Jurnal Elektronika dan Telekomunikasi (JET), Vol. 24, No. 1, August 2024, pp. 25-30
Accredited by KEMDIKBUDRISTEK, Decree No: **158/E/KPT/2021**
doi: 10.55981/jet.551

Two-Stage Object Detection for Autonomous Vehicles With VGG-16 Based Faster R-CNN

Arnetta Listiana Dewi [a], Hilman F. Pardede [b], Endang Suryawati [b,*], Hasih Pratiwi [a], Ana Heryana [b], Asri R. Yuliani [b], Ade Ramdan [b]

[a]*Department of Statistics*
Faculty of Mathematics and Natural Sciences
Universitas Sebelas Maret
Jl. Ir. Sutami 36 A Kentingan
Surakarta, Indonesia
[b]*Research Center for Artificial Intelligence and Cyber Security*
National Research and Innovation Agency (BRIN)
Jl Sangkuriang, KST Samaun Samadikun
Bandung, Indonesia

Abstract

The implementation of object detection for autonomous vehicles is essential because it is necessary to identify common objects on the street so that a proper response can be designed. While single-stage object detection may require fewer computations, two-stage object detection is preferred due to its ability to localize objects. Finding the optimum setup for multiple hyperparameters can enhance the performance of the two-stage object detection method. In this paper, we propose using Faster region-based convolutional neural network (R-CNN) as a two-stage object detection method with a visual geometry group (VGG)-16 backbone for detecting objects on the street. We evaluate the method using an open image subset by selecting common street objects (traffic lights, traffic signs, and vehicles). We explore several hyperparameter setups, such as learning rate and the number of region of interest (RoI), to find the optimum configuration. We found that using a learning rate of 0.000001 with the Adam optimizer is the optimum value for this task. Additionally, we discovered that increasing the number of RoI may improve performance. This suggests that there is potential for achieving a higher mAP (mean Average Precision) by increasing the number of RoI.

Keywords: object detection, faster r-cnn, autonomous vehicles, convolutional neural networks.

I. INTRODUCTION

Developments in the field of artificial intelligence (AI) technology, especially deep learning, had a significant impact on the object detection field. Object detection can contain multiple classes of objects, unlike image classification which the input image contains only one class of objects [1]. In the case of object detection, it takes category information as well as the position of each target object so that detection is a classification by localization, which is usually represented by a rectangular box or called a bounding box [2]. Each bounding box will have a class label. In other words, object detection makes it possible to predict an image containing more than one object. Object detection can be implemented for detecting objects in traffic for autonomous vehicles [3], [4], detecting plants or crop conditions for smart farming and precision agriculture [5], detecting areas of diseases from medical images [6], [7], etc.

Deep learning-based object detection models are grouped into two types, namely, one-stage and two-stage. One-stage detection models such as YOLO (You Only Look Once) and SSD (Single Shot Multibox Detector) classify and localize objects in a single stage. Meanwhile, the two-stage detection model is a region-based object detection model, which uses the region proposal network (RPN) to generate a Region of Interest in the first stage and process the proposed regions in the second stage for classification and localization of objects with bounding box regression [8]. The two-stage object detection models include region-based convolutional neural network (R-CNN), Fast R-CNN, and Faster R-CNN.

The development of the R-CNN in 2014 became the basis of the two-stage object detection model [9]. This method replaced the sliding window approach used by OverFeat with selective search to create a region of proposals that would otherwise be processed on a convolution network. R-CNN won the 2013 imagenet large scale visual recognition challenge (ILSVR) with a mean average precision (mAP) value of 31.4%, outperforming OverFeat, which had a mAP value of 24.3%. Although R-CNN has made significant progress in the field of object detection, R-CNN has some drawbacks. The R-CNN algorithm has a complex multi-stage training process [10]. In addition, the computation of the vast number of features of the overlapping proposal, with more than 2000 boxes in a single image, causes the detection speed to be very slow [11]. Fast R-CNN [12] was introduced in 2015 to address the shortcoming.

* Corresponding Author.
Email: enda029@brin.go.id
Received: May 26, 2023 ; Revised: March 08, 2024
Accepted: April 01, 2024 ; Published: August 31, 2024

Fast R-CNN changed the training procedure that was initially multi-stage, now trained simultaneously so that the training process is faster with higher detection quality compared to R-CNN [12]. Furthermore, a faster object detection model was again proposed. This model is an improvement over R-CNN and Fast R-CNN, called Faster R-CNN [13]. In the Faster R-CNN algorithm, the creation of a proposal region with selective search is replaced with an RPN. It makes Faster R-CNN faster in detecting objects, its detection speed near real-time, and improves the quality of the proposal region so that detection accuracy increases [13].

Previous researchers have implemented Faster R-CNN in various applications. Some are used for equipment detection in electric power rooms [14] and for defect detection in powertrain assembly lines [15]. In the agriculture field, we found that Faster RCNN was used to inspect agricultural products [16] and detect multi-class apples [17]. Meanwhile, other studies applied this method for garbage detection [18] and automatic gun detection [19].

The choice of the backbone for Faster R-CNN may affect its performance. Song et al. [16] use Faster R-CNN with visual geometry group (VGG)-16 as the backbone to detect kiwifruit during harvest time. The resulting model yielded an average precision value of 87.61%, outperforming Zeiler and Fergus network (ZFnet)'s 72.50%. Gao et al. [17] also implement Faster R-CNN in plant detection for apples under different conditions: not covered by something, covered with leaves, covered with branches/wire, and covered with another fruit. In their experiments, the backbone used for feature extraction was VGG-16 and ZFnet. The Faster R-CNN with the VGG-16 backbone architecture outperforms ZFnet in terms of mAP value, reaching 0.879. Whereas ZFnet ZFnet is superior in detection speed with a value of 0.167 seconds/image.

As mentioned previously, one implementation of object detection is for autonomous vehicles. In autonomous vehicles, the systems must be able to detect various objects within one figure that can be used to mitigate the vehicles' directions. In this study, we have applied object detection modeling carried out using the Faster R-CNN method. The model is built with VGG-16 as the backbone architecture to perform feature extraction from object images and perform localization with bounding boxes. We explore the best configurations of Faster R-CNN in several aspects. Firstly, we use different levels of learning in the model. A learning rate level that produces a model with higher accuracy was used for the second model. In the second model, we varied the amount of region of interest (RoI). The model was evaluated using a metric called mAP. The mAP is a metric used to measure how well a model can precisely localize and classify objects within an image. The data used in this study was obtained from an open-source dataset, namely the Open Image Dataset. We used three object classes from this dataset, namely traffic lights, traffic signs, and vehicles, for detection purposes. One-stage and two-stage object detection methods have their trade-off. One stage is faster but less accurate, especially for small objects and crowded scenes. Two stages offer higher accuracy but are slower due to increased complexity. In our research, we prioritize better localization even if it slows down the process.

The rest of this paper is organized as follows: in section 2, the materials and methods used in this study are given. Section 3 contains the experimental setup, including datasets and hyperparameter settings for the model. In section 4, we calculate the mAP value in the test dataset and show some test results. Finally, the conclusion is presented in Section 5.

II. METHOD

In the Faster R-CNN algorithm, the creation of a proposal region with selective search is replaced with an RPN. This method combines Fast R-CNN as an object detector and RPN as a region proposer. In general, Faster R-CNN consists of five parts: deep-fully convolutional network, region proposal network, RoI pooling, fully connected network, bounding box regressor, and classifier. A number of candidate objects were proposed through a deep fully connected network and region proposal networks, then normalized through ROI pooling. Then, the necessary features were extracted on fully connected layers to perform classification and regression [20]. The overall architecture of Faster R-CNN is shown in Figure 1.

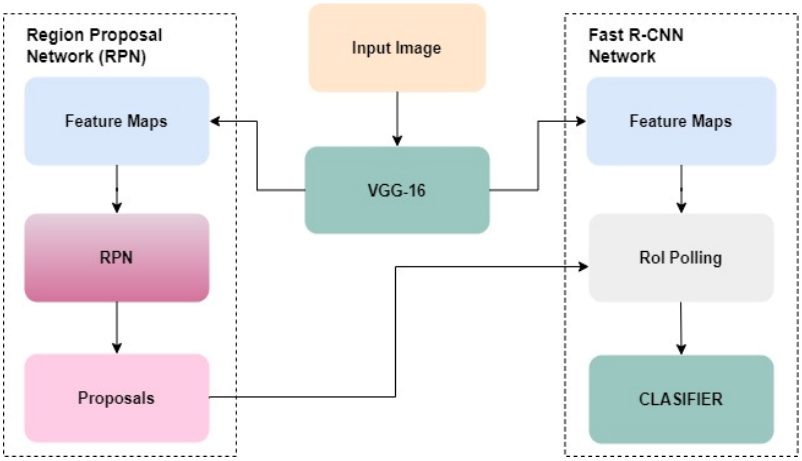

Figure 1. The overall architecture of Faster R-CNN.

RPN received feature maps that were produced by the backbones of CNN in the feature extraction process. This network was trained to predict the presence of objects in each anchor box and sublimate the location of the boxes to close-fit the object. A number of region proposals per image were produced in the RPN and then used as input in the Fast R-CNN Network. This network extracted features from each proposal using a RoI pooling layer.

The backbones used as feature extractors in the Faster R-CNN original paper were ZFnet and VGG-16. Based on research [16], [17] Faster R-CNN with the VGG-16 backbone architecture outperforms ZFnet in terms of mAP value. So, in this study, we used backbone networks VGG-16 for feature extraction. The architecture of VGG-16 can be seen in Figure 2. The architecture includes 13 convolutional layers, 13 rectified linear unit (ReLU) layers, and 4 pooling layers. The filter used on the convolution layer is 3×3, and the parameters used in the pooling layer are 2×2. There are 5 convolution blocks and each block consists of several convolution layers. The input image passed through a number of convolution layers and then connected with a pooling layer and a fully connected layer, then ended up in the output layer.

Input in the form of an image was processed using the CNN architecture. It generated a feature map that was sent to the RPN to determine the RoI of the image [18]. RPN is a convolution neural network with a kernel measuring 3×3 followed by two parallel layers of 1×1, namely for classification that determines whether a region is an object or not and predicts the appropriate bounding box against the position of the object [19]. RPN uses several boxes called anchors to predict the presence of objects. In this study, the used anchor size followed the original anchor size ($128^2, 256^2, 512^2$). Output from RPN is a number of object proposals, which is accompanied by an objectness score and bounding box. Objectness score is a score that indicates the presence of an object. While the bounding box is indicated by the coordinate points (x, y, w, h).

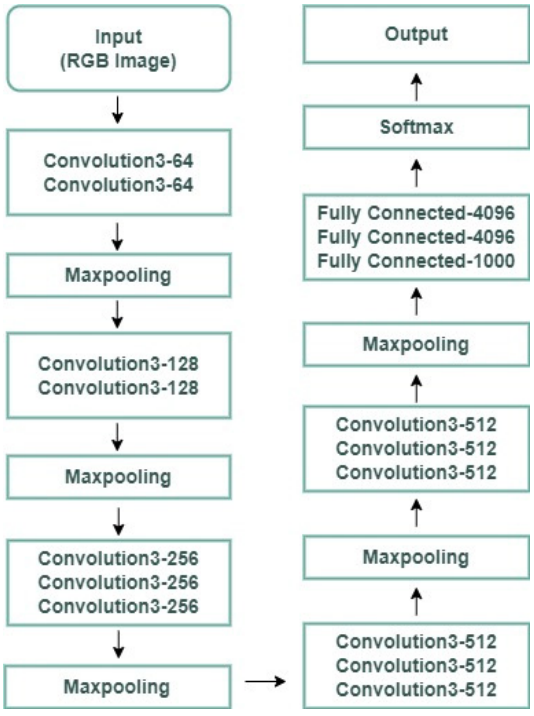

Figure 2. The VGG-16 architecture.

The proposal generated by RPN was then processed into the RoI pooling layer along with the feature map generated by the feature extraction network, namely VGG-16. The RoI pooling layer is used to map the coordinates of the proposal on the feature map back to the coordinates of the original image [15]. Feature map proposals were processed, including bounding boxes and probability scores, and then passed to the fully connected layer for classification and regression. The result of the classification process was the class label, and the result of the regression computation was the bounding box coordinates that localize the object.

III. Experimental Setup

The data used in this study is open source, namely Open Image Dataset. The class of objects to be detected were traffic lights, traffic signs, and vehicles. The dataset was then divided into 80% for the data train and 20% for the test data. The dataset contained a total of 3,988 images, with 3,535 images used for training data and 453 images for testing data. It is distributed across three classes, therefore defining the number of instances for each class: 5891 traffic lights, 3185 traffic signs, and 4694 vehicles. The running process was carried out in the Kaggle Notebook with the GPUP100 accelerator. Figure 3 shows the image samples, which are classified as a traffic light, traffic sign, and vehicle.

Our research investigates parameter variations in the Faster R-CNN model for object detection, specifically focusing on varying RoI amounts. However, the learning rate becomes the important parameter that must be taken into account to achieve the convergent model. The original Faster R-CNN method [16] uses a learning rate of 0.001 for 60,000 mini-batches and 0.0001 for the next 20,000 mini-batches on Pascal visual object classes (VOC) datasets. In this study, the learning rates used were 0.000001 and 0.00001 to carry out the training process in 30,000 batches. The optimizer used was Adam. The first model was trained with various learning rates. The learning rate that produces the model with the highest accuracy will be used to train the second model with varying amounts of RoI (4, 16, and 32).

In determining the RoI amount, we consider our available computing capacity. RoI is related to the amount of region of proposal (RoP). The more candidate regions generated can increase accuracy, but on the other

Figure 3. The samples from Open Image Dataset: (a) traffic light (b) traffic sign and (c) vehicle.

hand, it can reduce detection speed [3], [21]. For this reason, the amount of RoI must be regulated to control the number of object candidates.

Refer to (1), intersection over union (IoU) is used to measure the accuracy of the resulting bounding box by comparing the predicted bounding box with the ground truth box. The IoU calculation was carried out by comparing the area of the intersection or overlapping region with the combined area between the predicted bounding box and the ground truth box. In this study, the anchor belongs to positive if the IoU is greater than 0.7. If the IoU value is between 0.3 to 0.7, it is deemed ambiguous and not included in the objective.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (1)$$

Non-max suppression was used to overcome overlapping bounding boxes where there is more than one bounding box prediction. It works by reducing the number of bounding box candidates to one by ignoring redundant, overlapping bounding boxes. IoU threshold was applied here to determine it. The bounding box will be dropped if IoU in non-maximum suppression (NMS) exceeds this threshold. In this study, the IoU threshold was 0.7.

The performance of our object detection model was evaluated using the mAP method. As defined in the Introduction section, mAP can measure how well a model can precisely localize and classify objects within an image. It is defined by (2), which represents the formula for calculating mAP. The mAP value ranges from 0 to 1. The higher the mAP value, the better the model for detecting object targets.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (2)$$

Where N is the number of classes, and AP is the average precision for each class.

IV. Results and Discussions

The progression of accuracy for testing data for different learning rates is shown in Figure 3. Based on the graph shown in Figure 4, the model with a learning rate of 0.000001 achieves a higher level of classification accuracy compared to another one. But, according to the loss value that is shown in Figure 5, the model with a smaller loss value. Nevertheless, the loss value from both models is still relatively high. We performed model evaluation on test data by calculating mAP values to ensure better model performance, as shown in Table 1.

Table 1 shows the results of model testing using a smaller learning rate, resulting in a higher mAP value. This proves that the learning rate value has a significant influence on the level of accuracy. We then used the results of the learning rate testing for the second modeling by varying the number of RoIs. The number of RoI is the number of proposed regions to be selected to predict the bounding box.

Based on the results of our experiment, we observed that varying the amount of RoI can increase classification accuracy. The results show that the greater the number of RoI, the higher the accuracy, as shown in Figure 6. Likewise, with the loss level of the model where the greater the number of RoI, the lower the loss value, as shown in Figure 7. Figure 7 is the sum of the four losses, namely loss rpn classifier, loss rpn regression, loss detection classifier, and loss detection regression.

The results shown in Table 2 are the results of model evaluation on data testing by calculating mAP values. The results of calculating the mAP value show that the mAP value gets higher when the number of RoI increases. Where the model with a RoI of 32 has a higher mAP value.

Table 1
MAP values of different learning rate

Learning Rate	mAP
0.00001	0.668
0.000001	0.721

Table 2
MAP values of different numbers of RoI

Number of RoI	mAP
4	0.721
16	0.639
32	0.867

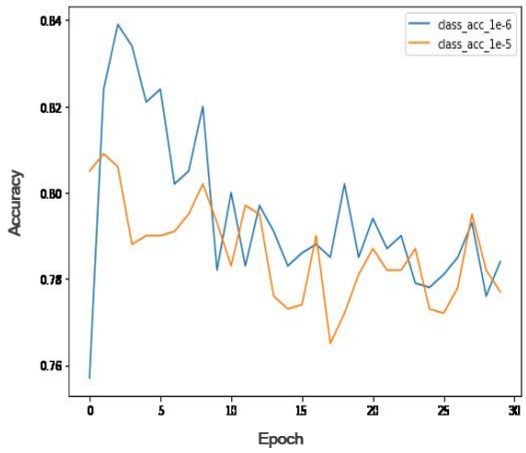

Figure 4. The progression of accuracy for Testing data for different learning rates.

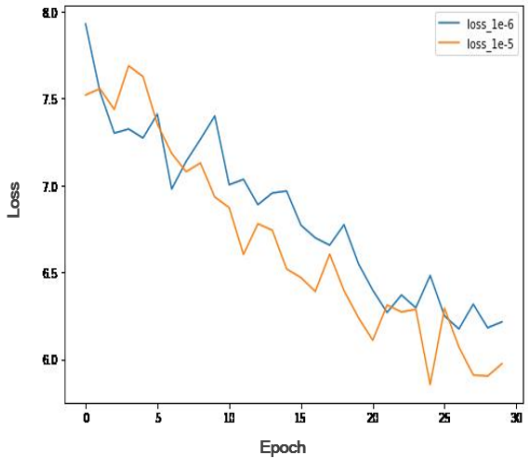

Figure 5. The progression of loss for Testing data for different learning rates.

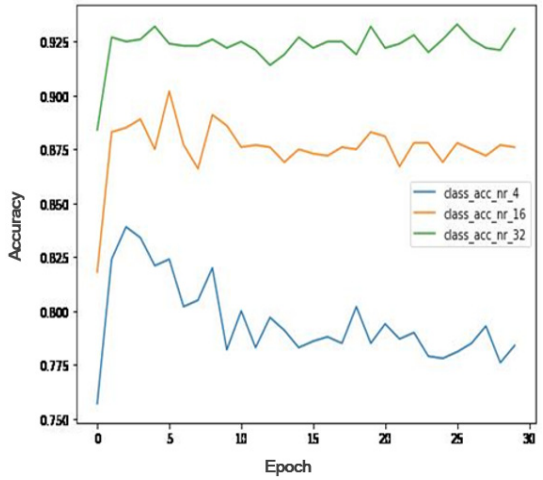

Figure 6. The progression of accuracy for Testing data for different numbers of ROI.

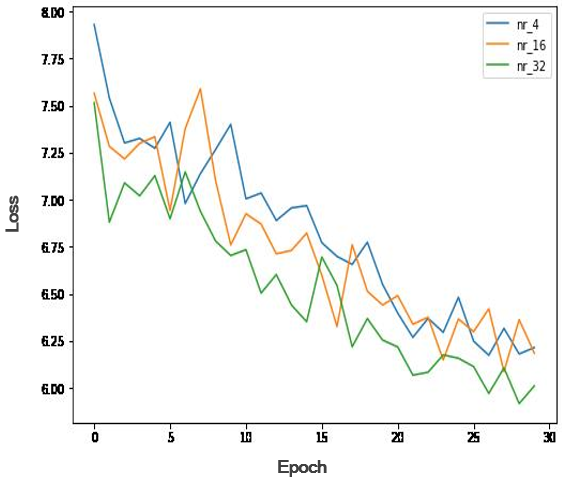

Figure 7. The progression of loss for testing data for different numbers of ROI.

V. CONCLUSION

In this study, Faster R-CNN has been used as a model for object detection. The model was built with VGG-16 as the backbone architecture to perform feature extraction. In the first experiment, we used different levels of learning rate. The results showed that models with a learning rate of 0.000001 have a higher classifier accuracy value on training and a higher mAP value on testing. In the second experiment, by using a learning rate of 0.000001, we increase the number of RoI. The results show that models with a RoI number of 32 have a higher mAP value. This shows that there is potential for getting a higher mAP with an increase in the amount of RoI.

For future work, we will explore whether increasing the number of RoIs above 32 still impacts performance improvement while acknowledging the potential need for increased memory. Therefore, our future research will focus on modifying the architecture to obtain a lighter model and more efficient feature selection. We hope to achieve faster and more accurate object detection.

DECLARATIONS

Conflict of Interest

The authors have declared that no competing interests exist.

CRediT Authorship Contribution

Arnetta Listiana Dewi: Conceptualization, Methodology, Data curation, Writing-Original draft; Endang Suryawati: Methodology; Hilman F. Pardede: Methodology, Supervision; Hasih Pratiwi: Methodology, Supervision; Ana Heryana: Software, Visualization, Investigation, Writing-Reviewing and Editing; Ade Ramdan; Software, Visualization, Investigation, Writing-Reviewing and Editing; Asri R. Yuliani: Software, Visualization, Investigation, Writing-Reviewing and Editing.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Acknowledgment

This research was conducted through a collaboration between the Research Organization for Electronics and Informatics (OREI) - National Research and Innovation Agency and the Faculty of Mathematics and Natural Sciences at Sebelas Maret University in the MBKM program.

REFERENCES

[1] C. Peng, K. Zhao, and B. C. Lovell, "Faster ILOD: Incremental learning for object detectors based on faster RCNN," *Pattern Recognit. Lett.*, vol. 140, pp. 109–115, Dec. 2020, doi: 10.1016/j.patrec.2020.09.030.

[2] A. Boukerche and Z. Hou, "Object detection using deep learning methods in traffic scenarios," *ACM Comput. Surv.*, vol. 54, no. 2, Art. no. 30, Mar. 2021, doi: 10.1145/3434398.

[3] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sens.*, vol. 13, no. 1, Art.no. 89, 2021, doi: 10.3390/rs13010089.

[4] Q. Fan, L. Brown, and J. Smith, "A closer look at Faster R-CNN for vehicle detection," in *Proc. 2016 IEEE Intell. Vehicles Symp.,* 2016, pp. 124–129, doi: 10.1109/IVS.2016.7535375.

[5] T. A. Shaikh, W. A. Mir, T. Rasool, and S. Sofi, "Machine learning for smart agriculture and precision farming: towards making the fields talk," *Arch. Comput. Methods Eng.*, vol. 29, pp. 4557–4597, 2022, doi: 10.1007/s11831-022-09761-4.

[6] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Imag. Anal.*, vol. 42, pp. 60–88, Dec. 2017. doi: 10.1016/j.media.2017.07.005.

[7] A. Rehman, M. Ahmed Butt, and M. Zaman, "A survey of medical image analysis using deep learning approaches," in *Proc. 5th Int. Conf. Comput. Methodologies Commun.,* 2021, pp. 1334–1342, doi: 10.1109/ICCMC51019.2021.9418385.

[8] P. Soviany and R. T. Ionescu, "Optimizing the trade-off between single-stage and two-stage deep object detectors using image difficulty prediction," in *Proc. 2018 20th Int. Symp. Symbolic Numer. Algorithms Sci. Comp.*, 2018, pp. 209–214, doi: 10.1109/SYNASC.2018.00041.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. 2014 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.

[10] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digit. Signal Process. A Rev. J.*, vol. 126, Jun. 2022, Art. no. 103514, doi: 10.1016/j.dsp.2022.103514.

[11] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: a survey," *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, Mar. 2023, doi: 10.1109/JPROC.2023.3238524.

[12] R. Girshick, "Fast R-CNN," in *Proc. 2015 IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[14] Q. Zhang, X. Chang, Z. Meng, and Y. Li, "Equipment detection and recognition in electric power room based on faster R-CNN," *Procedia Comput. Sci.*, vol. 183, pp. 324–330, 2021, doi: 10.1016/j.procs.2021.02.066.

[15] X. Liyun, L. Boyu, M. Hong, and L. Xingzhong, "Improved faster R-CNN algorithm for defect detection in powertrain assembly line," *Procedia CIRP*, vol. 93, pp. 479–484, 2020, doi: 10.1016/j.procir.2020.04.031.

[16] Z. Song, L. Fu, J. Wu, Z. Liu, R. Li, and Y. Cui, "Kiwifruit detection in field images using Faster R-CNN with VGG16," *IFAC-PapersOnLine*, vol. 52, no. 30, pp. 76–81, 2019, doi: 10.1016/j.ifacol.2019.12.500.

[17] F. Gao *et al.*, "Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN," *Comput. Electron. Agric.*, vol. 176, 2020, Art. no. 105634, doi: 10.1016/j.compag.2020.105634.

[18] M. F. Rahman and B. Bambang, "Deteksi sampah pada real-time video menggunakan metode Faster R-CNN," *Appl. Technol. Comput. Sci. J.*, vol. 3, no. 2, pp. 117–125, 2021, doi: 10.33086/atcsj.v3i2.1846.

[19] R. M. Alaqil, J. A. Alsuhaibani, B. A. Alhumaidi, R. A. Alnasser, R. D. Alotaibi, and H. Benhidour, "Automatic gun detection from images using Faster R-CNN," in *Proc. 2020 1st Int. Conf. Smart Syst. Emerg. Technol.,* 2020, pp. 149–154, doi: 10.1109/SMART-TECH49988.2020.00045.

[20] M.-C. Roh and J. Lee, "Refining faster-RCNN for accurate object detection," in *Proc. 2017 15th IAPR Int. Conf. Machine Vision Applications*, 2017, pp. 514–517. doi: 10.23919/MVA.2017.7986913.

[21] M. Lokanath, K. S. Kumar, and E. S. Keerthi, "Accurate object classification and detection by faster-RCNN," in *IOP Conf. Ser.: Mat. Sci. Eng.*, vol. 263, no. 5, 2017, Art. no. 052028. doi: 10.1088/1757-899X/263/5/052028.