Efficient Multi-Receptive Pooling for Object Detection on Drone

Jinsu An[1], Muhamad Dwisnanto Putro[2], Adri Priadana[1], and Kang-Hyun Jo[1]

[1] Department of Electrical, Electronic and Computer Engineering, University of Ulsan, Ulsan, South Korea
[2] Department of Electrical Engineering, Universitas Sam Ratulangi, Manado, Indonesia
jinsu5023@islab.ulsan.ac.kr, dwisnantoputro@unsrat.ac.id, priadana3202@mail.ulsan.ac.kr, acejo@ulsan.ac.kr

Abstract. Object detection is the most fundamental and important research in computer vision to discriminate the location and class of the object in the image. This technology has been continuously researched for the past few years. Recently, with the development of hardware such as GPU computing power and cameras, object detection technology is gradually improving. However, there are many difficulties in utilizing GPUs on low-cost devices such as drones. Therefore, efficient deep learning technology that can operate on low-cost devices is needed. In this paper, we propose a deep learning model to enable real-time object detection on a low-cost device. We experiment to reduce the amount of computation and improve speed by modifying the CSP Bottleneck and SPPF parts corresponding to the backbone of YOLOv5. The model has been trained on MS COCO and VisDrone datasets, and the mAP values are measured at 0.364mAP and 0.19mAP, which are about 0.07 and 0.04 higher than Refinedetlite and Refinedet, respectively. The speed is 23.010 frames per second on the CPU configuration, which is enough for real-time object detection.

Keywords: Object Detection · Drone Vision · Convolutional Neural Network (CNN) · Efficient Module · Attention Modules.

1 Introduction

Nowadays, drone technology has developed rapidly and guided to widespread use for many purposes. Drones, equipped with cameras, can capture images or videos and generate a variety of beneficial application scenarios, such as video surveillance [2], monitoring [34,11], tracking [41] and searching [33,31]. A drone even can enter difficult or dangerous areas that are impossible for humans to perform these works. This approach can also reduce the possibility of risks incurred.

Advances in computer vision have dramatically enhanced drone vision technology. Many works, such as object detection and classification, can be conducted based on video captured by the drone to support the intelligence system. It leads

the drone to localize and classify the objects based on its vision with high accuracy. It can even perform over enormous areas because the drone can capture extensive coverage only in a short period. It pushes drone vision technology to become increasingly popular.

Recently, the rapid development of Convolutional Neural Networks (CNNs) has improved object detection and classification tasks, providing improved results. Many researchers are developing deeper networks to achieve higher performance [20,30,28]. Unfortunately, it guides the architecture to produce enormous parameters and operate inefficiently. A drone practically uses a low-cost device to run its system. Therefore, it requires an efficient model to perform, especially in real-time.

The field of object detection has evolved over the past 20 years. it is generally divided into two methods. It is a traditional image processing method and a deep learning method. The deep learning method is also divided into two types, one-stage, and two-stage. The network proposed in this paper is an Improved one-stage YOLO(You Only Look Once) network. One-stage based YOLO has been presented as superior real-time object detection and brought much attention. YOLOv5 [12] appeared, which applies a Cross Stage Partial (CSP) [36] block with a bottleneck mechanism to make the network more efficient. This method offers many types based on size, which have various performances. Although the framework provides small versions with fewer parameters, the detector still suffers from infeasible results.

CNN architecture creates feature maps at different levels in each layer. The initial layer creates low-level features representing simple shapes, and as the layer deepens, mid and high-level features representing complex features are extracted. In general, small, medium, and large size objects are detected using low, mid, and high-level features. However, even when detecting large objects, for example, low-level features that respond strongly to edges or small instances are needed. We also need a high-level feature that captures the context of the image to detect small objects. To this end, it is possible to more accurately localize by effectively utilizing low, medium, and large features. In order to detect an object, these various feature information are essential. The existing Feature Pyramid Network (FPN) goes through more than 100 layers to deliver low-level information to high-level, but about 10 layers are sufficient in PANet. The detector used in YOLOv5 is applied to three layers of 80, 40, and 20 sizes of PANet. This layer is upsampled from the last layer of the backbone feature map and merged with the previous level feature map of the same size.

In this work, we adjusted the C3 and SPPF [10] layers to operate the object detection algorithm in real-time, the number of parameters of the network must be reduced. The C3 layer and SPPF layer used in the original YOLOv5 are lightened, and the C3 layer is composed of a bottleneck and 3 convolution layers as CSP bottleneck with 3 convolutions. The C3 layer is lightened by adjusting the convolution of the C3 layers from three to two and changing the order of the concatenation and addition operations of the feature map.

The SPPF layer consists of two convolution layers and three max-pooling layers. To lighten the layer, we reduced one max-pooling layer and added an addition operation. The contributions of this work are summarized as follows:

1. A real-time object detection method is proposed to localize the specific object quickly that can be operated on a low-cost device.
2. A new structure of the convolutional block is introduced by modifying the fusion operation on the CSP bottleneck module.
3. SPPF layer is improved to be more efficient. It supports the network to operate on a low-cost device without compromising its accuracy.

2 Related Work

CNN architectures as a backbone have been employed and developed to perform object detection and classification. It has offered outstanding results in extracting features equipped with many techniques to predict object locations with various sizes. Faster R-CNN [30] came to refine the previous version, R-CNN [9] and Fast R-CNN [8], proposed a Region Proposal Network (RPN) to locate the Region of Interest (RoI) and identify the class of objects. Another work, RetinaNet, offered a novel loss called Focal Loss to deal with the class imbalance problem. Meantime, YOLOv3 [25], YOLOv4 [3], and YOLOv5 [12] utilized the Feature Pyramid Network (FPN) [18] strategy to combine features with various levels.

Many researchers designed various efficient CNN architectures as a backbone to perform object detection. Fast-PdNet [27] offered a lightweight CNN architecture with multi-level contextual blocks that produce fewer parameters than general detectors. The detector is specially designed to perform person detection in supporting assistive robots. Another work [1] adjusted C3 module with a residual bottleneck mechanism on YOLOv5 [12] to make the model more efficient.

Several works modified the YOLO framework to perform efficient object detection applied in supporting drone vision. Pruned-YOLOv3/v5 [39] proposed an iterative channel pruning mechanism to design a lightweight network for YOLOv3 and YOLOv5. It gains a satisfactory balance between efficiency and accuracy on MS-COCO and VisDrone datasets. ECAP-YOLO [13], modified from YOLOv5 [12], offered an efficient channel attention pyramid method to deal with small object problems in aerial images. SPB-YOLO [38] also adjusted YOLOv5 [12] with Strip Bottleneck (SPB) module to build an efficient real-time detector for a drone. It achieves a good trade-off between speed and accuracy.

3 The Proposed Method

The proposed architecture has two main modules as shown in Fig. 1. Both are used in the backbone of YOLOv5, which corresponds to the baseline. The first is Efficient Residual Bottleneck (ERB), and the second is Efficient Multi-Receptive Pooling (EMRP).

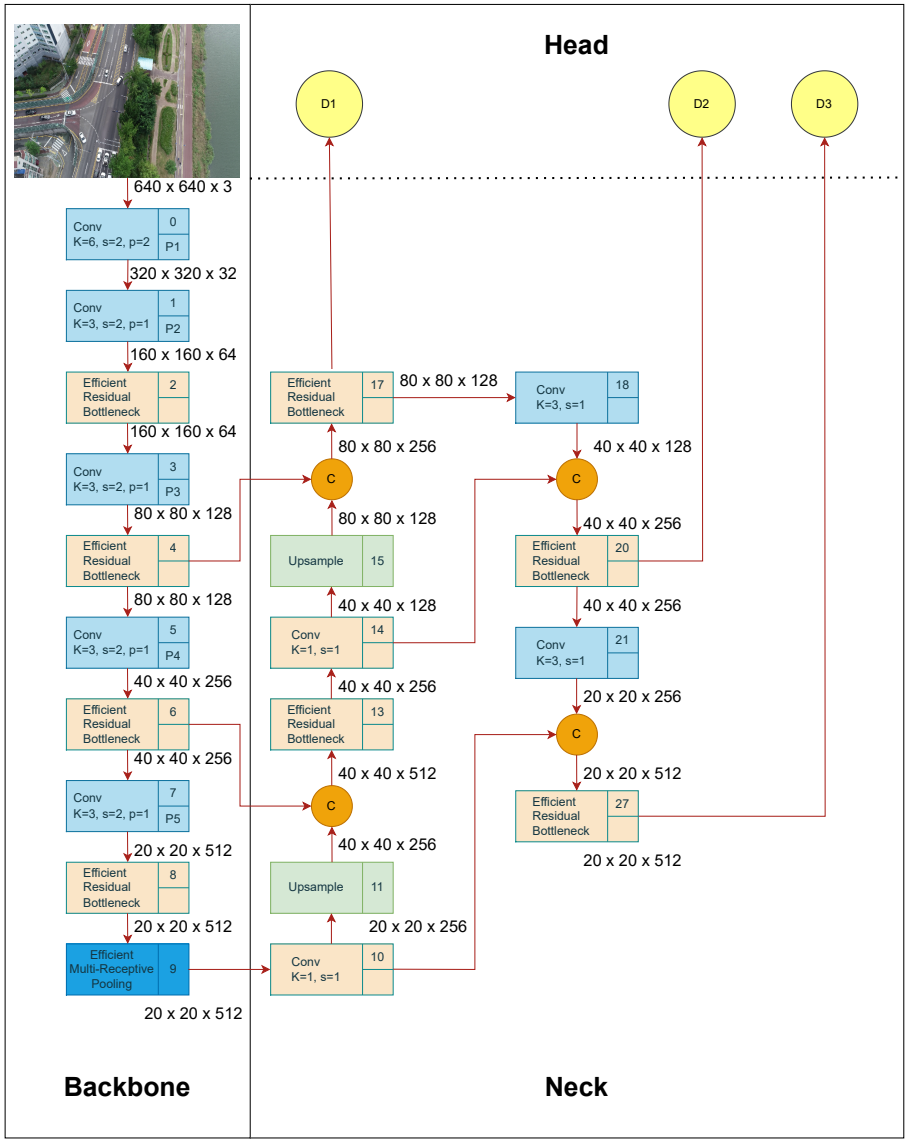

Fig. 1. The proposed architecture. A backbone module is used to extract object features with the proposed efficient methods. Besides, the PANet(Neck) and detection(Head) modules help the detector identify the location of the object in multi-scale variants.

3.1 The Backbone

The framework of YOLOv5 has three main components. It consists of Backbone, Neck, and Head. The Backbone extracts the features of the image and

transfers them to the Head through the Neck. Neck creates a feature pyramid by collecting feature maps extracted from the Backbone. Finally, it is composed of an output layer that detects objects in the Head. CSPDarknet53 [35] is used as the backbone, PANet(Path Aggregation Network) [23] is used for the Neck, and B x (5+C) output layer is used for the Head. B is the number of bounding boxes, and C is the class score. Among them, the C3 layer and SPPF [10] layer of CSPDarknet53 used in the backbone are modified to lighten the deep learning object detection model.

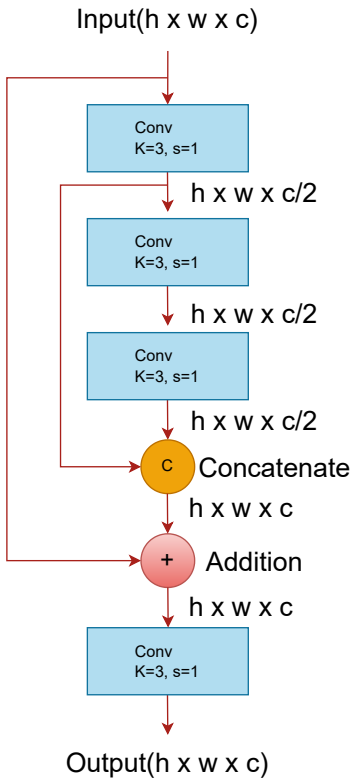

Fig. 2. Efficient Residual Bottleneck.

3.2 Efficient Residual Bottleneck

Efficient Residual Bottleneck (ERB) is an improved layer of the C3 layer used in YOLOv5. The C3 layer is CSP Bottleneck with 3 convolutions and consists of a bottleneck and 3 convolution layers. In order to operate the object detection algorithm in real-time on drones using low-cost devices, the number of parameters

of the deep learning object detection network must be reduced. To decrease the number of parameters, the convolution of the C3 layer is adjusted from three to two, and the order of concatenation and addition operations of the feature map is changed. The proposed network offers an improved backbone that extracts the object features and discriminates the essential elements from the background. It applies a set of convolution layers sequentially using an efficient module. Light blocks apply residual techniques to maintain the quality of the feature map to push high performance in the final prediction. To avoid gradient performance degradation and prevent saturation of the training process, SiLU activation and Batch Normalization are employed sequentially in each convolution operation.

3.3 Efficient Multi-Receptive Pooling

Improved from [10], the efficient multi-receptive pooling is introduced to capture the difference of spatial information that employs a cascade pooling and a simple convolution. It applies convolutional and two sequential pooling to provide various receptive areas. It can increase the options of feature selection from multi-perspective combinations. It uses simple convolution to obtain one spatial area. Two pooling with window size of 5x5 is employed sequentially to capture the maximum value of the features. Combining features from different receptive areas will increase the variety of information so that the network will learn more about the feature type. Then, it applies a convolution operation to mix the various information. The residual technique is used in this module to ensure that the different feature pooling results obtain the expected quality and reduce the error rate of the filtering process.

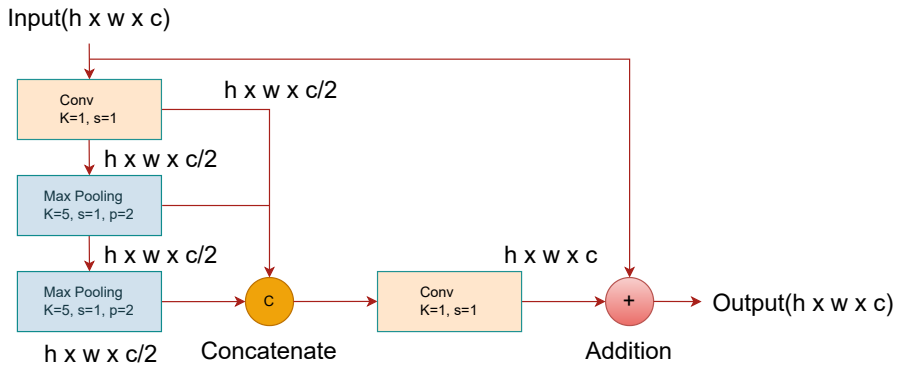

Fig. 3. Efficient Multi-Receptive Pooling, less complexity by double receptive pooling addition path ways

3.4 Loss Function

In YOLOv5, IoU loss, binary cross-entropy, and confidence loss were used as loss functions. Bounding-box regression is the most widely used method in object detection algorithms used to predict the position of an object to be detected using a bounding box. This method aims to correct the position of the predicted bounding box. Bounding box regression uses an overlapping region of the box of the real object and the predicted box location, called Intersection over Union (IOU). First, the IoU loss evaluates the difference between the predicted box position and the actual object's box's intersection, centroid distance, and aspect ratio. Second, we apply a confidence loss to evaluate whether or not there is an object in each cell. Finally, we use binary cross-entropy to measure the probability error of the predicted object class. Binary cross entropy is very effective for training models to solve many classification problems simultaneously. Combining the above three loss functions, the multi-box loss is expressed as:

$$L_{MB} = \lambda_{coord} \sum_{g=1}^{G^2} \sum_{a=1}^{A} g_{ga}^{obj} L_{coord} + lambda_{obj} \sum_{g=1}^{G^2} \sum_{a=1}^{A} 1_{ga}^{obj} L_{obj} + lambda_{cls} \sum_{g=1}^{G^2} \sum_{a=1}^{A} 1_{ga}^{obj} L_{cls} \tag{1}$$

4 Implementation Details

In this section, experiments with MS COCO [22] and VisDrone [41] datasets are described through the proposed architecture. As an experimental environment, the model is implemented using PyTorch in a Linux environment. When training the deep learning model, training is conducted using Intel Xeon Gold CPU and Nvidia Tesla V100 32GB GPU.

5 Experimental Results

5.1 Evaluation on Datasets

The proposed method tested the object detection performance on MS COCO 2017, VisDrone dataset. There are a total of 80 different classes in the COCO dataset, and it consists of a total of 143,575 image data. The COCO dataset contains objects of various sizes, complex backgrounds, and many obstacles, and the proposed model is trained with 118,287 image data. The model is evaluated with 5000 images, and the model is tested with the remaining 20,288 images. The VisDrone dataset consists of 288 video clips (261,908 images) and 10,209 static photos were collected from multiple cameras mounted on drones and has a total of 10 classes (pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus, motor). Among them, the proposed model is trained with 6,471 image data, and the model is evaluated with 1,610 images and tested with 548

Table 1. Detection Result Comparisons on MS COCO Dataset, where Time@CPU1 and Time@CPU2 mean Running Time Tested on Intel I7-6700@3.40GHZ and Intel I5 6600@3.30GHZ, respectively.

Model	**mAP 0.5:.95**	**Backbone**	**Time@CPU1**	**Time@CPU2**
SSD[24]	0.193	MobileNet	128ms	**-**
SSDLite[32]	0.222	MobileNet	125ms	-
SSDLite[32]	0.221	MobileNetV2	120ms	-
Pelee[37]	0.224	PeleeNet	140ms	-
Tiny-DSOD[15]	0.232	DDB-Net+D-FPN	180ms	-
SSD[24]	0.251	VGG	1250ms	-
SSD[24]	0.28	ResNet101	1000ms	-
YOLOv3[29]	0.282	DarkNet53	1300ms	-
RefineDetLite[7]	0.268	Res2NetLite72	130ms	-
RefineDetLite++[7]	0.296	Res2NetLite72	131ms	-
YOLOv5s-ERB	**0.367**	**Improved CSPDarknet53**	-	**43ms**
YOLOv5s-ERB_wosppf	**0.334**	**Improved CSPDarknet53**	-	**36ms**
YOLOv5s-ERB_conv3	**0.366**	**Improved CSPDarknet53**	-	**40ms**
YOLOv5s-ERB_EMRP	**0.364**	**Improved CSPDarknet53**	-	**-**

Table 2. Detection Result Comparisons on VisDrone Dataset.

Model	**mAP 0.5:.95**	**Backbone**
Cascade R-CNN+[5]	0.183	SEResNeXt-50
EnDet	0.178	ResNet101-fpn
DCRCNN[6]	0.178	ResNeXt-101
Cascade R-CNN++[5]	0.177	ResNeXt-101
ODAC	0.174	VGG
DA-RetianNet[26]	0.171	ResNet101
MOD-RETINANET	0.169	ResNet50
DBCL	0.168	Hourglass-104
ConstraintNet	0.161	Hourglass-104
CornetNet*[14]	0.174	Hourglass-104
Light-RCNN*[16]	0.165	ResNet101
FPN*[19]	0.165	ResNet50
Cascade R-CNN*[4]	0.161	ResNeXt-101
DetNet59*[17]	0.153	ResNet50
RefineDet*[40]	0.149	ResNet101
RetinaNet*[21]	0.118	ResNet101
YOLOv5s-vis-c3	**0.195**	**Improved CSPDarknet53**
YOLOv5s-vis-esppf	**0.193**	**Improved CSPDarknet53**
YOLOv5s-vis-c3esppf	**0.190**	**Improved CSPDarknet53**

images. An object detection model is evaluated through a dataset by extracting and learning the features of various objects included in the dataset. To evaluate the model, we use Average Precision (AP) to measure the accuracy of the predicted bounding box, derive AP for each class, and finally calculate the mean Average Precision (mAP) value for all classes. As a result, the mAP values of the proposed method are calculated as 0.364 and 0.190, respectively.

6 Conclusion

This paper proposes efficient residual bottleneck and efficient multi-receptive pooling for a deep learning algorithm capable of real-time object detection. In order to reduce the complexity, the existing CSP Bottleneck and SPPF are improved. And the proposed network is trained on MS COCO and VisDrone datasets. The mAP value on the MS COCO dataset is measured at 0.364, and when compared to RefineDetLite++, the performance increased by about 0.07 mAP difference. The mAP value on the VisDrone dataset is measured at 0.190, and when compared to RefineDet+, the value is about 0.04 mAP higher. In the future, we plan to use the additional detector to increase the object detection rate. As the number of layers in the network increases, the number of parameters required for computation increases. It is expected that the method proposed in this paper can be used to reduce the number of parameters and increase the object detection rate by using additional detectors.

7 Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the government(MSIT).(No.2020R1A2C2008972)

References

1. An, J., Putro, M.D., Jo, K.H.: Efficient residual bottleneck for object detection on cpu. In: 2022 International Workshop on Intelligent Systems (IWIS). pp. 1–4. IEEE (2022)
2. Bera, B., Das, A.K., Garg, S., Piran, M.J., Hossain, M.S.: Access control protocol for battlefield surveillance in drone-assisted iot environment. IEEE Internet of Things Journal **9**(4), 2708–2721 (2021)
3. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
4. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. CoRR **abs/1712.00726** (2017), http://arxiv.org/abs/1712.00726
5. Cai, Z., Vasconcelos, N.: Cascade r-cnn: High quality object detection and instance segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**(5), 1483–1498 (2021). https://doi.org/10.1109/TPAMI.2019.2956516
6. Chakraborty, S., Aich, S., Kumar, A., Sarkar, S., Sim, J.S., Kim, H.C.: Detection of cancerous tissue in histopathological images using dual-channel residual convolutional neural networks (dcrcnn). In: 2020 22nd International Conference on Advanced Communication Technology (ICACT). pp. 197–202 (2020). https://doi.org/10.23919/ICACT48636.2020.9061289
7. Chen, C., Liu, M., Meng, X., Xiao, W., Ju, Q.: Refinedetlite: A lightweight one-stage object detection framework for cpu-only devices. CoRR **abs/1911.08855** (2019), http://arxiv.org/abs/1911.08855
8. Girshick, R.: Fast r-cnn. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 1440–1448. IEEE (2015)

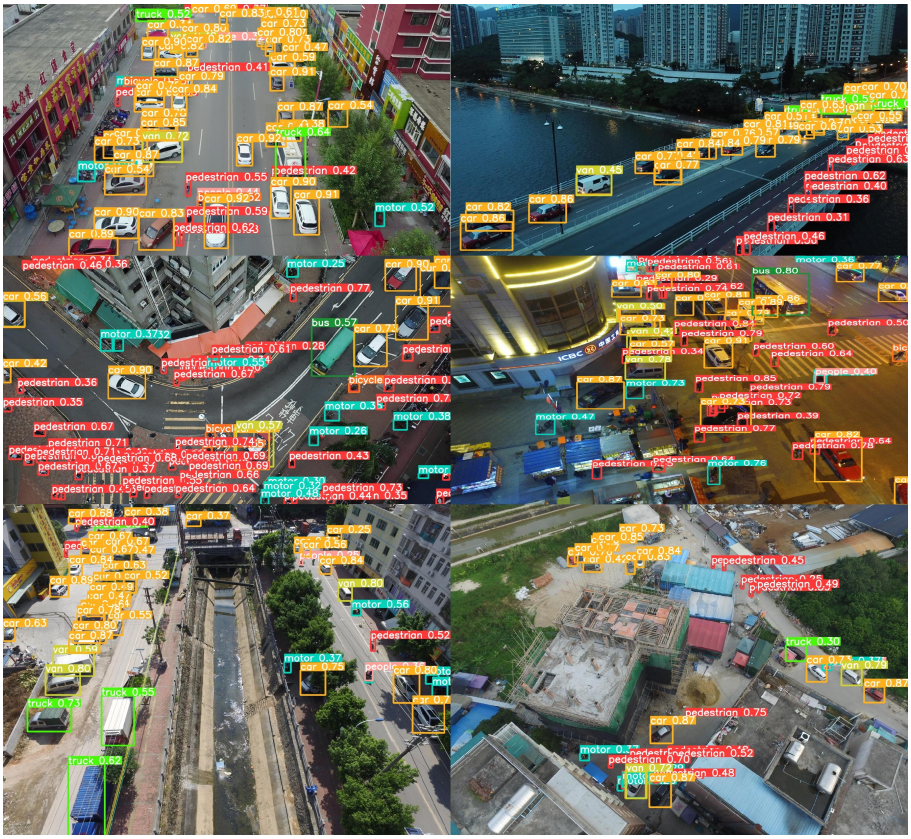

Fig. 4. Visualization of the Detection Result on VisDrone dataset.

9. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 580–587. IEEE (2014)
10. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE transactions on pattern analysis and machine intelligence **37**(9), 1904–1916 (2015)
11. Ikshwaku, S., Srinivasan, A., Varghese, A., Gubbi, J.: Railway corridor monitoring using deep drone vision. In: Computational Intelligence: Theories, Applications and Future Directions-Volume II, pp. 361–372. Springer (2019)
12. Jocher, G., Stoken, A., Borovec, J.: ultralytics/yolov5: v3.0, https://doi.org/10.5281/zenodo.3983579
13. Kim, M., Jeong, J., Kim, S.: Ecap-yolo: Efficient channel attention pyramid yolo for small object detection in aerial image. Remote Sensing **13**(23), 4851 (2021)
14. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. CoRR **abs/1808.01244** (2018), http://arxiv.org/abs/1808.01244
15. Li, Y., Li, J., Lin, W., Li, J.: Tiny-dsod: Lightweight object detection for resource-restricted usages. CoRR **abs/1807.11013** (2018), http://arxiv.org/abs/1807.11013

16. Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J.: Light-head R-CNN: in defense of two-stage object detector. CoRR **abs/1711.07264** (2017), http://arxiv.org/abs/1711.07264
17. Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J.: Detnet: A backbone network for object detection. CoRR **abs/1804.06215** (2018), http://arxiv.org/abs/1804.06215
18. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 936–944. IEEE (2017)
19. Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. CoRR **abs/1612.03144** (2016), http://arxiv.org/abs/1612.03144
20. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 318–327 (2018)
21. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. CoRR **abs/1708.02002** (2017), http://arxiv.org/abs/1708.02002
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
23. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8759–8768. IEEE (2018)
24. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., Berg, A.C.: SSD: single shot multibox detector. CoRR **abs/1512.02325** (2015), http://arxiv.org/abs/1512.02325
25. Murthy, C.B., Hashmi, M.F.: Real time pedestrian detection using robust enhanced yolov3+. In: 2020 21st International Arab Conference on Information Technology (ACIT). pp. 1–5. IEEE (2020)
26. Pasqualino, G., Furnari, A., Signorello, G., Farinella, G.M.: An unsupervised domain adaptation scheme for single-stage artwork recognition in cultural sites. Image and Vision Computing p. 104098 (2021). https://doi.org/https://doi.org/10.1016/j.imavis.2021.104098
27. Putro, M.D., Nguyen, D.L., Priadana, A., Jo, K.H.: Fast person detector with efficient multi-level contextual block for supporting assistive robot. In: 2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS). pp. 1–6. IEEE (2022)
28. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 779–788. IEEE (2016)
29. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR **abs/1804.02767** (2018), http://arxiv.org/abs/1804.02767
30. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(6), 1137–1149 (2016)
31. Sambolek, S., Ivasic-Kos, M.: Automatic person detection in search and rescue operations using deep cnn detectors. IEEE Access **9**, 37905–37922 (2021)
32. Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., Chen, L.: Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. CoRR **abs/1801.04381** (2018), http://arxiv.org/abs/1801.04381

33. Sibanyoni, S.V., Ramotsoela, D.T., Silva, B.J., Hancke, G.P.: A 2-d acoustic source localization system for drones in search and rescue missions. IEEE Sensors Journal **19**(1), 332–341 (2018)
34. Sun, W., Dai, L., Zhang, X., Chang, P., He, X.: Rsod: Real-time small object detection algorithm in uav-based traffic monitoring. Applied Intelligence **52**(8), 8448–8463 (2022)
35. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H.: Cspnet: A new backbone that can enhance learning capability of cnn. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 1571–1580. IEEE (2020)
36. Wang, C., Liao, H.M., Yeh, I., Wu, Y., Chen, P., Hsieh, J.: Cspnet: A new backbone that can enhance learning capability of CNN. CoRR **abs/1911.11929** (2019), http://arxiv.org/abs/1911.11929
37. Wang, R.J., Li, X., Ao, S., Ling, C.X.: Pelee: A real-time object detection system on mobile devices. CoRR **abs/1804.06882** (2018), http://arxiv.org/abs/1804.06882
38. Wang, X., Li, W., Guo, W., Cao, K.: Spb-yolo: An efficient real-time detector for unmanned aerial vehicle images. In: 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). pp. 099–104. IEEE (2021)
39. Zhang, J., Wang, P., Zhao, Z., Su, F.: Pruned-yolo: Learning efficient object detector using model pruning. In: International Conference on Artificial Neural Networks. pp. 34–45. Springer (2021)
40. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. CoRR **abs/1711.06897** (2017), http://arxiv.org/abs/1711.06897
41. Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., Ling, H.: Detection and tracking meet drones challenge. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(11), 7380–7399 (2021)