Object detection and classification have become active research areas for assessing palm fruit maturity. For instance, Xiao et al. (2023) demonstrated that YOLOv8 with a C2F module achieved 99.5% accuracy in fruit ripeness detection tasks [10]. Similarly, Alfredo and Suharjito [4] proposed enhancements to the AlexNet model by integrating image enhancement and hyperparameter tuning techniques, achieving 95.3% accuracy across six ripeness categories using 6,000 annotated bounding boxes.

In this study, an advanced object detection and classification framework is employed to ascertain the ripeness stage of palm fruit through the application of You Only Look Once version 8 (YOLOv8) and Convolutional Neural Network (CNN) models. To enhance the detection of small objects, CNNs are utilized as the backbone of YOLOv8 because they are known to improve feature representation and address YOLO's limitations in detecting small targets [11]. CNNs also enable the training of deeper networks, which helps mitigate the vanishing gradient problem and allows for transfer learning from large-scale datasets, such as ImageNet. The CNN architectures used as backbones in this study included ResNet50, GhostNetP2, GhostNetP6, and DarkNet53. This study aims to support the palm oil industry in identifying the maturity stages of palm fruits and to serve as a foundation for future research in this area.

1.1 Data Preparation

Data preparation involved organizing the dataset used for training, validation, and testing the model to conclude the research on identify the ripeness level of oil palm fruits. The dataset used in this study was secondary image data obtained from the Science Data Bank dataset [12]. This dataset was collected from an oil palm mill in South Kalimantan, Indonesia and consists of videos and images with various categories that reflect the actual conditions encountered by the grading section of the oil palm mill. The dataset was collected from 45 single-category videos of fresh fruit bunches (FFB) and 56 videos containing multiple FFB with several categories for each video. The videos were recorded using a smartphone with a resolution of 1280 × 720 pixels . mp4 format. The videos were then converted into 4,160 images with 14,559 labelled objects. An example of the research image data is shown in Fig. 1.

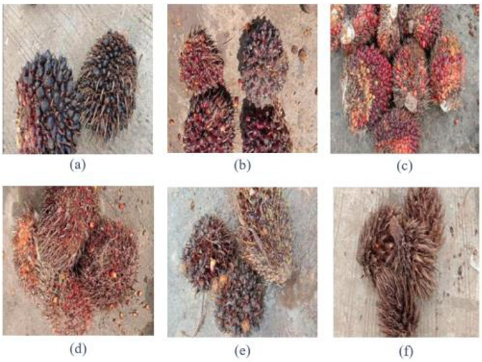

Fig. 1. Images of oil palm fruit: (a). Raw, (b). underripe, (c). ripe, (d). overripe, (e). abnormal, (f). empty bunch

The dataset was systematically annotated and classified into six distinct categories according to the ripening stages of oil palm fruits: unripe, under-ripe, ripe, overripe, abnormal, and empty bunches. Subsequently, the images in the dataset were resized to a resolution of 416 pixels × 416 pixels for processing consistency. Fig. 2 shows the number of images per category.

The dataset is partitioned into three discrete components: training data, validation data, and test data, following a proportional allocation of 8:1:1. The training data were used to train the model, the validation data were used to evaluate the model's performance and assist in hyperparameter tuning, and the test data were used to test the trained model to obtain reliable conclusions. This split ensured that the resulting model was accurate and capable of generalizing well to unseen data.

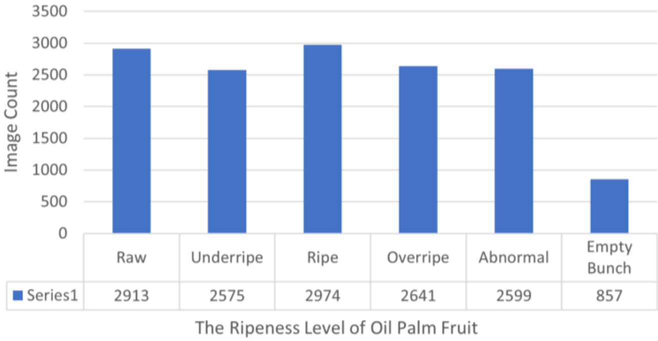

Fig. 2. Image Data of Oil Palm Fruit

1.2 Model Design

Backbone. The backbone Convolutional Neural Network (CNN) is responsible for extracting features from the input images. In the YOLOv8-CNN model, the backbone comprises several CNN layers along with Spatial Pyramid Pooling Fast (SPPF) modules. The CNN architectures employed as backbones included ResNet50, GhostNetP2, GhostNetP6, and DarkNet53. The selection of these architectures is motivated by their ability to capture fine-grained features, which helps address YOLO's limitations in detecting small objects [13]–[16]. The SPPF module, a variation of traditional Spatial Pyramid Pooling (SPP), is incorporated to efficiently aggregate features across multiple spatial scales while accelerating the computational process.

Neck. The neck is an architectural component that connects the backbone and head of the model. Its primary function is to aggregate and enhance the features extracted by the backbone to improve the accuracy of the predictions made by the head. YOLOv8 employs a Path Aggregation Network (PANet) as its neck structure. PANet facilitates the flow of information across different spatial resolutions, enabling the model to capture multi-scale features more effectively. The neck stages in the YOLOv8-CNN model are shown in Fig. 3 [17].

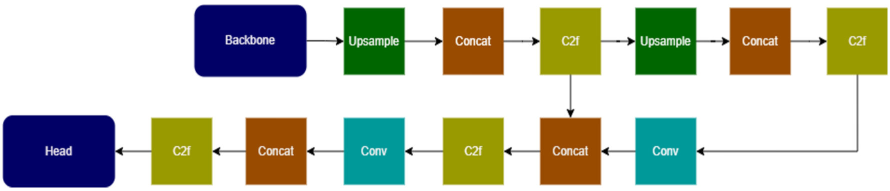

Fig. 3. Neck Stages in YOLOv8-CNN Model

Head. The head is responsible for generating predictions based on features extracted from the backbone and neck. YOLOv8 employs a variety of detection modules that estimate the bounding boxes, objectness scores, and class probabilities for each individual grid cell within the feature map. These predictions were subsequently combined to obtain the final detection. Detection modules P3, P4, and P5 correspond to different feature scales generated by the backbone network, representing outputs from specific layers with varying resolutions. The stages of the head in the YOLOv8-CNN model are shown in Fig. 4 [17].

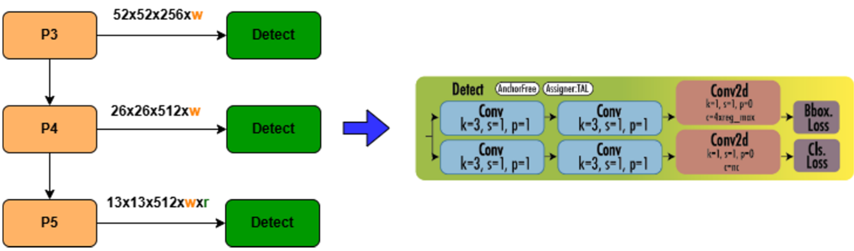

Fig. 4. Head stages in YOLOv8-CNN model.

YOLOv8 leverages features at multiple scales to enhance the detection of objects of various sizes. The P3 feature map, characterized by a high resolution, enables the detection of small objects that may be overlooked at lower resolutions. The P4 feature map balances the resolution and feature size, making it appropriate for the detection of medium-sized objects. Meanwhile, the P5 feature map has a lower resolution, but captures a more global context, which benefits the detection of large objects. These multi-resolution features (P3, P4, and P5) are extracted by the YOLOv8 backbone and utilized in the head for multiscale object detection. This approach improves overall detection accuracy by effectively addressing objects of different sizes.

1.3 Model Training Design

The model training process comprised two primary tasks: image classification and object detection, which were conducted concurrently under two experimental setups. These setups were designed to evaluate the impact of the backbone architecture and hyperparameter configuration on model performance. To assess the influence of the different backbone networks, the YOLOv8 model was trained using four CNN variants: ResNet50, GhostNetP2, GhostNetP6, and DarkNet53. For hyperparameter tuning, the best-performing backbone from the previous experiment was selected and trained with varying batch size, patience, learning rate, dropout rate, and optimizer type configurations. This dual-phase training strategy was intended to determine the effectiveness of the YOLOv8-CNN model in accurately detecting and classifying oil palm fruits based on their ripeness levels.

1.4 Model Testing Design

Model testing was conducted with three test scenarios, and all three scenarios were additional experiments of noise in the test data. The test scenarios included motion blur, overexposure, and under-exposure data scenarios, all of which are common in real-world settings. This study aimed to test how well the model can recognize and classify oil palm fruit objects under different visual conditions, imitating real conditions in the field. Furthermore, the model was evaluated using a confusion matrix to assess how well it could classify objects based on their class labels. In addition, the Mean Average Precision (mAP) value was used as an evaluation metric to measure the accuracy and reliability of the model in object detection. The combination of these two metrics provides a comprehensive picture of the performance of the model in dealing with variations in the visual conditions encountered in this test scenario.

1.5 Performance Evaluation

System performance evaluation is an assessment process that aims to determine the efficiency, effectiveness, and ability of a system to meet set objectives. The system performance measurement involves precision, recall, and accuracy. Precision refers to the degree of accuracy with which a system provides answers that match the information desired by the user. Recall, on the other hand, measures the extent to which the system can retrieve relevant information. *The accuracy* is the degree of agreement between the predicted value given by the system and the actual value. This performance evaluation not only provides an overview of the effectiveness of the system but also helps to improve and optimize the functions of the system to achieve better results.

Mean Average Precision. Mean Average Precision (mAP) is an evaluation metric commonly used in object detection tasks in image or video processing. This metric provides an overview of the extent to which a model can accurately detect and locate objects. Specifically, mAP measures the object detection accuracy by considering the Average Precision (AP) value of each object class tested. *Average precision* is a metric that describes how well a model can recover objects with a certain level of precision. Precision was measured against recall in the range 0–1. The mAP then took the average of the AP values for each object class, providing an overview of the overall detection performance of the model. In other words, mAP provides a comprehensive measurement of a model's ability to detect objects with high accuracy and good precision, and is often used as one of the main parameters in evaluating the quality of an object detection model [18]. The higher the mAP value, the better the detection ability of the model. The equation used to calculate the mAP value is as follows:

$$mAP = \sum_{i=1}^{k} \frac{AP_i}{k}$$

where k: Number of classes (data) Average Precision and AP_i : Average Precision on class-*i*.

Average Precision (AP) was calculated by determining the area under the precision-recall (PR) curve. More specifically, AP is defined as the average precision over a dataset that includes 11 recall values. The range of these recall values ranges from 0.0 to 1.0, with a distance of 0.1 between each value. Thus, AP provides an idea of how well the model can maintain a certain level of precision as the recall increases within a specified range, mathematically expressed by the following equation [19]:

$$AP = \frac{1}{11} \sum_{\substack{r=0 \\ step=0.1}}^{1} p(r)$$

Confusion Matrix. The confusion matrix functions as a key resource for analyzing performance in the pattern recognition and classification sectors. This provides a more detailed picture of how well a model or classification system can predict the class or label of the test data. A confusion matrix is commonly used in binary classification, where there are two classes or labels to be predicted: positive and negative. The confusion matrix consisted of four main cells.

1. True Positive (TP): The count of instances that genuinely belong to the positive category and are accurately classified as positive by the predictive model.
2. True Negative (TN): The count of instances that inherently belong to the negative category and are accurately classified as negative by the predictive model.
3. False Positive (FP): The count of instances that ought to have been categorized within the negative class but were erroneously classified as positive by the predictive model. This phenomenon is also referred to as type 1 error.
4. False Negative (FN): The count of instances that ought to have been categorized within the positive class but were erroneously classified as negative by the predictive model. This phenomenon is also referred to as a Type II error.

The following will show the equations were used to calculate the precision, recall, and accuracy in the system [20]:

$$Precision = \frac{TP}{TP + FP} x\ 100\%$$

$$Recall = \frac{TP}{TP + FN} x\ 100\%$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} x\ 100\%$$

An illustration of the two-class confusion matrix is provided in Table 1.

Table 1. Two-class confusion matrix

Prediction Value	Value	
	True	False
True	TP (True Positive) There is a correctly identified object	FP (False Positive) There is an object detected incorrectly
False	FN (False Negative) No object detected wrong	TN (True Negative) No objects were detected correctly

2 Results and Discussion

2.1 Training Results and Model Evaluation

The training stage was conducted to optimize the model weights, which were subsequently used during the testing phase. The analysis of the training outcomes and model evaluation is organized into two scenarios: backbone variation and hyperparameter tuning. These two scenarios resulted in six trained models, namely YOLOv8-ResNet50 (A), a YOLOv8 model with a ResNet50 backbone using the first hyperparameter tuning configuration; YOLOv8-ResNet50 (B), a YOLOv8 model with a ResNet50 backbone using the second hyperparameter tuning configuration; YOLOv8-GhostNetP2, YOLOv8 model with a GhostNetP2 backbone, YOLOv8-GhostNetP6, YOLOv8 model with a GhostNetP6 backbone, YOLOv8-DarkNet53 (A), a YOLOv8 model with a DarkNet53 backbone using the first hyperparameter tuning configuration, and YOLOv8-DarkNet53 (B), a YOLOv8 model with a DarkNet53 backbone using the second hyperparameter tuning configuration.

During the training process, the box loss and classification loss were recorded at each epoch. Box loss refers to the error in predicting object locations (bounding boxes), whereas classification loss reflects the errors in predicting object classes. Analyzing these losses is essential for identifying overfitting and underfitting and evaluating the overall generalization performance of the YOLOv8-CNN model.

The performance of the model was further assessed using the mean Average Precision (mAP) at an Intersection over Union (IoU) threshold of 0.50, along with the confusion matrix. The training and evaluation results for both scenarios were summarized based on these metrics. Fig. 5 and Table 2 present a comprehensive overview of the model performance for each training configuration.

The results of training and evaluating the model using the two scenarios show that the loss value is not directly related to the mAP value. One reason is that the loss function used is a combination of several losses. The combination of coefficients and inequality of the values of each loss can affect the overall loss value produced by the training process. The effect of the loss value on the mAP level in this model requires further analysis based on variations in the backbone and hyperparameters used. Table 3 shows the precision, recall, F1-score and accuracy of each model studied.

The best model was selected based on the highest mAP and F-1 Score values from the confusion matrix calculation in the model evaluation process. Therefore, the YOLOv8-

DarkNet53 model was selected as the best model because it had an mAP value of 91.43% and F-1 score of 99.08%.

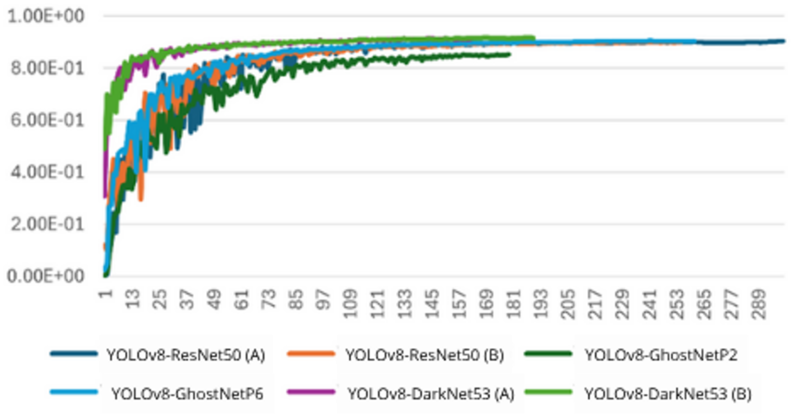

Fig. 5. mAP Training Results and Evaluation in Each Scenario

Table 2. Summary of Training Results and Evaluation

Model Name	Smoothed (%)	Value (%)	Epoch	Time (h)
YOLOv8-ResNet50 (A)	90.38	90.43	300	6.573
YOLOv8-ResNet50 (B)	90.2	90.49	255	6.098
YOLOv8-GhostNetP2	85.49	86.06	179	3.529
YOLOv8-GhostNetP6	90.42	90.31	261	5.442
YOLOv8-DarkNet53 (A)	91.27	91.43	163	2.995
YOLOv8-DarkNet53 (B)	91.8	91.19	190	3.535

Table 3. Summary of Confusion Matrix Calculation Results

Model Name	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
YOLOv8-ResNet50 (A)	97.86	99.62	98.64	99.54
YOLOv8-ResNet50 (B)	94.9	99.48	97.13	99.08
YOLOv8-GhostNetP2	86.76	97.55	91.83	97.26
YOLOv8-GhostNetP6	98.61	99.86	99.23	99.73
YOLOv8-DarkNet53 (A)	98.36	99.74	99.08	99.69
YOLOv8-DarkNet53 (B)	98.18	99.83	98.96	99.62

2.2 Model Testing Results

At this stage, a model trial was conducted using three test scenarios based on the data subjected to motion blur, overexposure, and underexposure. This test aims to assess the robustness of the model in identifying the ripeness level of oil palm fruits. Fig. 6. shows the modified image data.

Fig. 6. Image Test Data Modification Results

First-scenario testing (Motion Blur). Model testing on the motion blur scenario aims to analyze the effect of blurring on the image to identify the maturity stage of the oil palm fruit. The test results for this scenario are presented in Fig. 7 and Table 4.

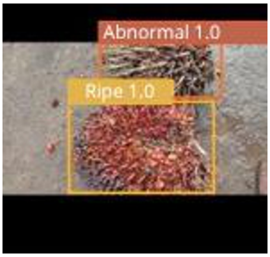

Fig. 7. Test Results of Motion Blur Image Data

Table 4. Summary of model performance test results for motion-blurred image.

Maturity Level	Precision (%)	Recall	F1-Score (%)	mAP50 (%)
Empty Bunch	97.78	1	98.88	99.5
Underripe	98.88	1	99.43	99.5
Abnormal	99.25	1	99.62	99.5
Ripe	99.59	1	99.79	99.5
Raw	99.25	1	99.63	99.5
Overripe	98.61	1	99.29	99.5

Utilizing the confusion matrix derived from the empirical evaluation of the model, the F-1 Score metric can be computed to evaluate the model's performance in a classification task characterized by an imbalanced class distribution. The F-1 score of the YOLOv8-DarkNet53 model is 99.44%. This shows that the model performed very well in handling the class imbalance. The mAP value obtained from the model test is 99.5%.

Second Scenario Testing (overexposed). Model testing in the overexposure scenario aims to analyze the effect of excess light on the image to identify the ripeness level of the oil palm fruit. The results of testing the overexposure image data in the form of mAP values and confusion matrix calculations are shown in Fig. 8 and Table 5.

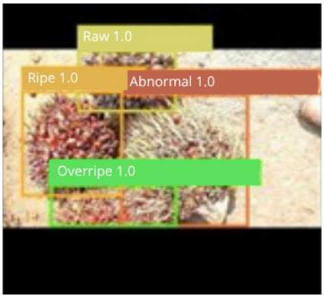

Fig. 8. Over Exposed Image Data Test Results

Table 5. Summary of Model Performance Test Results on Over Exposures Images.

Maturity Level	Precision (%)	Recall	F1-Score (%)	mAP50 (%)
Empty Bunch	98.88	1	99.44	99.5
Underripe	99.25	1	99.62	99.5
Abnormal	96.69	1	98.32	99.5
Ripe	86.32	1	92.65	99.5
Raw	97.79	1	98.89	99.5
Overripe	96.92	1	98.43	99.5

In reference to the confusion matrix derived from the outcomes of model testing, the F-1 Score value can be calculated to assess the model in a classification task with an unbalanced class. The F-1 Score of the YOLOv8-DarkNet53 model is 97.89%. This shows that the model exhibits excellent performance in handling class imbalances. The mAP value obtained from the model test is 99.5%. Based on these results, the overexposure effect has almost no significant impact on the classification and object detection processes using the proposed model.

Third-scenario testing (Inder Exposed). Model testing in the underexposure scenario aims to analyze the effect of excess light on the image to identify the ripeness level of oil palm fruit. The results of testing the underexposure image data in the form of mAP values and confusion matrix calculations are presented in Fig. 9 and Table 6.

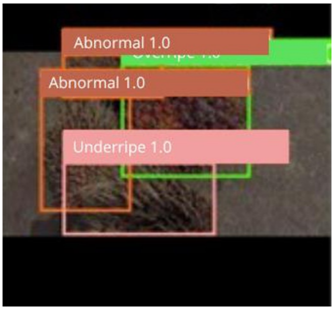

Fig. 9. Under Exposed Image Data Test Results

Table 6. Summary of model performance test results for underwater images.

Maturity Level	Precision (%)	Recall	F1-Score (%)	mAP50 (%)
Empty Bunch	97.78	1	98.88	99.5
Underripe	98.88	1	99.43	99.5
Abnormal	99.25	1	99.62	99.5
Ripe	99.59	1	99.78	99.5
Raw	99.25	1	99.63	99.5
Overripe	98.26	1	99.13	99.5

In reference to the confusion matrix derived from the outcomes of model testing, the F-1 Score value can be calculated to assess the model in a classification task with an unbalanced class. The F-1 Score of the YOLOv8-DarkNet53 model is 99.41%. This shows that the model performed very well in handling the class imbalance. The mAP value obtained from the model test is 99.5%. Based on these results, it can be concluded that the under-exposure effect has almost no significant impact on the classification and object detection processes using the proposed model.

3 Conclusion

Based on the research results, this study successfully applied a combination of You Only Look Once Version 8 (YOLOv8) and Convolutional Neural Network (CNN) models to identify the ripeness level of oil palm fruit. The identification process began with data preparation, divided into training, validation, and test data, with noise added to the test data as part of the testing scenario. Model training was performed with two scenarios, namely, the influence of the backbone and the influence of hyperparameter tuning, using CNN models such as ResNet50, GhostNetP2, GhostNetP6, and DarkNet53 as the backbone. The best model was selected based on the mean Average Precision (mAP) value and the confusion matrix. The test results showed that the YOLOv8 model with the DarkNet53 backbone provided the best performance, with an mAP value of 91.43% at an IoU threshold of 0.50, and an F-1 Score of 99.08%. This model also excels in computational time efficiency, with a computational time of 2.99 hours for 163 epochs. As suggestions for further research, it is recommended that the YOLOv8-CNN model with ResNet50, GhostNetP2, GhostNetP6, and DarkNet53 backbones be further developed with modifications to the neck. In addition, the variation of hyperparameters in training must be increased to obtain a more optimal performance.

Acknowledgments. This research was funded by the Directorate of Research, Technology, and Community Service, Ministry of Education, Culture, Research, and Technology, based on the Master Contract for the Implementation of the State University Operational Assistance Program for Regular Fundamental Research. Master Contract Number: 038/E5/PG.02.00.PL/2024, dated June 11, 2024. Research Contract Number: 1750/PKS/ITS/2024, dated June 12, 2024.

References

1. A. Septiarini, A. Sunyoto, H. Hamdani, A. A. Kasim, F. Utaminingrum, and H. R. Hatta, "Machine vision for the maturity classification of oil palm fresh fruit bunches based on color and texture features," Scientia Horticulturae, vol. 286, p. 110245, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304423821003526
2. Badan Pusat Statistik (BPS). *Statistical Yearbook of Indonesia 2022*. https://www.bps.go.id. Accessed 2 February 2024.
3. Nugroho A. *[Palm Oil Agroindustry Technology]*. Banjarmasin: Lambung Mangkurat University Press; 2019.
4. I. Alfredo and S. Suharjito, "Improvement of alexnet model to detect oil palm FFB ripeness using image enhancement and hyperparameter tuning," Jurnal Ilmiah Teknologi Dan Rekayasa, vol. 27, no. 1, pp. 56–68, 2022.
5. C. Valero, I. Homer, J. Ortiz-Canavate, M. Ruiz-Altisent, G. Ramos ˜ et al., "Non-destructive fruit firmness sensors: A review", Spanish Journal of Agricultural Research (Espana), 2023.
6. S. Saifullah, D. B. Prasetyo, Indahyani, R. Drezewski, and ˙ F. A. Dwiyanto, "Palm oil maturity classification using k nearest neighbors based on rgb and l*a*b color extraction," Procedia Computer Science, vol. 225, pp. 3011–3020, 2023, 27th International Conference on Knowledge Based and Intelligent Information and Engineering Systems (KES 2023). [On line]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923014527
7. E. Salim and Suharjito, "Hyperparameter optimization of yolov4 tiny for palm oil fresh fruit bunches maturity detection using genetics algorithms," Smart Agricultural Technology, vol. 6, p. 100364, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772375523001922
8. Elwirehardja GN, Prayoga JS, et al. Oil palm fresh fruit bunch ripeness classification on mobile devices using deep learning approaches. *Comput Electron Agric*. 2021;188:106359.
9. M. Ahmed and A. Ahmed, "Palm tree disease detection and classification using residual network and transfer learning of inception resnet," Plos one, vol. 18, no. 3, p. e0282250, 2023.

10. B. Xiao, M. Nguyen, and W. Q. Yan, “Fruit ripeness identification using yolov8 model,” Multimedia Tools and Applications, pp. 1–18, 2023.
11. Ali ML, Zhang Z. The YOLO framework: a comprehensive review of evolution, applications, and benchmarks in object detection. *Computers*. 2024;13:336. https://doi.org/10.3390/computers13120336
12. Suharjito and F. A. Junior, “Annotated Video Dataset of Oil Palm Fruit Bunch Piles for Ripeness Grading,” Feb. 2023.
13. Anggoro VK. Application of Faster Region-Based Convolutional Neural Network for Disease Identification on Rice Leaf Images. Doctoral dissertation, Institut Teknologi Sepuluh Nopember; 2023. *(Unpublished)*.
14. M. K. N. Mubarok, “Application of convolutional neural network and transformer for image captioning on diseased chili plants,” Ph.D. dissertation, Institut Teknologi Sepuluh Nopember, 2024.
15. K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, “Ghostnet: More features from cheap operations,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 1580– 1589.
16. N. N. Ramanathan, S. Darapaneni, B. K. Sil, A. S. R. K. Paduri, and A. Manuf, “Deep convolutional neural network (cnn) design for pathology detection of covid-19 in chest x-ray images,” in Machine Learning and Knowledge Extraction: 5th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2021, Virtual Event, August 17–20, 2021, Proceedings, vol. 12844. Springer Nature, 2021, p. 211.
17. J. Terven, D.-M. Cordova-Esparza, and J.-A. Romero-Gonz ´ alez, “A ´ comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolov9,” Machine Learning and Knowledge Extraction, vol. 5, no. 4, pp. 1680–1716, 2023.
18. H. R. Yudistira, L. Lina et al., “Human activity recognition from surveillance video recordings using the yolo method,” Journal of Computer Science and Information Systems, vol. 10, no. 1, 2022.
19. A. Anwar, "What is average precision in object detection & localization algorithms and how to calculate it?" towardsdatascience, retrieved, vol. 1, 2022.
20. S. L. H. Oktavian, “Application of Faster Regional–Convolutional Neural Network (Faster R–CNN) Method for Classification of Road Damage Types,” Ph.D. dissertation, Institut Teknologi Sepuluh Nopember, 2020.