

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage: www.joiv.org/index.php/joiv



Optimization of Vehicle Object Detection based on UAV dataset: CNN Model and Darknet Algorithm

Abdul Haris Rangkuti^{a,*}, Varyl Hasbi Athala^a

^a Informatics Department, School of Computer Science, Bina Nusantara University, Jakarta,11480, Indonesia Corresponding author: *rangku2000@binus.ac.id

Abstract— This study was conducted to identify several types of vehicles taken using drone technology or Unmanned Aerial Vehicles (UAV). The introduction of vehicles from above an altitude of more than 300-400 meters that pass the highway above ground level becomes a problem that needs optimum investigation so that there are no errors in determining the type of vehicle. This study was conducted at mining sites to identify the class of vehicles that pass through the highway and how many types of vehicles pass through the road for vehicle recognition using a deep learning algorithm using several CNN models such as Yolo V4, Yolo V3, Densenet 201, CsResNext –Panet 50 and supported by the Darknet algorithm to support the training process. In this study, several experiments were carried out with other CNN models, but with peripherals and hardware devices, only 4 CNN models resulted in optimal accuracy. Based on the experimental results, the CSResNext-Panet 50 model has the highest accuracy and can detect 100% of the captured UAV video data, including the number of detected vehicle volumes, then Densenet and Yolo V4, which can detect up to 98% - 99%. This research needs to continue to be developed by knowing all classes affordable by UAV technology but must be supported by hardware and peripheral technology to support the training process.

Keywords—UAV; vehicle; CNN model; CsresNext-Panet50; Densenet201; Yolo V3; Yolo V4.

Manuscript received 31 Aug. 2022; revised 30 Oct. 2022; accepted 15 Nov. 2022. Date of publication 30 Jun. 2023. International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. INTRODUCTION

Drones or Unmanned Aircraft are developing rapidly as a technology for data acquisition of objects on the earth's surface through the air. Drone technology was initially used in the military and then widely applied in the civilian field [1]. In recent years researchers have become interested in conducting image processing research on datasets generated from drones using deep learning. The application of drones in various fields, such as agriculture, aerial surveys, mapping, photography, surveillance, and others, impacts the data explosion and the abundance of drone datasets [2]. Consequently, processing datasets to extract information automatically becomes necessary, and computer vision becomes one of the relevant information technologies to do the job [3]. Drones can acquire thousands of high-resolution images during a single flight, which the operator must analyze. For example, in the case of an object search operation, it is necessary to find small objects (e.g., cars) in the image. The size of such an object will not exceed 50×50 pixels in an image size of 5000 \times 3000 pixels. This task cannot be completed for one person without automation due to the

accumulation of additional and more complex image data [4]. Therefore, we need a computational engine capable of automatically, accurately, and quickly performing object detection analysis.

At first, drones were only used in the coastal and marine fields, but now they have grown in plantations and mining areas. In addition, the use of drones is also increasingly widespread. If its use was only used for documentary activities at the beginning, now it can be more analytical [5]. This becomes very easy thanks to follow-up information in the form of metadata recorded on each photo produced by the drone. This metadata stores valuable information in the form of x and y coordinates and relative elevation points. This metadata can be processed through a photogrammetric scheme to produce informative aerial photo images. At first, the drone's flying height at the preparation stage for flying will affect the quality. As an initial stage of setting, the flight height can be adjusted to vary from 300 meters to 400 meters above ground level. Conversely, if the drone flies higher, it can cover a wider area, but the spatial resolution will decrease so that the map scale detail and the resulting accuracy are also lower [6].

UAV-captured images and their post-analysis are two major categories that fall in commercial applications of aerial vehicles. Applications in aerial images include landslide mapping, search, and rescue, wildlife monitoring, the creation of digital elevation maps, and the utilization of mounted cameras for many purposes. The technology behind innovation in aerial applications is responsible for digital video stabilization, autonomous navigation, and terrain analysis [5]. One of the attractions is that many researchers apply and use deep learning to handle and process drone data acquisition results and analyze and detect vehicle object class systems. Several researchers state that deep learning technology is one of the state-of-the-art in the field of artificial intelligence and computer vision for the domains of image classification, object detection, and natural language processing (NLP). The data obtained from drone acquisitions are mostly in the form of images and videos, where drones are flown at various altitudes ranging from low altitudes (10–99m) and medium altitudes (100-400m) [7].

Object detection in low-altitude UAV datasets has been performed using deep learning with some CNN models and examples of detections in Fig. 2. Object detection is a technique of identifying variable objects in a given image and inserting a boundary around them to provide localization coordinates. Object detection in aerial images has gained the attention of researchers working in this field, as aerial vehicles provide stereo views from a camera mounted on them. Deep learning-based object detection approaches rapidly revolutionize autonomous navigation vehicles' capabilities [8]. The work presented in the paper is intended to offer detection accuracy in a wide-ranging indication of the use of deep learning-based object detection approaches, specifically on low-altitude aerial datasets. It will serve as a repository of all current growth in deep learning-based object detection in low-altitude datasets and also help young researchers consult research issues for further perusal in this field.



Fig. 1 Examples of object detection in UAV datasets [9], [10]

Our research focuses on the need to find several methods based on the convolution network model for object detection in low-altitude UAV datasets and group them into three classes. This research is expected to determine the exact accuracy based on the height and low of the object depicted on the drone/UAV dataset. So that when determining the class for each object, it becomes optimal and reduces errors due to objects that are not too clearly depicted on the UAV Dataset. In addition, from the deepening of several experiments in detecting objects, the role of the deep learning method becomes very dominant in increasing the recognition of objects in the UAV Dataset. Such as how much the average accuracy is to recognize the detected object, although in general, the visible object ranges between 500 - 600 meters above the ground surface.

The main objectives of this research are as follows:

- To review the taxonomy of deep learning object detection algorithms using multiple CNN models.
- Can find out the type of vehicle class seen from an altitude of 500-600 meters above ground level more optimally based on drone data.
- Can find out the optimal MAP of each CNN model used so that it can be a guideline for other researchers in detecting objects from Drone/UAV data.
- This research can be one of the raw models for detecting objects visible from the UAV Dataset.

In addition, the motivation in object detection research on UAV Datasets using deep learning is to achieve the best deep learning algorithm model concerning two main factors: object detection accuracy and data processing speed. One of the state-of-the-art deep learning algorithms in object detection models is Yolo V4, Yolo V3, Densenet 201, and CsResNext-Panet 50. The performance test of the accuracy of object detection results on UAV Datasets has been carried out as presented in the table above, using a performance metric called mean average precision (mAP) or sometimes called simply AP (Average Precision) [11], with a formula such as under.

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}$$
(1)

Where Q is the number of queries in the dataset and AveP(q) is the average precision (AP) for a particular query, q. For a given query, q, the corresponding AP is computed, and then the average of all these AP scores will give a single number called MAP, which measures how well our model is querying. Average precision (AP) is calculated for each class from those under the area precision curve, where AP calculates the average precision value for values between 0 and 1. The display of the detection results using one of the CNN models from the Deep learning algorithm can be seen in Figs 2 and 3.

Figs 2 and 3 illustrate the MAP results from processing UAV Datasets in determining the class of the object that passes through public highways. The training process carried out by the Darknet algorithm will characterize the object class detected in an object in the bounding box. The object class detection concept from UAV Datasets in the videos forms a new approach that can continue to be developed for MAP. The difference in 2 (the two) Figures occurs at the initial time detection of objects of each class for the CsResNext-Panet 50 model; the percentage of accuracy starts at more than 80% - 85%, while in the CNN model, the average accuracy percentage starts from 60-65%, after that, it only reaches

100%, especially to detect motorcycle is too small if we take a look the dataset drone in video form.



Fig. 2 MAP results from object class detection from UAV Datasets on public highways using Densenet 112



Fig. 3 MAP results from object class detection from UAV Datasets on public highways using CsResNext -Panet 50 $\,$

II. MATERIAL AND METHODS

Drones or Unmanned Aircraft are developing; basically, the existence of UAV technology in the world of aviation continues to experience increasing development in recent years. However, as air transportation, it is also used in commercial and military circles, including technology with other functions such as regional mapping, the film industry, maritime patrols, disaster, medical assistance, and forest fire detection [8]. One of the technologies mentioned is the UAV (Unmanned Aerial Vehicle). UAV is a pilotless aircraft operated using remote or automatic control. UAVs have various shapes, sizes, configurations, and characters and are controlled remotely. The data collection results from Drones/UAVs are used to implement several deep learning algorithms for object detection in processing drone-generated datasets. He also uses a UAV Dataset that is flown at low altitudes, and in his research, the dataset used is the result of drone acquisition at an altitude of 350m [12]. There are at least four categories in computer vision on the UAV Dataset: image object detection, video object detection, single object The tracking, and multi-object tracking. dataset's characteristics are categorized into two types, namely urban and sub-urban areas. Several researchers said that there are several issues and problems in the UAV Dataset, namely small objects, occlusion, spatial scale/resolution variations, and class imbalance [13].

Another research revealed that there are many obstacles that UAVs mostly face, is the difficulty of landing on a base. This difficulty can be solved by the renewal of UAVs, which is the development of landing vision by detecting the helipad to prevent the risk of accidents that could be harmful and could lead to death [15]. Another study used UAVs to detect forest and land fires so that they have an impact on ecosystem damage. Besides, forests in Indonesia continue to shrink every year due to forest fires. One solution to this problem is using a UAV (Unmanned Aerial Vehicle) to direct observations through the camera. In detecting fire by producing an average accuracy of 0.92. The best accuracy was obtained on the 3rd test with a precision score of 0.96, a recall score of 0.98, and an accuracy score of 0.96. This research can continue to be developed [16]. In another paper, an approach for vehicle detection is presented with virtual line-based sensors, which are just straight detection lines that are first set on-road lanes. The proposed method has an outstanding advantage in any condition, such as excellent traffic jams, sunny, cloudy, and rainy days, nighttime, or even tunnels with complex illumination [17].

Several studies on UAVs have been widely published in international journals and conferences in different application areas, such as search and rescue [18], air security and monitoring [19], disaster management planning [20], plant management vision [21], and mission communication [22]. Air the vehicle can fly at different speeds to hover over the target, perform outdoor flights, and maneuver at a close range of objects over a suitable place [23]. These features make it suitable for replacing humans in operations where human intervention becomes difficult to perform completely. Some of the major challenges in low-altitude UAV based object detection when compared to standard images such as largescale variety, dense distribution of objects, arbitrary orientation, objects Relative motion and turbulence of atmospheric conditions cause objects to become blurry [24]. All these challenges lead to object development detection in low-altitude aerial images using low-level scene features and immersive features to process. Some other important critical issues in object detection on drone platforms due to differences in mAP can be seen [25]. An overview of the percentage of drone technology utilization in several types of activities supported by deep learning algorithms. All implementations can be seen in Fig. 4.



Fig. 4 Some examples in some fields of the use and utilization of object detection [1]

It is quite evident in recent years that a boost in research publications happened due to the emergence in the field of deep learning-based object detection, but a high value of accuracy cannot be achieved in the case of low-altitude UAVs. Object detection is infinite if we consider each and every development, but we would strictly stick to algorithms that have scope in low-altitude aerial images [26]. The literature on object detection in aerial images has been classified into two categories: classical and modern object detection approaches. categorization The classical includes conventional techniques, including vision-based and machine classifier-based approaches. Whereas modern deals with deep learning-based algorithms, which is our focus area. Classical object detection approaches include all major developments in aerial images using handcrafted features based on machinelearning approaches [27].

A. Measuring Model Performance

In types of vehicles, the detection study uses different CNN models to improve recognition; each model will perform differently. This happens because each CNN model has a different architecture that makes each model unique. This is why some models could work well in certain situations, especially in detecting the vehicle object. Therefore, by using the data collected throughout the training process, it will be possible to compare the four CNN models. Several calculation metrics like mAP, Precision, Recall, and F1-score are used as comparison variables. The comparison variables between each CNN model will determine the best approach to detect vehicle objects on highways based on UAV Dataset in Video form. The samples needed for measuring the CNN model's performance are split into two categories. The first category is the positive samples, which have the targeted object in them. The second category is the negative samples, which have none of the targeted objects.

1) Precision and Recall: There are two necessary variables to calculate the precision value. The first variable is the number of positive samples the model correctly classified and the last one is the total number of samples classified as positive samples (whether the model correctly classified them or not). The range value of precision is from 0 to 1, with 0 as its lowest score and 1 as its highest score. This precision value reflects the model's reliability when classifying the positive

samples. The result of precision is obtained by dividing only correctly classified positive samples by the total number of positive samples. Compared to precision, a recall is calculated by dividing the number of positive samples the model correctly classified and the number of total positive samples [28], [29]. Recall completely ignores the negative samples and only focuses on the result of the positive samples. With the range the same as precision, a recall measures how many of the model correctly classifies positive samples. Both precision and recall formulas are illustrated below:

$$Precision = \frac{IRUE_{positive}}{TRUE_{positive} + FALSE_{positive}}$$
(2)

$$Recall = \frac{TRUE_{positive}}{TRUE_{positive} + FALSE_{negative}}$$
(3)

Where:

 $TRUE_{positive}$ = total of positive samples that the model correctly classified.

FALSE_{positive}= total of negative samples that the model mistakenly classified as positive samples.

 $FALSE_{negative} = total of positive samples that the model could not be classified.$

2) Intersection over Union (IoU): Intersection over Union or IoU has two things that need to be addressed because the two are defined later in the IoU formula. Those two values are the predicted bounding box and the truth bounding box. The predicted bounding box is the box the model predicts to have one of the targeted objects or items. Meanwhile, the truth bounding box is the box the tester initially marked as the targeted object before the measuring process. Finally, the definition of IoU is the ratio between the intersection of the predicted bounding box and the truth bounding box with the combined area or union of the two boxes (see Fig. 5). The more the predicted box overlay the area of the truth box, the higher the accuracy of the model. In return, the IoU score would be near the value of 1, which is the highest accuracy score [28].



Fig. 5 IoU Formula and Illustration

3) F1-Score and mAP (mean Average Precision): F1 score used the two previous metrics: precision and recall. F1-Score is a metric that combines the precision and recall metrics into a single metric. The formula for the F1-score is defined as the average of precision and recall [28], [29]. Besides F1-score that summarizes the two previous metrics, the mean Average Precision(mAP) is the metric that shows the mean value of average precision for the detection process of all the previously determined classes [17]. Average Precision, or AP, is the average of the precision metric across all recall values between 0 and 1 at various IoU thresholds [28]. The mAP model is one of the core metrics to determine which model has the best overall performance because it considers all previously mentioned metrics. The formula's output will give an F1-score value ranging from 0 to 1, where 1 is the highest accuracy value.

F1 score =
$$2 * \frac{Precision * Recall}{Precision + Recall}$$
 (4)

B. Methodology (Stages)

In research for vehicle object recognition on UAV Datasets, the distance between surfaces is between 300 - 400 meters. Three objects are detected: Car, Truck, and Motorcycle Objects. The truck category is divided into 2, namely trailer trucks and ordinary trucks, and the car is not divided into the type of vehicle. The motorcycle object looks exceedingly small, so it is not easy to distinguish it from a bicycle.

This research is divided into three main parts, namely the preparation stage, the training stage, and the testing stage. The preparation stage is the process of collecting video data and analyzing the video dataset. For the training stage, all data collected is processed, and each data's weight is calculated to be recognized in the testing process. In the testing process, the test data is recognized based on the training data that has been collected. An overview of the processes that occur at the preparation stage can be seen in Fig 6.



Fig. 6 The Preparation stages of in-vehicle object detection

Fig. 5 is the preparation stage. At this stage, video data is collected through Drone/UAV technology and put in the drone video dataset as training data. All data collected will be carried out into object classes, including processing the initial data, which will be processed as the training dataset to support the training stage. Another process in this stage is pre-trained convolutional weight, including conFig. The video file uses dataset labeling directly to carry out the training dataset's image label. It is necessary to prepare things for the experiment. It is required to have data video taken by drone technology, including image labels, path paths, and train data.

In this case, all data were obtained from UAV Dataset. The next stage is the training process, which can be seen in Fig 7.



Fig. 7 The Training stage in vehicle object detection

Fig. 7 informs the training stage, which outputs weight files. As stated before, it requires files which are images for train, image label, class identity, train data, model pre-trained weight, and a matching model configuration before training. This stage aims to use the drone data in video form that were previously prepared in the preparation stage to be trained using the Darknet algorithm. The vehicle detection process will continue until the iteration modulus is completed or the 1000th iteration modulus finishes. The results of the training phase are weight files. Weights are purposed by choice of a target-object-space, which depends heavily on the nature of the objects in the training set and the predicted property". So that to get the most accurate data possible, the training process can be done using the same device and training data set, and the result would be in a less ambiguous weight file. To support the experiment, the Darknet framework was used to help only in the training process, which carried out some CNN models.

After the training process is carried out, it is continued with the testing process on vehicle object data taken from other UAV Datasets. Based on the training data, all testing data will be detected optimally. In this experiment, 4 CNN models were also used to know which is optimal in carrying out the vehicle object detection process. The stages of the testing process can be seen in Fig. 8.

Fig. 8. illustrates that after the training is complete, the system can start the testing phase by importing the required files for testing. These are the trained weight, configuration, and trainer data files. The testing process starts by taking a frame from the UAV Dataset in video form. Then, calculate the prediction using non-max suppression (NMS). The other process is drawing the bounding box, and the system will determine call traffic type and calculate the volume of the

object class as the target object. The bounding box usually comes with other information like class and coordinate. This information is important for vehicle object detection and Keep ID. It is to generate the appropriate virtual key for the UI to receive after the system succeeds in detecting the vehicle object. Then, some condition cases algorithm decides which behavior to apply within the case, including showing the accuracy starting with the lowest percent will grow up after detecting the vehicle. Finally, the bounding box of the vehicle should respond to the Id key and accuracy percent according to the predefined bindings.



Fig. 8 The Training stage in vehicle object detection

Two choices are made if the different objects are too close and have the same id. If yes, a new id is assigned, but if not, the old id is assigned; however, for the assigned id and old id. At the same time, the object is shown from the processed frame. In this testing stage, the output is accuracy, precision, F1 score, and others, so that the UAV Dataset can acquire the accuracy percentage in vehicles detection in the video form in the show frame process with the output testing is the performance of vehicle detection on the video UAV Dataset.

III. RESULT AND DISCUSSIONS

A. Training Results

The training process used the Darknet algorithm model to support, and all of the models resulted in a good loss average result. The training uses PCs with the latest CPU and GPU technology. The use of PC technology will not affect the aftermath use of the models. This method has a benefit in accelerating the training duration of the model because the Darknet framework supports the GPU Acceleration method for the training phase. Thus, reducing the training time when compared to using the CPU for training. If the training is done with much less advanced technology, it will take more time to finish because the output weight file would result in the same file. The bigger the model architecture, the slower the machine can train. The size of the model also affects the model output size. Fortunately, all of the tested models are designed for small devices and have a small architecture that could

Fig. 8 shows the graph training curve can vary depending on the model. At first, most of the model's loss declined in the first 1000 iterations. However, it did not happen to Yolo V3 and Yolo V4 as they have similar architecture, with one being smaller than the other. It will not have any effects as long as it declines to a level. After a steep decline at the start, the loss starts to stabilize in a gentle curve. It shows that the model is starting to understand the given dataset. Finally, the graph shows that the loss stabilizes until the end of the iteration. This result means the trained model has learned the given hand gestures dataset without a problem. Each model architecture is unique and has a beneficial impact in certain cases. Therefore, the experiment can go on using the generated train weights. A more detailed training result can be seen in Table 5.



TRAINING RESULT OF EACH CNN MODELS USING AVERAGE LOSS PARAMETER

Model	Average loss (%)
Yolo V3	0.1546
Yolo V4	0.2984
Densenet 201	0.8129
CsResNext –Panet 50	0.2985

Table 1 shows the average loss of the CNN model to determine how it will perform. Thus, carrying out it is one of the key parameters that could affect the test. As explained before, the lower the average loss, the better the machine understands the dataset. This way, it could potentially affect the performance of detecting objects. If the machine does not understand, it will not detect the object as expected. Table 5 shows that all models have an average loss below 0.2. This value is pretty low enough and acceptable for the experiment. In detecting vehicle objects in the highways CNN model, Yolo V3 has the highest average of 0.1546 with an Approximation time of 0.06. Next, Yolo V4 has the secondhighest average loss at 0.2984 with an Approximation time of 0.09. However, the difference between them is more than 0.143, which is a lot. CsResNext-Panet 50 model comes third with an average loss of 0.2985 with an Approximation time of 0.14. Then, Model Densenet201 has an average loss of 0.8129 with an Approximation time of 0.08. All the vehicle objects must do with proper testing and analysis. The performance of a model cannot be determined just by using loss value. Therefore, the following section will explain the performance in other aspects.

B. Simulation and Results

Before testing the weights trained in the self-service application, the Vehicle object Detection algorithm must import the supporting files. The supporting files are the training label, image path, model configuration, and a .data file type called the trainer. Data. These supporting files are necessary to execute the testing process, which uses the OpenCV library as the inference. OpenCV is an open-source library mainly used for image processing [20]. Then, the selfservice of accuracy and vehicle class detection will automatically be initiated simultaneously. Next, an examination of Vehicle object Detection is performed. This examination was needed to carry out that the trained object detection works properly. In this case, the machine's frames captured and processed were examined in a separate window.



Fig. 10 Vehicle class object detection simulation on UAV dataset

When the application detects a vehicle on the highway, it can run some objects and calculations flawlessly with aboveacceptable performance. That means a great response feel and fast processing speed. This is required for real-time object detection to make sure everything is processed without delay between one frame to another frame and interactions around the bounding box with accuracy number.



Fig. 11 CNN Models Average IoU After Optimization Comparison

However, smaller size comes at the price of processing performance. When it was tested to run the same Object detection, the response and processing speed were unacceptable. The model's network input sizes in the experiment depend on object classes such as trucks, cars, and motorcycles. This is to reduce the processing load, which could increase processing time. The result of the truth bounding box calculation with the combined area or union of the two boxes is in Fig. 11.

Fig. 11 informs the result of calculating the truth bounding box with the combined area or union of the two boxes. In calculating the IoU Performance for the highest on the CNN model, namely CsResNext -Panet at 91.4%, followed by the CNN Yolo V4 model at 86.11%. Meanwhile, the lowest IoU performance was on the CNN Yolo V3 model at 73.6%. So, in this experiment, it is shown that the CsResNext - Panet model has the highest IoU performance. So, this CNN model can be a guideline for future research. While the system will calculate the mean of Procession (MaP) and time processing for each vehicle can be seen in Fig. 12.

Combined Model Inference Time Graph



Fig. 12 Combined Inference Time model Graph Before and After Optimization

In Fig. 12, the inference time model is carried out to determine whether to reduce network size and include to reduce the inference time. Inference time calculates the time between the captured frame and the process until it results in data in terms of object detection [21]. AS information that the bigger the inference time become the slower the detection. This also worsens the experience of using this detection technology. In Fig. 12, In CNN Model show an additional of percent of about more than half the amount of original inference time. In the vehicle object detection process, the UAV Dataset shows that the average inference time is more than 50% compared to the average image accuracy predicted by the system. Even the CsResNext-Panet 50 model has 100% accuracy, but the inference time does not turn out to be 50% or even more than 100%, which is 130.86 ms.

Furthermore, the Yolo V4 model's accuracy is 99.19%, while the inference time is more than 50%, which is 65.8 ms. The Yolo V3 model's average accuracy is 95.75%, while the inference time is more than 50%, which is 56.3 ms. This model CNN is the most balanced in these metrics after optimization. In order to calculate recall, precision, and F1

scores with deep learning algorithms on several CNN models, can be seen in Fig. 13.

Based on this experiment, the highest accuracy is in the CNN CsResNext-Panet 50 model, where the percentage of precision, recall, and F1 score reaches 100%. Followed by the Yolo v4 model, where the average is up to 99%, and the lowest accuracy is the average Yolo V3 model is 96%. So it can be concluded that the approach of the deep learning algorithm with several CNN models in the testing process would be supported by the Darknet algorithm. So, the minimum average accuracy is only 96%. So, it can be concluded that object detection with this approach has worked well and could detect all objects almost perfectly.



Fig. 13 Precisions, Recall, and F1 score of Each Model CNN

Meanwhile, the combined average precision for each vehicle class for each CNN model on the UAV Dataset can be seen in Fig. 14.



Fig. 14 The combination of the average accuracy for each vehicle class on the CNN model $% \left(\mathcal{A}^{\prime}\right) =\left(\mathcal{A}^{\prime}\right) \left(\mathcal{A}$

Fig. 14 informs the average accuracy in recognizing three vehicle class objects using several CNN models in this experiment. It is concluded that the CsResNext-Panet 50 model can recognize all vehicle objects ranging from trucks (including trailers), cars (several types of cars), and motorcycles (including bicycles) on the UAV Dataset where the distance between the surface and the top position of the

drone is between 300-400 meters with the ground moving vehicles for each class up to 100%. The Yolo V4 model can also detect all classes of vehicle objects, such as trucks and cars, up to 100%, while motorcycles (bikes) can detect up to 97.6%. The introduction is continued by using the Densenet 201-Yolo model, where the accuracy of trucks can be recognized well, but for cars, there are only a few errors where the accuracy reaches 99%, and motorcycle objects can be recognized up to 95.5%. For the recognition of moving objects on the UAV Dataset, the Yolo V3 model is depicted; although it has a smaller accuracy than other models, such as motorcycle objects which is only 91.29%, it is still very robust to recognize this image because it is still more than 90%. This experiment also proves that the approach for vehicle object recognition on the UAV Dataset can be recognized on average more than 90%.

In the experiment, four CNN models were tested for their performance in object detection technology on the UAV dataset. Each CNN model has a unique architecture, producing different metric values from each other. This difference will be a key component for comparing the four CNN models and determining which one is most suitable for detecting vehicle object classes. In Figure 13, all CNN models depict the average values of Precision, Recall, F1-Score, and IoU. The CsResNext-Panet 50 model got the highest average IoU value, followed by Yolo V4, Densenet, and yolo V3, thus affecting Precision, Recall, and F1-Score on each CNN model. Changes in the way each model detects objects in the window frame after optimizing statistically depicted accuracy.

IV. CONCLUSION

In this study, the right approach is needed to optimize the detection of the three classes of vehicle objects depicted in the UAV dataset. For example, a motorcycle class is similar to a bicycle, a car class is almost similar to several types of cars, and a truck class is almost the same for trailers and general trucks. In detecting three vehicle class objects on the UAV dataset, the deep learning algorithm with 4 CNN models and Darknet algorithms would be used to support the training process. The experimental results can be concluded that the CsResNext-Panet 50 and Yolo V4 as the solution to recognize the three-vehicle class in UAV datasets such as car, truck, and motorcycle. Based on the experiment results on the UAV dataset, it is illustrated that the CsResNet50-Paket model has produced precision, recall, and F1 Scores with a percentage of up to 100% followed by an average IoU of more than 90%. Furthermore, the Yolo V4 model has an accuracy percentage is more than 98% with an average IoU of more than 85%.

REFERENCES

- Mittal, P., Sharma, A., & Singh, R. (2020). Deep learning-based object detection in low-altitude UAV datasets: A survey. Image and Vision Computing, 104046. https://doi.org/10.1016/j.imavis.2020.104046.
- [2] Madawalagama, S. L., N. Munasinghe, S. D. P. J. Dampegama, and L. Samarakoon. 2016. Low Cost Aerial Mapping With Consumer-Grade Drones.
- [3] Martin, C., Parkes, S., Zhang, Q., Zhang, X., McCabe, M. F., & Duarte, C. M. (2018). Use of unmanned aerial vehicles for efficient beach litter monitoring. Marine Pollution Bulletin, 131, 662–673. doi:10.1016/j.marpolbul.2018.04.045.
- [4] Konoplich, G. V., Putin, E. O., & Filchenkov, A. A. (2016). Application of deep learning to the problem of vehicle detection in UAV images. Proceedings of the 19th International Conference on

Soft Computing and Measurements, SCM 2016, 4–6. https://doi.org/10.1109/SCM.2016.7519666.

- [5] I. Colomina, P. Molina, Unmanned aerial systems for photogrammetry and remote sensing: a review, ISPRS J. Photogramm. Remote Sens. 92 (2014) 79–97.
- [6] Udin, W. S., and A. Ahmad. 2014. Assessment of Photogrammetric Mapping Accuracy Based on Variation Flying Altitude Using Unmanned Aerial Vehicle. IOP Conference Series: Earth and Environmental Science. 18: 12027.
- [7] Brian Dipert, Vision Processing Opportunities in Drones, [Online] Available: https://www.embedded-vision.com/platinummembers/embedded-vision-alliance/embeddedvisiiontraining/documents/pages/drones 2017 (Accessed: 24- Jan-2019).
- [8] A.R. Pathak, M. Pandey, S. Rautaray, Application of deep learning for object detection, Procedia Comput. Sci. 132 (2018).
- [9] K. Mok, Deep Learning Drone Detects Fights, Bombs, Shootings in Crowds, [Online]. Available: https://thenewstack.io/deep-learningdrone-detects-fights-bombsshootings-in-crowds/ (Accessed: 13-Jan-2019).
- [10] Detection and Counting of Arabian Oryx from Aerial Image, [Online]. Available: https://blogs.flytbase.com/ai-drones/ (Accessed: 18-Sep-2018).
- [11] R. O. Prakash and C. Saravanan, "Autonomous robust helipad detection algorithm using computer vision," Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016, pp. 2599–2604, 2016.
- [12] M. Tzelepi, A. Tefas, Human crowd detection for drone flight safety using convolutional neural networks, Signal Processing Conference (EUSIPCO), 2017 25th European 2017, pp. 743–747.
- [13] Arezoo Nazerdeylami, Babak Majidi, Ali Movaghar.(2021).Autonomous litter surveying and human activity monitoring for governance intelligence in coastal eco-cyber-physical systems,Ocean & Coastal Management, Volume 200, 105478,ISSN 0964-5691. doi:10.1016/j.ocecoaman.2020.105478.
- [14] Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of Deep Learning for Object Detection. Procedia Computer Science, 132(Iccids), 1706-1717. https://doi.org/10.1016/j.procs.2018.05.144.
- [15] Wahyudin, A., & Husin, Z. (2020). Sistem Deteksi Objek Helipad Pada Pergerakan Autonomous Fixed Wing Drone Berbasis Algoritma Yolo (Doctoral dissertation, Sriwijaya University).
- [16] Prasanta, M. R., Pranata, M. Y., Firnanda, M. A., & Sendari, S. (2022). Rancang Bangun Quadcopter Drone Untuk Deteksi Api Menggunakan YOLOv4. CYCLOTRON, 5(1).

- [17] Tian, Y., Wang, Y., Song, R., & Song, H. (2015, October). Accurate vehicle detection and counting algorithm for traffic data collection. In 2015 International Conference on Connected Vehicles and Expo (ICCVE) (pp. 285-290). IEEE.
- [18] S. Rathinam, et al., Autonomous searching and tracking of a river using an UAV, 2007 American Control Conference 2007, pp. 359–364.
- [19] P.-M. Olsson, J. Kvarnström, P. Doherty, O. Burdakov, K. Holmberg, Generating UAV communication networks for monitoring and surveillance, 2010 11th International Conference on Control Automation Robotics & Vision 2010, pp. 1070–1077.
- [20] S.M. Adams, C.J. Friedland, A survey of unmanned aerial vehicle (UAV) usage for imagery collection in disaster research and management, 9th International Workshop on Remote Sensing for Disaster Response, 8, , 2011.
- [21] E.A. George, G. Tiwari, R.N. Yadav, E. Peters, S. Sadana, UAV systems for parameter identification in agriculture, 2013 IEEE Global Humanitarian Technology Conference: South Asia Satellite (GHTC-SAS) 2013, pp. 270–273.
- [22] B. Xu, X. Xu, C.-M. Own, On the feature detection of nonconforming objects with automated drone surveillance, Proceedings of the 3rd International Conference on Communication and Information Processing 2017, pp. 484–489.
- [23] S. Campbell, W. Naeem, G.W. Irwin, A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres, Annu. Rev. Control. 36 (2) (2012) 267–283.
- [24] J. Zhou, C.-M. Vong, Q. Liu, Z. Wang, Scale adaptive image cropping for UAV object detection, Neurocomputing 366 (2019) 305–313.
- [25] P. Zhu, L. Wen, X. Bian, L. Haibin, Q. Hu, Vision meets drones: a challenge, arXiv Prepr. arXiv1804.07437 (2018) 1–11.
- [26] S. Sun, C. Salvaggio, Aerial 3D building detection and modeling from airborne LiDAR point clouds, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 6 (3) (2013) 1440–1449.
- [27] Z. Li, W. Shi, P. Lu, L. Yan, Q. Wang, Z. Miao, Landslide mapping from aerial photographs using change detection-based Markov random field, Remote Sens. Environ. 187 (2016) 76–90.
- [28] Esri, "How the Compute Accuracy For Object Detection tool works," ArcGIS Pro, Esri. 2020, [Online]. Available: https://pro.arcgis.com/en/pro-app/latest/tool-reference/imageanalyst/how-compute-accuracy-for-object-detection-works.htm.
- [29] A. F. Gad, "Accuracy, Precision, and Recall in Deep Learning." 2020, [Online]. Available: https://blog.paperspace.com/deep-learningmetrics-precision-recall-accuracy/.