YOLOv5 with Combination of Coordinate Attention and CBAM for Object Detection on Drone

Jinsu An
Department of Electrical, Electronic and Computer Engineering
University of Ulsan
Ulsan, Korea
jinsu5023@islab.ulsan.ac.kr

Muhamad Dwisnanto Putro
Department of Electrical Engineering
Universitas Sam Ratulangi
Manado, Indonesia
dwisnantoputro@unsrat.ac.id

Adri Priadana
Department of Electrical, Electronic and Computer Engineering
University of Ulsan
Ulsan, Korea
priadana3202@mail.ulsan.ac.kr

Youlkyeong Lee
Department of Electrical, Electronic and Computer Engineering
University of Ulsan
Ulsan, Korea
yklee00815@gmail.com

Junmyeong Kim
Department of Electrical, Electronic and Computer Engineering
University of Ulsan
Ulsan, Korea
kjm7029@islab.ulsan.ac.kr

Kang-Hyun Jo
Department of Electrical, Electronic and Computer Engineering
University of Ulsan
Ulsan, Korea
acejo@ulsan.ac.kr

IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society | 979-8-3503-3182-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/IECON51785.2023.10312634

***Abstract*—Object detection is an important study in computer vision to discriminate the position and class of an object in an image. Object detection in drone images is a technology that automatically detects and classifies objects using deep learning algorithms in flight images taken by drones. Object detection using drone images can rescue human life in disaster situations, grasp the situation at the disaster site, and identify the growth status of crops or pests in agriculture. In addition, it can be used in various fields such as infrastructure management, roads and railways, and city planning. A quick calculation is required. Although rapid computation is possible due to recent hardware development, there are many difficulties in using GPUs in industrial settings. In order to utilize drones in industrial sites, an object detection algorithm capable of real-time operation in a low-cost device is required. In this paper, we propose YOLOv5 with the combination of Coordinate Attention and CBAM for Object Detection on Drone for an algorithm capable of real-time operation in a low-cost device. The proposed architecture makes the model lighter by reducing the number of parameters and improves the object detection rate of the model through Coordinate Attention and CBAM. The model is trained using the VisDrone dataset, and the object detection rate, mAP, increased by about 10% to 22.2mAP, and the number of parameters decreased by about 70% to 2,147,589.**

***Index Terms*—Object Detection, Drone Vision, Attention Module, Deep Learning**

I. Introduction

Along with technological advances, drones have become essential for monitoring and surveillance operations, especially in industries. Drones equipped with high-resolution cameras and other sensors can provide an extensive view of an area, making them ideal for monitoring and surveillance that cover large or remote areas. Moreover, the support of artificial intelligence technology makes drones usable for vision works such as object detection and identification. Many practitioners and researchers strive to extend this technology as a primary tool in various fields, such as factories [1], mining [2], transportation [3], conservation [4], etc.

Deep learning is a technique that has been very successful in object detection tasks, particularly in computer vision. Object detection involves identifying objects within an image or video, and deep learning techniques such as convolutional neural networks (CNNs) are highly effective for this task. In recent years, several deep learning models have been developed that have achieved state-of-the-art performance in object detection. These models typically use a combination of convolutional layers for feature extraction, followed by fully connected layers for classification and bounding box regression. Faster R-CNN [5] become one of the most popular deep-learning models for object detection. It uses a two-stage approach, where regions of interest (ROIs) are first identified using a region proposal network, and then these regions are classified using a second CNN architecture.

YOLO (You Only Look Once), especially the fifth version (YOLOv5) [6], is another popular model that applies a single-stage object detector that processes the entire image in one pass, using a CNN to predict class probabilities and bounding box coordinates directly. As a result, YOLO can perform at a higher detection speed even though it provides a little bit of lower accuracy. Therefore, YOLO is ideal for vision drone that requires fast detection methods.

Several efforts have focused on enhancing a YOLOv5 architecture to boost performance or improve efficiency, especially on the Vision Drone (VisDrone) dataset [7]. Zhang et al. [8] utilizing a channel pruning approach to improve YOLOv5. Wang et al. [9] also tried to enhance YOLOv5 by applying Strip Bottleneck (SPB) block. Both gain satisfactory accuracy and efficiency. The emergence of the attention mechanism also boosted YOLOv5's performance. Squeeze-and-excitation

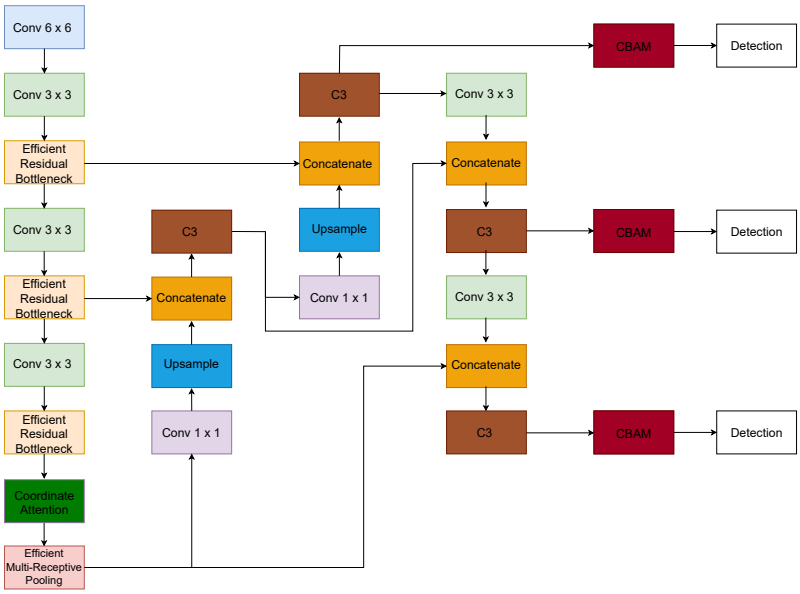

Fig. 1: The proposed architecture. Coordinate attention and CBAM are applied to improve the performance of the original YOLOv5 network.

(SE) module, one of the most attention modules, is used to encourage the detection precision of YOLOv5 in [10]. In another work, Kim et al. [11] proposed an efficient channel attention pyramid module, unifying with YOLOv5, to detect small objects in the VisDrone dataset. Zhu et al. [12] also improved YOLOv5 by utilizing a transformer mechanism and convolutional block attention model (CBAM) for object detection on images captured by drones.

In this work, we propose an enhancement of YOLOv5, which utilizes the combination of some attention modules for object detection on drones. The main contributions of this work are outlined as follows:

1) A real-time object detection method that can operate in a low-cost device by applying an efficient method.
2) Attention Module Combination is introduced by applying Coordinate Attention and Convolutional Block Attention Module (CBAM) to the original YOLOv5 network.

II. The Proposed Architecture

As can be seen in Fig. 1, the proposed architecture has two Attention Modules. The first Coordinate Attention is used before the EMRP layer corresponding to the Backbone of YOLOv5, and the second Convolutional Block Attention Module is applied to the PANet corresponding to the Neck. CBAM is applied before each detector, which is part from the Neck to the Head.

A. The Backbone

YOLOv5's framework has three main components: It is composed of Backbone, Neck, and Head. YOLOv5's Backbone is a network structure responsible for basic functions for object detection. YOLOv5 uses the CSPDarknet53 backbone architecture. CSPDarknet53 introduced a channel division method based on the Darknet53 architecture to increase network efficiency. Through this, it is possible to configure a deeper network with a smaller amount of computation and improve object detection performance. The backbone extracts the features of an image and transfers them to the Head through the Neck. Neck uses the Path Aggregation Network (PANet) architecture. PANet collects feature maps extracted from Backbone to create a feature pyramid and improves object detection accuracy. Lastly, Head uses two Heads for object detection. The Head consists of a $B \times (5 + C)$ output layer that predicts the object's bounding box and class. B is the number of bounding boxes, and C is the class score.

B. Efficient Residual Bottleneck

ERB (Efficient Residual Bottleneck) is an enhanced layer of the C3 used in YOLOv5. The C3 layer in YOLOv5 exhibits a bottleneck phenomenon with three convolutional layers. To operate object detection algorithms in real-time on low-cost devices such as drones, it is necessary to reduce the number of parameters in the deep learning object detection network. To reduce the number of parameters, the C3 layer's convolutions are adjusted from three to two and the sequence of feature map concatenation and additional operations is modified. The proposed network provides an improved backbone that extracts object features and distinguishes essential elements from the background. It applies a series of convolutional layers sequentially using efficient modules. The lightweight blocks employ residual techniques to maintain the quality of feature maps and deliver high performance in the final predictions. To prevent gradient performance degradation and alleviate the saturation of the training process, SiLU activation and batch

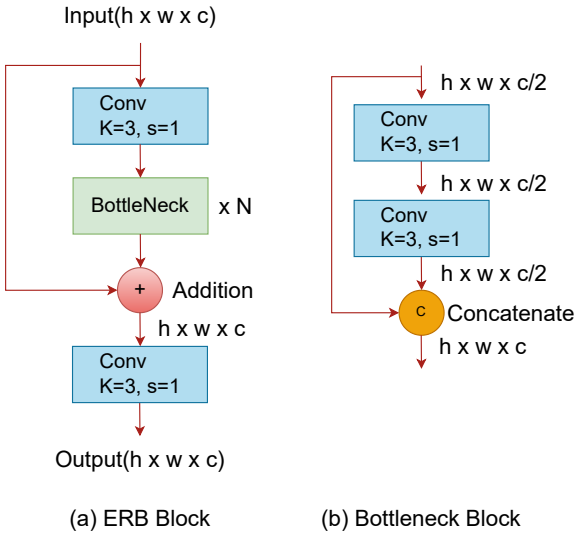

Fig. 2: Efficient Residual Bottleneck. (a) is a module that has improved the C3 layer more efficiently. (b) is the bottleneck structure included in ERB.

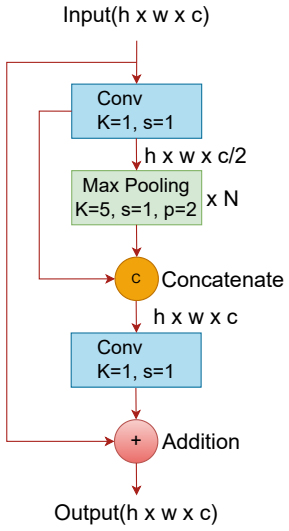

Fig. 3: Efficient Multi-Receptive Pooling. Less complexity by double receptive pooling & addition pathways.

normalization are used sequentially in each convolutional operation.

C. Efficient Multi-Receptive Pooling

EMRP (Efficient Multi-Receptive Pooling) introduces an improved and efficient multi-scale pooling to capture the spatial information difference between cascade pooling and simple convolution. By applying convolutional and two sequential pooling operations, it provides diverse receptive fields. It allows for an increased feature selection option in multi-perspective combinations. A simple convolution is used to obtain a single spatial region. Two pooling layers with a window size of 5×5 are sequentially applied to capture the maximum values of features. By combining features from different receptive fields, the diversity of information increases, allowing the network to learn more about different types of features. Then, convolutional operations are applied to mix the diverse information. The residual technique is used in this module to ensure that the feature pooling results from different receptive fields achieve the expected quality and reduce the error rate in the filtering process.

D. Coordinate Attention

Coordinate Attention [13] is one of the self-attention mechanisms used in the field of computer vision, and is a method of performing efficient attention by utilizing the coordinate information of an object. Coordinate Attention separates the input data into two tensors. The first tensor is the feature map of the input data, and the second tensor is the coordinate information of the input data. Afterward, each of the two tensors is converted to one-dimensional, and the attention weight is calculated using the coordinate information tensor. This weight pays attention to the required position based on the coordinate information of the input data. Since attention is applied only to the required location using coordinate information, the amount of calculation is very small and the size of the model can be reduced. Coordinate Attention is a method for better looking at the location information of an object. It can expect performance improvement in images with many small objects or dense objects. It can increase The EMRP layer is used to handle different sizes and aspect ratios of objects. Therefore, if Coordinate Attention is applied before the EMRP layer, more accurate object detection and classification becomes possible because the location information and size information of the object can be considered together. By adding Coordinate Attention, you can increase performance without significantly affecting speed.

E. Convolutional Block Attention Module

CBAM [14] consists of two Attention Modules. The first step is to encode which channel to focus on with the Channel Attention Module. A value is obtained by performing global max pooling and global average pooling, and nonlinearity is applied by performing MLP on the two vectors encoded by each pooling. After being added, it is finally encoded as a randomized value through sigmoid. The final encoding value Mc is a value expressed as a probability of which feature map it is, considering among different C feature maps. Multiply Mc by the input feature map F to generate F'. The second step is to encode which part of the $C \times H$ number of pixels to focus on with the Spatial Attention Module. After average pooling and max pooling are performed on the channel axis, a feature map of $H \times W \times 2$ is created by concatenation. Then, Ms of $H \times W \times 1$ is generated by performing 7×7 convolution for spatial attention. Ms is multiplied by F' generated by the Channel Attention Module to generate F''. By sequentially applying Channel Attention and Spatial Attention Modules through CBAM, it is possible to consider both inter-channel

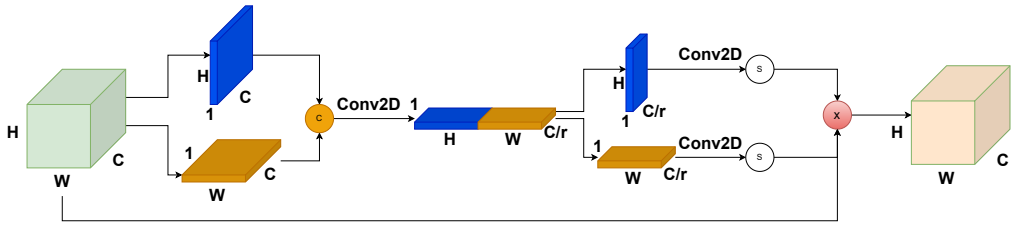

Fig. 4: Coordinate Attention Architecture. Since the attention is focused only on the required location using the coordinate information, the amount of calculation is very small and the size of the network can be reduced.

and spatial relationships, enabling more accurate and efficient feature map extraction.

F. Loss function

The loss function of YOLOv5 is used to improve the model's prediction during training by calculating the difference between the model's predicted bounding box and the actual ground truth bounding box during the object detection task. YOLOv5 uses three main loss functions. It consists of Localization loss, Confidence loss, and Class loss. Localization loss calculates the position difference between the bounding box of the object predicted by the model and the ground truth bounding box. YOLOv5 predicts the center coordinates, width, and height values of the bounding box, and Localization loss is used to improve location accuracy by using Mean Square Error (MSE). Confidence loss calculates the Intersection over Union (IoU) difference between predicted bounding boxes. Calculate the binary cross-entropy loss function between the confidence of the predicted bounding box and the confidence of the ground truth to help accurately detect objects. Finally, class loss helps to predict the correct class an object belongs to, and is calculated as a multi-class Cross-Entropy loss function. The three loss functions are combined to finally calculate the loss of the model's prediction result. Train the model to minimize this value.

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

III. Training and Testing Configuration

In this session, we describe the experiments of the YOLOv5 network with Coordinate Attention and CBAM on the Visdrone dataset. 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 A100 40GB GPU. We train our model for 200 epochs on VisDrone dataset.

IV. Experimental Results

A. Evaluation on Datasets

The VisDrone dataset is a large-scale object detection and tracking dataset based on high-resolution video images captured by multiple cameras mounted on drones. This dataset contains video images taken in various environments, mainly in cities, coastal areas, agricultural lands, and mountainous areas. There are a total of 10 classes (pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus, motor), and it consists of 288 (261,908 images) video clips and 10,209 static photos. The dataset contains video images in a variety of conditions, including day and night, sunny and cloudy, sharp and sunken, and supports up to 1080p. The VisDrone 2019 dataset can be used to solve various computer vision problems such as object detection, tracking, and velocity estimation. Since it contains videos taken on a large scale, high resolution, and under various conditions, it can be usefully used for the development and performance evaluation of object detection and tracking algorithms.

The proposed method tested the object detection performance on the VisDrone dataset. The VisDrone dataset contains many objects of tiny size. In order to detect small-sized objects, a high-resolution image is required or a method capable of extracting features of the object well is required. 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 proposed method shows 22.2map with about 10% higher mAP compared to the original YOLOv5s, and the number of parameters is 2,147,589, which is about 70% less. Fig. 6 presents the detection result of our model. It successfully identified small objects, even occluded challenges

B. Runtime Efficiency

In this paper, ERB and EMRP are applied to YOLOv5 to make the network more efficient. ERB and EMRP are created by improving the C3 and SPPF layers, which correspond to the Backbone of YOLOv5, and the number of parameters of the network could be effectively reduced by removing the C5 block. In addition, when Coordinate Attention and CBAM are applied, the parameters of the network increase, but the

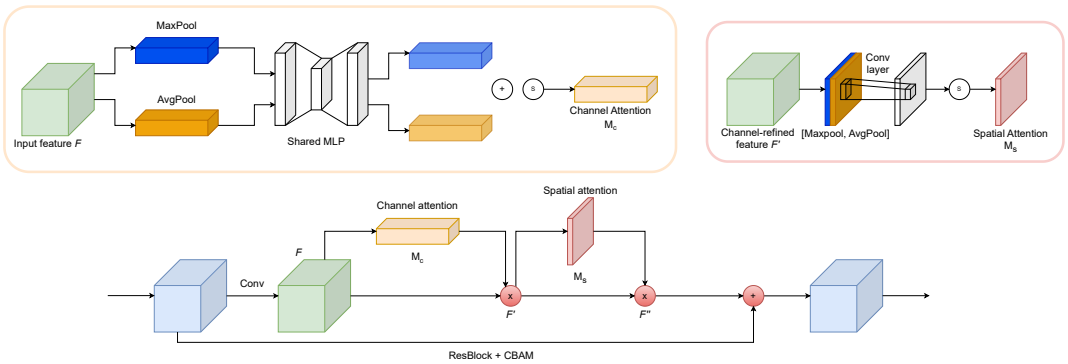

Fig. 5: Convolutional Block Attention Module. CBAM consists of Channel Attention and Spatial Attention sequentially.

TABLE I: Detection Result Comparisons on VisDrone Dataset

Model	AP	AP50	Backbone
retinaplus [15]	20.57	40.57	ResNeXt-101
ERCNNs [16]	20.45	41.2	ResNeXt-101
SAMFR-Cascade RCNN [17]	20.18	40.03	SERexNeXt-50
Cascade R-CNN++ [17]	18.33	33.5	SERexNeXt-50
EnDet	17.81	37.27	ResNet101-fpn
DCRCNN [18]	17.79	42.03	ResNeXt-101
Cascade R-CNN+ [17]	17.67	34.89	ResNeXt-101
ODAC	17.42	40.55	VGG
DA-RetianNet [19]	17.05	35.93	ResNet101
MOD-RETINANET [15]	16.96	33.77	ResNet50
DBCL [20]	16.78	31.08	Hourglass-104
ConstraintNet [21]	16.09	30.72	Hourglass-104
CornetNet* [22]	17.41	34.12	Hourglass-104
Light-RCNN* [23]	16.53	32.78	ResNet101
FPN* [24]	16.51	32.2	ResNet50
Cascade R-CNN* [25]	16.09	31.91	ResNeXt-101
DetNet59* [26]	15.26	29.23	ResNet50
RefineDet* [27]	14.9	28.76	ResNet101
RetinaNet* [15]	11.81	21.37	ResNet101
YOLOv5s	**20.1**	**35.7**	**Improved CSPDarknet53**
Proposed Method	**22.2**	**38.6**	**Improved CSPDarknet53**

TABLE II: Proposed Method Result

Model	# parameter	GFLOPs	AP
YOLOv5	7,046,599	15.9	20.1
YOLOv5s 4det w Coord & CBAM	6,816,684	31.5	20.5
YOLOv5s 4det w ERB & Coord & CBAM	6,641,644	31.0	20.1
YOLOv5s 4det w ERB & ERMP & Coord & CBAM	6,510,572	30.9	20.6
YOLOv5s 3det w Coord & CBAM	6,745,629	14.5	20.0
YOLOv5s 3det w ERB & Coord & CBAM	6,570,589	14.1	19.9
YOLOv5s 3det w ERB & EMRP & Coord & CBAM	6,439,517	**14.0**	19.3
YOLOv5s 3det w ERB & EMRP & Coord & CBAM wo C5	**2,147,589**	16.7	**22.2**

increase in parameters is prevented by reducing the number of iterations of ERB performed in Backbone to a minimum. As a result, the number of parameters is reduced through ERB and EMRP, and performance is improved through Coordinate Attention and CBAM. Compared to the original YOLOv5s, the number of parameters is reduced by about 70%, and the performance is increased by about 10% to 22.2mAP.

V. Conclusion

In this paper, we propose a YOLOv5 network using Combination of Coordinate Attention and CBAM that enables real-time object detection and shows higher performance. The proposed network improved the C3 layer to ERB and the SPPF layer to EMRP to reduce the number of parameters. Then, the C5 block in Backbone is removed, and a feature map is created to detect tiny, small, and medium-sized objects. To improve the performance of object detection, Coordinate Attention is added before the EMRP layer, and CBAM is applied before the final detector. The network is trained on the VisDrone dataset. The mAP value is 22.2mAP, about 10% higher than the original YOLOv5, and the number of parameters is about 70% less, 2,147,589.

In future works, we plan to further improve the EMRP to increase the object detection rate. In order to lighten the network through EMRP, the number of Max Pooling layers is reduced to two, but instead of the Max Pooling layer, a convolution layer is added to extract the features of the object more effectively. As the number of layers in the network increases, the number of parameters required for calculation will increase, but since it has about 70% less number of parameters than the original YOLOv5, it is expected that the object detection rate can be increased by adding layers.

VI. Acknowledgment

This result was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE)(2021RIS-003)

References

[1] O. Maghazei, T. H. Netland, D. Frauenberger, and T. Thalmann, "Automatic drones for factory inspection: The role of virtual simulation," in *Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems: IFIP WG 5.7 International Conference, APMS 2021, Nantes, France, September 5–9, 2021, Proceedings, Part IV*. Springer, 2021, pp. 457–464.

[2] J. Shahmoradi, E. Talebi, P. Roghanchi, and M. Hassanalian, "A comprehensive review of applications of drone technology in the mining industry," *Drones*, vol. 4, no. 3, p. 34, 2020.

[3] Y. Lee, Q. Tang, J. Choi, and K. Jo, "Low computational vehicle re-identification for unlabeled drone flight images," in *IECON 2022–48th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2022, pp. 1–6.

[4] J. Jiménez López and M. Mulero-Pázmány, "Drones for conservation in protected areas: present and future," *Drones*, vol. 3, no. 1, p. 10, 2019.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[6] G. Jocher, A. Stoken, and J. Borovec, "ultralytics/yolov5: v3.0." [Online]. Available: https://doi.org/10.5281/zenodo.3983579

Fig. 6: Object Detection Result on VisDrone 2019 dataset.

[7] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.

[8] J. Zhang, P. Wang, Z. Zhao, and F. Su, "Pruned-yolo: Learning efficient object detector using model pruning," in *International Conference on Artificial Neural Networks*. Springer, 2021, pp. 34–45.

[9] X. Wang, W. Li, W. Guo, and K. Cao, "Spb-yolo: An efficient real-time detector for unmanned aerial vehicle images," in *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2021, pp. 099–104.

[10] W. Zhan, C. Sun, M. Wang, J. She, Y. Zhang, Z. Zhang, and Y. Sun, "An improved yolov5 real-time detection method for small objects captured by uav," *Soft Computing*, vol. 26, pp. 361–373, 2022.

[11] M. Kim, J. Jeong, and S. Kim, "Ecap-yolo: Efficient channel attention pyramid yolo for small object detection in aerial image," *Remote Sensing*, vol. 13, no. 23, p. 4851, 2021.

[12] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. IEEE, 2021, pp. 2778–2788.

[13] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 13 708–13 717.

[14] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[15] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *CoRR*, vol. abs/1708.02002, 2017. [Online]. Available: http://arxiv.org/abs/1708.02002

[16] N. Xie, S. Li, and J. Zhao, "Ercnn: Enhanced recurrent convolutional neural networks for learning sentence similarity," in *Chinese Computational Linguistics*, M. Sun, X. Huang, H. Ji, Z. Liu, and Y. Liu, Eds. Cham: Springer International Publishing, 2019, pp. 119–130.

[17] Z. Cai and N. Vasconcelos, "Cascade r-cnn: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1483–1498, 2021.

[18] S. Chakraborty, S. Aich, A. Kumar, S. Sarkar, J.-S. Sim, and H.-C. Kim, "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)*, 2020, pp. 197–202.

[19] G. Pasqualino, A. Furnari, G. Signorello, and G. M. Farinella, "An unsupervised domain adaptation scheme for single-stage artwork recognition in cultural sites," *Image and Vision Computing*, p. 104098, 2021.

[20] Y. Wu, Z. Cheng, Z. Xu, and W. Wang, "Segmentation is all you need," *CoRR*, vol. abs/1904.13300, 2019. [Online]. Available: http://arxiv.org/abs/1904.13300

[21] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *CoRR*, vol. abs/1904.07850, 2019. [Online]. Available: http://arxiv.org/abs/1904.07850

[22] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," *CoRR*, vol. abs/1808.01244, 2018. [Online]. Available: http://arxiv.org/abs/1808.01244

[23] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Light-head R-CNN: in defense of two-stage object detector," *CoRR*, vol. abs/1711.07264, 2017. [Online]. Available: http://arxiv.org/abs/1711.07264

[24] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," *CoRR*, vol. abs/1612.03144, 2016. [Online]. Available: http://arxiv.org/abs/1612.03144

[25] Z. Cai and N. Vasconcelos, "Cascade R-CNN: delving into high quality object detection," *CoRR*, vol. abs/1712.00726, 2017. [Online]. Available: http://arxiv.org/abs/1712.00726

[26] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Detnet: A backbone network for object detection," *CoRR*, vol. abs/1804.06215, 2018. [Online]. Available: http://arxiv.org/abs/1804.06215

[27] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," *CoRR*, vol. abs/1711.06897, 2017. [Online]. Available: http://arxiv.org/abs/1711.06897