**International Journal of Advances in Data and Information Systems**
Vol. 6, No. 3, October 2025, pp. 528~543
ISSN: 2721-3056, DOI: 10.59395/ijadis.v6i3.1391 

# Automated Oil Palm Health Assessment Using Object-Based Deep Learning and High-Resolution UAV Imagery in Indonesia

**Atut Pindarwati[1], Arie Wahyu Wijayanto[2], I Putu Agus Karmawan[3], Ardhan Yeza[3], Asriadi Sakka[4,5]**
[1]Department of Computer Science, University Indonesia, Indonesia
[2]Department of Statistical Computing, Politeknik Statistika STIS, Jakarta, Indonesia
[3]Department of Geospatial Planning & Development, Bumitama Gunajaya Agro, Jakarta, Indonesia
[4]Korea Institute of Energy Research (KIER), Daejeon, Republic of Korea
[5] Korea National University of Science and Technology (UST), Daejeon, Republic of Korea

**Article Info**
***Article history:***

Received May 21, 2025
Revised Jul 19, 2025
Accepted Oct 15, 2025

***Keywords:***
Palm oil health status,
Remote-sensing,
Precision agriculture,
Object-based deep learning,
Aerial images

**ABSTRACT**
Indonesia, as the world's largest crude palm oil (CPO) producer, faces challenges in plantation monitoring due to reliance on manual data collection methods that are time-consuming, costly, and prone to human error. This study proposes an automated approach for assessing oil palm tree health using high-resolution UAV imagery (5–10 cm) and object-based deep learning models. We evaluate five state-of-the-art detectors—YOLOv5s, Faster R-CNN, Mask R-CNN, SSD, and RetinaNet—to classify individual trees into four health categories: Healthy, Moderately Healthy, Needs Improvement, and Urgent Condition. Using a dataset of 14,749 labeled trees from Kendawangan, Indonesia, YOLOv5s achieved the highest performance with a precision of 0.784, recall of 0.752, and mAP of 0.764. Our findings demonstrate the potential of AI-driven monitoring to enhance plantation management through rapid, accurate, and cost-effective health assessments—contributing a scalable solution to support precision agriculture and sustainable CPO production.

*This is an open access article under the CC BY-SA license.*

***Corresponding Author:***
Arie Wahyu Wijayanto,
Department of Statistical Computing,
Politeknik Statistika STIS,
Jakarta, Indonesia.
Email: ariewahyu@stis.ac.id

## 1. INTRODUCTION

The plantation sector is a significant subsector of agriculture in Indonesia, contributing 3.94% to the total gross domestic product (GDP) in 2021 [1]. The dominant crops in this subsector are palm oil, cocoa, rubber, coffee, and sugarcane. Among these crops, palm oil has an advantage in terms of production cost, as it requires lower production costs than other commodities [1]. The oil palm (Elaeis guineensis) serves as a primary source of vegetable oil, extensively utilized in the food industry and beyond.(68%), industrial (27%), and bioenergy (5%) sectors. It also contributes to the Sustainable Development Goals (SDGs), particularly Goal 13 (climate action) through ecological impact, and Goal 1 (no poverty) via socio-economic development [2].

Indonesia ranks as the largest global producer of crude palm oil (CPO) within the agricultural industry, according to the United States Department of Agriculture (USDA) [3]. The amount of CPO production is influenced by the productive age of palm trees, which can reach more than 25 years. From an economic perspective, data on the age of palm trees can be used in precision agriculture, which focuses on developing agriculture to increase cultivation efficiency. The data collection still uses conventional survey methods, which have limitations in terms of cost, time, and human resources. Therefore, breakthroughs are needed by utilizing the technology that is currently developing, namely remote sensing. Remote sensing has been widely used in the field of agriculture and plantations to observe plants, such as research on estimating rice production [4]. The use of remote sensing can also be felt in various research fields, including land detection,

environmental analysis, poverty index analysis, forest fires, and object counting [5]. Some previous studies provide strong reasons for utilizing remote sensing technology in the form of UAVs and artificial intelligence, especially deep learning, in processing data to become a solution in estimating the health status of palm trees. The proposed detection model in this study can be applied and developed to support and improve Plantation Statistics in Indonesia, especially the Plantation Company Survey. For future research, other object-based deep learning models can be used to estimate the age and health status of palm trees, as well as utilizing satellite and UAV images with higher resolution to obtain better detection results [6].

The productivity of oil palm is directly influenced by the health condition of the trees; healthier oil palm trees generally produce a higher yield of fruit. Factors that can cause a decrease in the health of oil palm trees include low soil fertility, hilly topography, and poor drainage systems in peat areas. Spatial data is a type of data that contains location information in the form of coordinates, attribute information that contains explanations and characteristics of objects/phenomena, and temporal information related to the time of occurrence. Spatial data describes one or more specific phenomena. Remote sensing is a technology that is currently developing and has been widely used in the field of agriculture and plantations to observe plants, such as research on estimating rice production. The use of remote sensing can also be felt in various research fields, including land detection, environmental analysis, poverty index analysis, forest fires, and object counting. Some previous studies provide strong reasons for utilizing remote sensing technology in the form of UAVs and artificial intelligence, especially deep learning, in processing data to become a solution in estimating the health status of oil palm trees. The proposed detection model in this study can be applied and developed to support and improve Plantation Statistics in Indonesia, especially the Plantation Company Survey. For future research, other object-based deep learning models can be used to estimate the age and health status of oil palm trees, as well as utilizing satellite and UAV images with higher resolution to obtain better detection results.

This study focuses on a major oil palm company in Kalimantan, Indonesia, managing approximately 180,000 hectares of plantation land. In the past year, crude palm oil (CPO) production has increased by 23.36%, yet the health status of oil palm trees is still assessed manually—an approach that is time-consuming, labor-intensive, and costly. Given the scale of operations and the rising production demands, there is a critical need for automated, scalable monitoring systems. Timely identification of unhealthy trees enables rapid intervention, such as targeted fertilization, which can significantly enhance productivity. Automated monitoring offers a cost-effective solution to support this goal by improving decision-making efficiency and operational responsiveness.

Table 1. Primary Contribution of the Study Compared to the Existing Literature

| Relevant Studies | Purpose | Methods | Data Source | Data Type | Location |
|---|---|---|---|---|---|
| Nurmasari & Wijayanto, 2022 [7] | Detection Oil Palm Plantation in Indonesia | Decision Tree, Random Forest, Support Vector Machine, and Naive Bayes | Sentinel-2 and Landsat-8 | Medium Resolution Optical Satelite Imagery | Rokan Hulu, Riau, Indonesia |
| Zheng et al, 2021 [8] | Observation status of oil palm growth. | *Multi-class oil PAlm detection* (MOPAD) | Drone | Very High-Resolution | Kalimantan & Papua, Indonesia |
| Mubin et al., 2019 [9] | Detection of oil palm age | *Deep learning*: CNN | Drone | Very High Resolution | Selangor, Malaysia |
| Rizeei et al., 2018 [10] | Estimating the age of oil palms | *SVM (Support Vector Machine)* | WorldView-3 & LiDAR | Very High-Resolution | Selangor, Malaysia |
| This study | Detection of Oil Palm Tree Health | Object Based Deep Learning (YOLOv5s, Single Shot Detector, Retina-Net, Fast R-CNN, Mask R-CNN) | Drone/ UAV | Very High Resolution | Kalimantan, Indonesia |

To support this effort, Table 1 reviews recent advancements in oil palm monitoring using remote sensing and machine learning, including plantation detection (Nurmasari & Wijayanto, 2022), growth stage assessment (Zheng et al., 2021), and age estimation using deep learning and

high-resolution imagery (Mubin et al., 2019; Rizeei et al., 2018). Although those existing studies in Indonesia and Malaysia have utilized remote sensing to examine oil palm plantations, they have yet to address the critical aspect of assessing the health status of the oil palm trees. The primary contribution of this study lies in offering an application-oriented solution for classifying the health status of individual oil palm trees using high-resolution UAV imagery. This is achieved by benchmarking the performance of five state-of-the-art object-based deep learning models—YOLOv5s, SSD, RetinaNet, Fast R-CNN, and Mask R-CNN—within a real-world context in Kalimantan, Indonesia. Trees are categorized into four health levels, enabling precise and scalable monitoring. The results show that deep learning-based detection significantly improves intervention timing and reduces monitoring costs, supporting more efficient and sustainable plantation management. The proposed automated monitoring framework enhances plantation management by reducing operational costs and enabling more efficient, data-driven decision-making. The overall research framework is illustrated in Figure 1.

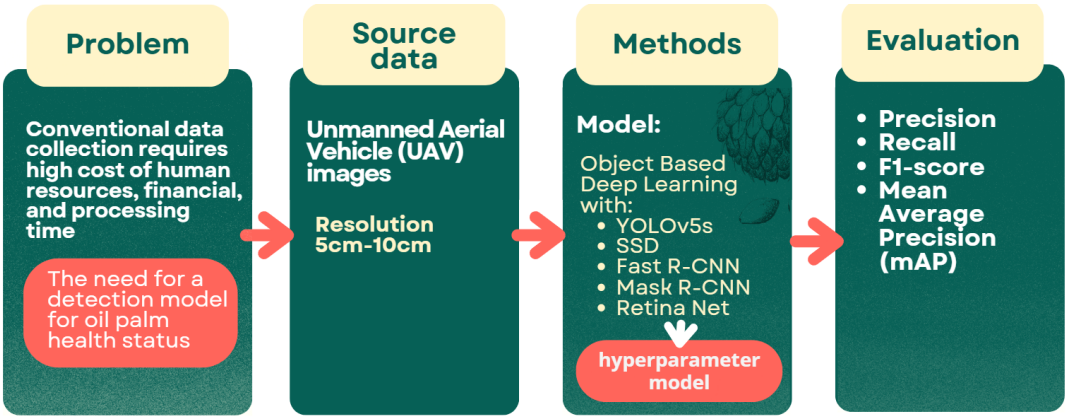



Figure 1. Proposed Research framework

## 2. RESEARCH METHOD

### *2.1 Data*

This study was conducted in Kendawangan, West Kalimantan, Indonesia. Four distinct classes represent the condition of oil palm trees: "Green," denoting very healthy and vibrant trees with lush green leaves; "Moderate Green," representing a state in which the oil palm trees' green leaves exhibit signs of decline, turning yellow; "Needs Improvement," signifying oil palm trees experiencing a decline in health and displaying yellow leaves while maintaining an overall healthy status; and "Urgent Health Condition," indicating trees undergoing a health decline with yellow leaves but still maintaining an overall healthy condition. These class labels served as essential criteria for evaluating and monitoring the condition of the oil palm plantations. Manual labeling was conducted by trained personnel from the Geospatial Information Division, equipped with standardized expertise in oil palm health assessment, under domain expert supervision. Label accuracy was validated using very high-resolution UAV imagery.

Table 2. List of the four predicted classes

| Class Type | Characteristic | Sample Size |
|---|---|---|
| Green | The oil palm trees are very healthy and green | 553 trees |
| Moderate Green | The oil palm trees, which green leaves are beginning to experience a decline (almost turning yellow) | 5,550 trees |
| Need Improvement | The oil palm tree, which is experiencing a decline in health, has turned yellow, but in terms of its overall condition, it's still healthy. | 6,126 trees |
| Urgent Health Condition | The oil palm tree, which has turned yellow and very yellow, has an unhealthy appearance (different from the others/stunted, dry) | 2,397 trees |

Figures 2 illustrates the comparison of the health status of oil palm trees as observed from an aerial view and compared to the ground-level perspective, respectively.




a) Health Status of Oil Palm tree (Aerial View)




b) Health Status of Oil Palm Tree (Ground View)

Figure 2. Health Status of Oil Palm Tree from Aerial and Grund View

At the study location, PT X Oil Palm Company collected data using a drone, capturing images during three distinct times: morning, afternoon, and evening. These different times of acquisition yielded various types of imagery, as depicted in Figure 3 below.



a) Image samples taken in the morning



b) Image samples taken in the afternoon or evening with cloudy interference



c) Broken Images

Figure 3. Unique Dataset

Our findings indicate that aerial imagery captured in the morning under optimal conditions—minimal cloud cover, clear skies, and stable weather—produces the most reliable results. This underscores the importance of selecting high-quality, artifact-free images for labeling and analysis. In this study, we utilized a dataset of 14,749 oil palm trees in Kendawangan, Indonesia, acquired via UAV imagery in March 2023 with a ground sampling resolution of 5–10 cm. These high-resolution images formed the basis for annotating trees into four health categories, serving as a critical foundation for the object-based deep learning models employed in our research.

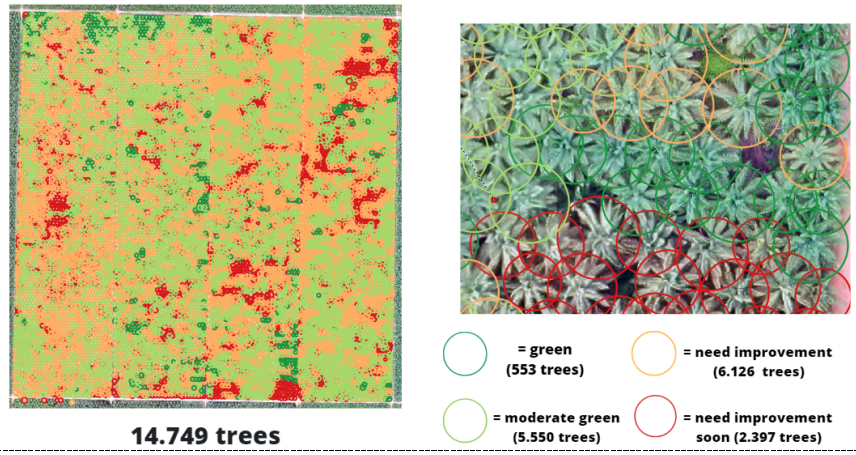



Figure 4. Unique Dataset

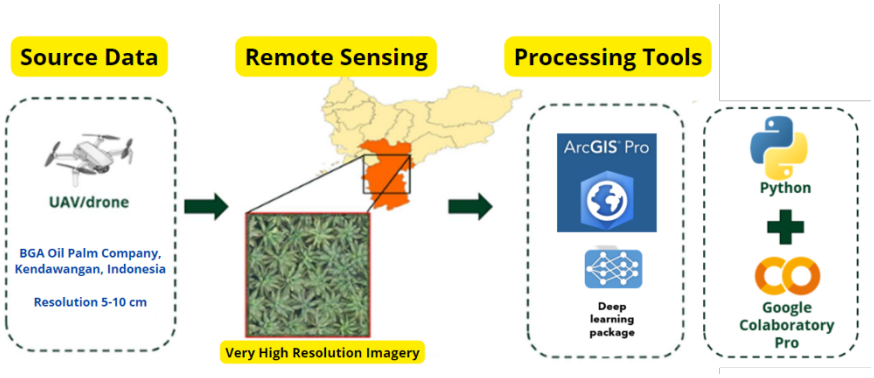



Figure 5. Research Framework

The research framework combines UAV (Unmanned Aerial Vehicle) technology with the object-based deep learning architecture for high-resolution imagery analysis. UAVs or drones equipped with high-resolution sensors (5 cm to 10 cm) serve as the data source, capturing detailed spatial imagery of targeted regions, such as agricultural fields or vegetation areas. The acquired images are processedusing Python programming and computational resources provided by Google Colab Pro. The research study compares five object-based deep learning methods, namely YOLOv5s, Single Shot Detector (SSD), Fast R-CNN, RetinaNet, and Mask R-CNN. These methods are used for object detection tasks. Several advanced methods are widely used for object detection tasks. Single Shot Detector (SSD) is a real-time, single-pass detector that directly predicts bounding boxes and class probabilities using a fully convolutional network [29]. Fast R-CNN, an enhancement over R-CNN, employs a Region Proposal Network (RPN) to generate candidate object regions and then extracts features from the entire image, improving both speed and accuracy. RetinaNet is a single-stage detector that addresses class imbalance—a major challenge in object detection—through the introduction of Focal Loss, and incorporates Feature Pyramid Networks (FPN) for multi-scale feature extraction [30]. Mask R-CNN extends Faster R-CNN by adding a parallel branch for segmentation masks, enabling high-performance instance and image segmentation [31]. This study aims to evaluate and compare the performance of these four models with respect to accuracy and inference speed.

### *2.2 You Only Look Once (YOLO) Detection Model*

The YOLOv5s model is employed for object detection and analysis. The framework begins by feeding high-resolution imagery into the model. The backbone of YOLOv5s extracts essential features using convolutional layers and spatial pyramid pooling (SPP), which facilitates hierarchical feature extraction. These features are further refined in the neck of the model through

upsampling, concatenation, and additional convolutional layers, ensuring robust detection across multiple scales. The head of the architecture predicts bounding boxes and object classifications at three scales (P3, P4, and P5), corresponding to different levels of detail. The final output is an annotated image with detected objects classified and localized within the frame. This framework is highly applicable in domains such as precision agriculture, environmental monitoring, and real-time object detection, leveraging the efficiency and accuracy of YOLOv5s.

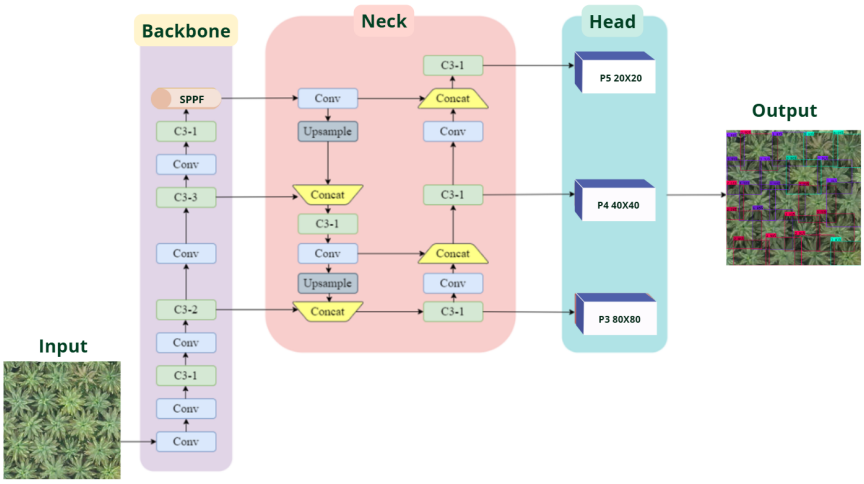



Figure 6. Architecture of YOLOv5s for object detection tasks

YOLOv5s extracts hierarchical features while minimizing computational redundancy via cross-stage partial (CSP) connections [32]. A PANet neck fuses multi-scale features (from 80×80 to 20×20 grid resolutions), improving the detection of small objects such as partially diseased crowns. The detection head outputs bounding boxes, confidence scores, and class probabilities using three anchor boxes, optimized for varying palm crown sizes.

$$L_{\{YOLO\}} = \lambda_{\{coord\}}\sum L_{\{CIoU\}} + \lambda_{\{obj\}}\sum L_{\{obj\}} + \lambda_{\{cls\}}\sum L_{\{cls\}}$$ [32].

- $L_{\{CIoU\}}$: Complete IoU loss, considering overlap, center distance, and aspect ratio.
- $L_{\{obj\}}$: Binary Cross-Entropy (BCE) loss for objectness score.
- $L_{\{cls\}}$: Binary Cross-Entropy (BCE) loss for class prediction.
- $\lambda_{\{coord\}}, \lambda_{\{obj\}}\lambda_{\{cls\}}\sum L_{\{cls\}}$: Loss weights to balance components.

Detection Steps

a) Input images (640×640) are divided into a grid.
b) Each cell predicts bounding boxes and class probabilities.
c) Low-confidence predictions (confidence < 0.5) are discarded.
d) Non-maximum suppression (NMS) is applied (IoU threshold = 0.45).

*2.3 Single Shot Detector (SSD) Detection Model*

Single Shot Detector (SSD) utilizes a truncated VGG16 backbone**, a** Visual Geometry Group **(VGG)** based convolutional neural networks (CNNs)**,** followed by six convolutional layers that generate feature maps at multiple scales (from 38×38 to 1×1) [33]. Each layer predicts class scores and bounding box offsets using default boxes with varying aspect ratios to accommodate diverse palm crown geometries.

$$L_{\{SSD\}} = \frac{1}{N}\left(\sum_i L_{\{loc\}}^{\{(i)\}} + \alpha\backslash sum_i L_{\{conf\}}^{\{(i)\}}\right)$$ [33]

- $L_{\{loc\}}^{\{(i)\}}$: Smooth L1 loss for bounding box regression.
- $L_{\{conf\}}^{\{(i)\}}$: Cross-entropy loss for class confidence scores.

Detection Steps

a) Input images are passed through the backbone and multi-scale feature extractors.
b) Default boxes are matched to ground truth (IoU ≥ 0.5).
c) Predictions with confidence ≥ 0.5 are retained and filtered via NMS.

*2.4 RetinaNet Detection Model*

RetinaNet is a detection model that integrates a ResNet-50 backbone with a Feature Pyramid Network (FPN) to capture features across five scales (P3–P7) [34]. Two subnets operate in parallel: a classification subnet with sigmoid activation and focal loss, and a regression subnet for bounding box adjustment.

$$L_{\{FL\}} = -\alpha(1 - p_t)^{\gamma \backslash log(p_t)} \text{ [34]}$$

- $p_t$: Predicted probability for the true class.
- $\alpha$: Class balancing factor (typically 0.25).
- $\gamma$: Focusing parameter to down-weight easy examples (typically 2).

Detection Steps

a) Anchors (5 scales × 3 aspect ratios) are assigned to ground truth boxes.
b) Features from each FPN level feed into the classification and regression subnets.
c) Outputs are filtered by confidence ≥ 0.5 and NMS is applied.

*2.5 Fast R-CNN*

Fast R-CNN uses a ResNet-50 backbone to extract dense feature maps from the entire image [35]. Region proposals (typically ~2,000) are generated via Selective Search and pooled into fixed-size (7×7) features using RoI Pooling. Fully connected layers then predict class probabilities and bounding box coordinates.

$$L_{\{Fast,R\backslash text\{-\}CNN\}} = L_{\{cls\}} + \lambda L_{\{loc\}} \text{ [35]}$$

- $L_{\{cls\}}$: Softmax cross-entropy for classification.
- $L_{\{loc\}}$: Smooth L1 loss for bounding box regression.

Detection Steps

a) Proposals are extracted from the input image.
b) Features from each RoI are pooled and passed through FC layers.
c) Outputs are filtered based on confidence ≥ 0.5 and refined via NMS.

*2.6 Mask R-CNN*

R-CNN extends Faster R-CNN by incorporating a segmentation branch and employing RoIAlign for high-precision feature extraction [36]. Using a ResNet-50-FPN backbone and a Region Proposal Network (RPN), the model performs multi-task learning with classification, localization, and instance segmentation.

$$L_{\{Mask,R\backslash text\{-\}CNN\}} = L_{\{cls\}} + L_{\{loc\}} + L_{\{mask\}} \text{ [36]}$$

- $L_{\{mask\}}$: Binary cross-entropy loss applied pixel-wise to the predicted segmentation mask.

Detection Steps

a) RPN generates candidate RoIs.
b) RoIAlign ensures precise spatial alignment.
c) Masks are generated per RoI and thresholded (≥ 0.5) to segment affected regions.
d)

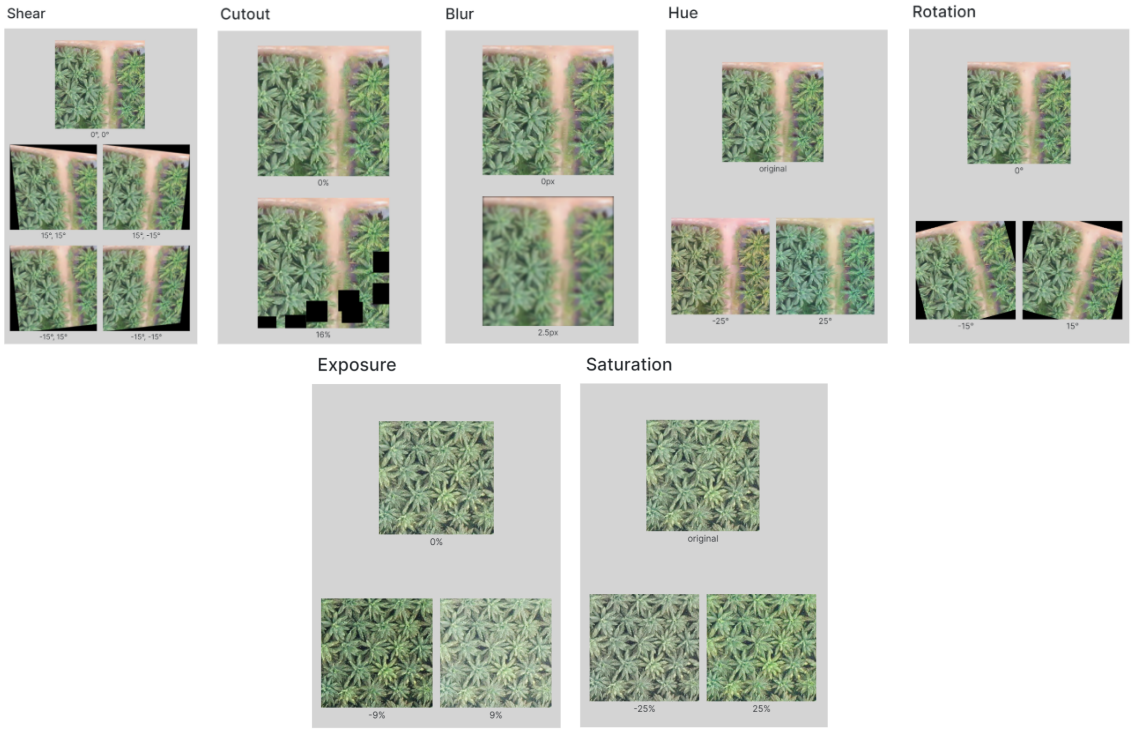
Table 3. Model Architecture Summary

| Model | Backbone | Neck/Enhancement | Detection Head | Key Innovation | Architecture Summary |
|---|---|---|---|---|---|
| **YOLOv5s** | CSPDarknet | PANet | Anchor-based (3 anchors) | Multi-scale fusion, CIoU loss | Input → CSPDarknet → PANet → Detection Head |
| **SSD** | VGG16 | Multi-scale convs | Default boxes (6 ratios) | Single-pass multi-scale detection | Input → VGG16 → Multi-Scale Feature Maps → Prediction Heads |
| **RetinaNet** | ResNet-50 + FPN | FPN | Focal Loss classification | Focal Loss for class imbalance | Input → ResNet-FPN → Classification + Regression Subnets |
| **Fast R-CNN** | ResNet-50 | RoI Pooling | FC layers | Region-based feature pooling | Input → ResNet → RoI Pooling → FC Layers |
| **Mask R-CNN** | ResNet-50-FPN | RoIAlign | Mask head (4 conv layers) | Pixel-wise mask prediction | Input → ResNet-FPN → RPN → RoIAlign → Class/BBox/Mask Heads |

To summarize the technical specifications and innovations of the evaluated models, Table 3 provides a comparative overview of their architectures, key components, and loss functions. This table highlights critical distinctions such as the use of CSPDarknet in YOLOv5s for computational efficiency, Focal Loss in RetinaNet to address class imbalance, and RoIAlign in Mask R-CNN for precise mask prediction. The backbone networks, detection strategies, and optimization objectives are systematically contrasted to underscore the strengths and limitations of each model in the context of oil palm health detection.

We applied data augmentation techniques to enhance the model robustness by increasing the diversity of the training dataset. Shear was used to tilt images along the x or y-axis, distorting them to create slanted perspectives and simulate different viewpoints. Cutout augmentation randomly masked or removed regions of the image, simulating occlusions to encourage the model to focus on unmasked areas. Blur, often applied using Gaussian blur, simulated out-of-focus conditions to improve the model's tolerance to image quality variations. Hue adjustment shifted the image colors within the hue spectrum to mimic lighting changes. Rotation was implemented by rotating images within a specific degree range, simulating various orientations. Additionally, we adjusted exposure by varying brightness levels to simulate dark and bright conditions, while saturation modification altered the intensity of colors, ranging from grayscale to highly vivid colors, to account for different lighting and color conditions. Flip augmentation was applied by mirroring images horizontally or vertically, creating variations in directional views.

Traditional rebalancing techniques such as Synthetic Minority Oversampling Technique (SMOTE) were not applied in this study due to their unsuitability for object detection tasks involving high-resolution UAV imagery. Unlike structured data, object detection relies on spatial and visual context, and generating synthetic samples may introduce unrealistic artifacts that degrade model performance. Instead, the study leveraged deep learning models with built-in mechanisms—such as data augmentation, class-balanced sampling, and optimized loss functions—to address class imbalance more effectively without compromising spatial integrity or generalization.



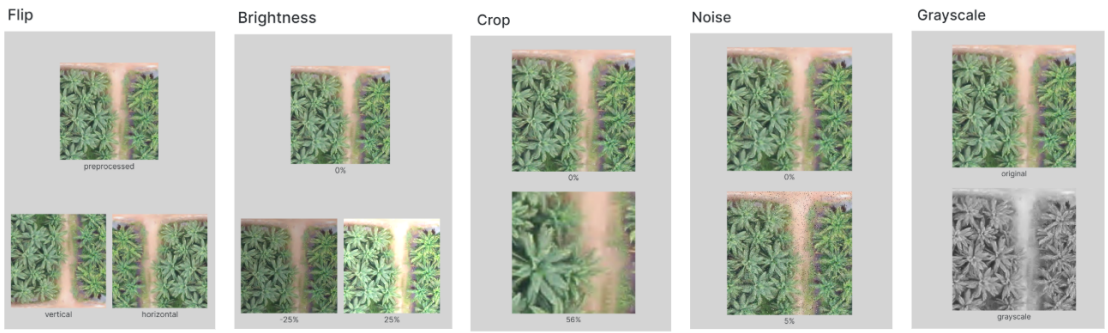



Figure 7. The various data augmentation techniques for Model YOLOv5s

Brightness adjustment modified the light levels of images uniformly to account for lighting inconsistencies. We also employed crop to extract specific sections of images, simulating zoomed-in perspectives or focusing on certain areas. Noise was added to simulate sensor imperfections or low-quality images, making the model more robust to real-world data. Finally, grayscale conversion was used to remove all color information, helping the model focus on texture and structural details in the absence of color. These augmentation techniques collectively improved the model's ability to generalize across diverse scenarios.

YOLO-based object detection models divide the input image into an S×S grid, with each cell responsible for predicting objects whose centers lie within it. Each cell outputs bounding boxes—defined by center coordinates, dimensions, and confidence scores—alongside class probabilities. During inference, low-confidence and poorly localized predictions are filtered using confidence thresholds and Intersection over Union (IoU) criteria. Non-Maximum Suppression (NMS) is then applied to eliminate redundant boxes, retaining only the most confident detections. This streamlined, single-pass architecture enables efficient and accurate real-time object detection.

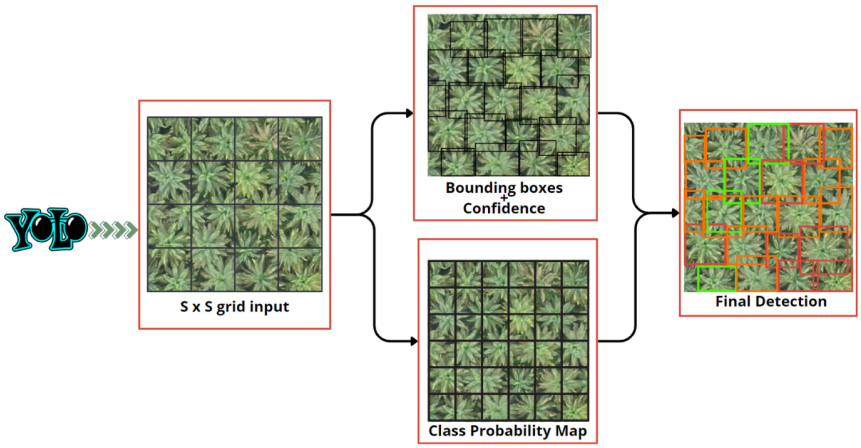



Figure 8. YOLO detection model

*2.7. Evaluation*

In object-based detection, key evaluation metrics include Precision, Recall, and F1-score [17]. In some cases, performance is assessed using only point-level associations—such as object centroids—without considering spatial overlap via Intersection over Union (IoU). This may limit boundary accuracy assessment. In this study, we used overall accuracy for global evaluation and the F1-score of the positive (building) class for classification performance. Comprehensive object detection evaluation typically includes Precision, Recall, F1-score, IoU, and Mean Average Precision (mAP). True positives, false positives, and false negatives are represented in the confusion matrix, as summarized in Table 3.

$$Precision = \frac{True\ Positives}{(True\ Positives+False\ Positives)} \quad (1)$$

$$Recall = \frac{True\ Positives}{(True\ Positives+False\ Negatives)} \quad (2)$$

$$F1-score = \frac{2\times Precision\times Recall}{(Precision+Recall)} \quad (3)$$

$$Mean\ Average\ Precision\ (mAP) = \frac{1}{N}\sum_{i=1}^{N}\int_{0}^{1} P_{i(r)dr} \quad (4)$$

Table 3. Confusion Matrix

| *Confusion Matrix* | | *Predicted Class (from Model)* | |
|---|---|---|---|
| | | *Positive* | *Negative* |
| *Actual Class (Manual Counting)* | *Positive* | True Postive | False Negative |
| | *Negative* | False Positive | True Negative |

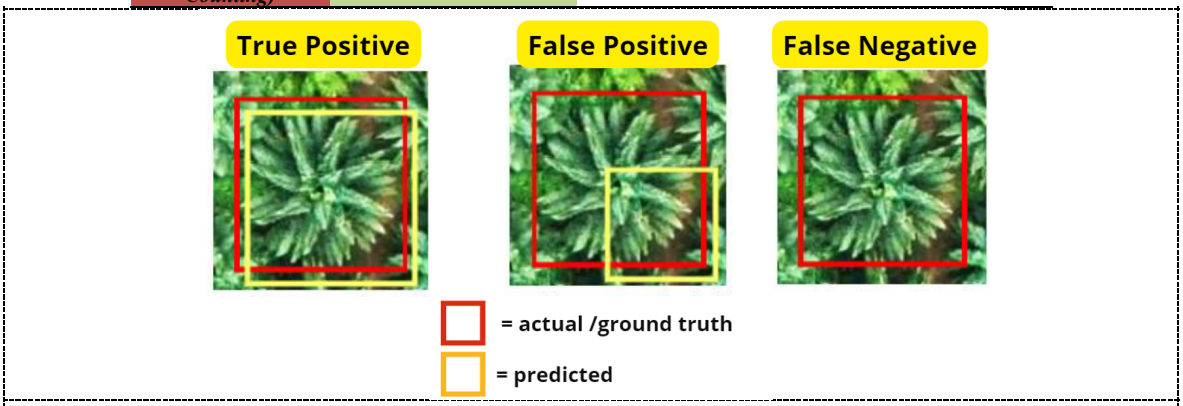



Figure 9. Illustrations of True positive, false positive, and false negative

True positives occur when the model correctly identifies the presence of a building object. False positives arise when the model incorrectly classifies a non-building object as a building, while false negatives refer to instances where a building is present but goes undetected by the model. This study focuses on mAP, precision, and recall, as they are the most informative and This study evaluates the performance of five object-based deep learning models for object detection: YOLOv5s, Single Shot Detector (SSD), Fast R-CNN, RetinaNet, and Mask R-CNN. Figure 10 presents the hyperparameter configurations used for Faster R-CNN, RetinaNet, SSD, and Mask R-CNN, including settings for maximum epochs, batch size, backbone architecture, and learning rate [29]. The maximum epoch defines the number of complete passes through the training dataset. Batch size denotes the number of samples processed before the model is updated. The backbone model serves as the feature extractor that underpins the network's detection capabilities. The learning rate is a critical hyperparameter that controls the step size during optimization to minimize the loss function. Additionally, Non-Maximum Suppression (NMS) is applied post-processing to eliminate redundant overlapping bounding boxes, retaining only the most confident detections.

## 3. RESULTS AND DISCUSSION

In this study, we compare four object-based deep learning models to detect the health status of oil palm trees. The models used were YOLOv5s, Faster R-CNN, RetinaNet, Single Shot Detector, and Mask R-CNN. Figures 11 and 12 illustrates the detection result of oil palm tree health. The evaluation results showed that YOLOv5s demonstrating strong generalization, achieving an overall precision of 0.784, recall of 0.752, mAP@50 of 0.764, and mAP@50-95 of 0.612, the RetinaNet model had the highest confidence value of 57.1%, with a training value of 14.729 and a testing value of 4.021, and there was no overlap. The Single Shot Detector model had the highest confidence value of 100%, but the model's performance was not good, with a training value of 14.729 and a testing value of 11.093. The Faster R-CNN model had overlapping, with the highest confidence value of 57.4%, a training value of 14.729, and a testing value of 28.111. The Mask R-CNN model received a low ranking compared to the other models because its performance was stuck at an average precision of only 0.37% even after hyperparameter setting at 60, 100, and

200 epochs. The study highlights that future research can enhance model accuracy by refining parameter settings—particularly the Intersection over Union (IoU), which quantifies the overlap between predicted and ground-truth bounding boxes relative to their combined area. Further improvements can be achieved by incorporating more diverse datasets and applying data augmentation techniques, which can enhance the model's generalizability and robustness across varying conditions.

standard metrics for object detection, effectively capturing both classification and localization performance. Metrics like per-class F1-scores and confusion matrices, while useful in classification tasks, are less applicable here due to the spatial nature of object detection and the complexity of overlapping bounding boxes. Future work may explore these for deeper class-specific insights.

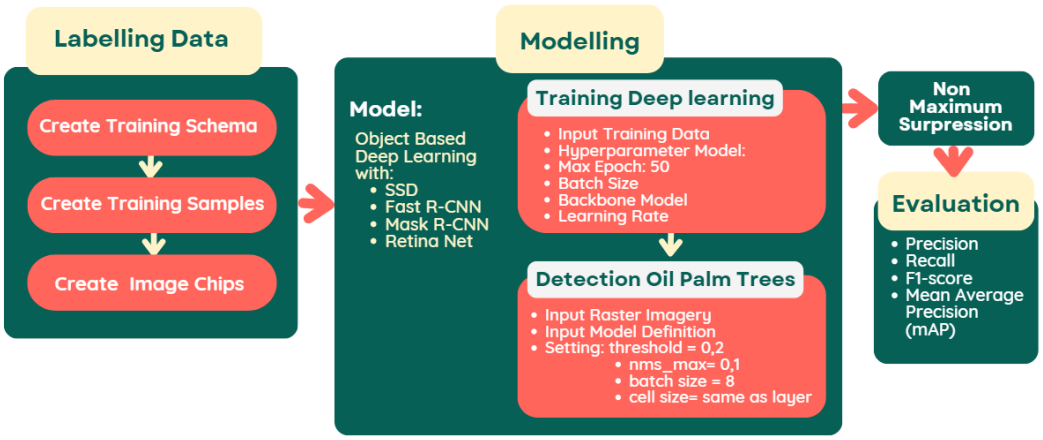



Figure 10. Processing Hyperparameter Model and Evaluation

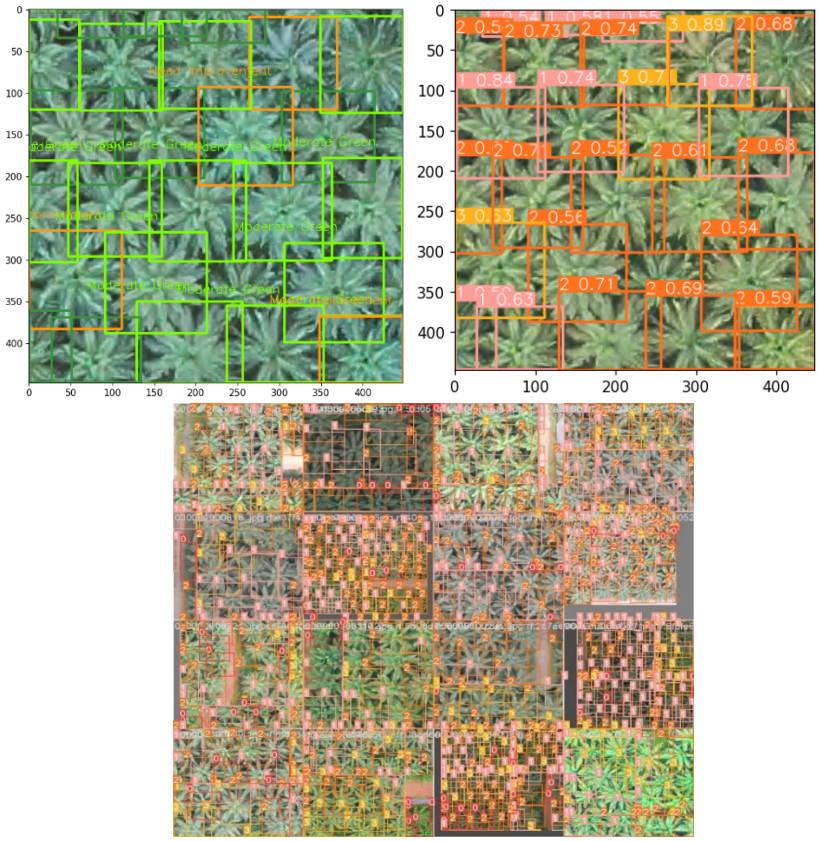


Figure 11. *Output of Result testing* model YOLOv5s in every condition

Several parameters were defined prior to building the YOLOv5s model to achieve the best model with optimal evaluation results. The model with the best evaluation performance was obtained after 200 iterations during the training process. The model with the highest evaluation performance was then used for the next stage, which is testing. The detailed parameter values for model development and the evaluation results can be seen in Table 4.

Table 4. Parameters For Model Training and Testing

| Parameter | Value |
|---|---|
| Image Size | 256*256 |
| Batch Size | 32 |
| Momentum | 0,937 |
| Iteration Training | 200 |
| Learning Rate | 0,01 |
| IoU Threshold | 0,50 |
| Confidence Threshold | 0,50 |

By applying an IoU threshold of 0.50 and a confidence threshold of 0.50, as shown in the table, the results were satisfactory, as the model successfully classified oil palm objects. The detection results included bounding boxes and confidence scores corresponding to each object class. The following figure illustrates the testing output of the model.

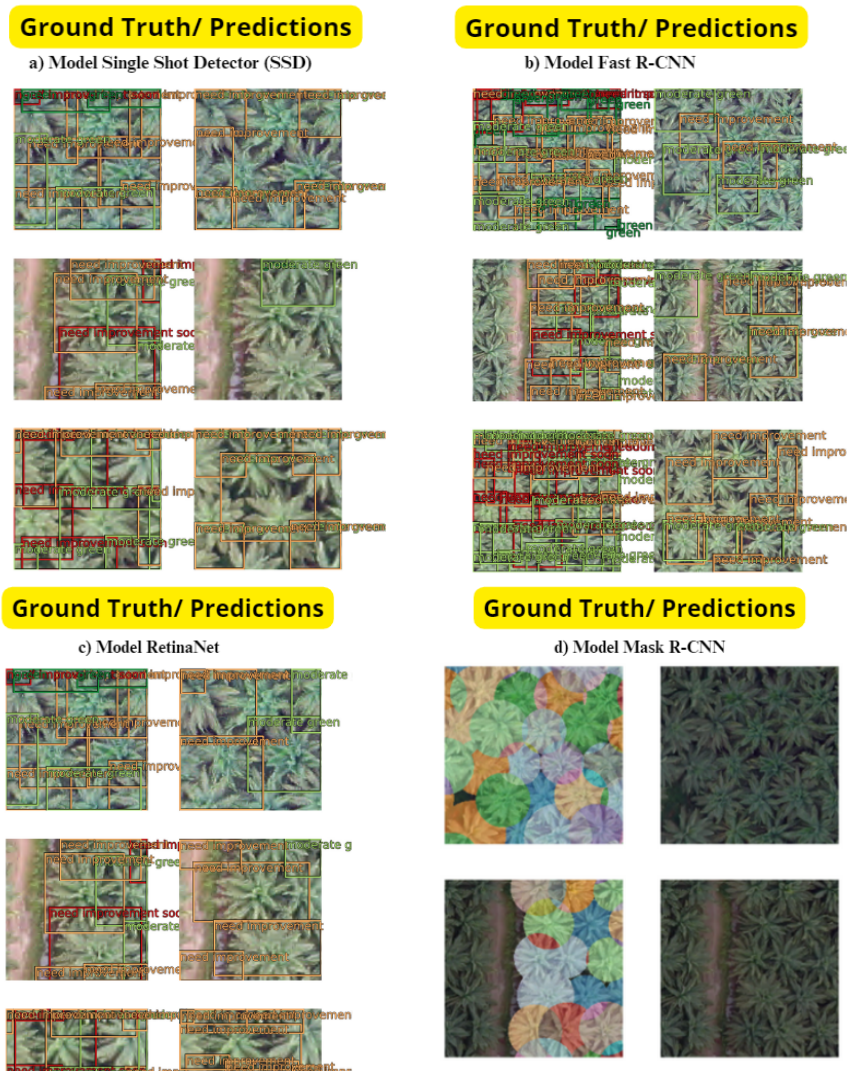



Figure 12. Illustrations Result of Detection Health Status of Oil Palm by Model

The YOLOv5s model's performance evaluation across four classes demonstrates its strong capability to classify diverse conditions accurately. The overall metrics indicate a precision (P) of 0.784**,** recall (R) of 0.752**,** mAP@50 of 0.764**,** and mAP@50-95 of 0.612**,** showcasing its robust generalization and effectiveness across all classes. The models used were Faster R-CNN, RetinaNet, Single Shot Detector, and Mask R-CNN. The evaluation results showed that the RetinaNet model had the highest confidence value of 57.1%, with a training value of 14.729 and a testing value of 4.021, and there was no overlap.

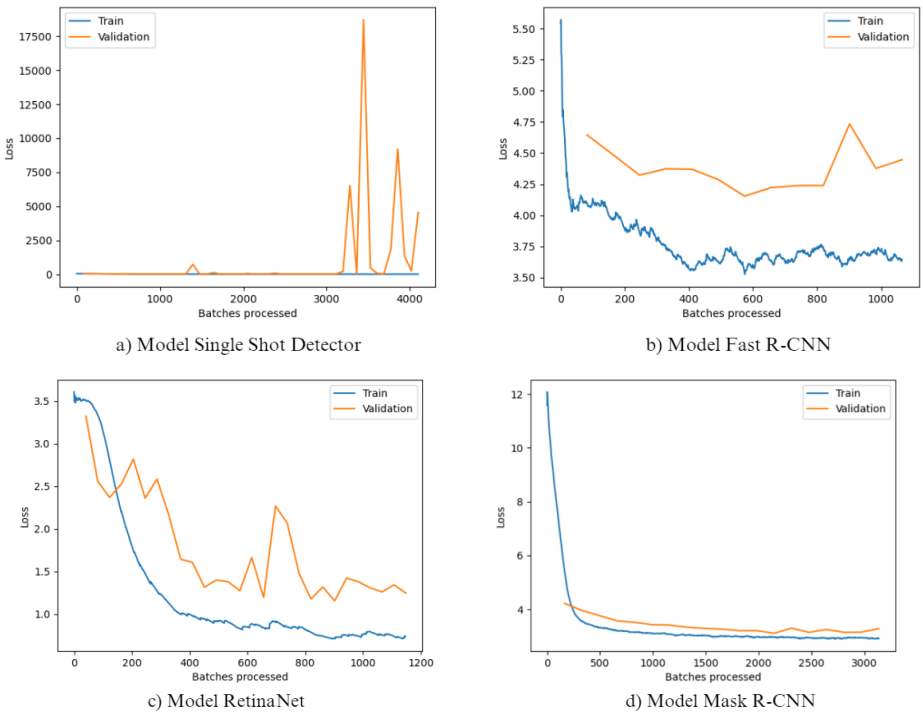



Figure 13. The graphical representation of the processing of the four models using ArcGIS

The Faster R-CNN model had overlapping, with the highest confidence value of 57.4%, a training value of 14.729, and a testing value of 28.111. The Single Shot Detector model had the highest confidence value of 100%, but the model's performance was not good, with a training value of 14.729 and a testing value of 11.093. The Mask R-CNN model received a low ranking compared to the other models because its performance was stuck at an average precision of only 0.37% even after hyperparameter setting at 60, 100, and 200 epochs. Based on the evaluation results, the RetinaNet and Faster R-CNN models performed better than the Single Shot Detector and Mask R-CNN models. The study suggests that future research can improve the model's accuracy by enhancing the Parameter Model Determination. The method offers significant potential for enhancement through the inclusion of more diverse datasets and the application of data augmentation strategies, which can improve the model’s robustness and overall detection accuracy.

Table 5. Summary of Model's performance in Comparing all the Methods

| No | Model | Precision | Recall | F-1 Score | Average Precision |
|---|---|---|---|---|---|
| 1. | YOLOv5s | 78.4% | 75,2% | - | 76,4% |
| 2. | Single Shot Detector | 16% | 12,6% | 14,4% | 42,9% |
| 3. | Retina-Net | 46% | 12,8% | 20,2% | 67% |
| 4. | Faster R-CNN | 26% | 51% | 34,9% | 19,2% |
| 5. | Mask R-CNN | | - | | 0,37% |

The study also includes experiments on the model parameters to assess their performance. The proposed method is applicable to other locations and has the potential for rapid and accurate estimation of oil palm tree health, leading to appropriate treatments. The implementation of automated monitoring can help reduce costs and improve plantation management efficiency. The results show promising performance, with YOLOv5s demonstrating strong generalization, achieving an overall precision of 0.784, recall of 0.752, mAP@50 of 0.764, and mAP@50-95 of 0.612. The RetinaNet model achieved an average precision of 67%. By refining model parameters using Intersection of Union (IoU) and incorporating more diverse datasets and data augmentation techniques, the accuracy of the models can be further improved.

In summary, while YOLOv5s succeeded due to its balance of speed, optimized architecture, and effective default configurations, the other models—RetinaNet, Faster R-CNN, Single Shot Detector (SSD), and Mask R-CNN—underperformed primarily due to poor generalization, overfitting, suboptimal hyperparameter settings, and sensitivity to class imbalance or dataset limitations. Future work involving enhanced data augmentation, more diverse training datasets, and refined parameter tuning could significantly improve the performance of these models.

## 4. CONCLUSION

In this study, we propose an object-based deep learning framework to automatically assess the health status of oil palm trees using high-resolution drone imagery. The primary contribution lies in the comparative evaluation of five state-of-the-art object detection models—YOLOv5s, Single Shot Detector (SSD), RetinaNet, Faster R-CNN, and Mask R-CNN—for multi-class classification of individual trees into four health categories: Healthy (Green), Moderately Healthy (Moderate Green), Needs Improvement, and Urgent Health Condition. High-resolution UAV imagery enables fine-grained detection, while the deep learning models automate the classification process, significantly reducing the need for manual intervention. Among the evaluated models, YOLOv5s demonstrated robust performance and generalization capability. This approach offers a scalable and efficient solution for precision agriculture, with potential to enhance plantation monitoring, decision-making, and resource allocation.

Each object detection model demonstrates unique advantages tailored to specific detection challenges. Faster R-CNN consistently delivers superior accuracy, SSD offers an optimal trade-off between speed and performance, and RetinaNet performs exceptionally well in identifying small objects. This study presents a practical framework for accurately and efficiently assessing the health status of oil palm trees using object-based deep learning, contributing to more informed and timely plantation management decisions. The approach is scalable and adaptable, with strong potential for application across other geographic regions. Furthermore, model performance can be further enhanced through the integration of more diverse datasets and advanced data augmentation strategies, supporting broader generalization and robustness in real-world deployment.

## REFERENCES

[1] BPS - Statistics Indonesia, "Statistik Kelapa Sawit Indonesia 2021," 2021. [Online]. Available: https://www.bps.go.id/publication/2020/11/30/36cba77a73179202def4ba14/statistik-kelapa-sawit-indonesia-2019.html.

[2] E. Agasta, "Prediksi Jumlah Produksi Kelapa Sawit Dengan Menggunakan Metode Extreme Learning Machine (ELM) (Studi kasus: PT. Sandabi Indah Lestari Kota Bengkulu)," J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 2, no. 11, pp. 5751–5759, 2018.

[3] Y. C. Putra, A. W. Wijayanto, and G. A. Chulafak, "Oil palm trees detection and counting on Microsoft Bing Maps Very High Resolution (VHR) satellite imagery and Unmanned Aerial Vehicles (UAV) data using image processing thresholding approach," Ecol. Inform., 2022.

[4] Y. C. Putra and A. W. Wijayanto, "Automatic detection and counting of oil palm trees using remote sensing and object-based deep learning," Remote Sens. Appl. Soc. Environ., vol. 29, no. July 2022, p. 100914, 2023, doi: 10.1016/j.rsase.2022.100914.

[5] A. Syahza, "Effort to Spur Economic Growth in Rural Areas," Int. Reseacrh J. Bus. Stud., vol. 4, no. 3, pp. 171–188, 2011.
[6] D. Djawardi, "Kelayakan Perkebunan Kelapa Sawit Di Papua," J. Sains dan Teknol. Indones., vol. 11, no. 3, pp. 199–204, 2009, doi: 10.29122/jsti.v11i3.833.
[7] Y. Nurmasari and A. W. Wijayanto, "Oil Palm Plantation Detection in Indonesia using Sentinel-2 and Landsat-8 Optical Satellite Imagery (Case Study: Rokan Hulu Regency, Riau Province)," Int. J. Remote Sens. Earth Sci., pp. 1–18, 2021, [Online]. Available: http://dx.doi.org/10.30536/j.ijreses.2021.v18.a3537.
[8] J. Zheng et al., "Growing status observation for oil palm trees using Unmanned Aerial Vehicle (UAV) images," ISPRS J. Photogramm. Remote Sens., vol. 173, no. August 2020, pp. 95–121, 2021, doi: 10.1016/j.isprsjprs.2021.01.008.
[9] N. A. Mubin, E. Nadarajoo, H. Z. M. Shafri, and A. Hamedianfar, "Young and mature oil palm tree detection and counting using convolutional neural network deep learning method," Int. J. Remote Sens., vol. 40, no. 19, pp. 7500–7515, 2019, doi: 10.1080/01431161.2019.1569282.
[10] H. M. Rizeei, H. Z. M. Shafri, M. A. Mohamoud, B. Pradhan, and B. Kalantar, "Oil palm counting and age estimation from WorldView-3 imagery and LiDAR data using an integrated OBIA height model and regression analysis," J. Sensors, vol. 2018, 2018, doi: 10.1155/2018/2536327.
[11] S. R. Putri and A. W. Wijayanto, "Estimating Rice Production using Machine Learning Models on Multitemporal Landsat-8 Satellite Images (Case Study: Ngawi Regency, East Java, Indonesia)," IEEE Int. Conf. Cybern. Comput. Intell., vol. 18, no. 1, pp. 280–285, 2022, doi: 10.1109/CyberneticsCom55287.2022.9865364.
[12] A. W. Wijayanto, N. Afira, and W. Nurkarim, "Machine Learning Approaches using Satellite Data for Oil Palm Area Detection in Pekanbaru City, Riau," Mach. Learn. Approaches using Satell. Data Oil Palm Area Detect. Pekanbaru City, Riau, pp. 84–89, 2022, doi: 10.1109/CyberneticsCom55287.2022.9865301.
[13] Caesarendra, R. Kurniawan, A. M. W. Saputra, and A. W. Wijayanto, "Eco-environment vulnerability assessment using remote sensing approach in East Kalimantan, Indonesia," Remote Sens. Appl., vol. v. 27, pp. 100791--2022 v.27, 2022, doi: 10.1016/j.rsase.2022.100791.
[14] K. Aprianto, A. W. Wijayanto, and S. Pramana, "Deep Learning Approach using Satellite Imagery Data for Poverty Analysis in Banten, Indonesia," in 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), 2022, pp. 126–131, doi: 10.1109/CyberneticsCom55287.2022.9865480.
[15] S. R. Putri, A. W. Wijayanto, and A. D. Sakti, "Developing Relative Spatial Poverty Index Using Integrated Remote Sensing and Geospatial Big Data Approach: A Case Study of East Java, Indonesia," ISPRS Int. J. Geo-Information, vol. 11, no. 5, 2022, doi: 10.3390/ijgi11050275.
[16] S. R. Putri, A. W. Wijayanto, and S. Pramana, "Multi-source satellite imagery and point of interest data for poverty mapping in East Java, Indonesia: Machine learning and deep learning approaches," Remote Sens. Appl. Soc. Environ., vol. 29, p. 100889, 2023, doi: https://doi.org/10.1016/j.rsase.2022.100889.
[17] N. Afira and A. W. Wijayanto, "Mono-temporal and multi-temporal approaches for burnt area detection using Sentinel-2 satellite imagery (a case study of Rokan Hilir Regency, Indonesia)," Ecol. Inform., vol. 69, p. 101677, 2022, doi: https://doi.org/10.1016/j.ecoinf.2022.101677.
[18] W. Nurkarim and A. W. Wijayanto, "Building footprint extraction and counting on very high-resolution satellite imagery using object detection deep learning framework," Earth Sci. Informatics, no. 0123456789, 2022, doi: 10.1007/s12145-022-00895-4.
[19] A. C. Fitrianto, A. Darmawan, K. Tokimatsu, and M. Sufwandika, "Estimating the age of oil palm trees using remote sensing technique," IOP Conf. Ser. Earth Environ. Sci., vol. 148, no. 1, 2018, doi: 10.1088/1755-1315/148/1/012020.

[20] A. Tridawati, S. Darmawan, and A. Armijon, "Estimation the oil palm age based on optical remote sensing image in Landak Regency, West Kalimantan Indonesia," IOP Conf. Ser. Earth Environ. Sci., vol. 169, no. 1, 2018, doi: 10.1088/1755-1315/169/1/012063.
[21] A. Chemura, I. van Duren, and L. M. van Leeuwen, "Determination of the age of oil palm from crown projection area detected from WorldView-2 multispectral remote sensing data: The case of Ejisu-Juaben district, Ghana," ISPRS J. Photogramm. Remote Sens., vol. 100, pp. 118–127, 2015, doi: 10.1016/j.isprsjprs.2014.07.013.
[22] I. Carolita, J. Sitorus, J. Manalu, and D. Wiratmoko, "Growth Profile Analysis of Oil Palm By Using Spot 6 the Case of North Sumatra," Int. J. Remote Sens. Earth Sci., vol. 12, no. 1, p. 21, 2017, doi: 10.30536/j.ijreses.2015.v12.a2669.
[23] N. Kamiran and M. L. R. Sarker, "Exploring the potential of high resolution remote sensing data for mapping vegetation and the age groups of oil palm plantation," IOP Conf. Ser. Earth Environ. Sci., vol. 18, no. 1, 2014, doi: 10.1088/1755-1315/18/1/012181.
[24] C. S. Hamsa, K. D. Kanniah, F. M. Muharam, N. H. Idris, Z. Abdullah, and L. Mohamed, "Textural measures for estimating oil palm age," Int. J. Remote Sens., vol. 40, no. 19, pp. 7516–7537, 2019, doi: 10.1080/01431161.2018.1530813.
[25] K. P. Tan, K. D. Kanniah, and A. P. Cracknell, "Use of UK-DMC 2 and ALOS PALSAR for studying the age of oil palm trees in southern peninsular Malaysia," Int. J. Remote Sens., vol. 34, no. 20, pp. 7424–7446, 2013, doi: 10.1080/01431161.2013.822601.
[26] I. H. Al Amin and F. H. Arby, "Implementation of YOLO-v5 for a Real Time Social Distancing Detection," J. Appl. Informatics Comput., vol. 6, no. 1, pp. 01–06, 2022, doi: 10.30871/jaic.v6i1.3484.
[27] J. Bai, J. Dai, Z. Wang, and S. Yang, "A detection method of the rescue targets in the marine casualty based on improved YOLOv5s," Front. Neurorobot., vol. 16, no. November, pp. 1–14, 2022, doi: 10.3389/fnbot.2022.1053124.
[28] A. Ma'ruf, "Materi Kelapa Sawit 3: Pemeliharaan Tanaman," no. June. 2018. [Online]. Available: https://www.researchgate.net/publication/325961511.
[29] A. Aslam, A. Irtaza, and N. Nida, "Object detection and localization in natural scenes through single-step and two-step models," *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, 2020. doi:10.1109/icetst49965.2020.9080728
[30] B. Jaison, A. J. G, J. J, and D. P. C, "You only look once(yol O) object detection with coco using machine learning," *2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC)*, 2022. doi:10.1109/iihc55949.2022.10059737
[31] B. Sun, Y. Wang, and S. Wu, "An efficient lightweight CNN model for real-time fire smoke detection," *Journal of Real-Time Image Processing*, vol. 20, no. 4, 2023. doi:10.1007/s11554-023-01331-6
[32] Ultralytics, "YOLOv5 Documentation," 2021. [Online]. Available: https://github.com/ultralytics/yolov5. [Accessed: 10-Oct-2024].
[33] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in Proc. European Conference on Computer Vision (ECCV), Amsterdam, Netherlands, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.
[34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in Proc. IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.324.
[35] R. Girshick, "Fast R-CNN," in Proc. IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.
[36] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in Proc. IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2961–2969, doi: 10.1109/ICCV.2017.322.