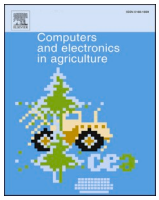
Computers and Electronics in Agriculture 227 (2024) 109530

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag



Oil palm tree detection in UAV imagery using an enhanced RetinaNet

Sheng Siang Lee [a,b], Lam Ghai Lim [c,*], Shivakumara Palaiahnakote [d], Jin Xi Cheong [b], Serene Sow Mun Lock [e,f], Mohamad Nizam Bin Ayub [a,**]

[a] Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[b] Aonic Sdn. Bhd., 9, Jalan TP 6, Taman Perindustrian UEP, 47600 Subang Jaya, Selangor, Malaysia
[c] Department of Electrical and Robotics Engineering, School of Engineering, Monash University Malaysia, Jalan Lagoon Selatan, 47500 Bandar Sunway, Selangor, Malaysia
[d] School of Science, Engineering and Environment, University of Salford, UK
[e] Department of Chemical Engineering, Universiti Teknologi PETRONAS, 32610 Seri Iskandar, Perak Darul Ridzuan, Malaysia
[f] Centre of Carbon Capture, Utilisation and Storage (CCCUS), Universiti Teknologi PETRONAS, 32610 Seri Iskandar, Perak Darul Ridzuan, Malaysia

ARTICLE INFO

Keywords:
Convolutional neural network
Deep learning
Object detection
Oil palm tree
Unmanned aerial vehicle

ABSTRACT

Accurate inventory management of oil palm trees is crucial for optimizing yield and monitoring the health and growth of plantations. However, detecting and counting oil palm trees, particularly young trees that blend into complex environments, presents significant challenges for deep learning models. While current methods perform well in detecting mature oil palm trees, they often struggle to generalize across the diverse variations found in both young and mature trees. In this study, we propose an enhanced RetinaNet model that incorporates deformable convolutions into the ResNet-50 backbone, deeper feature pyramid layers, and an intersection-over-union-aware branch in a multi-head configuration to improve detection performance. The model was evaluated using a diverse dataset of unmanned aerial vehicle imagery from multiple regions, encompassing oil palm and coconut trees, as well as banana plants. To refine detection, confidence thresholding and non-maximum suppression were applied during inference, filtering out low-confidence predictions and eliminating duplicate detections. Experimental results demonstrate that our method outperforms state-of-the-art models, achieving F1-scores of 0.947 and 0.902 for single- and dual-species detection tasks, respectively, surpassing existing approaches by 1.5–6.3%. These findings highlight the model's ability to accurately detect oil palm trees, particularly young ones in complex backgrounds, offering a reliable solution to support sustainable agriculture and improved land management.

1. Introduction

Oil palm tree cultivation is a cornerstone of the global agricultural economy, playing a pivotal role in the prosperity of many tropical countries, particularly Malaysia and Indonesia (Quezada et al., 2019; Taheripour et al., 2019). Accurate and efficient detection of oil palm trees is critical for monitoring plantation health, optimizing yields, and promoting sustainable agricultural practices. Traditional monitoring methods, such as manual counting, are not only labor-intensive but often lack the precision required for effective management, especially on large plantations. Recent advancements in unmanned aerial vehicle (UAV) technology, combined with the growing capabilities of computer vision, have revolutionized agricultural monitoring (Bouguettaya et al., 2022; Boursianis et al., 2022; Wakchaure et al., 2023). For instance, high-resolution UAV imagery, when analyzed using machine learning or deep learning techniques, enables accurate estimation of crop biomass, such as in the case of potato fields (Liu et al., 2024a; Liu et al., 2024b; Liu et al., 2023; Liu et al., 2024c).

The detection of oil palm trees is particularly challenging due to their varying growth stages, which can be classified into four main categories. In the seedling stage (age < 3 months), young oil palms develop their first leaves under controlled nurtured conditions. As saplings (ages 3–12

* Corresponding author at: Department of Electrical and Robotics Engineering, School of Engineering, Monash University Malaysia, Jalan Lagoon Selatan, 47500 Bandar Sunway, Selangor, Malaysia.
** Corresponding author at: Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia.
E-mail addresses: lim.lamghai@monash.edu (L.G. Lim), nizam_ayub@um.edu.my (M.N.B. Ayub).

https://doi.org/10.1016/j.compag.2024.109530
Received 15 July 2024; Received in revised form 30 September 2024; Accepted 5 October 2024
Available online 11 October 2024

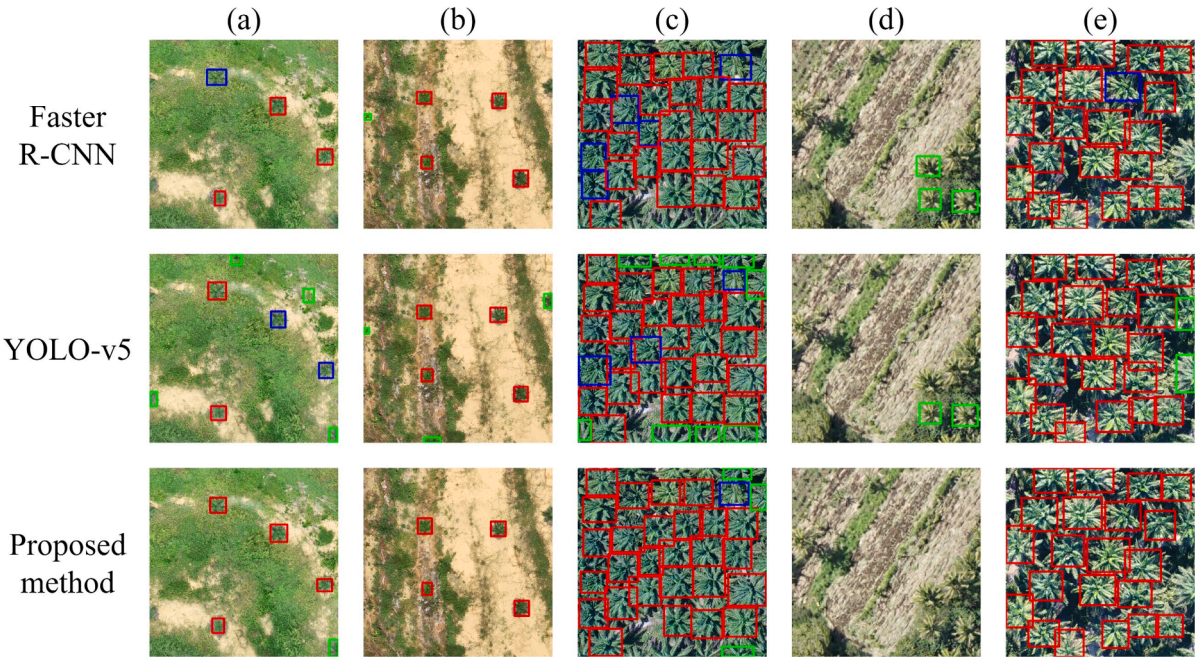


Fig. 1. The performance of Faster R-CNN, YOLO-v5 and our proposed method in detecting oil palm trees under various challenging conditions: (a) young trees blended into vegetation backgrounds, (b) young trees blended into varying soil backgrounds, (c) densely overlapped trees, (d) presence of coconut trees, and (e) influenced by weather conditions. Red, blue, and green bounding boxes represent correct detections, missed detections, and incorrect detections of oil palm trees, respectively.

months), they are typically transplanted into the field, showing robust growth in fronds and root systems. The young tree stage (ages 1–3 years) involves substantial growth in both height and canopy, but these trees are still not fully productive. Finally, mature trees (age > 3 years) enter the productive phase, forming well-defined trunks and extensive canopies, which produce fresh fruit bunches.

Despite progress in deep learning-based object detection models, identifying oil palm trees across diverse environments remains a formidable challenge. Oil palm trees vary significantly in size and shape based on age and environmental conditions. Mature trees feature large and overlapping canopies, while younger trees present fewer spatial features, making them more difficult to detect, especially when set against complex backgrounds, such as mixed vegetation (Fig. 1(a)) or varied soil textures (Fig. 1(b)). In densely planted areas (Fig. 1(c)), overlapping canopies further complicate detection tasks. Additionally, plantations often contain a mixture of oil palm and other species, such as coconut trees and banana plants (Fig. 1(d)), requiring models to discern subtle differences in crown shape.

Moreover, UAV-captured imagery introduces additional challenges due to variability in angle, altitude, lighting, and weather conditions (Fig. 1(e)). Oblique camera angles and varying flight altitudes distort the appearance of trees, while changes in lighting—resulting from weather or time of day—impact the visibility of critical features. These variations, coupled with imbalances in training datasets, often bias models towards detecting more common features, compromising their ability to identify less distinctive trees. These factors underscore the limitations of current deep learning models, which frequently struggle to generalize effectively across the diverse conditions encountered in oil palm plantations, leading to missed detections or false positives.

In this study, we propose a novel detection method aimed at accurately identifying both young and mature oil palm trees across diverse complexities in UAV imagery. Our main contributions are as follows:

- We introduce an enhanced version of RetinaNet, specifically designed to capture subtle variations in UAV imagery, with a focus on detecting young oil palm trees amidst complex backgrounds. The architecture integrates deformable convolutions, deeper feature pyramid levels, and an intersection-over-union (IoU)-aware branch, improving its adaptability to geometric variations and complex visual features.
- During inference, we apply advanced post-processing methods, including confidence thresholding and non-maximum suppression (NMS), to filter out low-confidence predictions and eliminate duplicate bounding boxes, resulting in significantly improved detection performance.
- We conduct a comprehensive evaluation of state-of-the-art object detection models, including faster region-based convolutional neural networks (R-CNN) and you only look once (YOLO), utilizing datasets that encompass a wide array of variations in both young and mature oil palm trees, as well as multi-species detection tasks involving coconut trees and banana plants. To our knowledge, this study represents the first direct comparison of these advanced models for oil palm tree detection. The findings provide critical insights into the relative strengths and limitations of each model, offering a robust framework for future research in precision agriculture and automated tree detection.

The remainder of this paper is organized as follows: Section 2 reviews existing deep-learning models for oil palm tree detection in satellite and UAV imagery. Section 3 describes the proposed method for tree detection. Section 4 presents the experimental results and validation, while Section 5 concludes the study with insights and future directions.

2. Related work

Over the past decades, numerous deep learning-based object detection models, primarily leveraging convolutional neural networks (CNNs) as their backbone, have been explored for detecting tree crowns using remote sensing imagery (Bouguettaya et al., 2022). Initial research focused on satellite images due to their wide area coverage and lower operational costs. Li et al. applied a one-stage CNN for detecting oil palm

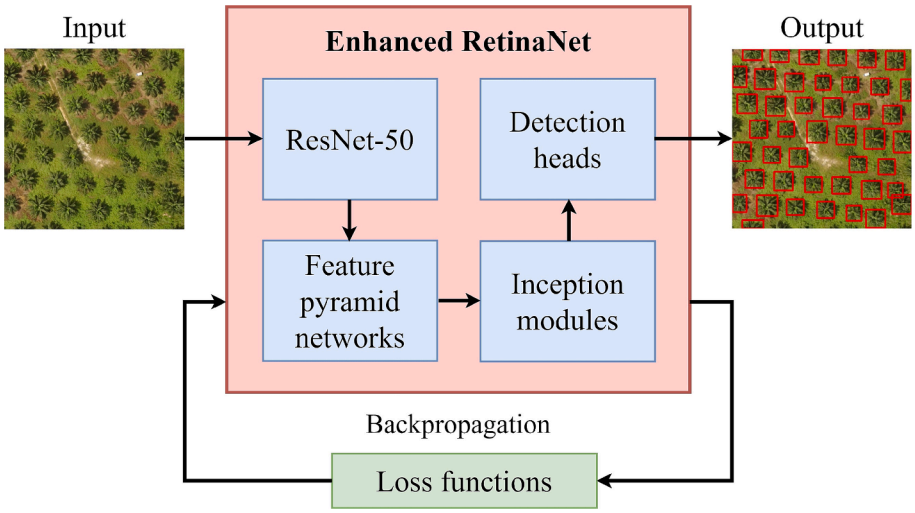


Fig. 2. The workflow of the proposed method.

trees (Li et al., 2016) and later improved detection performance by employing a two-stage CNN (Li et al., 2018). Freudenberg et al. explored the use of U-Net for oil palm tree detection in satellite imagery (Freudenberg et al., 2019), while Mubin et al. utilized LeNet to classify young and mature oil palm trees (Mubin et al., 2019). Although these methods achieved some success, the extracted features proved insufficient to address the full range of challenges posed by the low resolution of satellite imagery.

In recent years, research on oil palm tree detection has increasingly focused on UAV imagery due to its high resolution. Liu et al. implemented Faster R-CNN, which generates region proposals likely to contain oil palm trees in UAV imagery (Liu et al., 2021). The multi-head configuration of Faster R-CNN facilitates the classification of these proposals and the refinement of their bounding boxes in a separate stage. Zheng et al. proposed an enhanced detection method based on Faster R-CNN by incorporating a refined pyramid feature module and a hybrid class-balanced loss module (Zheng et al., 2021). Using an extensive dataset comprising nearly 300,000 oil palm trees, they detected individual trees to assess various growth statuses, such as healthy, dead, mismanaged, smallish, and yellowish. In addition, Yarak et al. reported detection errors arising from the presence of coconut and other trees with physical characteristics similar to those of oil palm trees, highlighting the challenges in detecting young oil palm trees with smaller crowns (Yarak et al., 2021). Leveraging CNNs as the backbone, UAV imagery has also been used for the detection of citrus (Osco et al., 2020), forest (Arce et al., 2021; Beloiu et al., 2023; Miyoshi et al., 2020; Onishi and Ise, 2021), apple (Wu et al., 2020a), Amazonian palm (Ferreira et al., 2020), and date palm trees (Gibril et al., 2021; Jintasuttisak et al., 2022), as well as banana (Neupane et al., 2019) and ornamental plants (Bayraktar et al., 2020).

On the other hand, the single-stage object detection model, YOLO, has gained popularity due to its balance of speed and accuracy, making it highly suitable for real-time processing (Badgujar et al., 2024). Wibowo et al. trained and validated various versions of YOLO models using a dataset of 56,614 oil palm trees captured from UAV imagery (Wibowo et al., 2022), achieving excellent performance in detecting a wide range of variations in oil palm trees. Chowdhury et al. introduced a generalized gradient vector flow to detect dominant points representing the edges of leaf stems and then used YOLO-v5 to eliminate false candidate points (Chowdhury et al., 2022). Although integrating both methods demonstrated good performance in detecting oil palm trees under different conditions, a single deep learning model that can function independently is preferable for simplicity and robustness. Putra and Wijayanto also achieved strong performance in oil palm tree detection using YOLO-v3, though their dataset had limited variation (Putra and Wijayanto, 2023). Among the studies utilizing UAV images, only one explored multi-species detections, incorporating coconut trees as a secondary species (Chowdhury et al., 2022).

To address the issue of duplicate predicted bounding boxes, Li et al. proposed merging the coordinates corresponding to the same ground truth label into a single coordinate (Li et al., 2016). Unlike YOLO, studies employing Faster R-CNN typically do not apply post-processing methods to remove duplicate predictions (Liu et al., 2021; Yarak et al., 2021; Zheng et al., 2021). This is because Faster R-CNN generally produce high-confidence predictions through its combination of region proposal refinement, multi-task loss optimization, deep feature extraction, and hard negative mining (Ren et al., 2016). In contrast, studies using YOLO often implement confidence thresholding and NMS to filter out low-confidence prediction and eliminate duplicate bounding boxes (Chowdhury et al., 2022; Putra and Wijayanto, 2023; Wibowo et al., 2022), thereby improving detection accuracy.

Detecting mature oil palm trees is relatively easier than detecting younger ones due to the larger crowns of mature trees, which provide richer spatial information. However, current studies using UAV images for oil palm tree detection have often overlooked the challenges associated with detecting young oil palm trees, primarily due to limited dataset variability (Liu et al., 2021; Putra and Wijayanto, 2023). It remains unclear how well state-of-the-art methods like Faster R-CNN and YOLO-v5 perform in detecting young oil palm trees when they blend into complex backgrounds, such as dense vegetation or varying soil textures. Moreover, no comparative analysis has specifically evaluated the performance of these models for oil palm tree detection. Thus, there is a clear need to develop and investigate a robust model capable of effectively detecting oil palm trees across diverse variations and complex environments.

Meanwhile, RetinaNet (Lin et al., 2017b), a single-stage model with multi-head configurations, has demonstrated strong performance by incorporating feature pyramid networks (FPNs) and the focal loss function. Although a study applied RetinaNet to oil palm tree detection (Pribadi et al., 2023), it was limited to satellite imagery and did not address all the challenges previously discussed. While its specific application in oil palm tree detection remains underexplored, prior research has shown RetinaNet's effectiveness in detecting banana plants (Selvaraj et al., 2020) and forest trees (Weinstein et al., 2019; Weinstein et al., 2020) using UAV imagery. These findings suggest that RetinaNet holds significant potential for adaptation to oil palm tree detection as well.

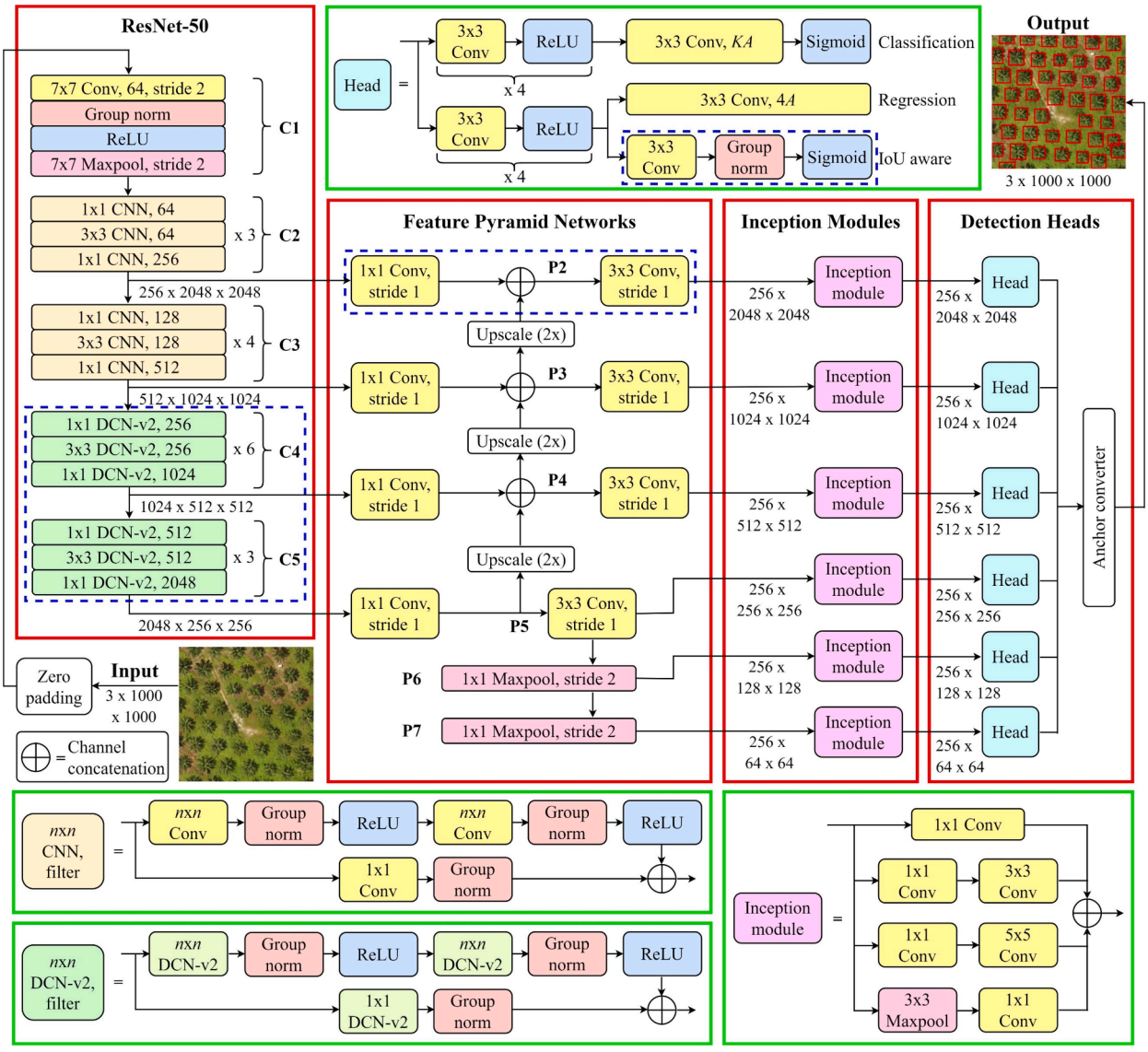


Fig. 3. Illustration of the enhanced RetinaNet architecture. Red boxes indicate the primary components of RetinaNet, while green boxes provide detailed views of its subcomponents. The proposed enhancements to the architecture are highlighted within dashed blue boxes.

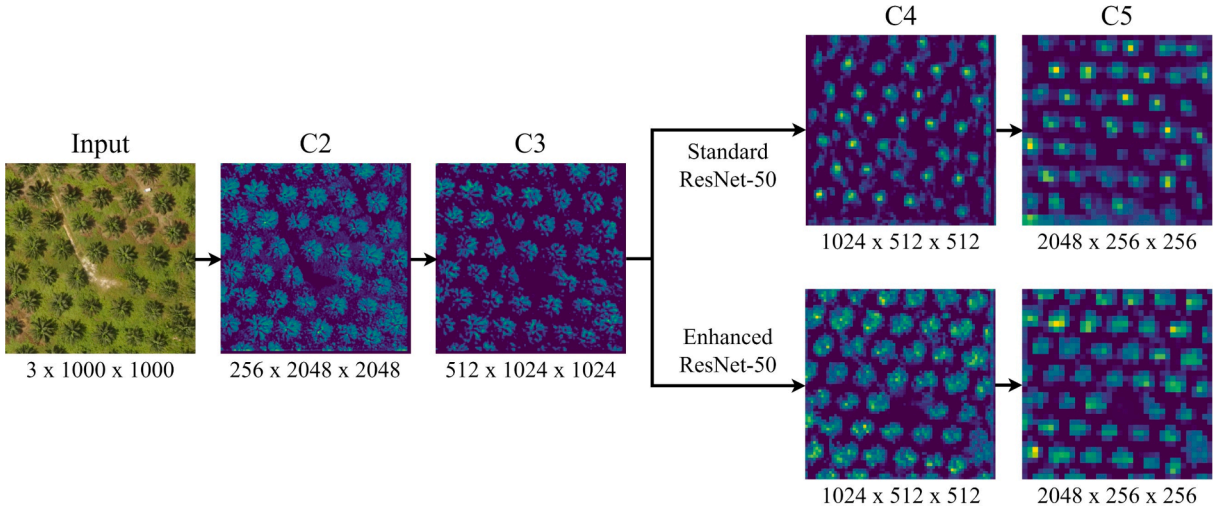


Fig. 4. Hierarchical feature representations in the enhanced ResNet-50.

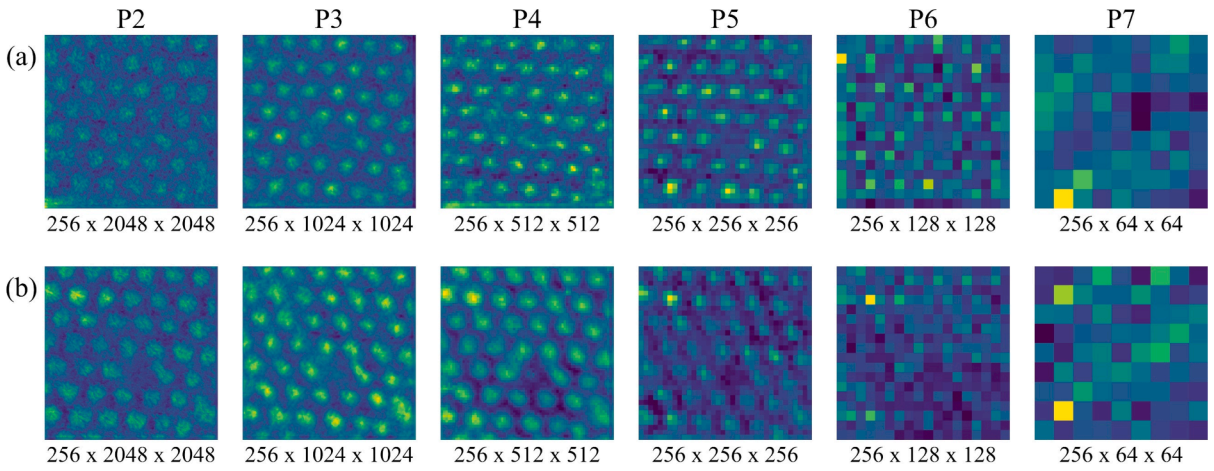


Fig. 5. Comparison of feature maps generated by FPNs from (a) the standard ResNet-50 and (b) the enhanced ResNet-50.

3. Proposed method

As mentioned in previous sections, the main objective of this study is to develop an effective method for detecting oil palm trees in UAV imagery. To achieve this, we build upon the strengths of RetinaNet (Lin et al., 2017b), a model recognized for its robust performance in detecting objects within complex and challenging environments. Fig. 2 provides an overview of the proposed method, illustrating how the RetinaNet architecture is utilized to capture subtle object variations.

3.1. Overall architecture

The core components of RetinaNet include ResNet-50, FPNs, inception modules, and detection heads. Inspired by Zhu et al.'s enhancements to RetinaNet (Zhu et al., 2020), which offer a strong yet simple baseline for face detection, we incorporate the latest deformable convolutions (Zhu et al., 2019) into the RestNet-50 backbone and utilize six feature pyramid levels to enhance feature extraction (Fig. 3). Additionally, to improve detection localization, we introduce an IoU-aware branch in a multi-head configuration (Wu et al., 2020b).

3.2. Feature extraction

3.2.1. ResNet-50

ResNet-50 is integrated as the backbone for feature extraction, which is crucial for training very deep neural networks. Developed by He et al. (He et al., 2016), ResNet-50 addresses the degradation problem caused by vanishing gradients, where deeper networks initially converge but subsequently degrade in accuracy as depth increases. This architecture utilizes residual blocks with identity shortcut connections that skip one or more layers, enabling the network to learn residual functions relative to the layer inputs instead of learning unreferenced functions. Fig. 4 illustrates the progressive stages from C2 to C5 of ResNet-50, showcasing the hierarchical feature representations computed at each stage.

The strength of ResNet-50 lies in its ability to compute hierarchical representations of features across stages. However, conventional convolutional operations rely on a fixed and rigid grid structure over the input feature maps, applying the same filter uniformly across the entire image. While effective for many applications, this fixed geometric structure struggles with objects that exhibit significant variability in shape, size, and pose (Dai et al., 2017). This issue becomes more pronounced in the deeper layers of ResNet-50, where the resolution of the feature map decreases, causing features to shift and become difficult to detect with conventional convolutions (Fig. 4).

To overcome these limitations, deformable convolution can be applied, allowing convolutional filters to dynamically adjust their receptive fields based on the input data. This adaptability is achieved by introducing learnable offsets to the regular grid sampling locations used in convolutional operations. These offsets, learned during training, enable the network to spatially deform the convolutional filters, enhancing its ability to capture variations in object shapes, sizes, and poses.

In this study, we integrate the latest deformable convolutional networks, known as DCN-v2 (Zhu et al., 2019), into the C4 and C5 stages of ResNet-50. This strategic placement targets the deeper layers of the network, where capturing higher-level semantic information is crucial and where conventional convolutional operations may struggle to capture complex patterns. Mathematically, DCN-v2 is formulated as follows (Zhu et al., 2019):

$$y(p) = \sum_{k=1}^{K} \omega_k \bullet x(p + p_k + \Delta p_k) \bullet \Delta m_k \tag{1}$$

where K is the number of sampling locations in the convolutional filter, ω_k is the weight for the k-th location, p_k is the pre-defined offset for the k-th location, $x(p)$ represents the features at location p from the input feature maps x, $y(p)$ denotes the features at location p from the output feature maps y, Δp_k is the learnable offset for the k-th location, and Δm_k is the modulation scalar for the k-th location.

In addition, deformable pooling extends the concept of deformable convolution to pooling operations, enabling the network to dynamically learn offsets for pooling regions. For a given region of interest, deformable pooling divides the region into k spatial bins. Within each bin, sampling grids with uniform spatial intervals are utilized. The values sampled from these grids are then averaged to produce the output for that bin. The resulting output feature for each bin, denoted as $y(k)$, is calculated as follows (Zhu et al., 2019):

$$y(k) = \sum_{j=1}^{n_k} x\left(p_{kj} + \Delta p_k\right) \bullet \Delta m_n / n_k \tag{2}$$

where p_{kj} is the sampling location for the j-th grid cell in the k-th bin, and n_k is the number of sampled grid cells.

As DCNs often operate in environments with significant variations in object shapes and poses, batch normalization can be sensitive to batch size and less effective at capturing the intricate patterns introduced by deformable convolutions. In contrast, group normalization has proven more effective in such scenarios by providing flexible and reliable normalization, ensuring consistent performance regardless of batch size (Wu and He, 2018). Therefore, we replace all batch normalization layers in ResNet-50 with group normalization. With the implementation of deformable convolution, deformable pooling, and group normalization

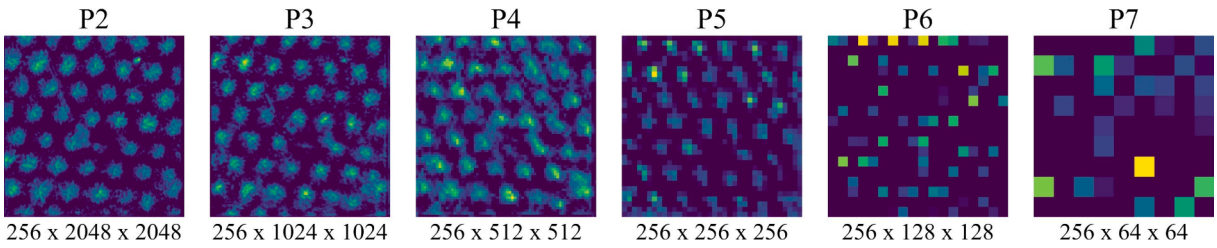


Fig. 6. Hierarchical feature extraction across multiple scales using inception modules.

in ResNet-50, the feature maps at C4 and C5 exhibit enhanced correlations, which are more effectively utilized in subsequent FPN processing.

3.2.2. *Feature pyramid networks*

FPNs generate feature maps at multiple levels on top of ResNet-50, constructing high-level semantic features across all scales via top-down pathways and lateral connections (Lin et al., 2017a). This architecture enhances the network's ability to detect objects of varying sizes more effectively. While previous studies have utilized five feature pyramid levels (P3 to P7) for detecting tree crowns (Pribadi et al., 2023; Selvaraj et al., 2020; Weinstein et al., 2019; Weinstein et al., 2020), we extend this to six levels (P2 to P7) to better accommodate oil palm trees of varying shapes and sizes in complex backgrounds.

First, a 1x1 convolution is applied to C5 to generate the coarsest resolution map, which is then upsampled by a factor of two to increase spatial resolution. This upsampled map is merged with the corresponding bottom-up map through element-wise addition. Following this fusion, a 3x3 convolution is applied to each merged map to mitigate the aliasing effect of upsampling, resulting in the final feature maps at levels P2, P3, P4, and P5. These levels correspond to C2, C3, C4, and C5, respectively, and are optimized for detecting small to medium-sized objects. Meanwhile, P6 and P7 are generated through downsampling to further enhance detection accuracy for larger objects. To ensure uniformity, all final feature maps are standardized to 256 channels. Fig. 5 illustrates how the deeper layers of the enhanced ResNet-50 provide detailed features crucial for generating the final feature maps in FPNs, in contrast to the standard ResNet-50, which tends to produce noisier results.

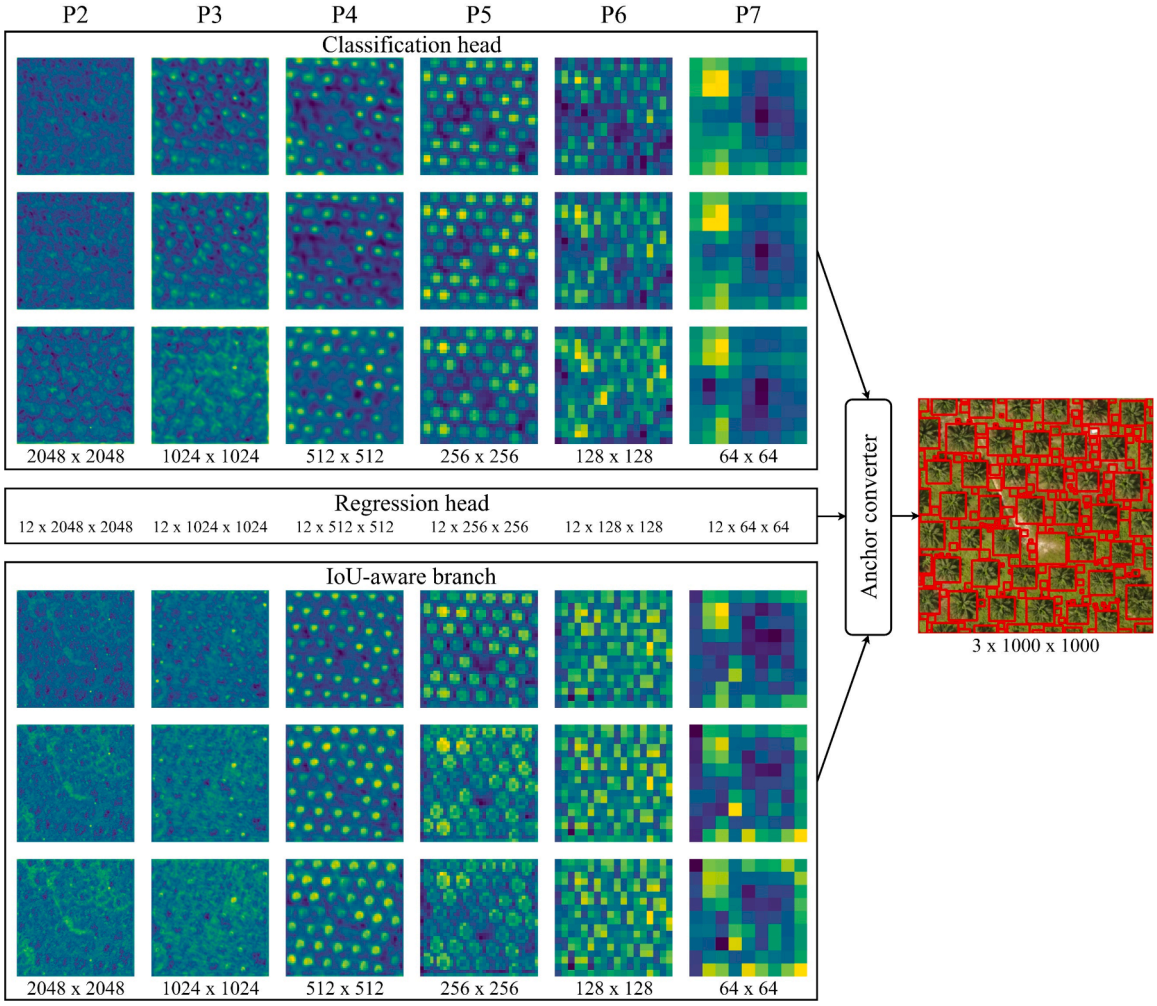


Fig. 7. Final detection bounding boxes, confidence scores, and classifications are generated from the outputs of the classification, regression, and IoU-aware subnetworks through an anchor converter. The regression head outputs are omitted for simplicity.

3.3. Inception modules

The inception module is integrated to enhance the network's capacity for multi-scale processing, allowing for the simultaneous analysis of visual information at various scales and resolutions. This capability is critical in object detection scenarios where objects may vary in size, be partially occluded, or intersect with other objects in the scene (Zhu et al., 2020). Configured with an input channel size of 256 across six feature pyramid levels, the inception module incorporates group normalization to ensure stable training, with shared parameters across levels to maintain consistency and reduce computational complexity.

Featuring multiple parallel paths, each equipped with convolutions of varying filter sizes (1x1, 3x3, and 5x5), the module captures diverse levels of spatial detail. Additionally, max-pooling operations within these paths introduce translation invariance and capture contextual information. The outputs from these paths are concatenated along the channel dimension, providing subsequent network layers with a rich, multi-scale feature representation of the original input (Szegedy et al., 2015). Fig. 6 illustrates how these enhanced features improve the discernibility of oil palm trees, facilitating their intuitive identification.

3.4. Detection heads

Anchor boxes are employed to accommodate a range of object sizes and aspect ratios, enabling effective detection across different scales. The network evaluates the presence of objects and adjusts these anchor boxes by calculating confidence scores and refining coordinates to align with ground-truth bounding boxes. This process involves two specialized subnetworks: one for object classification and another for bounding box regression.

The classification subnet predicts the probability of object presence across A anchors and K object classes using a fully convolutional network. It consists of four 3x3 convolutions with 256 filters each, followed by rectified linear unit (ReLU) activations. The final layer uses a 3x3 convolution with KA filters ($A = 3$), and a sigmoid activation converts the outputs into confidence scores. Simultaneously, the regression subnet employs four 3x3 convolutional layers, producing $4A$ linear outputs per spatial location, which focus on regressing offsets from each anchor box to nearby ground-truth bounding boxes.

Discrepancies between classification confidence and bounding box localization accuracy can result in high-confidence predictions that do not align with actual objects. To mitigate this issue, an IoU-aware branch operates alongside the two subnetworks to compute the IoU value between predicted and ground-truth bounding boxes (Wu et al., 2020b). The IoU-aware head, designed to handle all predicted object classes and anchor boxes, processes 256 input channels. It refines the features through four 3x3 convolutional layers with 256 filters each, followed by ReLU activations and group normalization for stability during training. The final layer uses a sigmoid activation to output probabilities within the IoU value range of 0 to 1.

Throughout this process, the spatial dimensions (width x height) are preserved for each level as the data passes through the detection heads. The classification and regression subnetworks produce outputs with depths of KA and $4A$, respectively, while the IoU-aware branch generates an output with a depth of A. Finally, outputs from all three subnetworks are passed into an anchor converter, which produces the final detection bounding boxes, confidence scores, and classifications (Fig. 7).

3.5. Loss functions

Focal loss, specifically the α-balanced variant (Lin et al., 2017b), is employed in the classification head to effectively manage the class imbalance often encountered in object detection tasks. This variant of focal loss, denoted as $\mathscr{L}_{\text{focal}}$, is defined by the following equation:

$$\mathscr{L}_{\text{focal}} = -\alpha_t(1-p_t)^{\gamma}\log(p_t) \tag{3}$$

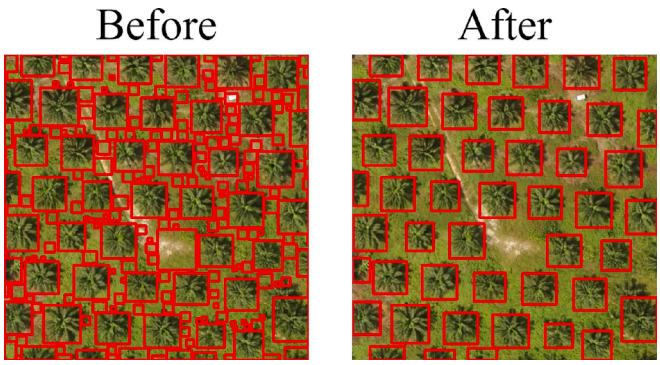


Fig. 8. Refinement of raw predictions through confidence thresholding and NMS.

where p_t is the model's estimated probability for the ground-truth class, α_t is a weighting factor for class t to balance positive and negative examples, and γ is a tunable parameter that controls the down-weighting of easy examples.

For the regression head, smooth L1 loss is commonly used to refine bounding box coordinates. However, it may not fully capture the IoU metric, which is crucial for evaluating bounding box quality. To address this limitation, we employ the distance IoU loss function, $\mathscr{L}_{\text{DIoU}}$ (Zheng et al., 2020), which is defined as:

$$\mathscr{L}_{\text{DIoU}} = 1 - \text{IoU} + \frac{\rho^2(b, b_{gt})}{c^2} \tag{4}$$

where b is the center point of the predicted bounding box, b_{gt} is the center point of the ground-truth bounding box, $\rho(\bullet)$ represents the Euclidean distance between the two center points, and c is the diagonal length of the smallest enclosing box that covers both b and b_{gt}. This modification encourages the model to generate bounding boxes that not only localize objects more accurately but also maximize their overlap with ground-truth boxes. This approach leads to more precise and context-aware object detection, particularly in scenarios involving complex object shapes or significant occlusion.

3.6. Inference

During the inference phase, post-processing techniques are applied to refine the raw predictions generated by the network (Fig. 8). The primary methods employed are confidence thresholding and NMS. After the initial detection, each bounding box is assigned a confidence score, indicating the likelihood that it contains an object of interest. To filter out low-confidence detections and reduce false positives, a confidence threshold is set, retaining only those bounding boxes with scores exceeding this threshold for further analysis. This step ensures that the remaining detections are both accurate and relevant.

Despite applying confidence thresholding, multiple overlapping bounding boxes may still correspond to the same object. To resolve this, NMS is applied, which retains the bounding box with the highest confidence score while suppressing others with an IoU value greater than 0.5 (Salscheider, 2021). This process iteratively removes overlapping boxes until none remain, effectively reducing duplicate detections and ensuring that each object is represented by a single, high-confidence bounding box.

4. Experimental results

4.1. Datasets

In this study, the datasets, consisting of aerial images captured over oil palm plantations in Pahang and Johor, Malaysia, were primarily

Fig. 9. Samples of UAV images illustrating (a) diverse variations in oil palm trees and (b) the presence of other species such as coconut trees and banana plants.

Table 1
Number of ground-truth labels in the training, validation, and test sets for tree detection tasks.

Detection	Training set	Validation set	Test set
Single species	1735 oil palm trees	388 oil palm trees	386 oil palm trees
Dual species	1735 oil palm trees	388 oil palm trees	386 oil palm trees
	388 coconut trees	295 coconut trees	292 coconut trees
Triple species	1950 oil palm trees	502 oil palm trees	489 oil palm trees
	1639 coconut trees	335 coconut trees	315 coconut trees
	750 banana plants	184 banana plants	158 banana plants

provided by Aonic Sdn. Bhd.. To adhere to confidentiality agreements, the precise geographic coordinates of the study areas have been withheld. The imagery spans an area of approximately 5000 ha, covering plantations at varying stages of development, ranging from newly established (1–3 years old) to mature (>4 years old). The surveyed regions are characterized by flat to gently rolling topography within a tropical climate zone.

The aerial images were acquired using DJI Phantom 4 Pro UAVs equipped with FC6310 camera sensors. The FC6310 sensor is a 20-megapixel, 1-inch CMOS device, paired with an 8.8 mm/24 mm (35 mm equivalent) f/2.8-f/11 lens. Flight missions were conducted at altitudes between 120 and 150 m above ground level, with an average flight speed of 5 m/s. To ensure accurate photogrammetric reconstruction, the image acquisition settings included front and side overlaps of 75 % and 65 %, respectively.

The datasets offer extensive coverage of both young and mature oil palm trees, capturing the variability and complexity of these plantations. Additionally, they encompass other species such as coconut trees and banana plants (Fig. 9). To further enhance species diversity, a supplementary dataset focused specifically on coconut trees was incorporated from a public repository (Alonzo et al., 2014).

Ground truth annotations for oil palm and coconut trees, as well as banana plants, were meticulously created for all images using the labeling tool available at https://github.com/tzutalin/labelImg. This tool enables precise annotations by allowing bounding boxes to be drawn around the tree crowns, with the four corner points carefully marked. Trees located near the image edges were annotated if their centroids were visible, provided that at least 50 % of the tree crown remained within the image frame.

Table 1 presents the distribution of ground-truth labels across the training (~70 %), validation (~15 %), and test (~15 %) sets for three distinct detection tasks: (i) single-species detection, focusing solely on oil palm trees; (ii) dual-species detection, expanding to include both oil palm and coconut trees; and (iii) triple-species detection, which incorporates banana plants in addition to the other two species. These multiple detection tasks were designed to assess the robustness of object detection models across varying levels of complexity.

A comprehensive preprocessing pipeline was applied to the annotated images in the training set. To simulate diverse fields of view and varying object distances, images were randomly cropped to scales ranging from 30 % to 100 % of their original size. Next, brightness (±25 %), contrast (±20 %), saturation (±30 %), and hue (±15 %) adjustments were applied to enhance the model's robustness against real-world lighting variations. Additionally, geometric transformations, including horizontal flips and random rotations (±15°), were employed to introduce further variability and replicate different orientations.

The images were then resized to a standardized dimension of 1000 x 1000 pixels, a resolution chosen to balance detail retention with computational efficiency. Pixel values were normalized using the mean and standard deviation for each channel to aid model convergence during training. The preprocessed images, along with their corresponding annotations, were subsequently formatted for model training. For the validation and test sets, images were resized similarly to ensure consistency during model evaluation. The validation set was used to fine-tune hyperparameters and assess performance during training, while the test set served as the benchmark for evaluating final model performance.

4.2. Evaluation Metrics

To analyze the precision-recall curves, we retained the original predicted bounding boxes to ensure fair comparisons across different models without adjusting the confidence scores. Before applying NMS, all predicted bounding boxes with confidence scores below 0.001 were discarded as noise. The remaining predictions, overlapping with ground-truth bounding boxes, were then sorted by confidence score in descending order. Each ground truth instance was matched with the prediction that had the highest confidence score and an IoU value of at least 0.5, categorizing them as true positives. Any unmatched predictions were considered false positives.

Once true positives and false positives for each detection class were identified, the predictions were ranked from highest to lowest confidence. Precision and recall were then calculated iteratively at each rank. Precision is defined as the ratio of true positive detections to the total detections made up to that rank, while recall represents the proportion of true positive detections relative to the total ground truth instances.

Using 101 equally spaced recall levels [0, 0.01, 0.02, …, 1], the precision at each recall level r was interpolated by taking the maximum precision value where the corresponding recall was equal to or greater than r. The average precision (AP), which summarizes the shape of the precision-recall curve, was calculated as the area under the interpolated precision-recall curve. This approach offers a comprehensive view of model performance across the full range of recall values, rather than at a single point (Everingham et al., 2010). The mean average precision (mAP) was then obtained by averaging AP values across different detection tasks involving multiple plant species.

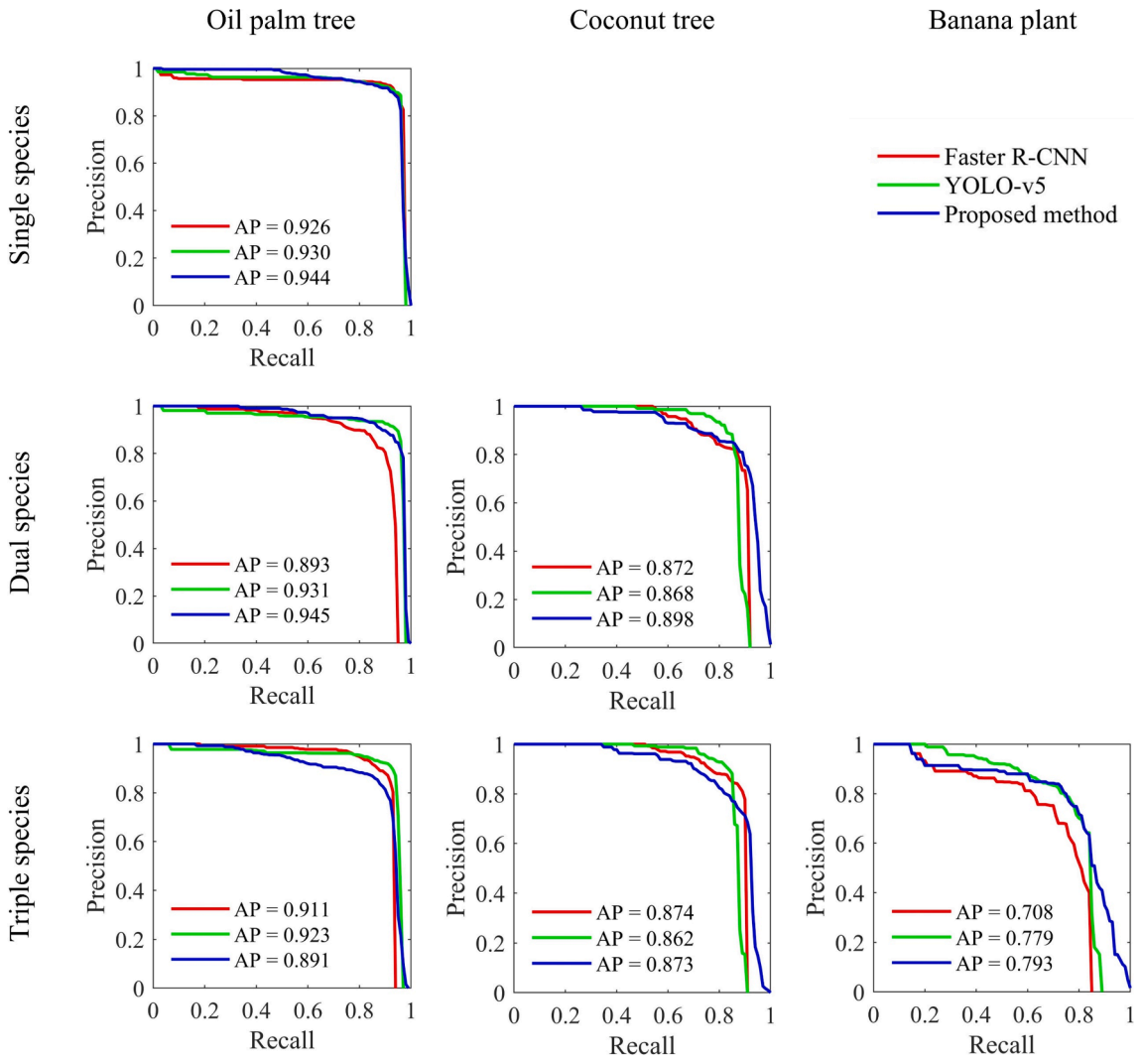


Fig. 10. Precision-recall curves comparing Faster R-CNN, YOLO-v5, and the proposed method for detecting oil palm trees, coconut trees, and banana plants.

To compare overall model performance, confidence scores were adjusted during the inference phase to remove low-confidence predictions. We evaluated precision, recall, and F1-score at 101 equally spaced confidence levels from 0 to 1, with a step size of 0.01. The F1-score, the harmonic mean of precision and recall, provides a balanced single metric. The confidence threshold yielding the highest F1-score was selected as the model's optimal threshold. This evaluation ensures a fair and effective comparison, as different models may return varying numbers of detections.

4.3. Implementation Details

In our implementation, we utilized the stochastic gradient descent optimizer with a momentum of 0.9 and a weight decay parameter of 5 x 10^{-4}. Training was executed on three GeForce RTX 3070 GPUs, using a batch size of 12 distributed as 3 x 4 across the GPUs. The learning rate was initialized at 3.75 x 10^{-3} and followed a cosine decay schedule, gradually reducing to 3.75 x 10^{-5} over 630 epochs, with adjustments every 30 epochs. A warm-up phase during the first 500 iterations incrementally increased the learning rate from 3.75 x 10^{-4} to 3.75 x 10^{-3}, fostering stable convergence as the model adapted to the training data.

4.4. Comparison with State-of-the-Art methods

Recent studies have employed Faster R-CNN (Liu et al., 2021; Yarak et al., 2021; Zheng et al., 2021) and YOLO-v5 (Chowdhury et al., 2022; Wibowo et al., 2022) for detecting oil palm trees. To evaluate the effectiveness of our proposed method, we conducted a comparative study with these state-of-the-art techniques. Our method detected approximately 240 times more objects than Faster R-CNN and 64 times more than YOLO-v5. As a single-stage detector, RetinaNet predicts both classes and bounding boxes directly from densely placed anchors. In contrast, two-stage detectors like Faster R-CNN use a region proposal network to filter low-confidence regions before making final detections (Ren et al., 2016). RetinaNet, however, processes a greater number of anchors without intermediate filtering, resulting in a higher number of predictions (Lin et al., 2017a).

Fig. 10 shows the precision-recall curves for both single- and multi-species detection tasks. Our method consistently achieves higher recall values compared to Faster R-CNN and YOLO-v5 across all detection tasks. While the increased number of predictions leads to more false positives, this is mitigated by the fact that most of these false positives have low confidence scores. As shown in Fig. 10, the proposed method also demonstrates superior AP values for both single- and dual-species detection tasks.

Although YOLO-v5 is optimized for high-speed detection and excels in real-time applications, it struggles to detect young oil palm trees that blend into complex backgrounds. Faster R-CNN, while effective at identifying objects with distinct and clear boundaries, also faces challenges in detecting less distinguishable features like young oil palm

Table 2
The mAP for tree detection tasks using Faster R-CNN, YOLO-v5, and the proposed method.

Detection	Faster R-CNN	YOLO-v5	Proposed method
Single species	0.926	0.930	0.944
Dual species	0.883	0.900	0.922
Triple species	0.831	0.855	0.852

trees. Our proposed method addresses these limitations by incorporating deformable convolutions and deeper feature pyramid layers, allowing the network to adapt to geometric variations and capture subtle features in complex environments. This significantly enhances detection performance, particularly in challenging scenarios.

For triple-species detection, the advantage of our method is less pronounced. However, Faster R-CNN performs significantly worse in detecting banana plants, with markedly lower AP values. This underperformance can be attributed to Faster R-CNN's reliance on standard cross-entropy loss (Ren et al., 2016), which struggles with class imbalance, as banana plants constitute only about 17 % of the dataset. In contrast, our proposed method utilizes focal loss, which effectively handles class imbalance by down-weighting the loss from well-classified examples (Lin et al., 2017b).

As shown in Table 2, the proposed method achieves superior performance in single- and dual-species detection tasks, outperforming Faster R-CNN and YOLO-v5 by a margin of 1.4–3.9 % in mAP. However, as the detection task complexity increases with the addition of more species, all models exhibit a decline in mAP, highlighting the inherent challenges in distinguishing between plant species with overlapping features.

Confidence thresholding during inference—a standard practice in object detection (Chowdhury et al., 2022; Putra and Wijayanto, 2023; Wibowo et al., 2022)—is crucial for enhancing model performance by filtering out low-confidence predictions. As expected, all models demonstrate substantial improvements in F1-scores after applying this thresholding (Table 3). Given that RetinaNet generates more predictions than Faster R-CNN and YOLO-v5, it necessitates lower confidence thresholds to achieve optimal performance. However, while removing low-confidence predictions boosts precision, it slightly reduces recall by discarding some true positives. This effect is particularly pronounced for objects with indistinct features or those that blend into complex backgrounds, which tend to receive lower confidence scores. Overall, our method excels in single- and dual-species detection, achieving F1-scores of 0.947 and 0.902, respectively, thereby outperforming Faster R-CNN and YOLO-v5 by 1.5 % to 6.3 %.

Fig. 11 illustrates the detection capabilities of Faster R-CNN, YOLO-v5, and our proposed method under challenging conditions. Faster R-CNN and YOLO-v5 struggle to detect young oil palm trees that blend into complex backgrounds, unlike our method (Fig. 11(a)–(c)). This difficulty arises from the smaller, less distinct crowns of young oil palm trees, which lack the detailed spatial information present in mature trees. Additional challenges, such as densely overlapped trees and adverse weather conditions, further hinder detection accuracy (Fig. 11(d) and 11(e)). In multi-species detection scenarios, the presence of coconut trees and banana plants—both of which share visual similarities with oil palm trees—results in an increased incidence of false positives across all models. False positives are also more frequent near image edges, where cropped trees—of which only 30 to 45 % of the crown is visible—are not included in the ground truth annotations.

4.5. Ablation study

Our ablation study examines the impact of deformable convolution, feature pyramid levels, and the IoU-aware branch on detection performance. Deformable convolution is evaluated against the standard ResNet-50, which uses conventional convolution and batch normalization. Table 4 presents mAP values for the single-species detection task, illustrating the effects of these modifications. As expected, the enhanced RetinaNet exhibits a significant 4.8 % increase in mAP compared to the standard model, indicating that deformable convolution and deeper feature extraction enhance hierarchical information crucial for accurate detection. However, expanding the FPN to seven levels does not lead to improvements, suggesting that the resolution at this level might not be appropriate for reliable detection. Furthermore, the IoU-aware branch contributes to improved performance by refining localization and reducing false positives.

4.6. Limitations and Future Works

Our study has several limitations. First, the high computational demands of both training and inference may limit the practicality of our model for real-time applications on large-scale datasets. While the model demonstrates strong performance in detecting oil palm and coconut trees, its generalizability to other plant species remains untested. Future work should address this by expanding the dataset to include a broader variety of species and further optimizing the model for real-time, large-scale deployments. Additionally, although YOLO has advanced to version 10 (Alif and Hussain, 2024), offering improvements over YOLO-v5, our comparisons were conducted using YOLO-v5 (Chowdhury et al., 2022; Wibowo et al., 2022), which was the most established and widely adopted model for oil palm tree detection at the time of this study. Extending the comparison to include later versions would necessitate extensive additional experimentation, which falls outside the scope of this study.

While the proposed method achieves near-perfect recall, it also produces a substantial number of false positives. Although confidence thresholding and NMS help optimize performance during inference, they can inadvertently filter out low-confidence true positives. Future work could explore more advanced post-processing techniques, such as soft-NMS (Bodla et al., 2017), robust and efficient post-processing (Sabater et al., 2020), or density-aware adaptive thresholding (Lee et al., 2024). These methods could improve the balance between reducing false positives and retaining true positives, particularly in complex detection scenarios. Increasing the sample size of young oil

Table 3
Optimized model performance for tree detection tasks using confidence thresholding and NMS during the inference phase.

Detection	Method	Baseline			Optimized performance			
		Precision	Recall	F1-score	Confidence threshold	Precision	Recall	F1-score
Single species	Faster R-CNN	0.751	0.975	0.848	0.95	0.927	0.920	0.923
	YOLO-v5	0.753	0.922	0.829	0.66	0.931	0.925	0.928
	Proposed method	0	0.991	0	0.51	0.949	0.946	0.947
Dual species	Faster R-CNN	0.611	0.955	0.734	0.84	0.867	0.813	0.839
	YOLO-v5	0.441	0.895	0.585	0.60	0.899	0.876	0.887
	Proposed method	0.003	0.996	0.006	0.41	0.892	0.913	0.902
Triple species	Faster R-CNN	0.580	0.897	0.687	0.60	0.817	0.814	0.815
	YOLO-v5	0.475	0.891	0.569	0.57	0.858	0.838	0.848
	Proposed method	0.006	0.997	0.012	0.23	0.841	0.844	0.842

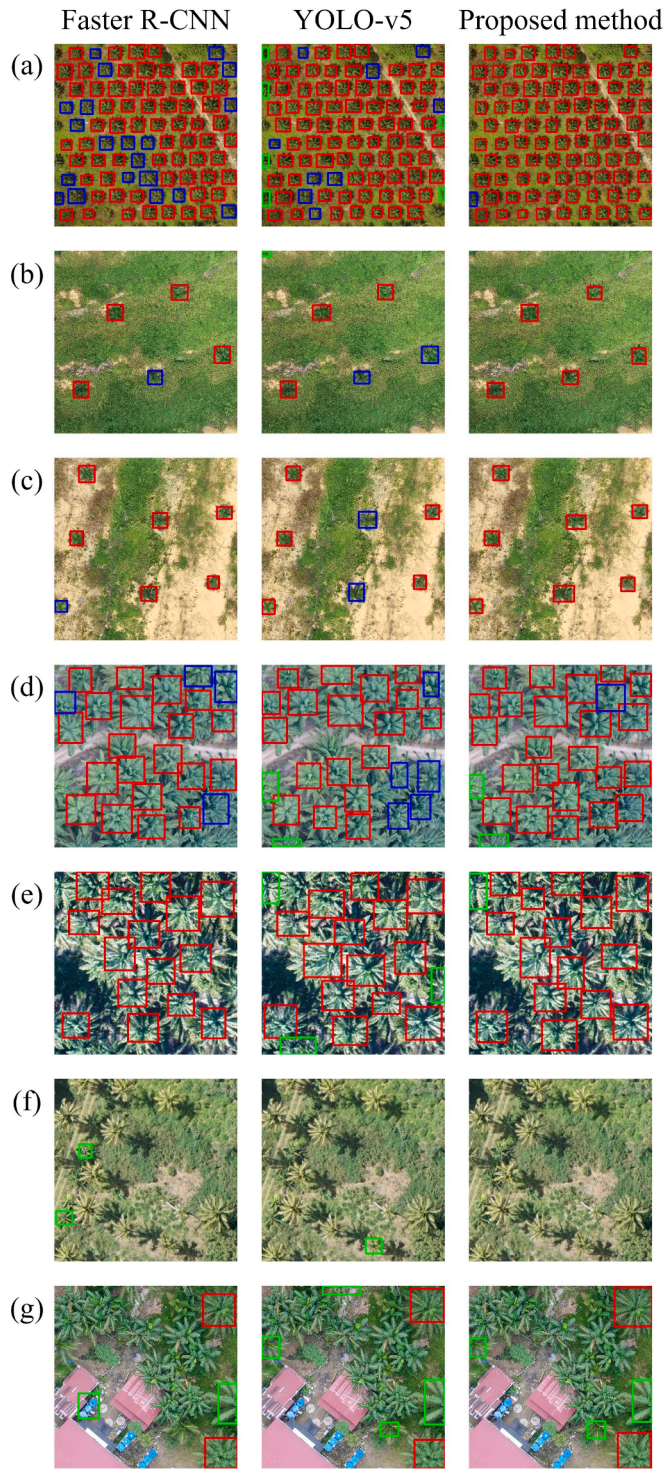


Fig. 11. Samples of oil palm tree detection using Faster R-CNN, YOLO-v5, and our proposed method under diverse challenging conditions: (a) oriented and organized young trees, (b) young trees blended into vegetation backgrounds, (c) young trees blended into varying soil backgrounds, (d) densely overlapped trees, (e) influenced by weather conditions, (f) presence of coconut trees, and (g) presence of coconut trees and banana plants. Red, blue, and green bounding boxes indicate correctly detected, missed, and incorrect detections of oil palm trees, respectively.

Table 4
Ablation study evaluating the effects of deformable convolution, feature pyramid levels, and the IoU-aware branch on the single-species detection task.

FPN level	ResNet-50	IoU aware	mAP
5	Standard	Yes	0.904
		No	0.896
	DCN-v1	Yes	0.918
		No	0.912
	DCN-v2	Yes	0.923
		No	0.915
6	Standard	Yes	0.924
		No	0.918
	DCN-v1	Yes	0.938
		No	0.932
	DCN-v2	Yes	0.944
		No	0.937
7	Standard	Yes	0.921
		No	0.920
	DCN-v1	Yes	0.937
		No	0.935
	DCN-v2	Yes	0.942
		No	0.940

palm trees in complex backgrounds could also improve confidence scores, reducing the reliance on confidence thresholding to minimize true positive loss.

Another limitation is the prevalence of false positives near image edges, particularly for trees that are only partially visible. Our method does not fully account for these edge cases, where cropped tree crowns lead to detection errors. Future research could focus on developing specialized algorithms for detecting objects near image boundaries. Approaches like box boundary-aware vectors (Yi et al., 2021) or a boundary-aware self-consistent framework (Xu et al., 2023) could be explored to adapt detection models for edge proximity, thereby reducing false positives caused by incomplete object visibility.

5. Conclusion

We presented a novel enhancement to the RetinaNet architecture for the accurate detection of oil palm trees in UAV imagery, effectively addressing diverse variations and complexities inherent in plantation environments. Our approach significantly improves detection performance by integrating deformable convolutions into the ResNet-50 backbone, leveraging deeper feature pyramid levels, and incorporating an IoU-aware branch in a multi-head configuration. This enables our model to detect not only mature oil palm trees but also young trees that are often camouflaged within complex backgrounds. The application of confidence thresholding and NMS during the inference phase further improves precision without substantially compromising recall.

The proposed method achieves superior detection results, with F1-scores of 0.947 and 0.902 for single- and dual-species detection tasks, respectively, outperforming state-of-the-art methods by 1.5–6.3 %. These findings underscore the robustness and adaptability of our model, addressing key challenges in oil palm tree detection and providing a reliable and efficient solution for optimizing yield and monitoring plantation health. By enhancing detection performance and improving the capacity to handle complex detection tasks, our method contributes to sustainable land management and agricultural practices, positioning it as a valuable tool for supporting precision agriculture and resource optimization.

CRediT authorship contribution statement

Sheng Siang Lee: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Lam Ghai Lim:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Formal analysis, Conceptualization. **Shivakumara Palaiahnakote:**

Writing – review & editing, Supervision, Methodology, Conceptualization. **Jin Xi Cheong:** Writing – review & editing, Resources, Funding acquisition. **Serene Sow Mun Lock:** Writing – review & editing. **Mohamad Nizam Bin Ayub:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This work was supported by Aonic Sdn. Bhd., Malaysia, who also provided the data for this study.

References

Alif, M.A.R., Hussain, M., 2024. YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain. arXiv preprint arXiv: 2406.10139.
Alonzo, M., Bookhagen, B., Roberts, D.A., 2014. Urban tree species mapping using hyperspectral and lidar data fusion. Remote Sens. Environ. 148, 70–83.
Arce, L.S.D., Osco, L.P., Arruda, M.d.S.d., Furuya, D.E.G., Ramos, A.P.M., Aoki, C., Pott, A., Fatholahi, S., Li, J., Araujo, F.F.d., 2021. Mauritia flexuosa palm trees airborne mapping with deep convolutional neural network. Sci. Rep. 11, 19619.
Badgujar, C.M., Poulose, A., Gan, H., 2024. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. Comput. Electron. Agric. 223, 109090.
Bayraktar, E., Basarkan, M.E., Celebi, N., 2020. A low-cost UAV framework towards ornamental plant detection and counting in the wild. ISPRS J. Photogramm. Remote Sens. 167, 1–11.
Beloiu, M., Heinzmann, L., Rehush, N., Gessler, A., Griess, V.C., 2023. Individual tree-crown detection and species identification in heterogeneous forests using aerial RGB imagery and deep learning. Remote Sens. (Basel) 15, 1463.
Bodla, N., Singh, B., Chellappa, R., Davis, L.S., 2017. Soft-NMS–improving object detection with one line of code. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5561–5569.
Bouguettaya, A., Zarzour, H., Kechida, A., Taberkit, A.M., 2022. Deep learning techniques to classify agricultural crops through UAV imagery: a review. Neural Comput. & Applic. 34, 9511–9536.
Boursianis, A.D., Papadopoulou, M.S., Diamantoulakis, P., Liopa-Tsakalidi, A., Barouchas, P., Salahas, G., Karagiannidis, G., Wan, S., Goudos, S.K., 2022. Internet of things (IoT) and agricultural unmanned aerial vehicles (UAVs) in smart farming: a comprehensive review. Internet of Things 18, 100187.
Chowdhury, P.N., Shivakumara, P., Nandanwar, L., Samiron, F., Pal, U., Lu, T., 2022. Oil palm tree counting in drone images. Pattern Recogn. Lett. 153, 1–9.
Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 764–773.
Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A., 2010. The pascal visual object classes (voc) challenge. Int. J. Comput. Vis. 88, 303–338.
Ferreira, M.P., de Almeida, D.R.A., de Almeida Papa, D., Minervino, J.B.S., Veras, H.F.P., Formighieri, A., Santos, C.A.N., Ferreira, M.A.D., Figueiredo, E.O., Ferreira, E.J.L., 2020. Individual tree detection and species classification of Amazonian palms using UAV images and deep learning. For. Ecol. Manage. 475, 118397.
Freudenberg, M., Nölke, N., Agostini, A., Urban, K., Wörgötter, F., Kleinn, C., 2019. Large scale palm tree detection in high resolution satellite images using U-Net. Remote Sens. (Basel) 11, 312.
Gibril, M.B.A., Shafri, H.Z.M., Shanableh, A., Al-Ruzouq, R., Wayayok, A., Hashim, S.J., 2021. Deep convolutional neural network for large-scale date palm tree mapping from UAV-based images. Remote Sens. (Basel) 13, 2787.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
Jintasuttisak, T., Edirisinghe, E., Elbattay, A., 2022. Deep neural network based date palm tree detection in drone imagery. Comput. Electron. Agric. 192, 106560.
Lee, E., Jung, M., Kim, A., 2024. Toward Robust LiDAR based 3D Object Detection via Density-Aware Adaptive Thresholding. arXiv preprint arXiv:2404.13852.
Li, W., Fu, H., Yu, L., Cracknell, A., 2016. Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. Remote Sens. (Basel) 9, 22.
Li, W., Dong, R., Fu, H., Yu, L., 2018. Large-scale oil palm tree detection from high-resolution satellite images using two-stage convolutional neural networks. Remote Sens. (Basel) 11, 11.
Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017a. Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125.
Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017b. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988.
Liu, Y., Feng, H., Yue, J., Fan, Y., Bian, M., Ma, Y., Jin, X., Song, X., Yang, G., 2023. Estimating potato above-ground biomass by using integrated unmanned aerial system-based optical, structural, and textural canopy measurements. Comput. Electron. Agric. 213, 108229.
Liu, Y., Fan, Y., Feng, H., Chen, R., Bian, M., Ma, Y., Yue, J., Yang, G., 2024a. Estimating potato above-ground biomass based on vegetation indices and texture features constructed from sensitive bands of UAV hyperspectral imagery. Comput. Electron. Agric. 220, 108918.
Liu, Y., Feng, H., Fan, Y., Yue, J., Chen, R., Ma, Y., Bian, M., Yang, G., 2024b. Improving potato above ground biomass estimation combining hyperspectral data and harmonic decomposition techniques. Comput. Electron. Agric. 218, 108699.
Liu, Y., Feng, H., Yue, J., Jin, X., Fan, Y., Chen, R., Bian, M., Ma, Y., Li, J., Xu, B., 2024c. Improving potato AGB estimation to mitigate phenological stage impacts through depth features from hyperspectral data. Comput. Electron. Agric. 219, 108808.
Liu, X., Ghazali, K.H., Han, F., Mohamed, I.I., 2021. Automatic detection of oil palm tree from UAV images based on the deep learning method. Appl. Artif. Intell. 35, 13–24.
Miyoshi, G.T., Arruda, M.d.S., Osco, L.P., Marcato Junior, J., Gonçalves, D.N., Imai, N.N., Tommaselli, A.M.G., Honkavaara, E., Gonçalves, W.N., 2020. A novel deep learning method to identify single tree species in UAV-based hyperspectral images. Remote Sens. (Basel) 12, 1294.
Mubin, N.A., Nadarajoo, E., Shafri, H.Z.M., Hamedianfar, A., 2019. Young and mature oil palm tree detection and counting using convolutional neural network deep learning method. Int. J. Remote Sens. 40, 7500–7515.
Neupane, B., Horanont, T., Hung, N.D., 2019. Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV). PLoS One 14, e0223906.
Onishi, M., Ise, T., 2021. Explainable identification and mapping of trees using UAV RGB image and deep learning. Sci. Rep. 11, 903.
Osco, L.P., De Arruda, M.d.S., Junior, J.M., Da Silva, N.B., Ramos, A.P.M., Moryia, É.A.S., Imai, N.N., Pereira, D.R., Creste, J.E., Matsubara, E.T., 2020. A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery. ISPRS J. Photogramm. Remote Sens. 160, 97–106.
Pribadi, D.O., Rustiadi, E., Nurdin, M., Saad, A., Pravitasari, A.E., Mulya, S.P., Ermyanyla, M., 2023. Mapping smallholder plantation as a key to sustainable oil palm: a deep learning approach to high-resolution satellite imagery. Appl. Geogr. 153, 102921.
Putra, Y.C., Wijayanto, A.W., 2023. Automatic detection and counting of oil palm trees using remote sensing and object-based deep learning. Remote Sens. Appl.: Soc. Environ. 29, 100914.
Quezada, J.C., Etter, A., Ghazoul, J., Buttler, A., Guillaume, T., 2019. Carbon neutral expansion of oil palm plantations in the Neotropics. Sci. Adv. 5, eaaw4418.
Ren, S., He, K., Girshick, R., Sun, J., 2016. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149.
Sabater, A., Montesano, L., Murillo, A.C., 2020. In: Robust and Efficient Post-Processing for Video Object Detection, pp. 10536–10542.
Salscheider, N.O., 2021. Featurenms: Non-maximum suppression by learning feature embeddings. In: 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, pp. 7848–7854.
Selvaraj, M.G., Vergara, A., Montenegro, F., Ruiz, H.A., Safari, N., Raymaekers, D., Ocimati, W., Ntamwira, J., Tits, L., Omondi, A.B., 2020. Detection of banana plants and their major diseases through aerial images and machine learning methods: a case study in DR Congo and Republic of Benin. ISPRS J. Photogramm. Remote Sens. 169, 110–124.
Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9.
Taheripour, F., Hertel, T.W., Ramankutty, N., 2019. Market-mediated responses confound policies to limit deforestation from oil palm expansion in Malaysia and Indonesia. Proc. Natl. Acad. Sci. 116, 19193–19199.
Wakchaure, M., Patle, B., Mahindrakar, A., 2023. Application of AI techniques and robotics in agriculture: a review. Artificial Intelligence in the Life Sciences 3, 100057.
Weinstein, B.G., Marconi, S., Bohlman, S., Zare, A., White, E., 2019. Individual tree-crown detection in RGB imagery using semi-supervised deep learning neural networks. Remote Sens. (Basel) 11, 1309.
Weinstein, B.G., Marconi, S., Bohlman, S.A., Zare, A., White, E.P., 2020. Cross-site learning in deep learning RGB tree crown detection. Eco. Inform. 56, 101061.
Wibowo, H., Sitanggang, I.S., Mushthofa, M., Adrianto, H.A., 2022. Large-scale oil palm trees detection from high-resolution remote sensing images using deep learning. Big Data and Cognitive Computing 6, 89.
Wu, Y., He, K., 2018. Group normalization. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19.
Wu, S., Li, X., Wang, X., 2020b. IoU-aware single-stage object detector for accurate localization. Image Vis. Comput. 97, 103911.
Wu, J., Yang, G., Yang, H., Zhu, Y., Li, Z., Lei, L., Zhao, C., 2020a. Extracting apple tree crown information from remote imagery using deep learning. Comput. Electron. Agric. 174, 105504.

Xu, B., Liang, H., Liang, R., Chen, P., 2023. Synthesize boundaries: A boundary-aware self-consistent framework for weakly supervised salient object detection. IEEE Transactions on Multimedia.

Yarak, K., Witayangkurn, A., Kritiyutanont, K., Arunplod, C., Shibasaki, R., 2021. Oil palm tree detection and health classification on high-resolution imagery using deep learning. Agriculture 11, 183.

Yi, J., Wu, P., Liu, B., Huang, Q., Qu, H., Metaxas, D., 2021. Oriented object detection in aerial images with box boundary-aware vectors. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2150–2159.

Zheng, J., Fu, H., Li, W., Wu, W., Yu, L., Yuan, S., Tao, W.Y.W., Pang, T.K., Kanniah, K.D., 2021. Growing status observation for oil palm trees using Unmanned Aerial Vehicle (UAV) images. ISPRS J. Photogramm. Remote Sens. 173, 95–121.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., Ren, D., 2020. Distance-IoU loss: Faster and better learning for bounding box regression. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 12993–13000.

Zhu, Y., Cai, H., Zhang, S., Wang, C., Xiong, Y., 2020. Tinaface: Strong but simple baseline for face detection. arXiv preprint arXiv:2011.13183.

Zhu, X., Hu, H., Lin, S., Dai, J., 2019. Deformable convnets v2: More deformable, better results. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9308–9316.