JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 40, 993-1004 (2024)
DOI: 10.6688/JISE.202409_40(5).0004

Computer Vision-Based Surveillance of Oil Palm Trees using YOLOv5 and Aerial Imagery Investigation on Stochastic Optimised Hyperparameters

YOHANES NUWARA[1,+], WEI KITT WONG[2] AND FILBERT H. JUWONO[3]
[1]*Corporate IT Division, Asia Pulp and Paper Sinarmas*
[2]*Department of Electrical Engineering, Curtin University Malaysia*
[3]*Department of Electrical and Electronics Engineering*
Xi'an Jiaotong – Liverpool University, 215123 P.R. China
E-mail: yohanes_nuwara@app.co.id[+]

Oil palm trees (*Elaeis guineensis*) is an important species for bio-energy agribusiness. Despite the rapid growth of oil palm tree plantations mostly in tropical countries to support global demand for biofuels, problems such as diseases can reduce the productivity and survival rate of palm trees which have adverse impact to the business. Therefore, palm plantation needs regular tree counting for inventory and health monitoring. Thanks to the rapid development of remote sensing technology deep learning-based computer vision, these two intertwined technologies help to automate tree counting. Continuous improvement in this domain is expected to improve classification. In this study, YOLOv5 model was implemented for tree counting using the palm aerial imagery dataset from Papua, Indonesia. UAV images were used to classification of trees into five distinct classes, namely healthy, smallish, yellowish, mismanaged, and dead palms. We achieved average F1-score of 0.895 for 5 classes, which outperformed Faster R-CNN (0.706) and CNN ResNet-101 (0.493). The strength of our YOLOv5 model is high precision for all 5 classes above 0.961. In the effort to further optimise YOLOv5, further improvements can be achieved by optimising the parameters. This was achieved using Genetic Algorithm to optimise the parameters. The final average F1-score of this model on the five palm classes achieves 0.915. This application provides fast, robust, and accurate oil palm tree counting that can be applied elsewhere in the world.

Keywords: automation, computer vision, hyperparameter evolution, oil palm, tree count, unmaned aerial vehicle, YOLOv5

1. INTRODUCTION

Elaeis guineensis, also known as oil palm tree, is a species native to West and Central Africa and flourishing in tropical countries [1]. Palm trees produce Crude Palm Oil (CPO) which is largely used as feedstock for biofuel production, a clean energy substitute to fossil fuel used for transportation as fuel mixtures.

Indonesia is the world's largest producer with 48.3 million of tonnes of CPO production in 2020, followed by Malaysia as the second largest producer [2]. In fact, according to

Received December 30, 2022; revised March 22 & July 12, 2023; accepted July 27, 2023.
Communicated by King Hann Lim.
[+] Corresponding author.

the report by National Center for Biotechnology Information [3], Indonesia and Malaysia supplied 85% of global palm oil supply. Its wet tropical climate provides ideal conditions for oil palm growth and yield. Sumatra has the largest oil palm plantation in Indonesia, but is expanding rapidly in Kalimantan and further east to Papua [1].

Many problems may occur during the biological growth of oil palm trees, such as pests and diseases which reduce productivity and yield therefore impacting the business. According to [4], some important diseases on palm trees are blast, crown disease, *Fusarium oxysporum* and basal trunk rot, while main pests include palm weevil, slug caterpillar and the red ring nematode.Therefore, monitoring the growth status and disturbances of oil palm trees are crucial for businesses in the industries.

To perform oil palm plantation management, tree count is needed to do inventory of number of trees grown on areal croplands. The process of tree counting done manually is very resource-intensive and time-demanding. Therefore, this process needs to be automated [5]. This automation is called precision forestry technology. Remote sensing using Unmaned Aerial Vehicles (UAVs) has played important role in these precision forestry activities for example tree productivity monitoring, age of trees, tree mapping, nursery inventory management, and individual plant detection, as stated in both [6, 7]. In recent years, computer vision methods, especially deep learning, have become core method in many researches in the detection of trees in forest [8].

Formally, the objective of this research report is dual fold. Firstly, the report seeks to investigate the efficiency of classifying the palm states using YOLOv5. This can be further compared with the state of the art approaches. Subsequently, an optimisation of YOLOv5 was further explored using a well known stochastic optimisation, Genetic Algorithm (GA). The proposed stochastic algorithm is a well implemented optimisation algorithm that thrives even in difficult and highly multi modal problems. As such, the report will first discuss some relevant works in Literature review section. This is followed by the YOLOv5 implementation along with the optimisation schemes. The results are discussed and analysed in the results section followed by some conclusion and take home message in the conclusion section

2. RELATED WORKS

In this study, we restrict our discussion on object detection methods only for deep learning computer vision models. Region Based Convolutional Neural Network (R-CNN) model is the State-of-the-art (SOTA) object detector family widely used for tree counting. The two most popular R-CNN models used for tree counting are Faster R-CNN [9, 10]. In [11], authors used Faster R-CNN to perform detection on banana trees from high resolution RGB images with 4,000×3,000 pixels on multiple altitudes. Image processing methods were used to generate multiple variants from the image based on vegetation indices. The Faster R-CNN used pretrained Inception v2 layer to train on these processed images. Their method achieved F1-score of 0.857. The work by [12] used Faster R-CNN (FRCNN) to perform detection on oil palm trees. The images were produced by Skywalker X8 fixed-wing UAV with 60,000×30,000 pixels size. They modified the FRCNN by proposing Refined Pyramid Feature (RPF) module and used ResNet-101 as backbone. This method achieved F1-score up to 0.911. Authors in [13] used FRCNN to detect co-

conut trees from Google Earth images and achieved F1-score of 0.771.

Authors in [14] used Mask R-CNN to perform detection on young Chinese fir plants (*Cunninghamia lanceolata*) from multiband UAV images. The images had 2-cm/pixel resolution. They derived 6 different combinations from image bands and trained the Mask R-CNN on each combination. The method achieved the highest F1-score of 0.847 on NDVI-CHM combination. Interestingly, they also did tree height prediction using these results with DTM (Digital Terrain Model) and DSM (Digital Surface Model) data. Both DTM and DSM are elevation data obtained from satellite images, UAVs, or LiDAR (Light Detection and Ranging) system. In [15], authors used Mask R-CNN to detect individual trees of an urban forest from Google Earth satellite images. This image has 28,062×6,377 pixel size and 0.27-m spatial resolution. They used a pretrained Mask R-CNN on Microsoft Common Objects in Context (COCO) datasets and achieved F1-score of 0.9.

From the previous discussions, we could see that the two-stage R-CNN models performed better than one-stage detector such as RetinaNet. However, few works reported that another very popular one-stage detector called the You Only Look Once (YOLO) models outperformed the R-CNN. For example, authors in [16] compared YOLOv3 and YOLOv4 with Faster R-CNN and EfficientDet-D5 to perform detection on palm trees from a DJI Mavic Pro image with 4,000×3,000 pixel size. YOLOv3 gave slightly better F1-score of 0.88 compared to Faster R-CNN with F1-score of 0.85 for palm class. YOLOv4 showed the highest Recall while EfficientDet showed the highest Precision. They concluded that YOLOv4 appeared as good trade-off between mAP and inference speed. In [17], authors compared YOLOv2 with Boosted Cascade algorithms, namely Viola-Jones [18] and Aggregate Channel Features (ACF) [19] to perform detection on coconut trees from UAV RGB image with 10,000×10,000 pixel size. Using Darknet-19 as backbone, YOLOv2 achieved F1-score of 0.927 compared to the Boosted Cascade models.

3. METHODOLOGY

Two sites with total area of 29,042 Ha in Papua province, Indonesia, were captured using Skywalker X8 First Person View UAV aircraft flying at altitude of 425 m and cruising at speed of 15-20 m/s. The coordinate for these sites is (140°29'17"E, 6°57'42"S). The two sites have image size of [64273, 27839] pixels and [85957, 31976] pixels, respectively, then cut into smaller tile images with size of [1024, 1024] pixels. These images have spatial resolution of 8 cm and RGB channel. Using 8 cm/pixel spatial resolution, the real extent of one tile image is therefore [81.92,81.92] m or 0.671 Ha of area.

The source of the images used in this study is from [12] which the authors developed MOPAD (Multi-Class Oil Palm Detection), a computer vision model for multiclass oil palm tree counting based on Faster R-CNN with Refined Pyramid Feature module (RPF) and Region Proposal Network (RPN) modules. These datasets are available in GitHub at https://github.com/rs-dl/MOPAD.

These images were annotated into five different classes depending on the health status of the palm trees, namely healthy palm, smallish palm, yellowish palm, mismanaged palm, and dead palm. Healthy palms are those which are normally cultivated and properly grown. Smallish palms have diameter of the palm crown relatively smaller than healthy palms. Two main reason for the difference in size are that the trees are under a

seeding stage or under abnormal growing stage. Yellowish palms have yellowish color in appearance and yellow spots on their leaves that may be caused by pests or diseases. Mismanaged palms are surrounded by weeds or other vegetation, therefore having difficulty to obtain sufficient soil nutrients and sunlight. Dead palms do not have vitality and usually have gray tree crowns.

Two inputs were prepared for training. These inputs were training image as much as 4,000 photos and annotations for 5 palm classes provided as JSON file format. The JSON file followed the convention of Microsoft COCO (Common Object in Context) dataset. Since YOLOv5 by default could not directly process JSON annotation files, we did conversion to ASCII text files with 5 columns, namely name of class, x and y center coordinates of bounding box, width, and height of bounding boxes.

Furthermore, the training images were randomly split into training and validation set using with a composition of 70% and 30% for training and validation. We generated 2,800 training images and 1,200 validation images.

Using batch size of 16, the training took 2.57 hours on NVIDIA GPU Tesla T4. We have set to save the history callbacks and the weights from training at every 3 epochs. After training was finished, we evaluated the bounding box loss and objectness to determine which epoch gave the best detection performance. We found that the best detection performance happened at epoch 65. Using the trained weight at epoch 65, we did inference on 500 tiles photos with pixel size of [1024,1024]. The IoU parameter for inference post-processing using Non-Maximum Suppression (NMS) was set to be 0.45 by default. In YOLOv5, the IoU (Intersection over Union) threshold is used to determine whether a predicted bounding box is considered a true positive or a false positive during object detection. The IoU is a measure of overlap between the predicted bounding box and the ground truth bounding box.meaning that if the IoU between a predicted bounding box and the ground truth box is equal to or greater than 0.45, it is considered a true positive detection. Anything below 0.45 is considered a false positive. The decision to implement the due to the fact that just slight bias to positive is preferred from the 0.5 threshold. This value is highly considered to be a executive decision for classification.

To evaluate the accuracy of our model predicted on every class, we used Precision, Recall, and F1-score metrics. We also used the Mean Average Precision (mAP) as another metric to evaluate the performance of our model.

Genetic Algorithm (GA) is a type of optimization algorithm inspired by the process of natural selection in biology. It is used to find the optimal solution to a problem by searching through a large space of possible solutions and selecting the best ones.

The GA algorithm can be used for selecting the optimal hyperparameters, called the hyperparameter evolution. The GA algorithm for hyperparameter evolution in YOLOv5 consists of two main processes, namely parent selection and hyperparameter mutation. The algorithm will select one or more parent sets of hyperparameters and use them to create a child set of hyperparameters through mutation. The parent selection method can either be single or weighted, and is specified by the parent variable.

The fitness is defined in the Eq. (1) as follows, -0.018in

$$f = w_1 \cdot P + w_2 \cdot R + w_3 \cdot \overline{AP}_{0.5} + w_4 \cdot \overline{AP}_{[0.5,0.95]}, \tag{1}$$

0.02in Where f is fitness, w_n is the weights, P is precision metric, R is recall metric, $\overline{AP}_{0.5}$ is the Mean Average Precision metric at 50% confidence (MAP@0.5), and $\overline{AP}_{[0.5,0.95]}$ is

Table 1. Weight values for genetic algorithm fitness.

Weight	Value
w_1	0
w_2	0
w_3	0.1
w_4	0.9

the Mean Average Precision metric over interval from 50% to 95% confidence (MAP@ 0.5:0.95). These metrics will be explained more in the later chapter. The values of w_n is tabulated in Table 1.

After selecting the parent(s), the algorithm mutates the hyperparameters by randomly adjusting each value with a probability of mp (mutation probability) and a standard deviation of s (sigma). The mutation is performed using a random Gaussian distribution centered at the original value of the hyperparameter. The resulting values are then clipped to a minimum of 0.3 and a maximum of 3.0. The optimisable parameters are shown in Table 2.

4. RESULTS

Training on this multiclass oil palm dataset was challenging because of the highly imbalanced class, especially the mismanaged and dead palms that consist only 17% in total of the whole training annotations. Fig. 1 shows the inference result on 4 samples out of 500 test images. The prediction is annotated with bounding boxes, name of the class, and confidence score. We could see the prediction is dominated by two classes, namely healthy palms (palm 2) and smallish palms (palm 4). The confidence score mostly ranges from 0.90 to 0.96. There are some trees with lower confidence scores which means the predicted class of the object is not really distinct with the other classes, probably because of the palm shape or color similarity.

The Intersection over Union (IoU) is calculated by dividing the area of intersection between prediction and ground truth bounding boxes by the sum of both areas, which is expressed as follows,

$$IoU = \frac{area(GT \cap PD)}{area(GT \cup PD)}, \tag{2}$$

where GT and PD are ground truth and prediction bounding boxes.

The IoU is set at a certain value to be the threshold for deciding whether a predicted bounding box is a true positive, from which we can then calculate Precision and Recall and the area under Precision-Recall curve (AUC). For example, the IoU threshold is 0.5. The AUC at 0.5 threshold is called AP@0.5. The Average Precision (AP) is expressed as follows,

$$AP = \int_0^1 p(r)\,dr. \tag{3}$$

The AP can also be calculated over a range of IoU thresholds from 0.5 to 0.95 which is then called the AP@0.5:0.95. If there are N classes, the AP can be averaged, which is then called mAP.

Table 2. Optimisable features using GA.

Commonly used Annotation	Details
lr0	initial learning rate
lrf	final OneCycleLR learning rate (lr0 * lrf)
momentum	SGD momentum/Adam beta1
weight decay	optimizer weight decay
warm up epochs	warmup epochs
warm up momentum	warmup initial momentum
warmup bias lr	warmup initial bias lr
box	box loss gain
cls	cls loss gain
cls pw	cls BCELoss positive$_{w}$*eight*
obj	obj loss gain (scale with pixels)
obj pw	obj BCELoss positive$_{w}$*eight*
iou	IoU training threshold
anchor$_{t}$	anchor-multiple threshold
anchors	anchors per output layer (0 to ignore)
fl_gamma	focal loss gamma
hsv$_{h}$	image HSV-Hue augmentation (fraction)
hsv_s	image HSV-Saturation augmentation (fraction)
hsv_v	image HSV-Value augmentation (fraction)
degrees	image rotation (+/– deg)
translate	image translation (+/– fraction)
scale	image scale (+/– gain)
shear	image shear (+/– deg)
perspective	image perspective (+/– fraction),
flipud	image flip up-down (probability)
fliplr	image flip left-right (probability)
mosaic	image mosaic (probability)
mixup	image mixup (probability)
copy paste	segment copy-paste (probability)

The YOLOv5 object detection model calculates box loss using this IoU, called the Complete IoU loss or CIoU. It is calculated as follows,

$$CIoU = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v, \tag{4}$$

where b and b^{gt} each represent the center locations of the prediction and ground truth boxes, ρ represents the Euclidean distance between the two center locations, c represents the diagonal distance of the smallest closed area that contains the prediction and ground truth boxes, and both α and v are impact factors. In addition to box loss, YOLOv5 also uses objectness loss using binary cross-entropy (BCE). Objectness determines whether an object exists at an anchor.

All metrics result at epoch 65 are tabulated in Table 3 The mAP@0.5 at epoch 65 achieved 0.915 and the ROC AUC is 0.922. Both Precision (0.882) and Recall(0.860) are quite similar indicating selected IoU is proper. However, further tuning of other parameters may effect these results as they are trade-off values.

Fig. 1. YOLOv5 prediction results on 4 sample test images.

Table 3. Training and validation result.

Parameter	Description
Best epoch	65
Training CIoU	0.031
Validation CIoU	0.027
mAP@0.5	0.915
mAP@0.5:0.95	0.557
Precision	0.882
Recall	0.860

Fig. 2 shows the comparison of ground truth and prediction result on one sample test image. In this image we could see 5 classes from the color of bounding boxes, namely healthy palms (red), smallish palms (blue), yellowish palms (yellow), mismanaged palms (cyan), and dead palms (black). The prediction shows exactly the similarly detected palm class with the ground truth. The model was also able to predict palm trees on the edge of this photo where they were not labeled as ground truth.

Table 4 shows the metrics result after the inference was done on the 500 total test images. There were in total 25,511 supports or trees labeled as ground truth which is divided into 16,670 healthy palms, 8,149 smallish palms, 468 yellowish palms, 158 mismanaged palms, and 66 dead palms. Precision (P), Recall (R), and F1-score ($F1$) are calculated by the following formulas respectively,

$$P = \frac{TP}{FP+TP} = \frac{TP}{PD}, \tag{5}$$

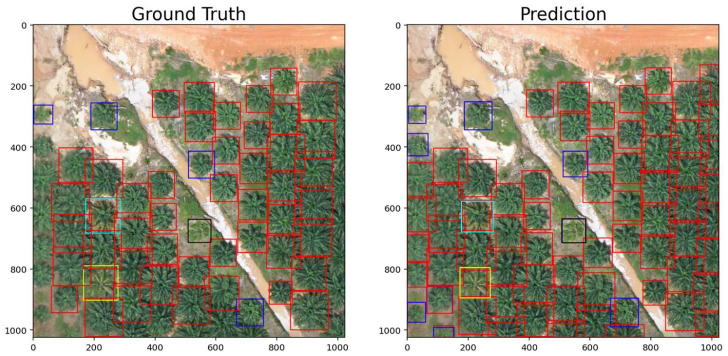


Fig. 2. Ground truth annotations (a) and YOLOv5 predictions (b) on a sample [1024,1024] test image.

Table 4. F1-score on 6 plots on sample test image.

Class No.	Class name	Number of labels	P	R	AP@0.5
1	Dead palm	66	1.000	0.681	0.673
2	Healthy palm	16,670	0.986	0.991	0.994
3	Mismanaged palm	158	1.000	0.675	0.706
4	Smallish palm	8,149	0.961	0.977	0.991
5	Yellowish palm	468	1.000	0.689	0.703
		Average	0.989	0.817	0.813

$$R = \frac{TP}{FN+TP} = \frac{TP}{GT}, \tag{6}$$

$$F1 = \frac{2 \cdot P \cdot R}{P+R}, \tag{7}$$

where PD is the number of predicted instances, GT is the number of ground truths (labeled instances), TP is the number of true positives, FP is the number of false positives, FN is the number of false negatives.

The strength of our YOLOv5 model is the very high precision. The precision for all five classes appears to be very high, ranging from 0.961 to 1. The very high precision means the number of over-detection or false positives is almost negligible. The average precision for all classes is therefore 0.989. Now, we focus on Recall. In this study using YOLOv5 model, two classes, namely the healthy palm and smallish palms have the largest recall of 0.991 and 0.977, respectively. However, the other three classes, namely yellowish, mismanaged, and dead palms have lower recall ranging from 0.675 to 0.751. This means there are more miss-detection or false negatives in these three classes compared to the healthy and smallish palms. This is understood as the difference between two classes such as yellowish and dead palms sometimes can be indistinguishable. Assuming a dead palm that is dying or not entirely dead, the tree can look yellowish in color. Also, the classes were highly imbalanced. As spoken, further improvement was proposed as detailed in previous section. This was achieved vi hyperparameter tuning using GA. As discussed previously, this process is called hyperparameter evolution. After 120 iterations

of Genetic Algorithm process, we obtain the set of most optimum hyperparameters to use for training this palm dataset. Compared to other computer vision models that have been used before for multiclass oil palm detection, our hyperparameter tuned-YOLOv5 model had the winning performance.

Table 5. F1-score benchmark of object detection models on multi-class oil palm case.

Model	Healthy	Dead	Mismanaged	Yellowish	Smallish	Average
RF	0.810	0.000	0.012	0.428	0.160	0.251
SVM	0.861	0.000	0.073	0.409	0.236	0.260
CNN (R-101)	0.871	0.325	0.263	0.4657	0.444	0.288
FRCNN	0.985	0.794	0.174	0.7825	0.440	0.492
Grid RCNN	0.985	0.279	0.251	0.793	0.513	0.552
GA FRCNN	0.983	0.265	0.423	0.797	0.498	0.593
Cascade RCNN	0.986	0.179	0.410	0.803	0.674	0.610
Libra FRCNN	0.984	0.226	0.418	0.826	0.666	0.624
MOPAD	0.994	0.432	0.445	0.895	0.760	0.705
YOLOv5 (fixed param.)	**0.988**	**0.810**	**0.806**	**0.969**	**0.816**	**0.878**
YOLOv5 (Optimised param.)	**0.995**	**0.863**	**0.856**	**0.974**	**0.885**	**0.915**

Table 5 shows the comparison between YOLOv5 (before and after hyperaparameter tuning) with other models used by author in [12]. The MOPAD model has an average F1-score of only 0.705. It can be seen that only two classes have F1-score above 0.8, namely Healthy Palm (0.994) and Yellowish Palm (0.895). It is no surprise, as we have discussed previously, that the other classes tend to have lower F1-score because they are the minority classes. With the proposed YOLOv5 model, the F1-scores of the other classes significantly improve, such as Dead Palm (YOLOv5 0.810 v. MOPAD 0.432) and Mismanaged Palm (YOLOv5 0.806 v. MOPAD 0.445). However, the F1-score for Healthy Palm is scored a slightly lower than the MOPAD model (YOLOv5 0.988 v. MOPAD 0.994). Our approach with hyperparameter tuning successfully improved the F1-scores of all the minority classes. The F1-score of Healthy Palm achieves 0.995. On a comparison between Optimised parameter models vs non- optimised parameters, this is clearly shown in Table.5. The detection F1-Score is consistently improved in every single category. This is clear indication on the improvement achieved. The improvement can be attributed to the well optimised hyperparameter. At least in the case of palm oil aerial images, the difference between non- optimised (fixed) hyperparameter and intervention using some form of optimisation on the hyperparameter is obvious in which approximately 5% can be expected. to visualise the effect, the IoU threshold can be a good example. In a particular setting, the particular IoU threshold affects the tradeoff between sensitivity and precision, and subsequently the F1-score. small changes on the hyperparameter can have huge effect on the F1-score. Manually tuning the hyperparameters is not feasible as there increment in a single objective may effect another. However, it needs to be highlighted that computation training time increases at scale of *GA population* $\times$ *generational iterations*. At an increased computation cost, the results showed a significant improvement of approximately 5% on F1-score.

5. CONCLUSION

In this study, we have implemented YOLOv5 object detection model for counting and classification of tree health on oil palm tree plantation UAV orthophotos. We showed that YOLOv5 was robust enough to classify five palm classes, namely healthy, smallish, yellowish, mismanaged, and dead trees. The strength of this model was its precision above 0.961 for all classes. Due to the imbalanced number of classified classes, the recall values were lower in the 3 minority classes, namely the yellowish, mismanaged, and dead palm, as the recall values were in the range of 0.67 and 0.69. Despite low recall, our YOLOv5 model successfully achieved recall (0.813) that is better than other models, such as the MOPAD (Multi-Class Oil Palm Detection) model which runs on Faster R-CNN with with Refined Pyramid Feature (RPF) module. Comparing the overall detection accuracy measured by F1-score, the average F1-score from YOLOv5 (0.878) was better than MOPAD (0.705) and other models implemented on the same dataset. Therefore, the YOLOv5 was proven to be successful for implementation in the precision agriculture where the objective is to automate tree counting and tree health surveillance to maximize yield and production.

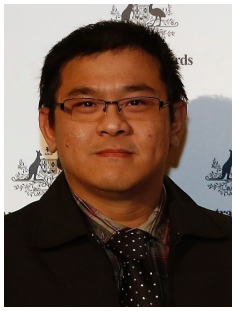
REFERENCES

1. D. Sheil, A. Casson, E. Meijaard, M. van Noordwijk, J. Gaskell, J. Sunderland-Groves, K. Wertz, and M. Kanninen, "The impacts and opportunities of oil palm in southeast Asia," Center for International Forestry Research (CIFOR), Indonesia, 2009.
2. D. O. Darkwah and M. Ong-Abdullah, "Sustainability of the oil palm industry," *Elaeis Guineensis*, H. Kamyab, ed., Intech Open, 2021.
3. N. C. for Biotechnology Information, *The Nexus of Biofuels, Climate Change, and Human Health: Workshop Summary*, Ch. Case Study: The Palm Oil Example, National Academies Press, Washington, 2014.
4. W. Verheye, *Land Use, Land Cover and Soil Sciences, Encyclopedia of Life Support Systems (EOLSS)*, Chapter, Growth and Production of Oil Palm, UNESCO-EOLSS Publishers, 2010.
5. X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, Vol. 5, 2017, pp. 8-36.
6. O. Guldogan, J. Rotola-Pukkila, U. Balasundaram, T.-H. Le, K. Mannar, T. M. Chrisna, and M. Gabbouj, "Automated tree detection and density calculation using unmanned aerial vehicles," in *Proceedings of Conference on Visual Communications and Image Processing*, 2016, pp. 1-4.
7. F. Gnädinger and U. Schmidhalter, "Digital counts of maize plants by unmanned aerial vehicles (UAVs)," *Remote Sensing*, Vol. 9, 2017, p. 544.
8. B. G. Weinstein, S. Marconi, S. Bohlman, A. Zare, and E. White, "Individual tree-crown detection in RGB imagery using semi-supervised deep learning neural networks," *Remote Sensing*, Vol. 11, 2019, p. 1309.

9. S. Ren and K. He, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, Vol. 28, 2015, pp. 1-9.
10. K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of IEEE International Conference on Computer Vision*, 2017, pp. 2961-2969.
11. B. Neupane, T. Horanont, and N. D. Hung, "Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV)," *PLoS ONE*, Vol. 14, 2019, pp. 1-22.
12. J. Zheng, H. Fua, W. Li, W. Wu, L. Yu, S. Y. Wai, Y. W. Tao, T. K. Pang, and K. D. Kanniah, "Growing status observation for oil palm trees using unmanned aerial vehicle (UAV) images," *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 173, 2021, pp. 95-121.
13. J. Zheng, W. Wu, L. Yu, and H. Fu, "Coconut trees detection on the tenarunga using high-resolution satellite images and deep learning," in *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*, 2021, pp. 6512-6515.
14. Z. Hao, L. Lin, C. J. Post, E. A. Mikhailova, M. Li, Y. Chen, K. Yua, and J. Liu, "Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (mask R-CNN)," *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 178, 2021, pp. 112-123.
15. M. Yang, Y. Mou, S. Liu, Y. Meng, Z. Liu, P. Li, W. Xiang, X. Zhou, and C. Peng, "Detecting and mapping tree crowns based on convolutional neural network and google earth images," *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 108, 2022, p. 102764.
16. A. Ammar, A. Koubaa, and B. Benjdira, "Deep-learning-based automated palm tree counting and geolocation in large farms from aerial geotagged images," *Agronomy*, Vol. 11, 2021, p. 1458.
17. S. Puttemans, K. van Beeck, and T. Goedemé, "Focal loss for dense object detection," in *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, Vol. 5, 2018, pp. 230-241.
18. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 5, 2001, pp. 230-241.
19. E. Ohn-Bar and M. M. Trivedi, "To boost or not to boost? on the limits of boosted trees for object detection," in *Proceedings of the 23rd International Conference on Pattern Recognition*, 2016, pp. 3350-3355.

Yohanes Nuwara graduated with a B.Eng in Geophysical Engineering in 2019. He is currently an Expert Data Scientist at Asia Pulp and Paper Sinarmas in Indonesia. His work mainly focuses on computer vision methods for Light Detection and Ranging (LiDAR) photogrammetry, tree, and weed detection from aerial images. He previously worked with oil and gas geophysical consulting services, OYO Corporation, in Japan, from 2020-2021. He had research in Carbon Capture, Utilization, and Storage (CCUS) technology. He also frequently lectured machine learning and data science in universities, such as University of Oklahoma (USA), Marietta College (USA), and Chandigargh University (India).

Wei Kitt Wong received the M.Eng and Ph.D. degrees from Universiti Malaysia Sabah in 2012 and 2016, respectively. Prior to joining academia, he was with the telecommunication and building services industry. He is currently serving as an Associate Professor with the Department of Electrical and Computer Engineering, Curtin University Malaysia. His research interests include embedded system development, machine learning applications, and image processing.



Filbert Hilman Juwono received the B.Eng. degree in Electrical Engineering and the M.Eng. degree in Telecommunication Engineering from the University of Indonesia, Depok, Indonesia, in 2007 and 2009, respectively, and the Ph.D. degree in Electrical and Electronic Engineering from The University of Western Australia, Perth, WA, Australia, in 2017. He is currently an Assistant Professor with University of Southampton Malaysia. His research interests include signal processing for communications, wireless communications, power-line communications, machine learning applications, and biomedical engineering. He was a recipient of the prestigious Australian Awards Scholarship, in 2012. He serves as an Associate Editor for IEEE Access, a Review Editor for Frontiers in Signal Processing, and the Editor-in-Chief for a newly established journal Green Intelligent Systems and Applications.