Multi-class Oil Palm Trees Condition Detection from UAV Images using Faster R-CNN with EfficientNetV2

Arinal Haq
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
6025231064@student.its.ac.id

Eko Sugeng Cahyadi
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
6025231038@student.its.ac.id

Muhammad Shafhi Kasyfillah
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
6025231053@student.its.ac.id

Riyanarto Sarno
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
riyanarto@if.its.ac.id

Agus Tri Haryono
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
7025231020@student.its.ac.id

2024 2nd International Conference on Technology Innovation and Its Applications (ICTIIA) | 979-8-3503-5161-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICTIIA61827.2024.10761449

***Abstract*— In regions like Indonesia, known for extensive Oil Palm tree production, tree counting and tree condition detection accuracy are pivotal in assessing and forecasting the country's oil palm tree production. Faster R-CNN emerges as a deep learning method suitable for tree condition detection to increase detection precision. This paper proposes modifications to the backbone of Faster R-CNN to enhance tree condition detection performance, particularly when applied to UAV images for assessing tree conditions. In this study, the EfficientNetV2, especially the EfficientNetV2-S backbone, has stood out in performance for evaluation results and has become the utmost model in this study. The F1 Score from this model reached 80% and 73% for IoU@50 and IoU@75, respectively. This result is preferable to the other models used in this study.**

Keywords— Oil Palm, Faster R-CNN, UAV Images, Tree Detection.

I. Introduction

Oil palm plantations in Indonesia have rapidly grown over the past few decades [1], as shown by the increased area of oil palm plantations every year. In order to maximize crop yields and maintain tree health, plantation owners or the oil palm industry need to conduct regular monitoring. Monitoring by conventional methods has proven inefficient and ineffective, especially for large. In this way, Artificial Intelligence (AI) is a promising solution.

One of the deep learning methods is object detection, which can be utilized in the agricultural domain to examine tree counting and monitor the health of oil palm trees [2]. Tree counting by object detection can use remote sensing satellite imagery and aerial imagery using Unmanned Aerial Vehicles (UAV). This study chose UAV imagery because it has a higher resolution and allows for more detailed object identification compared to satellite imagery.

Several studies have been conducted to detect tree objects using UAV, such as Multi-class Oil Palm Detection using Faster R-CNN with ResNet101 getting 70% of F1-score [1], improved Non-Maximum Suppression based max intersection over portion (NMS-MIoP) to overcome significant error boundaries in high Mean Average Precision (MAP) [3], developing a new method to map Amazonian palm species at the individual tree crown (ITC) using UAV RGB [4], using Fully Convolutional Regression Networks and Multi-Task Learning can detect individual full-grown trees, tree seedlings, and tree gaps in citrus orchard [5], and developing architecture Convolutional Neural Network (CNN) with UAV to overcome biased images and detecting plants and row on a one-step basis [6].

This study conducted a comparative study to develop and proposed new Faster R-CNN model with backbone modification for oil palm tree condition detection. Several backbones used in this study are EfficientNetV2 and MobileNetV3. The best backbone model obtained from comparative study will be proposed to improve the performance of oil palm tree detection from UAV images.

This study is categorized as follows. Section 2 describes several previous works on object detection and tree-counting architecture. The method of multi-class detection Faster R-CNN is shown in section 3. The experiment and discussion are presented in section 4. The last section gives the conclusion of this paper.

II. Related Works

Ever since the inception of object detection through R-CNN [6], its performance has shown great promise, leading to continuous improvements over time. This evolution is evident in advancements like Fast R-CNN [7] and Faster R-CNN [8]. Even with the adaptation of Faster R-CNN for specialized tasks such as Palm Tree Oil Detection [1], [9], ongoing enhancements persist for these targeted objects.

A. Faster R-CNN

The foundation of Faster R-CNN lies in the Regional Convolutional Neural Networks (R-CNN) approach to object detection. In the study [6], R-CNN involves segmenting the input image into approximately 2000 region proposals, extracting features from each proposal using deep convolutional neural networks (CNNs), and subsequently classifying these regions using class-specific linear Support Vector Machines (SVMs).

In [7], R-CNN underwent enhancements to improve processing efficiency, reduce storage requirements, and enhance accuracy. Fast R-CNN employs multiple convolutional and max pooling layers to process the image, generating a convolutional feature map. Subsequently, a region of interest (RoI) pooling layer extracts a fixed-length feature vector from the feature map for each object proposal. The feature vectors undergo processing through a sequence of fully connected layers, branching into two output layers. The first layer generates softmax probability estimates across K object classes, encompassing a general "background" class. The second layer produces four real-valued numbers for each of the K object classes, where each set of four values represents refined bounding-box positions for one of the K classes.

Based on research [8], [10] Faster R-CNN mainly consists of two modules: the fully deep convolutional network that proposes regions and the Fast R-CNN detector. The Region Proposal Network (RPN) serves as the 'attention' to guide the Fast R-CNN in looking at essential information.

B. Oil Palm Tree Detection

Zheng et al. [1] tested several methods on oil palm trees for object detection, classifying the tree into five categories: healthy palm, dead palm, mismanaged palm, smallish palm, and yellowish palm. From the research produced the Average F1-score of the RF method is 28.98%, SVM method 23,19%, CNN (ResNet-101) 28.03%, Faster R-CNN 55.74%, Grid R-CNN 56.23%, GA Faster R-CNN 58.33%, Cascade R-CNN 60.71%, Libra Faster R-CNN 62.46%, and MOPAD 72.83%. This paper proposes a Multi-class Oil Palm Detection approach (MOPAD) based on Faster R-CNN with ResNet101 backbone, which combines a Refined Pyramid Feature (RPF) module and a hybrid class-balanced loss module to achieve satisfactory observation of individual oil palms' growing status.

Mubin et. al. [11] conducted a deep-learning study to predict and count oil palms in satellite imagery, focusing on young and mature ones. The research uses two CNN models and GIS for data processing and result storage. The architecture is based on LeNet, with an adaptive gradient algorithm for model training. The method achieves 95.11% and 92.96% accuracy in detecting young and mature oil palms.

Yarak et. al. [12] conducted a study and proposed an alternative method for oil palm tree management from high-resolution imagery and using Faster-R-CNN for tree detection and health classification. The model uses 4172 bounding boxes of healthy and unhealthy palm trees, with crown size, colour, and density as key characteristics. The optimal altitude for image capture is 100m, with Resnet-50 achieving better F1 scores than VGG-16.

Putra et. al [13] conducted a study to investigate the use of remote sensing data from Microsoft Bing Maps VHR satellite imagery and UAV data for detecting and counting oil palm trees. The proposed image segmentation approach achieves the results with an average true-positive rate (TPR) of 88-97% in best-case data input and an estimation error of up to 78%. The result is that UAV imagery has better results than satellite imagery.

Kent et al [14] conducted a study to present a modified U-Net architecture called the Multi Convolution Residual (M-CR) U-Net, combined with the overlapped contour separation (OVCS) algorithm for early symptom detection of oil palm trees BSR disease in a crowded area. The M-CR U-Net enhances segmentation performance by using multiple convolution kernels and skipping connections. The OVCS algorithm allows for effectively segmenting individual oil palm trees in crowded regions, resulting in an average performance gain of 2.716%.

Liu et al. [15] proposed an end-to-end method based on deep learning for detecting and counting oil palm trees from UAV images. The method uses a Faster-R-CNN algorithm to extract features from the images, identify oil palm trees, and provide location information. The method achieved an overall accuracy of 97.06%, 96.58%, and 97.79% in detecting oil palm trees from three different sites, demonstrating its effectiveness and accuracy. However, this paper only focuses on the presence of the oil palm tree in the images.

III. Research Method

A. Data set

The dataset used in this study is the multi-class condition in oil palm trees dataset from a previous study [1], which is available in the public repository (https://github.com/rs-dl/MOPAD). This study uses some data from Site 2, which is in Papua, Indonesia (140◦29′ 17′′E, 6◦57′ 42′′S). The UAV platform used for collecting that data is Skywalker X8. The images were taken at an altitude of 425 cm using a Sony A6000 camera. UAV collected the image data with three bands: red, green, and blue (RGB) images with a spatial resolution of 8 cm.

The dataset images were annotated into five classes based on oil palm conditions. The classes are dead palms, healthy palms, mismanaged palms, smallish palms, and yellowish palms. The total image data in the dataset are 2303 images with 151586 annotations included in train and test data. The training data in this study is divided into 90% for training data and 10% for validation data. Details of the data distribution in the dataset are shown in Table I. The data distribution for each class in the dataset is imbalanced due to the small sample for certain oil palm conditions.

B. Model Architecture

The model architecture used in this study is based on Faster R-CNN [8]. The Faster R-CNN model contained a ConvNet (backbone) as a feature extractor, Region Proposal Network (RPN), ROI pooling, softmax classification, and bounding box regression. This study conducted a comparison study to modify the model backbone using several state-of-the-art ConvNet image classification models. Several backbone models used in this study are ResNet50v2 [2],[3], ResNet101 [16], MobileNetV3-Larg e[18], and EfficientNetV2-S [19]. The ResNet family model

TABLE I. Data Distribution in Oil Palm Dataset

Type	**Training**	**Validation**	**Test**
Images	***1622***	***181***	***500***
Healthy	80482	9476	24884
Mismanaged	396	35	95
Smallish	24061	2583	7385
Yellowish	1332	141	411
Dead	208	26	61
Total	106839	11901	32846

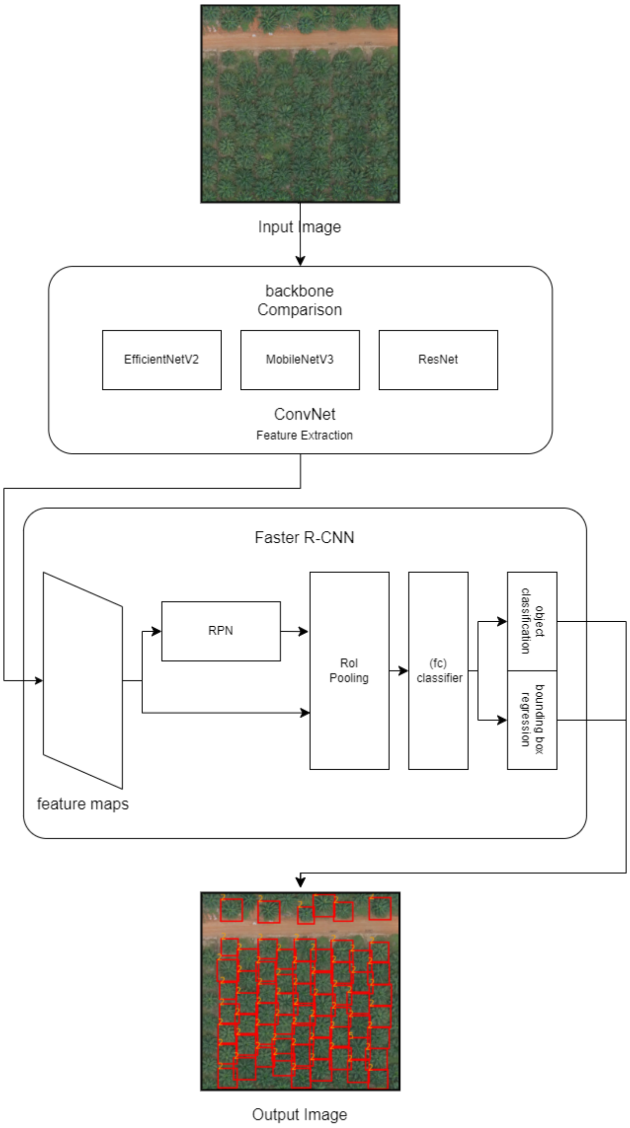


Fig. 1. Model Architecture

was used as a benchmark for model evaluation to compare with previous studies that used ResNet101 as a backbone. This study proposed comparing two newer backbone models for oil palm tree detection, which are smaller and faster in computation but can still maintain the model performance. RPN part used anchor size in $\{32^2, 64^2, 128^2, 256^2\}$ and aspect ratio in {1:2, 1:1, 2:1}. This modification was made to adjust the size of the oil palm bounding box from the dataset. The architecture design of Faster-R-CNN used in this study is shown in Figure 1.

Faster R-CNN was used in this study because this two-stage object detection method has been proven to perform better in oil palm tree detection [1]. Due to the longer training time problem faced in Faster R-CNN, this study used ConvNet backbone that introduces faster and more efficient models, such as MobileNetV3 and EfficientNetV2. MobileNetV3 and EfficientNetV2 prioritize efficiency and accuracy, but they achieve this through different architectural choices. MobileNetV3 enhances the previous version with squeeze-and-excitation layers and h-swish non-linearity [18], while EfficientNetV2 focuses on efficient scaling [19].

MobileNetV3 is a family of CNN architectures designed to balance model accuracy and computational efficiency [18]. MobileNetV3 use depthwise separable convolutions and incorporates squeeze-and-excitation blocks. Its lightweight design suits mobile devices, edge computing, and IoT applications. There are two variants in this family. This study uses MobileNetV3-Large, the variant with a more extensive backbone designed for better accuracy in the family but can still maintain computational efficiency. The paper [18] stated that MobileNetV3_large can achieve 25% faster training in the COCO Object Detection challenge than the previous version with similar accuracy.

EfficientNetV2 is a family of CNN architectures offering faster training speed and better parameter efficiency than previous models [19]. EfficientNetV2 extensively uses MBConv (MobileNetV2-style inverted residual blocks) and the newly added layer fused-MBConv in the early layers. This study chose to adopt EfficientNetV2-S, the baseline architecture from this family, due to computational resource reasons to construct a smaller model while still maintaining model performance. The EfficientNetV1 family is the backbone of the EfficientDet family, which has achieved state-of-the-art results on object detection challenges.

C. Model Evaluation

Evaluation Metrics used in this study to evaluate model performance are precision, recall, and F1 Score. These metrics can be calculated by (1) – (3). This study's calculation of True Positive, False Positive, and False Negative differs from other classifications or detections [20]-[23]. These are chosen and calculated following previous studies [1]. Precision detects the truly correct prediction out of all predictions. Recall detects the truly correct prediction out of all actual classes. F1 scores are harmonic mean between these two that measure overall model performance. These metrics are produced by calculating True Positive (TP), False Positive (FP), and False Negative (FN). In this case, TP is calculated from the correct prediction with the ground truth in the Intersection Over Union (IoU) Threshold range. FP is a false prediction with the ground truth but within the range of the IoU Threshold. FN is the ground truth label; the model prediction does not predict or miss. IoU is the intersection area calculated from ground truth and prediction bounding boxes. IoU can be calculated as (4). In this study, the IoU Thresholds used to calculate the metrics are 0.50 and 0.75.

$$Precision = \frac{TP}{(TP+FP)} \tag{1}$$

$$Recall = \frac{TP}{(TP+FN)} \tag{2}$$

$$F1\ Score = \frac{2 \times Precision \times Recall}{(Precision+Recall)} \tag{3}$$

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{4}$$

The metric mean Average Precision (mAP) is introduced in COCO dataset evaluation metrics and is widely used in object detection studies as a benchmark for evaluating and comparing model performance. The mAP provides a comprehensive evaluation of the object detection model. mAP considers precision and recall values across

TABLE II. EVALUATION RESULTS

Backbone	F1 Score IoU@50	F1 Score IoU@75	mAP	mAP@50	mAP@75
ResNet50v2	0.682	0.576	0.368	0.619	0.394
ResNet101	0.630	0.562	0.366	0.635	0.393
MobileNetV3-Large	0.736	0.634	0.487	**0.796**	0.564
EfficientNetV2-S	**0.805**	**0.735**	**0.521**	0.760	**0.620**

TABLE III. PRECISION, RECALL, AND F1 SCORE (IOU @ 50) FOR EACH CLASS

Backbone	Metrics	Healthy	Mismngd	Smallish	Yellowish	Dead	Average
MobileNetV3-Large	Precision	0.970	0.635	0.871	0.498	0.376	0.670
	Recall	0.987	0.695	0.924	0.886	0.770	**0.852**
	F1 Score	0.978	0.663	0.897	0.637	0.505	0.736
EfficientNetV2-S	Precision	0.985	0.865	0.929	0.894	0.662	**0.867**
	Recall	0.968	0.474	0.781	0.859	0.770	0.770
	F1 Score	0.977	0.612	0.849	0.876	0.712	**0.805**

various confidence thresholds and object classes. mAP calculated all object classes' mean of Average Precision (AP) values. AP is the area under the precision-recall curve, calculated from varying confidence thresholds for positive predictions in different precision-recall pairs. The formula to obtain AP and mAP can be calculated by (5) and (6)

$$AP_i = AP\ of\ class\ i \tag{5}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{6}$$

IV. RESULTS & DISCUSSION

A. Experimental Setup

The experiment in this study mainly uses a Kaggle kernel-free environment. The training process is carried out using the Nvidia T4 GPU. The optimizer used for the training process is Stochastic Gradient Descent (SGD), with a learning rate of 0.005. Each model was trained using a batch size of 4 with a maximum number of epochs of 25. The validation process is carried out after each training epoch to monitor model performance in the training phase. Early stopping callbacks are based on val mAP to prevent overfitting in the training phase. Dataset preprocessing used for data augmentation in the training phase is a Random Horizontal Flip with a probability of 0.5.

B. Evaluation Results

The evaluation results for F1 Score and mAP from the Faster R-CNN model in this study are shown in Table II, and the summary of these results is shown in Figure 2. The F1 score combines precision and recall metrics, providing a balanced model performance measure. For the IoU Threshold set to 0.50 and 0.75, the EfficientNetV2-S model stands out with an F1 Score of approximately 0.805 and 0.735. The second-best backbone for the F1 Score result is the MobileNetV3-Large model, which achieved 0.736 for F1 Score@50 and 0.634 for F1 Score@75, almost lower than the best model. The two ResNet family models performed relatively poorly compared to the two new backbones tested in the research. ResNet101, the backbone in the previous study, achieved slightly different performance and worse F1 score performance when compared to ResNet50v2 used in this study.

The best model for the overall mAP score in evaluation results is EfficientNetV2-S, with a value of around 0.521. The MobileNetV3-Large model performs well in the mAP@50 metric with a value of 0.796. The efficientNetV2-S model still performs well in mAP@50 even though it is not becoming the best, with the value achieved at 0.76. For mAP@75, the EfficientNetV2-S model is again the best model with the best performance with a value of around 0.62. MobileNetV3-Large became the second-best, with a score of around 0.56. The two ResNet models achieved similar performance with mAP, around 0.36, with ResNet50v2 showing slightly better performance for mAP and mAP@75 metrics and ResNet101 having a better result for mAP@50 with a score of 0.635.

From all evaluation results in this study, the EfficientNetV2-S model consistently outperforms the other backbone, except that using the mAP@50 metric does not become the best model. However, it can maintain being the second-best with slightly different features. MobileNetV3-Large also performs well for all evaluation metrics as the second-best model. The two ResNet models also have competitive performance, but compared to the others, the model performances fall behind. So, from this model

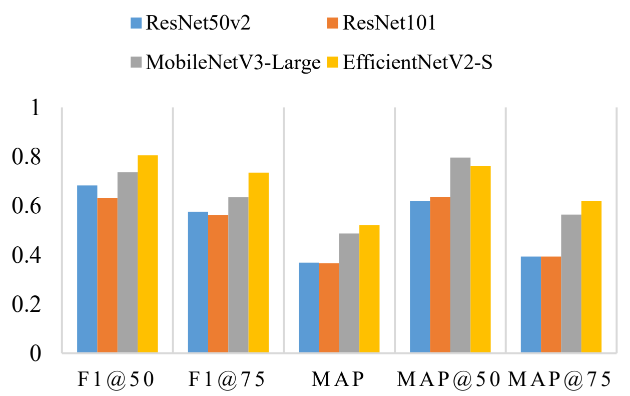
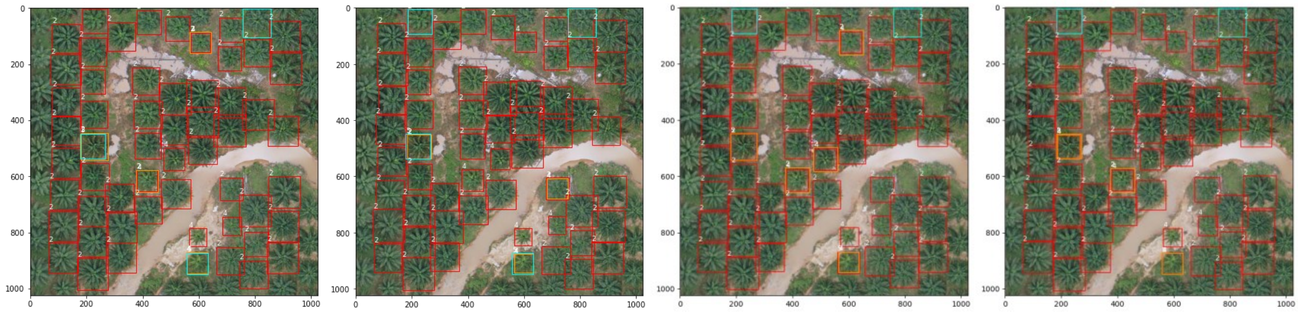


Fig. 2. Evaluation Results



a) ResNet50 b) ResNet101 c) MobileNetV3-Large d) EfficientNetV2-S

Fig. 3. Detection Results sample for EfficientNetV2-S (d) with indicate better result compared to other models; Bbox color: Red (TP), Orange (FP), and Cyan (FN)

evaluation, this study proposed the EfficientNetV2-S backbone as a new backbone for the Faster R-CNN model to improve the performance of multi-class oil palm condition detection.

C. Detection Results Analysis for Proposed Model

To take a closer look at the proposed model performance, Faster R-CNN with EfficientNetV2-S backbone model, its detection results for Precision, Recall, and F1 score from the model evaluation at IoU Threshold set to 0.50 compared to the MobileNetV3-Large shown in Table III. The table shows that the proposed model achieved good performance for each class even though it had to deal with imbalanced data in the dataset. Each class has a precision value above 60%, even for classes with small data such as 'mismanaged'. Even though the recall results still have values below 50%, which means there are still many missing values from the prediction results, the F1 score results show promising results with a score reaching 80%. The MobileNetV3-Large model can achieve good performance in recall metric but has poorer precision and F1 score metric performance, and some classes have precision values below 50%.

The sample of detection results from the proposed model, EfficientNetV2-S, compared to other models in the experiment shown in Figure 3. In this Figure, it can be seen that the FP error for MobileNetV3-Large is higher than that of EfficientNetV2-S. That aligns with the precision value, which is smaller than that of EfficientNetV2-S. In contrast, EfficientNetV2-S obtained a lower recall value than the MobileNetV3-Large, indicating this model has a high FN error. The most common error prediction from the two models is between the 'Healthy' and 'Smallish' classes because of the small difference between these two classes, apart from the fact that these two classes have more significant portions in the dataset.

The TIDE toolbox [24] is used to compute the detection error in object detection model performance for in-depth detection error analysis. The result shows a comprehensive analysis of the strengths and weaknesses of the model against six types of error, which are 'classification error (cls)', 'localization error (loc)', 'both cls and loc error (both)', 'duplicate detection error (dupe)', 'background predicted as an object (bkg)', and 'missing ground-truth class in the prediction result (miss)'. Each type is calculated to obtain delta AP (dAP), which is the contribution of the error type to the difference value to reach 100% mAP.

The error analysis results obtained from the proposed model are shown in Table IV. The highest obstacle or weakness of the proposed model is the 'miss' error. That means the amount of data that has not been predicted or is missing is still relatively high for the proposed model. This result is in line with the previous analysis for the recall value, which is smaller than the precision value, where the recall value is related to the ratio of the number of correct predictions with all ground-truth data. However, for other types of error, such as 'cls' error, the results are pretty good, with a value of around 5% for dAP@50 and dAP@75, and for other types of error, the proposed model has promising results with an error value of almost 0% for dAP@50. MobileNetV3-Large, on the other hand, has a lower error rate for 'miss' error, but for all the remaining types, it has a larger value than the proposed model, especially for 'cls' with a value around 12% for dAP@50 and 11% for dAP@75.

V. CONCLUSION

This study conducted a backbone modification comparison to propose a new Faster R-CNN model with backbone modifications that are lighter and more efficient in detecting the condition of multi-class palm oil trees. The backbones used in this study are ConvNet architecture, which prioritizes the trade-off between efficiency and accuracy. This study used two architectures that suit that objective in its development, which are MobileNetV3 and EfficientNetV2. The EfficientNetV2, especially the

TABLE IV. DETECTION ERROR RESULTS (IN PERCENT)

Backbone	mAP	Type	cls	loc	both	dupe	bkg	miss
MobileNetV3-Large	79.590	**dAP@50**	12.60	1.64	0.04	0.00	0.18	4.81
	56.369	**dAP@75**	10.69	22.97	0.40	0.00	0.15	4.72
EfficientNetV2-S	76.042	**dAP@50**	5.85	0.04	0.00	0.00	0.06	15.17
	61.955	**dAP@75**	4.70	13.75	0.06	0.00	0.05	11.65

EfficientNetV2-S backbone, has stood out in performance for evaluation results and has become the best model in this study. The F1 Score from this model is reaching 80% and 73% for IoU@50 and IoU@75, respectively. This result became superior among other models in this study. The mAP score reached 52%.

In conclusion, the proposed a new Faster R-CNN with EfficientNetV2 for Mult-class Oil Palm Condition Detection. However, the proposed model still has limitations and could be improved further. Further development of the detection model for this case study can be explored further to obtain a better model with improved performance or with more complex data sources and case studies, such as tree detection from multi-spectral images.

ACKNOWLEDGMENT

This research was funded by Institut Teknologi Sepuluh Nopember (ITS) under Penelitian Dana Departemen Program, Penelitian Keilmuan, Penelitian Flagship, Penelitian Kolaborasi Pusat and under the Project Scheme of the Publication Writing and Intellectual Property Rights (IPR) Incentive (Penulisan Publikasi dan Hak Kekayaan Intelektual/PPHKI) Program 2024.

REFERENCES

[1] J. Zheng *et al.*, "Growing status observation for oil palm trees using Unmanned Aerial Vehicle (UAV) images," *ISPRS J. Photogramm. Remote Sens.*, vol. 173, pp. 95–121, Mar. 2021, doi: 10.1016/j.isprsjprs.2021.01.008.

[2] Q. Jiang, Z. Huang, G. Xu, and Y. Su, "MIoP-NMS: Perfecting crops target detection and counting in dense occlusion from high-resolution UAV imagery," *Smart Agric. Technol.*, vol. 4, p. 100226, Aug. 2023, doi: 10.1016/j.atech.2023.100226.

[3] M. P. Ferreira *et al.*, "Individual tree detection and species classification of Amazonian palms using UAV images and deep learning," *For. Ecol. Manag.*, vol. 475, p. 118397, Nov. 2020, doi: 10.1016/j.foreco.2020.118397.

[4] L. E. C. La Rosa, M. Zortea, B. H. Gemignani, D. A. B. Oliveira, and R. Q. Feitosa, "FCRN-Based Multi-Task Learning for Automatic Citrus Tree Detection From UAV Images," in *2020 IEEE Latin American GRSS & ISPRS Remote Sensing Conference (LAGIRS)*, Santiago, Chile: IEEE, Mar. 2020, pp. 403–408. doi: 10.1109/LAGIRS48042.2020.9165654.

[5] L. P. Osco *et al.*, "A CNN approach to simultaneously count plants and detect plantation-rows from UAV imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 174, pp. 1–17, Apr. 2021, doi: 10.1016/j.isprsjprs.2021.01.024.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", doi: 10.1109/CVPR.2014.81.

[7] R. Girshick, "Fast R-CNN." arXiv, Sep. 27, 2015. doi: 10.48550/arXiv.1504.08083.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." arXiv, Jan. 06, 2016. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/1506.01497

[9] J. Zheng, W. Li, M. Xia, R. Dong, H. Fu, and S. Yuan, "Large-Scale Oil Palm Tree Detection from High-Resolution Remote Sensing Images Using Faster-R-CNN," in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan: IEEE, Jul. 2019, pp. 1422–1425. doi: 10.1109/IGARSS.2019.8898360.

[10] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-Based Models for Speech Recognition," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2015. Accessed: May 14, 2024. [Online]. Available: https://proceedings.neurips.cc/paper/2015/hash/1068c6e4c8051cfd4e9ea8072e3189e2-Abstract.html

[11] N. A. Mubin, E. Nadarajoo, H. Z. M. Shafri, and A. Hamedianfar, "Young and mature oil palm tree detection and counting using convolutional neural network deep learning method," *Int. J. Remote Sens.*, vol. 40, no. 19, pp. 7500–7515, Oct. 2019, doi: 10.1080/01431161.2019.1569282.

[12] K. Yarak, A. Witayangkurn, K. Kritiyutanont, C. Arunplod, and R. Shibasaki, "Oil Palm Tree Detection and Health Classification on High-Resolution Imagery Using Deep Learning," *Agriculture*, vol. 11, no. 2, p. 183, Feb. 2021, doi: 10.3390/agriculture11020183.

[13] Y. C. Putra, A. W. Wijayanto, and G. A. Chulafak, "Oil palm trees detection and counting on Microsoft Bing Maps Very High Resolution (VHR) satellite imagery and Unmanned Aerial Vehicles (UAV) data using image processing thresholding approach," *Ecol. Inform.*, vol. 72, p. 101878, Dec. 2022, doi: 10.1016/j.ecoinf.2022.101878.

[14] O. Win Kent, T. Weng Chun, T. Lee Choo, and L. Weng Kin, "Early symptom detection of basal stem rot disease in oil palm trees using a deep learning approach on UAV images," *Comput. Electron. Agric.*, vol. 213, p. 108192, Oct. 2023, doi: 10.1016/j.compag.2023.108192.

[15] X. Liu, K. H. Ghazali, F. Han, and I. I. Mohamed, "Automatic Detection of Oil Palm Tree from UAV Images Based on the Deep Learning Method," *Appl. Artif. Intell.*, vol. 35, no. 1, pp. 13–24, Jan. 2021, doi: 10.1080/08839514.2020.1831226.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[17] Y. Li, S. Xie, X. Chen, P. Dollar, K. He, and R. Girshick, "Benchmarking Detection Transfer Learning with Vision Transformers." arXiv, Nov. 22, 2021. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/2111.11429

[18] A. Howard *et al.*, "Searching for MobileNetV3." arXiv, Nov. 20, 2019. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/1905.02244

[19] M. Tan and Q. V. Le, "EfficientNetV2: Smaller Models and Faster Training." arXiv, Jun. 23, 2021. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/2104.00298

[20] Sarno, R., Inoue, S., Ardani, M.S.H., Purbawa, D.P., Sabilla, S.I., Sungkono, K.R., Fatichah, C., Sunaryono, D., Bakhtiar, A., Prakoeswa, C.R. and Tinduh, D., " Detection of infectious respiratory disease through sweat from axillary using an E-nose with stacked deep neural network," IEEE Access, vol. 10, p. 51285-51298, 2022, doi: 10.1109/ACCESS.2022.3173736

[21] D. Aulia, R. Sarno, S. C. Hidayati and M. Rivai, "Optimization of the Electronic Nose Sensor Array for Asthma Detection Based on Genetic Algorithm," IEEE Access, vol. 10, p. 74924-74935, 2023, doi: 10.1109/ACCESS.2023.3291451

[22] Sarno, R., Ghozali, I. and Sungkono, K.R., "Improving cyberbullying detection through multi-level machine learning," International Journal of Electrical & Computer Engineering, vol. 14, p. 2088-8708, 2024, doi: 10.11591/ijece.v14i2.pp1779-1787

[23] Purbawa, D.P., Sarno, R., Ardani, M.S.H., Sabilla, S.I., Sungkono, K.R., Fatichah, C., Sunaryono, D., Parimba, I.S. and Bakhtiar, A., " Adaptive filter for detection outlier data on electronic nose signal," Sensing and Bio-Sensing Research, vol. 36, p. 100492, 2022, doi: 10.1016/j.sbsr.2022.100492

[24] D. Bolya, S. Foley, J. Hays, and J. Hoffman, "TIDE: A General Toolbox for Identifying Object Detection Errors." arXiv, Aug. 31, 2020. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/2008.08115