VEHICLE DETECTION AND IDENTIFICATION WITH SMALL DATASET USING FEW-SHOT LEARNING

Muzakki Afandi
Department of Computer Science,
Sampoerna University,
Jakarta, Indonesia.
muzakki.afandi@my.sampoernauniversity.ac.id

Media Anugerah Ayu
Department of Computer Science,
Sampoerna University,
Jakarta, Indonesia.
media.ayu@sampoernauniversity.ac.id

Submitted: *Oct, 31, 2023* ***Revised:*** *Dec, 06, 2023* ***Accepted:*** *Dec, 13, 2023*

Abstract: Vehicle detection and identification serve an important role in employing autonomous vehicle classification. However, most deep learning methods for vehicle detection rely on large number of datasets for the training to perform well. The dataset shortage has become a main problem to acquiring a high accuracy detection. A detection method that could adapt with small samples could be a powerful solution for this problem. One of the popular methods is the Few-Shot Learning (FSL) algorithm. This method utilizes meta learning that can deliver more accurate detection with a small number of datasets. This research aimed to assess the effectiveness of the FSL algorithm in classifying road vehicles in Indonesia. Road vehicles categorization which based on the Decree of the Indonesian Minister of Public Works is unique, whereby it differentiates the trucks based on their axles. The datasets for these trucks are rarely available. This research work has employed a small amount of data to the FSL algorithm with reweighting module (FSRW) for classification. The results have demonstrated that FSRW method with fine tuning achieved better results in road vehicle identification with around 33% increase compared to the baseline YOLO method in categorizing road vehicles in Indonesia context.

Keywords: Few-Shot object detection, few-shot learning, vehicle identification, vehicle classification.

I. INTRODUCTION

Vehicle detection and identification is part of computer vision with the main task to locate and classify vehicle objects in images. The implementation of this system could efficiently improve the transportation system as many parts of the transportation system could be automated by using this detection and identification system. For this purpose, various deep learning methods have been proposed to cope with this field [1]. However, these deep learning methods require large number of samples to be able to accurately detect objects within their detection range. Moreover, there is always a condition where a large amount of data samples could not be obtained. This became a big concern of the current deep learning scenario.

One of the cases that could be studied is the detection scenario for the road vehicles in Indonesia. In Indonesia, the categorization of vehicles is mostly referred to the decree of the Minister of Public Works number 370/KPTS/M/2007 concerning the determination of types of motorized vehicles on toll road sections that are already in operation and the number of toll fees on several toll road sections in Indonesia. The highlighted difficult categorization scenario refers to one of its classifications where trucks are categorized by the difference number of its axles. This categorization can be said as a hard scenario since data samples of trucks with different number of axles are rarely available. Most of the available images are either not a full image or on different angles that made the system unable to identify how much axles there are. Not to mention that the task to compile these resources requires plenty of time and resources, also the cost is unlikely to be high [2].

Over the years, the development of vehicle detection and identification systems is mainly a combination of machine learning and data mining [3]. As a result, a wide range of algorithms have been developed and studied to cope with it, for example, support vector machine (SVM), neural network, K-means, Naïve Bayes classifier, cascade classifier, boosting classifier, etc [4]. In the development of classification algorithms, SVM is originally used to solve classification problems. However, the accuracy of the SVM algorithm is falling when it is used for a few samples of the labeled dataset [5]. Other algorithms mentioned also were failing to accurately classify data with few samples of labeled datasets, especially for neural networks algorithm [6-7]. It is very difficult to work with small data with neural networks [8].

As a result, an algorithm that could accurately identify an object with a few samples of the dataset to learn is needed. That is where Few-Shot Learning (FSL) algorithms come in. FSL algorithms have successfully done its task to classify an object with few samples of data [9]. Many approaches of FSL are introduced to help the algorithm perform better. For instance, there is an approach with meta-learning and prototype generation, which works by adding an auxiliary network to help the learning process, one of the approaches is the cosine-based distance classifier [10]. The approach is based on the two sequences of numbers of similarities.

The existing few-shot object detection uses a canonical dataset and is implemented in a very diverse scenario [11]. However, it has not yet implemented in Indonesia. Traditionally, developing a machine learning system for the detection and identification of vehicles involves the collection of large amounts of data and the training of the algorithm to produce a good result [12]. Therefore, the scarcity of data samples could pose a problem, especially when dealing with trucks that rely on how many axles it has to classify [13]. In this case, few-shots learning method plays a role to increase the detection accuracy while dealing with the low sample problem to do detection and identification of road vehicle, especially in Indonesia.

This research focuses on implementing the few-shot learning method to detect and identify road vehicles in Indonesia, specifically on trucks with different number of axles. The learning method used in this research is the Few-Shot with feature ReWeighting (FSRW). This research was conducted to investigate the capabilities of FSRW to detect and identify road vehicles in Indonesia using the Indonesia road vehicle dataset. The dataset was created based on a selection of vehicle types that fall under the 5 different classes listed in the Ministry of Public Works Decree number 370/KPTS/M/2007. This research demonstrates the performance of few shot learning algorithm in identifying vehicles within the Indonesian vehicle datasets, which involving car, small truck and the truck with different numbers of axles.

II. RELATED WORKS

Zhou et al., discussed few-shot video-based detection method that employs a feature scale selection pyramid and proposal contrastive learning. This study uses feature scale selection pyramid networks (FSSPN) to detect objects from the orbital satellite and add contrastive learning items in fine-tuning stage to get a more robust representation of the orbital satellite video [14]. The image needs to be enhanced because the object inside the video is a very small-sized object, which is why it needs a more robust representation.

Chen et al., discussed the classification model using the few-shot classifier. By using a generic object recognition scenario with a meta-learning algorithm, an average of 95% confidence interval results out of 600 randomly episodes point out that the few-shots classification problem could be solved by reducing the intra-class variation [15]. The adaptation of this solution could be applied to the meta-training stage to achieve a better meta-learning result.

Majee et al., implemented the few-shot learning to detect road objects. The research uses India Driving Dataset as a base sample; it contains less-occurring object samples to evaluate model capabilities to learn the context of road objects. By employing metric-learning and meta-learning few shot object detection methods to compare [9]. The result is that metric learning outperforms meta-learning and points out that metric learning is better to be used in the open-set scenario as meta-learning is more effective to detect similarities from the known object.

Fan et al., implemented the few-shot learning algorithm that uses an attention adaptive mechanism. The mechanism starts by putting a second-order statistics perspective based on the covariance matrix between the feature vectors to construct a class representation. Then the feature vector is adjusted by using an attention adaptive module to create a query sample to make the class representation closer to the corresponding classes [12]. The result is an average increase of accuracy by 6% from the base metric-learning method.

In conclusion, based on the compiled study to obtain a good result on a few shot learning algorithm experiments, it is important to implement several things. First, a robust representation of the image could be obtained by adding a fine-tuning stage. Second, reducing intra-class variation is a good adaptation to cope with the novel classes, achieving better performance on the classification task. Third, using meta-learning could achieve better results on objects that are not so different from known classes. Lastly, the use of query and support samples could make a better class representation for the corresponding classes. Implementing all of those methods could achieve better accuracy thus improving the performance of the few-shot classification models.

III. RESEARCH METHODOLOGY

To assess the capabilities of the few-shot learning algorithm on road vehicles in image format, a definitive step must be followed. Figure 1 depicts the overall flow of the research.

A. Data Preparation

Based on Figure 1 the data preparation processes include three main processes, which are the data collection, data pre-processing, and data division process. In data collection, the samples that were collected are road vehicles including cars, pickup trucks, small trucks, buses, big trucks, and trailer trucks.

Most of these sample were collected from Pattern Analysis, Statistical modeling and Computational Learning Visual Object Classes(PASCAL VOC) dataset, Common Objects in Context (COCO) dataset, Indonesia car dataset, Enes Umcu Car and Truck dataset, Zenodo Truck Image dataset, and the rest is from web-scrapping. A total of 350 samples were collected with each class consisting of 50 samples.

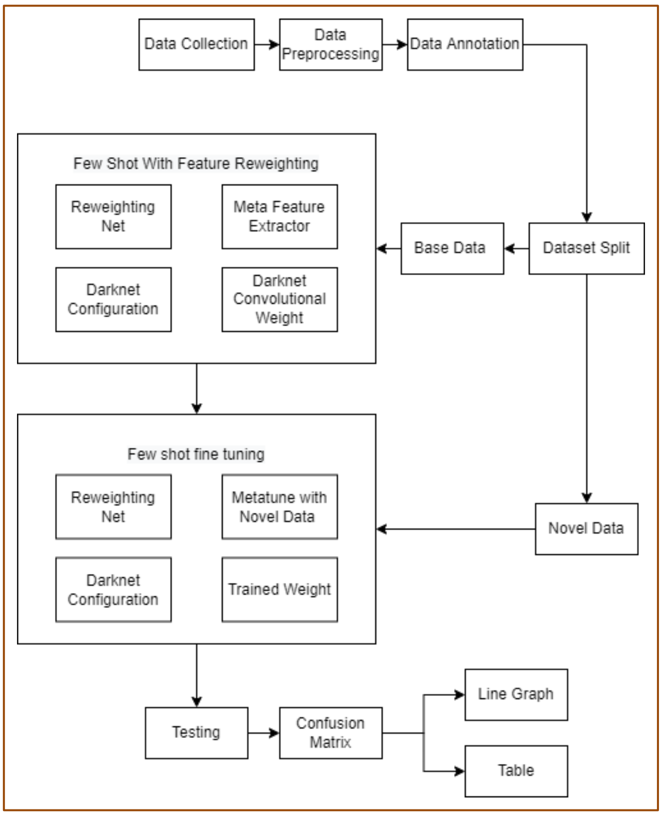

Fig. 1 Few-shot object detection via feature reweighting research flow

In data pre-processing, collected sample images are modified so that it could perform well in Darknet based model. Only single object is presented on each sample, the images also resized into 416 x 416 px with 72 ppi. The labeling process was following a predetermined format that based on the Indonesia Ministry of Public Works Decree number 370/KPTS/M/2007 (See Figure 2). The collected sample was also annotated in PASCAL VOC format. The noted factor for the annotation is the folder name where the file resides, filename, complete file path, image height, image width, image depth, class name, bottom left corner x-coordinates (x-min), bottom left corner y-coordinates (y-min), top right corner x-coordinates (x-max), top right corner y coordinates (y-max).

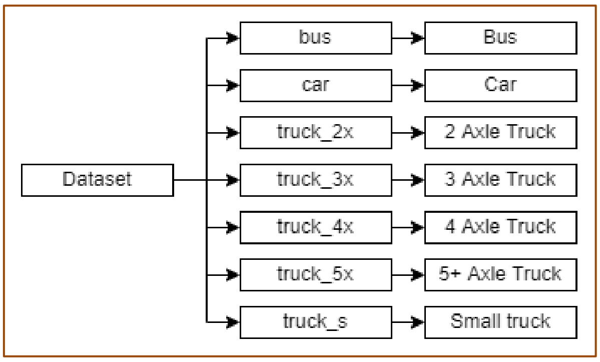

Fig. 2 The class label with its corresponding class for Indonesia road vehicle dataset

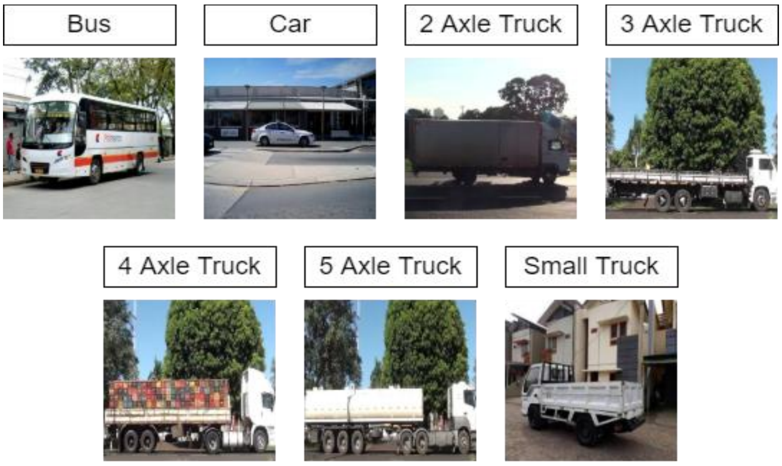

Fig. 3 Object sample for each class

Next is the data division process, this process will determine which sample is used for the particular test. There are three different tests to measure the model's capabilities. Each test uses different combinations of sample that defined by a set that called novel set. Based on Fig. 2 there are 7 sets of classes, the division will take 2 classes for the novel classes and the rest for the base classes. The first set uses two-axle and five-axle trucks for the novel set, the second set uses bus and car for novel set, and the last class use three-axle and four-axle trucks for the novel set. The details can be seen in Fig. 3.

TABLE 1. Base Training Sets and Number of Shots Used for Novel Sample

ID	**Class Name**	**Base Sample**	**Novel class**				
			K-Shots				
1	Bus	50	1	2	3	5	10
2	Car	50	1	2	3	5	10
3	2 Axle Truck	50	1	2	3	5	10
4	3 Axle Truck	50	1	2	3	5	10
5	4 Axle Truck	50	1	2	3	5	10
6	5+ Axle Truck	50	1	2	3	5	10
7	Small Truck	50	1	2	3	5	10

Each novel class was splitted based on the k-shots needed for each training (See Table 1). The base training evaluates the 40 data samples for each class. The fine-tuning process used the set of training images from annotated novel classes, each set contained equal to 1, 2, 3, 5 and 10 numbers of shots. Each set was chosen based on the difficulty level of detection (See Table 2). The first set was categorized as medium as from the view of collected sample, both classes have many different features that can represent their uniqueness. The second set was categorized as easy as each class has a very unique feature representative to each other, it was easy to differentiate. Then the last set was categorized as the hardest difficulty as each sample from each class has very little featured that can represent the uniqueness of each class. Details on novel classes are presented in Figure 4.

TABLE 2. Novel Sets With 2 Classes from the Indonesia Road Vehicle Dataset and Its Corresponding Class Labels

Novel Set	Difficulty Level	Novel Classes
1	Medium	2 Axle Truck, 5 Axle Truck
2	Easy	Bus, Car
3	Hard	3 Axle Truck, 4 Axle Truck

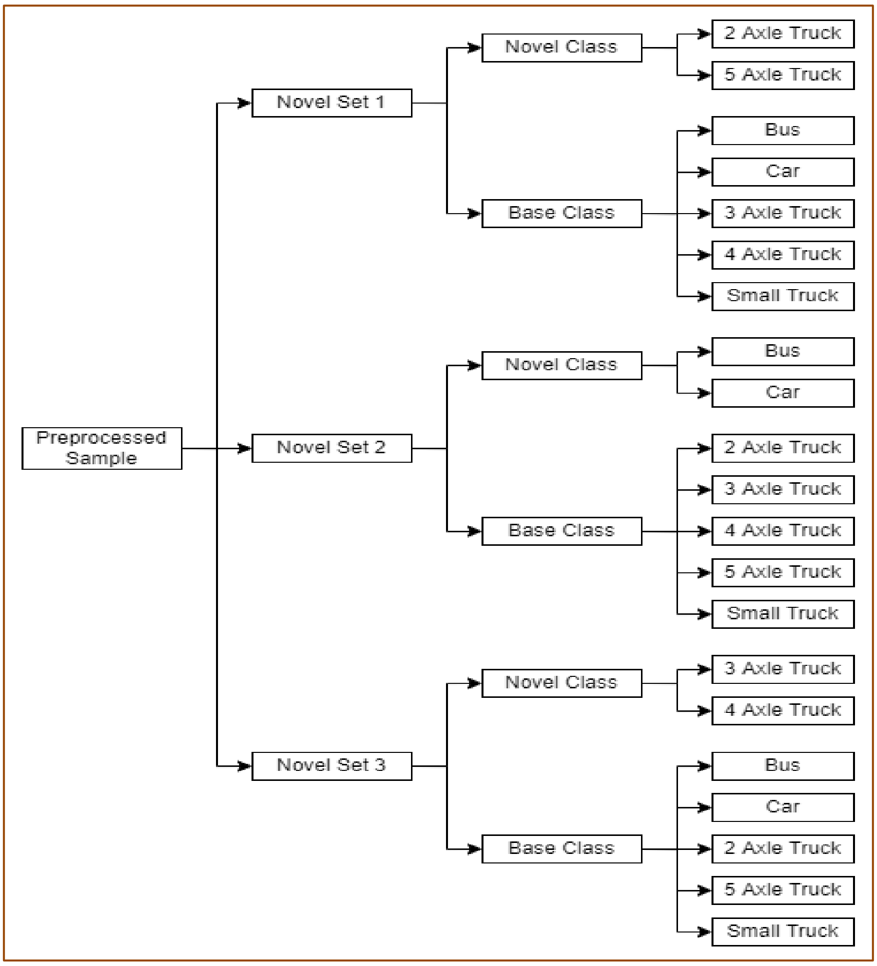

Fig. 4 Three novel setting used on each training generation for few-shot learning

B. Meta-Learning and Fine-Tuning

Each set of the samples goes through two processes before being evaluated which are the Meta-Learning process and Fine-Tuning process in Figure 5. The meta-learning processes extract the meta-feature of each sample to be learned by the model, it is also employing feature reweighting to adjust its weight for the best possible outcome. This process trains 5 base samples, the sample processed with You Only Look Once Version 2 (YOLOV2) alike process that uses the Darknet configuration with Darknet convolutional weight [16]. The results are fine-tuned within the next process. The fine-tuning process fine-tunes the parameters from the source without the target classes, which is the novel set. The process is repeated three times according to the amount of novel set. The results are 3 different weight files that come from three different novel sets. Each weight was evaluated by using a python implementation of MATLAB evaluation and also with YOLOv4 testing module. The test results were used as the base of comparison for the experiments [7].

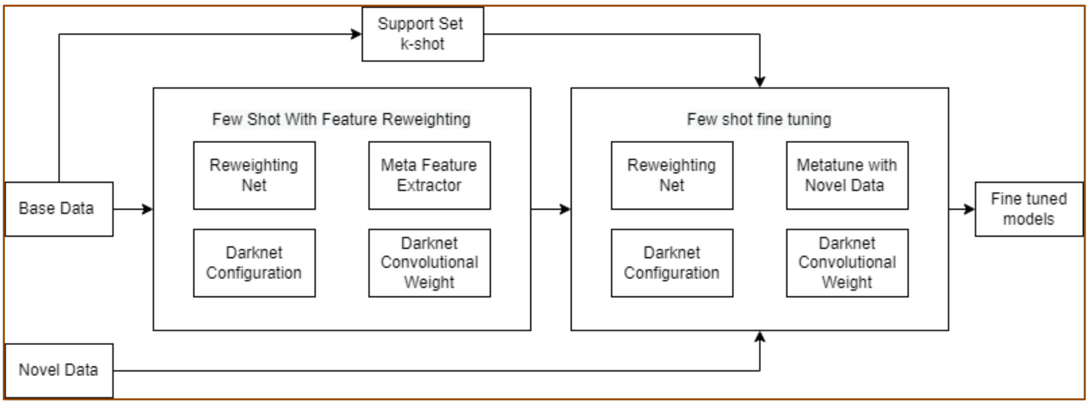

Fig. 5 Breakdown of Few-Shot Two Stage Fine Tuning with Feature Reweighting Approach

For the result of the main experiment, it was shown in the form of table which contains mean Average Precision (mAP) of each shot within each different novel set. The mAP depicted as mean Average Precision across prescribed class. The results of Baseline YOLO experiment with the sample dataset were also used for comparison. The value m refers to the value of k-annotated box, the value n refers to class id assigned to each class, the result rounded to 1 decimal value, the scoring was in percentage which is scaling from 0 to 100%. This presentation was meant to show how well the FSRW method improved from the baseline YOLO method. To attain the precise result despite low data, the result was the mean of the 10 times testing iteration. The result shows how well the learning method compares to its baseline method.

IV. RESULTS AND DISCUSSIONS

A. Preliminary Results

The preliminary phase of this research covers data preprocessing and data annotation parts. By using ROBOFLOW AI, a tool to annotate datasets, 7 classes had been annotated with bounding boxes and labeled accordingly. The models also had been successfully pre-trained with the PASCAL VOC and COCO dataset resulting in a pretrained model and a weight file. The weight file was a file containing the learned parameters from training with format of the Darknet convolutional format. The weight file was updated for every 10-epoch interval of training. In this preliminary experiment, the resulting weight file was named 00040 which means this file saved after around more than 40 epochs.

B. Research Results

The result of the research was visualized on Table 3. Note that to increase prediction accuracy, the result was the average of 10-time testing on the same trained model. This was done to cope with the small number of testing samples. The result in Table 3 was the mAP of every shot depicted using percentage scaling from 0 to 100%.

TABLE 3. Novel Sets With 2 Classes from the Indonesia Road Vehicle Dataset and Its Corresponding Class Labels

Novel Set 1					
Method/Shot	1	2	3	5	10
YOLO	0	0	1.8	1.8	1.8
YOLO + ft	6.3	3.3	3.6	7.8	7.8
FSRW	14.3	17.1	20.0	28.6	40.0
FSRW + Ft	19.1	23.1	27.1	35.7	52.6
Novel Set 2					
Method/Shot	1	2	3	5	10
YOLO	0	0	1.8	1.8	1.8
YOLO + ft	6.3	3.3	3.6	7.8	7.8
FSRW	14.3	17.1	20.0	28.6	40.0
FSRW + Ft	19.1	23.1	27.1	35.7	52.6
Novel Set 3					
Method/Shot	1	2	3	5	10
YOLO	0	0	0	1.8	0.9
YOLO + ft	5.7	3.6	3.9	3.9	7.8
FSRW	11.4	11.4	14.3	22.9	34.3
FSRW + Ft	21.7	24.6	29.7	38.9	49.9

From Table 3 it can be seen that the few tuning processes significantly improve average precision of the model that trained with the FSRW models. The result also shows consistent improvement of the models for each novel class set within the amount number of shots. While the YOLO baseline performs with low score on each shot of novel set, the fine-tuned version could perform better. FSRW method compared to the base YOLO method, the result FSRW was far superior to the baseline method. An increase of average mAP of around 30% was also achieved with the fine-tuned models.

C. Data Representation and Performance Analysis

This subchapter explained the performance and result that was tested with fine-tuned model. There were three novel set, and each set were analyzed through the result of *k*-annotation box result with their respective shots, which was 1, 2, 3, 5, and 10 of each set. Each result compiled in form of confusion matrix followed by table containing precision, recall, and F1 score. The scoring inside the table scales from 0 to 1.

C.1 Novel Set-1

The overall detection result of novel set-1 detection is shown in Figure 6 below which contains line graph depiction for each k-shot. Figure 6 shows a bit of instability on the 2-shot detection on small truck, there was also slower improvement on truck with 3 axles and 4 axles. The reason may be because both cases have very few differentiating features from each other, except for the number of axles. Other than that, there was stable improvement except for the bus class that spiking through score achieving 82% precision on 10-shot.

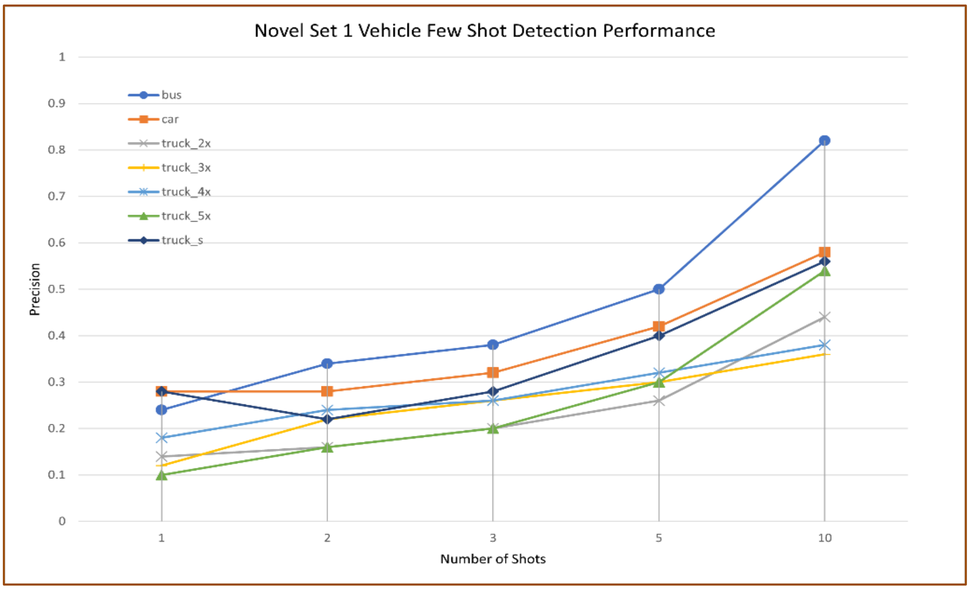

Fig. 6 Line graph showing the improvement on model precision in different k-shot detection on novel set 1

C.2 Novel Set-2

The overall detection result of novel set-2 detection was shown in Figure 7 which contains line graph depiction of each k-shot. The line graph in Figure 7 shows the 3 and 4 axle truck having the lowest score, just like on previous set. On the other hand, now there was no instability in terms of detection in all classes which means that the detection is now stable.

C.3 Novel Set 3

The overall detection result of novel set-3 detection is shown in Figure 8 below which contains line graph depiction of each k-shot. The line graph in Figure 8 shows that there was instability that occurred again in the small truck classes. The bus class achieved the highest precision just like in novel set 1 and the other classes have a stable improvement. Still, three axle and four axle truck stay the lowest.

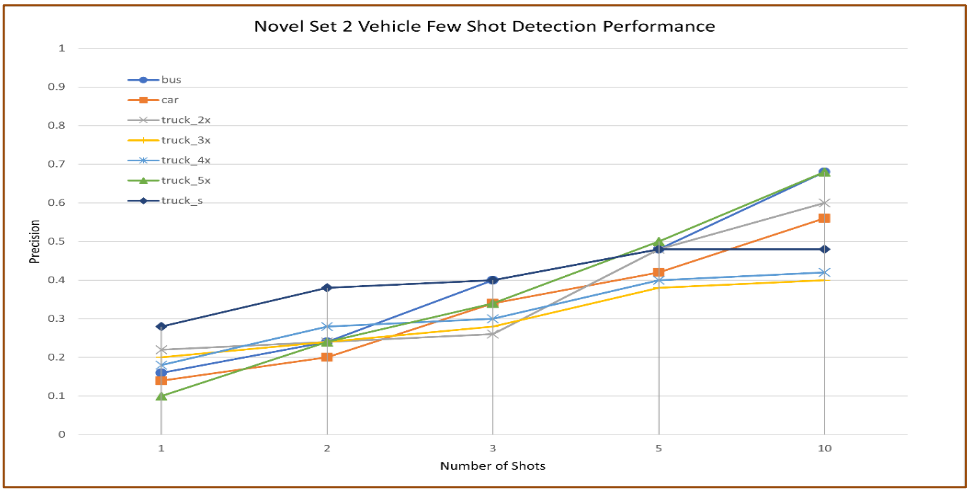

Fig. 7. Line graph showing the improvement on model precision in different k-shot detection on novel set 2

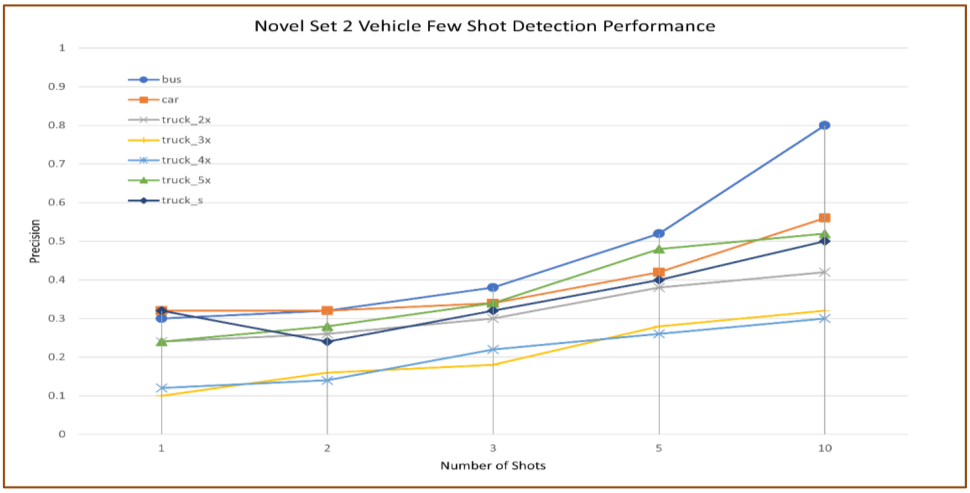

Fig. 8 Line graph showing the improvement on model precision in different k-shot detection on novel set 3

D. Comparison with the Other Baselines

The FSRW method initially made to works better on small amounts of data, however it should also work well with abundant data. That could make it more desirable. For that full-scale training using FSRW was required [17]. Table 4 provides the result of full-scale training and its comparison with another selected model. The other model was a similar system that employs Darknet Framework. All of them were used for comparison purposes. The models involved in the comparison are YOLO, YOLOv2, and YOLOv4. Table 21 below shows the comparison of involved model on mAP scoring, the score depicted in percentage scaling from 0 to 100%.

TABLE 4. Few Shot Detection Performance (in accuracy): Comparison on Base Categories Evaluated from Three Different Sets of Base Categories

Method/Set	Base Set 1	Base Set 2	Base Set 3
YOLO Baseline	70.3	72.4	70.2
YOLOv2 Baseline (2016)	73.3	74.1	70.6
YOLOv4 Baseline (2020)	75.3	77.9	73.2
FSRW (2020)	70.6	72.1	71.8

Based on Table 4, FSRW algorithm outperforms YOLO baseline mAP on base set 1 by 0.3% and base set 3 by 1.6%, it also outperforms YOLOv2 on base set 3 by 1.2% both on slight margin. But it was outperformed by YOLOv4 that deployed in the same year of FSRW method which means on large full-scale training FSRW managed to attain no higher value for another method that was developed to cope with larger dataset as well. Still the model was intended for a few labeled samples training.

V. CONCLUSION

Having to perform detection and identification by utilizing small amount of sample was a challenging problem. By employing FSRW algorithm, the base 1-shot on the first novel set obtained a good mAP with score of 14.3% even before its fine tuned, fine tuning the model resulting with 19.1% mAP score, which was around 30% increase on mAP. It also has steady improvement on increased shot. On the second novel set there was also a 20% increase of mAP on 1-shot. On the third novel set it achieved around 95% increase on mAP resulting in huge improvement. Based on the remarks, it can be concluded two things: First, the few-shots fine tuning positively increase the learning performance and mAP which state that the model was able to effectively increase the overall precision and performs well with the Indonesia case. Second, in dealing with scarcity of trucks with different number of axles the model was performing in a good way, even though there was instability on small truck sample, others were doing better. Especially on 1-shot the model could achieve more than 10% precision which was higher than any other baseline method. In conclusion, this research demonstrates the overall performance of the FSRW algorithm implemented in Indonesia road vehicle dataset. The result shows that FSRW algorithm achieves better accuracies compared to the baseline YOLO method with few samples.

Funding Statement: This work was supported by JSPS KAKENHI Grant Number JP 18K11350.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

REFERENCES

[1]. G.S. Dhillon, P. Chaudhari, A. Ravichandran and S. Soatto, “A baseline for few-shot image classification,” International Conference on Learning Representations, 2019, pp. 1–20.

[2]. Z. Moutakki, I.M. Ouloul, K. Afdel and A. Amghar, “Real-time system based on feature extraction for vehicle detection and classification,” Transport Telecommunication, vol. 19, no. 2, 2018, pp. 93–102.
[3]. K. Balaji, M. Rabiei, V. Suicmez, C.H. Canbaz, Z. Agharzeyva, S. Tek and C. Temizel, "Status of data-driven methods and their applications in oil and gas industry" 80th EAGE Conference and Exhibition, 2018, pp. 1-15.
[4]. W.Q. Zhang, “Numerical simulation in forward hot-extrusion and back hot-extrusion forming process for micro-gear,” DEStech Transactions on Engineering and Technology Research, 2017, pp.1-12.
[5]. R. Berwick, “An idiot’s guide to support vector machines (SVMs): A new generation of learning algorithms key ideas,” Massachusetts Institute of Technology, 2003, pp. 1–28.
[6]. H.K. Leung, X.Z. Chen, C.W. Yu, H.Y. Liang, J.Y. Wu and Y.L. Chen, “A deep-learning-based vehicle detection approach for insufficient and nighttime illumination conditions,” Applied Sciences, vol. 9, no. 22, 2019, pp. 1-27.
[7]. B. Mahaur, N. Singh and K.K. Mishra, “Road object detection: A comparative study of deep learning-based algorithms,” Multimedia Tools and Applications, vol. 81, no. 10, 2022, pp. 14247–14282.
[8]. A.R. Pathak, M. Pandey and S. Rautaray, “Application of deep learning for object detection,” Procedia Computer Sciences, vol. 132, 2018, pp. 1706–1717.
[9]. A. Majee, K. Agrawal and A. Subramanian, “Few-shot learning for road object detection,” 2021, pp. 1-8 [Online]. Available: http://arxiv.org/ abs/ 2101.12543
[10]. S. Gidaris, P. Paristech and N. Komodakis, “Dynamic few-shot visual learning without forgetting : Supplementary material,” Conference on Computer Vision and Pattern Recognition, vol. 9, 2018, pp. 4367–4375.
[11]. G. Chandan, A. Jain, H. Jain, and Mohana, “Real time object detection and tracking using deep learning and OpenCV,” International Conference on Inventive Research in Computing Applications, 2018, pp. 1305–1308.
[12]. Y. Fan, Y. Li and A. Zhu, “A few-shot learning algorithm based on attention adaptive mechanism,” Journal of Physics: Conference Series, vol. 1966, no. 1, 2021, pp. 1-7.
[13]. R. Zhang, T. Che, Y. Bengio, Z. Ghahramani and Y. Song, “Metagan: An adversarial approach to few-shot learning,” Advances in Neural Information Processing Systems, 2018, pp. 2365–2374.
[14]. Z. Zhou, S. Li, W. Guo and Y. Gu, “Few-shot aircraft detection in satellite videos based on feature scale selection pyramid and proposal contrastive learning,” Remote Sensing, vol. 14, no. 18, 2022, pp. 1-19.
[15]. Y. Wang, Q. Yao, J.T. Kwok and L.M. Ni, “Generalizing from a few examples: A survey on few-shot learning,” ACM Computing Surveys, vol. 53, no. 3, 2020, pp. 1–34.
[16]. W. Wu, H. Liu, L. Li, Y. Long, X. Wang, Z. Wang and Y. Chang, “Application of local fully convolutional neural network combined with YOLO v5 algorithm in small target detection of remote sensing image,” PLoS One, vol. 16, no. 10, 2021, pp. 1–15.
[17]. B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng and T. Darrell, “Few-shot object detection via feature reweighting,” International Conference on Computer Vision, 2019, pp. 8419–8428.