Improved Fast R-CNN with Fusion of Optical and 3D Data for Robust Palm Tree Detection in High Resolution UAV Images

Zi Yan Chen and Iman Yi Liao

Abstract—Palm density records are crucial for fertilizer, yield and biomass estimation. Traditionally, workers have to count the number of standing palms on the ground, which is physically arduous and costly. Remote sensing imageries such as unmanned aerial vehicle (UAV) data provide an efficient way to partially or completely eliminate the needs of physical counting. This paper proposes to fuse 3D digital surface model (DSM) and Red, Green and Blue (RGB) image to detect region of interest (ROIs) and subsequently classify palm trees based on Fast Region-based Convolutional Neural Network (Fast R-CNN) architecture. The proposed method reduced computation time by passing the ROI extracted from DSM using local maximum filtering (LM) to the convolutional feature map of the RGB image for bounding box regression and classification. Results showed that the proposed method detected palm trees in high resolution UAV images 5 times faster and 2.5 to 4.5% more accurate than the state-of-the-art Faster R-CNN. It successfully achieved 99.8%, 100% and 91.4% average accuracy in young, mature and mixed vegetation areas, respectively. Results also showed that unlike Faster R-CNN and YOLO V2, the accuracy of the proposed method was not affected by the input image size.

Index Terms—Convolutional neural network (CNN), digital surface model (DSM), palm tree detection, unmanned aerial vehicle (UAV), region proposal network (RPN).

I. INTRODUCTION

Effective and efficient management of large scale oil palm plantations still poses a challenge for many estate managers. The managers need to be equipped with adequate and updated information such as palm tree record, palm health, pest and diseases status of their plantations etc. in order to make sound decisions and implement yield improvement projects. One of the ways to increase palm oil production is to ensure that optimal number of palms is planted in fields. Knowing the updated palm tree record is also crucial to estimate fertilizer cost which accounts for approximately 24% of total production cost [1]. The most frequently used method in obtaining palm tree record is by manual inspection on the ground. However, this method is physically arduous, error-prone and costly. With the rapid development of remote sensing technologies, the need for ground inspection can be partially or completely eliminated while the speed of counting can be accelerated as workers can now identify and

count the palms easily on remote sensing images.

The use of unmanned aerial vehicle (UAV) has become very common in agricultural industry nowadays due to its affordability, ease of use, higher spatial resolution, cloud-free, and higher revisit frequency. Higher spatial resolution image allows detection of greater details and smaller objects in the image. However, it presents a bigger challenge in processing due to massive and high dimensional data. Convolutional neural network (CNN), as a new era in machine learning, has achieved state-of-the-art performance in image classification including palm tree detection in UAV images as reported in [2], [3] and [4]. Two types of data are usually exploited with UAV technologies for palm tree detection, i.e., optical data and 3D digital surface model (DSM). DSM comprises dense 3D point cloud derived from structure-from-motion (SfM) photogrammetry workflow from a set of 2D images [5]. Unlike Light Detection and Ranging (LiDAR) point cloud, SfM-derived point cloud can be obtained without additional cost during generation of orthomosaics [6].

In this paper, we propose a novel approach that incorporates SfM-derived DSM with state-of-the-art region-based CNN, namely Fast R-CNN [7] for detecting palm trees in high resolution UAV images. We hypothesize that using DSM as prior information about location of objects (palm trees), the search space in images can be reduced and hence detection time can be reduced. Also, the rate of false positives can be potentially decreased.

The rest of the paper is organized as follows. Section II gives a summary of previous works on palm tree detection and region-based CNN algorithms. Section III introduces the proposed method. The experimental results with analysis are shown in Section IV. Finally, conclusion is described in Section V.

II. RELATED WORK

A. Palm Tree Detection

Palm tree detection in an image refers to detection and localization of all instances of palm tree. Traditionally, this problem was addressed by using handcrafted visual features such as color histogram, gradient and textures on top of strong classifiers such as Support Vector Machine (SVM). For example, [8] extracted scale-invariant feature transform (SIFT) keypoints of date palms from UAV images and trained on an extreme learning machine to classify these features into palm and non-palm categories. In [9], shape feature called circular autocorrelation of the polar shape matrix was used to represent palm tree in UAV images and

Manuscript received July 20, 2019; revised December 17, 2019.

Zi Yan Chen is with the School of Computer Science, University of Nottingham Malaysia and Advanced Agricoecological Research Sdn. Bhd., Malaysia (e-mail: chenzy@aarsb.com.my).

Iman Yi Liao is with the School of Computer Science, University of Nottingham Malaysia (e-mail: Iman.Liao@nottingham.edu.my).

SVM was applied for classification. Similarly, histogram of gradients (HOG) [10], local binary patterns (LBP) [11] and HAAR-like features [12] were used to extract shape and texture features from satellite or UAV images. SVM was utilized in the studies for classification and achieved up to 100% detection accuracy especially in young palm areas.

The bottleneck of the handcrafted techniques requires careful design of features in the feature extraction stage and fine-tuning of parameters in the training stage. Thus, CNN based end to end object detection techniques were further exploited to further increase the efficiency of palm tree detection, as reported in [2] and [3]. The main benefit of utilizing CNN for object detection is that features can be effectively extracted from images without going through complex handcrafted feature extraction process, and without having prior knowledge about the feature representation of the object. In addition, the introduction of transfer learning [13] in deep learning models allows users to use a small dataset to fine-tune a network already pre-trained with millions of images. This is much faster than training a CNN from scratch.

The above-mentioned studies commonly used sliding window technique for providing proposals. The feature extraction and classification algorithm have to be applied on a large number of proposals. As significant amount of computational time is spent on regions with no objects, it has reduced the computational efficiency and increased the possibility of false detections. In view of this, 3D data derived from SfM point clouds were utilized to identify tree locations quickly using local maximum filtering (LM) [5], [14]. The common assumption that the pixel of tree top has higher elevation than its neighboring pixels thus can be localized as tree center. Computation of LM is faster and easier than extracting handcrafted features and does not require training. However, directly applying LM for tree detection can be erroneous because other non-tree objects can also be included.

Studies on the fusion of 3D model and 2D visual feature for palm tree detection are limited. One of the studies in [15] used image segmentation techniques to group pixels with homogenous properties as an individual object (palm) and used LiDAR data to estimate the palm height. However, the approach is parametric and requires heavy user input. Therefore, it is necessary to develop an effective and automated algorithm for searching proposals in UAV images to improve detection speed and decrease false positive rate.

B. Region-Based Convolutional Neural Networks

To eliminate the need of exhaustive selection of regions, [16] proposed a framework called R-CNN which combines a region proposal stage and CNN to selectively extract around 2000 region proposals in an image. The region proposal stage generates class-agnostic proposals in image to reduce the number of region of interest (RoI) and to maintain meaningful proposals. However, R-CNN uses three separate models, one for feature extraction, one for bounding box classification, and one for regressing and fine-tuning the bounding box position and size. Classification of CNN model will have to run around 2000 times per image, which compromises speed. An improved version called Fast R-CNN was proposed later by the same author [4]. Fast

R-CNN combines the feature extraction, classification and bounding regression stages in a unified framework. Instead of feeding each of the region proposals to CNN, the entire image is fed into the CNN to obtain a convolutional feature map and the features are shared across the 2000 region proposals for subsequent RoI pooling and detection. This saves significant time on performing forward pass. However, the selective search algorithm used in Fast R-CNN is slow and time-consuming. Later on, [17] proposed Faster R-CNN to substitute the selective search with a shallow CNN called Regional Proposal Network (RPN). It slides a small network over the feature maps by the last shared convolutional layer of CNN to generate proposals and their corresponding probabilities of containing object. This greatly improves the speed of modelling. The outputs of the RPN are passed to Fast R-CNN component for final classification and bounding box regression. The framework can be considered as a combination of Fast R-CNN and RPN and is trained end-to-end. YOLO [18] further improves the detection speed by unifying the bounding box prediction and CNN into a single neural network. The algorithm divides the input image into regions, then it predicts the coordinates and probabilities of bounding boxes. Nevertheless, for a feature map of size $W \times H$ with k region proposals at each sliding location, there are still WHk region proposals in total to be processed.

Although the more efficient one stage region-based CNNs such as YOLO and SSD [19] unify ROIs generation and computation in the same deep network by sharing convolutional layers of the same data, such network architecture is not flexible to allow non-image data sources. Furthermore, the one-stage region-based CNNs would resize all input images to a fixed size. The dimension of high resolution UAV images is usually very huge; resizing them to a fixed small dimension will cause problems in detecting small objects like palm trees. Although large images can be cropped into multiple smaller regions for detection, this is not an efficient approach because the small images have to pass through the convolutional network multiple times and overlap between two images is required to detect objects which appear at the image edges. The latter results in more images to be processed and subsequently requires higher computational cost than processing a bigger image at once. On the other hand, Fast R-CNN has separate region proposal and CNN networks, thus allowing more room for improvement, i.e. using additional data and works on any dimension of input images.

III. IMPROVED FAST R-CNN APPROACH WITH DATA FUSION

A. Proposal Generation Using 3D Data

We propose an improved technique to use Fast R-CNN as base network and DSM derived from SfM 3D point cloud to guide the generation of regions of interests (ROIs). The architecture is summarized in Fig. 1. The approach comprises of two modules. The first module is a simple LM filtering process that takes a DSM as input and generates potential palm tree locations as outputs. The LM filtering can be considered as a spatial constraint to guide the CNN model in searching for objects and making it faster for prediction. In

order to not miss out on any potential palm trees, the smallest kernel size (3×3) was used to search for local maxima. Unlike Selective Search [20] and RPN that extract many ROIs with or without objects, LM filtering provides higher quality ROIs that are difficult to be obtained from 2D images. Moreover, the operation of LM algorithm is very light and does not require training thus can compute palm tree proposals efficiently. The idea of the improved technique shares similar idea as in [21], i.e., using 3D LiDAR depth map and RGB image to increase detection accuracy. However, [21] extracted region proposals from both 3D and 2D data using Selective Search requires high computational cost.

To account for palm tree of varying sizes, the concept of anchor boxes as implemented in RPN was used in this study. Three anchor boxes of size 40×40 , 80×80 and 120×120 pixels corresponding to the minimum, average and maximum canopy sizes were introduced at each potential palm tree location. The second module, which is the Fast R-CNN network first convolved input RGB image with a series of a number of convolutional and max pooling layers to a feature map. Then, the three anchor boxes were projected onto the corresponding feature map for ROI extraction. Since there are three ROIs of different sizes, a ROI pooling layer was used to extract a fixed-length feature vector. Each feature vector was subsequently fed into a sequence of fully connected layers and finally two sibling convolution layers: a box-regression layer that outputs bounding box related values and a classification layer that outputs the softmax probability of palm tree and "background" classes.

B. Fine-tuning for Palm Tree Detection

The VGG16 model pre-trained on ImageNet was employed as the base network. 94 partitioned images and DSM of 400×400 pixel was randomly selected from the entire UAV image of $30,000 \times 30,000$ pixel. Each image contained 10-20 palm trees and the bounding box of each palm tree was manually drawn as ground-truth. This will bias towards positive samples as they are dominating. Thus, additional 100 locations were randomly sampled in each image to increase negative sample size. Training sample is considered as positive if its overlap degree with the ground-truth is greater than 0.6. If the overlap is less than 0.1, it is labelled as negative. The rest of the training samples with overlap between 0.1 and 0.6 are discarded. The losses of box-classification and box-regression layers are computed separately and combined as final loss:

$$L(p, p^*, t, t^*) = L_{cls}(p, u) + \lambda[u \geq 1]L_{reg}(t, t^*) \quad (1)$$

in which, $L_{cls} = -\log p_u$ is the classic cross-entropy log loss of true class u . The second task loss, L_{reg} is defined as:

$$L_{reg}(t, t^*) = \sum_{i \in x, y, w, h} \text{smooth}_{L1}(t_i - t_i^*) \quad (2)$$

in which

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{If } |x| < 1 \\ |x| - 0.5 & \text{otherwise,} \end{cases} \quad (3)$$

t and t^* are vectors defined over four parameterized

coordinates of predicted bounding box corrections and true bounding box corrections respectively where:

$$t_x = \frac{(x-x_a)}{w_a}, t_y = \frac{(y-y_a)}{h_a} \quad (4)$$

$$t_w = \log\left(\frac{w}{w_a}\right), t_h = \log\left(\frac{h}{h_a}\right) \quad (5)$$

$$t_x^* = \frac{(x^*-x_a)}{w_a}, t_y^* = \frac{(y^*-y_a)}{h_a} \quad (6)$$

$$t_w^* = \log\left(\frac{w^*}{w_a}\right), t_h^* = \log\left(\frac{h^*}{h_a}\right) \quad (7)$$

Variables x, y, w , and h denote the predicted box's center coordinates and its width and height. Likewise, x_a, y_a, w_a and h_a and x^*, y^*, w^* , and h^* correspond to the four parametrized coordinates of anchor box and ground-truth respectively. This can be interpreted as regressing a bounding box from an anchor box to a nearby ground-truth box. The Inversion bracket indicator function $[u \geq 1]$ evaluates to 1 if the anchor is positive (palm), and is 0 if the anchor is negative (background). L_{reg} is ignored for background ROIs since ground-truth bounding box is not labelled. λ controls the weights of both losses and it was set to 1. For more detailed discussion of this objective function and the recommended value of parameters, we refer readers to [7] and [17].

In this study, one image per mini batch was randomly sampled for training. Stochastic gradient descent (SGD) solver with base learning rate of 0.0001 was run for 10 epochs to minimize the objective function. Data augmentation was ignored since there is little visual difference between palm tree objects. The mean mini-batch accuracy obtained after fine-tuning was 96.88%.

C. Study Area

The study was carried out in an oil palm estate located in Kuala Ketil, a town about 80 km away from Penang town, Malaysia. The study area was planted with a mixture of oil palm trees (*Elaeis guineensis*) of different ages and on different terrain. An off-the-shelf quadcopter drone DJI Phantom 4 equipped with 12.4 mega pixel camera with RGB was used in capturing the images. The aerial photos were post-processed using Pix4D Desktop Professional. Afterwards, a 10 cm orthophoto was produced. DSM was also generated based on SfM photogrammetry point cloud. Three study areas were cropped from the entire UAV image. The first study site represents a young palm area where the canopy has not overlapped. The second site represents a mature palm area where the canopy is heavily overlapped and only a small part is bare ground, which is more challenging than the first case. The third case is the most challenging as it is mixed with other crops and buildings. The resolution of each cropped image is 1300×1300 pixels.

D. Implementation Details

To evaluate the performance of the proposed method, it was compared with Faster R-CNN, exhaustive Sliding Window method and one stage YOLO V2 approach. AlexNet and VGG16 were selected as base networks for comparison. The experiments were developed using MATLAB R2019a and utilized MATLAB Deep Learning Toolbox to build the deep learning architecture and Computer Vision Toolbox for

feature detection and extraction. They were performed in MATLAB Online, a web-based version of MATLAB.

Fig. 1. Architecture of the proposed method using AlexNet as base network.

At test time, the cropped test images (1300×1300 pixels) were partitioned into multiple sub-images with the size of 400×400 pixels each via a sliding window, with stride=150 pixels and ran through the trained models. The stride ensured all regions especially the image edges were analyzed, but also resulted in overlapping detections on the edges. Such problem was alleviated using non-maximum suppression (NMS).

This study also evaluated the effect of different image dimensions on the detection accuracy and speed. A new test image of size 2000×2000 pixels near Site 3 was cropped from the whole UAV image. Subsequently, this was then partitioned into 10 sub-images of different sizes, ranging from 200×200 pixels up to 2000×2000 pixels and fed into the proposed method, Faster RCNN, and YOLO V2 using VGG16 as base network for evaluation.

E. Evaluation Methods

The evaluation of tree detection was based on counting the palms correctly detected or missing. The measures were based on Precision (P), Recall (R) and $F1 - score$. P (8) is related to commission error while R (9) is related to omission error. $F1 - score$ (10) is the average of both measures. Evaluation was based on number of correctly detected palms (TP), number of falsely detected palms (FP), number of missing palms (FN) and number of correctly detected non-palms (TN).

$$P = \frac{TP}{TP+FP} \quad (8)$$

$$R = \frac{TP}{TP+FN} \quad (9)$$

$$F1 - score = \frac{2.P.R}{P+R} \quad (10)$$

IV. EXPERIMENTAL RESULTS

A. Detection Accuracy of the Proposed Method

Table I shows the detection accuracy in terms of precision, recall and F1-Score based on the proposed method in three study areas. Results showed that the proposed method

achieved F1-score of 100% in Site 2 and 99.8% in Site 1. The precision in Site 3 was low because all of the coconut trees were mistakenly identified as palm trees. The pictorial descriptions of the results are shown in Fig. 2(a)-(c). The recall was close to 1 showing that almost all palm trees were successfully detected. In future, providing hard samples for training can possibly decrease the number of false positives.

TABLE I: DETECTION RESULTS OF THE THREE STUDY AREAS BASED ON THE PROPOSED METHOD (VGG16)

Site	TP	FP	FN	Precision	Recall	F1-score
1	215	1	0	0.995	1.000	0.998
2	151	0	0	1.000	1.000	1.000
3	74	11	3	0.871	0.960	0.914

(a) Site 1: Young Palm Area

(b) Site 2: Mature Palm Area

(c) Site 3: Mixed Forest Area

Fig. 2 (a)-(c). Palm tree detection results at three sites using proposed method. Red circles indicates false positives and blue circles indicate false negatives.

B. Performance Comparison with Other Methods

Table II shows that the proposed method outperformed Sliding Window and Faster R-CNN methods in terms of F1-score and detection speed per image. The performance of this method was very close to the one-stage YOLO V2 that achieved the highest accuracy among all models and also tested 5 times faster than Faster R-CNN. The reason for the outstanding improvement in speed is that the proposed method only needed to evaluate RoIs at sparse locations identified using LM filtering, reducing the need of processing large number of 'background' proposals. Also, LM filtering successfully reduced the number of false positives. RPN in Faster R-CNN exhaustively analyzed all positions in feature map, thus increasing the possibility of false detections. The proposed method with AlexNet as the base network provided the fastest detection per image since its network architecture is shallower than VGG16, but at the cost of accuracy.

TABLE II: PERFORMANCE OF DIFFERENT APPROACHES IN RESPECTIVE TO AVERAGE RECALL PRECISION, F1-SCORE AND DETECTION TIME PER IMAGE (400×400 PIXELS) OVER THREE STUDY AREAS

Methods	Precision	Recall	F1-score	Time (s)
Sliding Window (AlexNet)	0.840	0.990	0.909	140.10
Sliding Window (AlexNet) + DSM	0.960	0.980	0.970	1.16
Faster R-CNN (AlexNet)	0.776	1.000	0.874	0.39
Proposed Method (AlexNet)	0.856	0.980	0.914	0.08
Faster R-CNN (VGG16)	0.897	1.000	0.946	1.37
Proposed Method (VGG16)	0.955	0.987	0.971	0.35
YOLO V2 (VGG16)	0.962	1.000	0.981	0.29

C. Effect of Input Image Size on Accuracy

Detecting palm trees in large-scale UAV images is a challenging task because each palm tree occupies only a minor portion of the images. The resolution of palm tree is reduced in deep convolutional feature map caused by repeated down-sampling of CNN and usually would be too small to contain discriminative information for reliable classification. Results in Fig. 3 showed that the accuracy of the proposed method was not affected by the image size whereas the accuracy of Faster R-CNN reduced gradually when the image size was higher than 1000×1000 pixels due

to the increase in false positive rate.

On the other hand, YOLO V2 experienced sudden decline in accuracy and resulted in 100% false negative rate when the image size was bigger than 800×800 pixels. This is because YOLO V2 resized all input images to the training sample resolution, i.e. 400×400 pixels to maintain fast detection speed. As such, the resolution of a single palm tree has become very low and smaller than the cell size of the feature map. Therefore, it is hard to be distinguish palm trees from generic clutter in the background. Although this can be improved by training the network using higher resolution images, it will increase the computation time and memory consumption quadratically. In contrast, both Faster R-CNN and proposed method did not resize input images to a fixed resolution thus provides greater flexibility in feeding image of various sizes to the networks.

Fig. 3. Comparison of palm detection F1 score with respect to different image dimensions.

Fig. 4. Comparison of palm detection time with respect to different image dimensions.

D. Effect of Input Image Size on Detection Time

Considering that using large-scale UAV images directly will result in high time and memory consumption for training and testing, the large images are usually cropped with a much smaller scale with some overlaps between sub-images and processing the sub-images asynchronously. However, results in Fig. 4 showed that too many sub-images will increase computational time. This is because each sub-image has to be analyzed independently and the network has to process redundant information due to the overlapping regions between sub-images.

Since the proposed method and Faster R-CNN could handle large images without cropping, therefore their rate of increase in detection time was similar to the rate of increase in image size. In contrast, multiple sub-images of size 400×400 pixels have to be cropped from the whole images for YOLO V2 detection to maintain high accuracy. Consequently, the rate of increase in detection time was faster than the rate of increase in the image size. Despite YOLO V2 had the fastest speed in detecting a single 400×400 pixels sub-image, its speed gradually deteriorated when the whole image size increased and eventually outperformed by the proposed method with a wide margin in 2000×2000 pixels image. This implies that the proposed method is still the best option to detect palm trees in high resolution UAV images.

V. CONCLUSIONS

This paper presents an improved Fast R-CNN that incorporated both DSM information and RGB data to increase detection accuracy and speed. Results showed that the proposed method detected images 5 times faster than Faster R-CNN and successfully achieved average accuracy of 91.4 to 100% in the three study areas. Its detection speed and accuracy were higher than the state-of-the-art one-stage YOLO V2 in the highest resolution UAV image. An advantage of the proposed method resides in the fact that it can be applied to images of any size and maintain the same accuracy. The class-agnostic proposals generated from DSM can also be utilized to detect other objects such as vehicles and buildings.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

ACKNOWLEDGMENT

The authors would like to thank Boustead Plantation Berhad and Kuala Lumpur Kepong Berhad for their financial backing given to the research activities leading to this article, and the permission to publish the findings.

REFERENCES

- [1] K. Goh, C. Teo, P. Chew, and S. Chiu, "Fertiliser management in oil palm-agronomic principles and field practices," *Fertilizer Management for Oil Palm Plantations*, vol. 20, p. 21, 1999.
- [2] E. K. Cheang, T. K. Cheang, and Y. H. Tay, "Using convolutional neural networks to count palm trees in satellite images," *arXiv preprint arXiv:1701.06462*, 2017.
- [3] N. A. Mubin, E. Nadarajoo, H. Z. M. Shafri, and A. Hamedianfar, "Young and mature oil palm tree detection and counting using convolutional neural network deep learning method," *International Journal of Remote Sensing*, pp. 1–16, 2019.
- [4] M. Xia, W. Li, H. Fu, L. Yu, R. Dong, and J. Zheng, "Fast and robust detection of oil palm trees using high-resolution remote sensing images," *Automatic Target Recognition XXIX*, vol. 10988, International Society for Optics and Photonics, p. 109880C, 2019.
- [5] D. Fawcett, B. Azlan, T. C. Hill, L. K. Kho, J. Bennie, and K. Anderson, "Unmanned aerial vehicle (UAV) derived structure-from-motion photogrammetry point clouds for oil palm (*Elaeis Guineensis*) canopy segmentation and height estimation," *International Journal of Remote Sensing*, pp. 1–23, 2019.
- [6] C. Stone, M. Webster, J. Osborn, and I. Iqbal, "Alternatives to LiDAR-derived canopy height models for softwood plantations: a review and example using photogrammetry," *Australian Forestry*, vol. 79, no. 4, pp. 271–282, 2016.

- [7] R. Girshick, "Fast R-CNN," presented at 2015 IEEE International Conference on Computer Vision (ICCV), Dec. 2015.
- [8] S. Malek, Y. Bazi, N. Alajlan, H. AlHichri, and F. Melgani, "Efficient framework for palm tree detection in UAV images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 12, pp. 4692–4703, Dec 2014.
- [9] A. Manandhar, L. Hoegner, and U. Stilla, "Palm tree detection using circular autocorrelation of polar shape matrix," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 3, p. 465, 2016.
- [10] Y. Wang, X. Zhu, and B. Wu, "Automatic detection of individual oil palm trees from UAV images using HOG features and an SVM classifier," *International Journal of Remote Sensing*, pp. 1–15, 2018.
- [11] C. Rueda, J. Miserque, and R. Laverde, "Validation of an oil-palm detection system based on a logistic regression model," in *Proc. 2016 IEEE ANDESCON*, Oct 2016, pp. 1–4.
- [12] S. Daliman, S. Abu-Bakar, and S. M. N. Azam, "Development of young oil palm tree recognition using HAAR-based rectangular windows," in *Proc. IOP Conference Series: Earth and Environmental Science*, 2016, vol. 37, no. 1, p. 012041.
- [13] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *Advances in Neural Information Processing Systems*, vol. 27, pp. 3320–3328, 2014.
- [14] T. Kattenborn, M. Sperlich, K. Bataua, and B. Koch, "Automatic single tree detection in plantations using UAV-based photogrammetric point clouds," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, no. 3, p. 139, 2014.
- [15] H. M. Rizzei, H. Z. Shafri, M. A. Mohamoud, B. Pradhan, and B. Kalantar, "Oil palm counting and age estimation from Worldview-3 imagery and LiDAR data using an integrated OBIA height model and regression analysis," *Journal of Sensors*, 2018.
- [16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.
- [17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.
- [18] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015.
- [19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: Single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015.
- [20] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.
- [21] T. Kim and J. Ghosh, "Robust detection of non-motorized road users using deep learning on optical and LiDAR data," in *Proc. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2016, pp. 271–276.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Zi Yan Chen is a PhD student in the School of Computer Science, University of Nottingham Malaysia. He holds a bachelor of science in geoinformatics from the Universiti Teknologi Malaysia. His current research interest emphasizes on machine learning, computer vision, remote sensing and GIS. He is a geospatial analyst and agronomist at Advanced Agriecological Research Sdn. Bhd.

Iman Yi Liao obtained her BSc, MSc, PhD all from School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an, China, in 1996, 1999, and 2006 respectively. She was a visiting researcher in Chinese University of Hong Kong from July 2001 to Dec 2001, and in CVSSP, University of Surrey, UK from Dec. 2003 to Sep. 2004, respectively. She joined the School of Computer Sciences, Universiti Sains Malaysia in 2008 as a post-doctorate research fellow and subsequently became a senior lecturer there until 2012. She is now an associate professor in the School of Computer Science, University of Nottingham Malaysia Campus.