Robust detection of small and dense objects in images from autonomous aerial vehicles

Joo Chan Lee,1 JeongYeop Yoo,2 Youngwoo Kim,3 SungTae Moon,4 and Jong Hwan Ko5

1Department of Artificial Intelligence, Sungkyunkwan University, Suwon, South Korea
2Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon, South Korea
3Department of System Semiconductor Engineering, Sangmyung University, Cheonan, South Korea
4Artificial Intelligence Research Division, Korea Aerospace Research Institute, Daejeon, South Korea
5College of Information and Communication Engineering, Sungkyunkwan University, Suwon, South Korea

Aerial images obtained from autonomous aerial vehicles have lots of small and densely distributed objects because of the capture distance. This paper proposes a deep neural network architecture and training/inference techniques for robust detection of objects in the aerial images. Based on cascade R-CNN, the proposed model adopts the recursive feature pyramid and switchable atrous convolution for robust detection of dense objects. A patch-level division and multi-scale inference techniques are applied to effectively detect small objects. The results show that the proposed approach achieves the highest performance on the VisDrone test-dev dataset, in the official ECCV VisDrone2020-DET challenge.

Introduction: Deployment of autonomous aerial vehicles such as drones and flying robots is getting more prevalent with their increasing applications including remote surveillance and object search. One of the essential capabilities in these applications is robust object detection in the images captured by aerial vehicles. While considerable research work has been done in the area of deep neural network-based object detection, most of the techniques are targeted for general datasets such as MS COCO [1] and Pascal VOC [2]. Meanwhile, detection performance on the aerial images is hindered by multiple challenges regarding the size and distribution of objects in an image. As the aerial images are taken from a very long distance, objects are captured in relatively small area in an image, and many objects are densely distributed and overlapped [3].

To address these challenges, this work presents a deep neural network architecture and its training/testing techniques for effective and robust detection of small and dense objects in aerial images. The proposed model structure is based on cascade R-CNN [4] and recursive feature pyramid (RFP) [5] according to the looking and thinking twice mechanism, and switchable atrous convolution (SAC) [5] is utilised to deal with high object density. To effectively extract features of small objects, we divide images into multiple patches during the training phase. When testing, soft-NMS (non-maximum suppression) [6] is applied to handle densely distributed objects, and multi-scale testing is performed to leverage the effect of ensemble. Using the proposed model and techniques, we won the first place in the ECCV VisDrone2020-DET challenge [7].

Related work: Object detection: The anchor-based object detection models can be divided into one stage detectors such as YOLO [8–10] and SSD [11], and two stage detectors using region proposal networks (RPNs) such as Faster R-CNN [12]. In general, two stage models are more complex, but they perform better. In the two-stage models, PANet [13], BiFPN [14] were proposed to reinforce multi-scale features by modifying FPN [15], and Nas-FPN [16] and Auto-FPN [17] were proposed to find the optimal structure. In addition, RFP [5] was proposed to allow recursive features to be identified by adding a feedback connection. DCN [18, 19], GCNet [20] and SAC [5] have been proposed to improve the convolution method in backbone to extract robust features. Some recent studies have proposed methods that do not use anchor such as CornerNet [21] and CenterNet [22].

Object detection on aerial images: Several studies have been proposed to detect objects in aerial images, considering the characteristics of objects captured in distance. Similar to the traditional cut and paste augmentation technique, one prior work was proposed to solve the class imbalance problem [24]. Another study proposed RRNet [25] to improve a augmentation method. There is also a study that detects objects in cluster units for efficient detection of dense objects [3].

Analysis of aerial images: VisDrone dataset: In this work, we utilise the VisDrone dataset (Figure 1), the most widely-used aerial image dataset for object detection. The VisDrone dataset consists of 288 video clips with 261,908 frames, captured by various drone-mounted cameras. To design a model and techniques optimised for this dataset, we analyse the object size and the number of objects in a frame in the dataset.

Average object size: We analysed the object size for each of 10 classes included in the VisDrone dataset. First of all, we resized all of the training images to the same size, 1360×765. To calculate the object size, we multiplied the width and height of all resized bounding boxes included in the training set. These values were averaged by classes and total. As shown in Table 1, the entire object occupies 1597 pixels on average. On the other hand, the average object size of the COCO dataset is 80,900, which is about 50.7× compared to VisDrone dataset. Also, the bus class with the largest average size is 5297.0, which is about 14.8× larger than 389.6 of the smallest class (person). This observation implies that we need to focus on detecting small objects, as well as consider the variance in the object size.

Average objects per image: Table 2 shows the average number of objects per image in VisDrone, COCO and Pascal VOC datasets. The VisDrone dataset has 53.03 objects on average, which is 7.3× and 22.3× more objects per image than COCO and Pascal VOC, respectively. It implies that the VisDrone dataset has much denser distribution than other datasets. Therefore, it is crucial to effectively deal with the scene that objects are densely distributed for the better performance.

Proposed method: Model architecture: The structure of the proposed model is based on cascade R-CNN [4], as shown in Figure 2. Region proposals are obtained from the RPN, where the backbone and neck extract the feature maps. Using the proposals, detection results are obtained through three stages of bounding box regression and classification operations. The detection performance is enhanced by gradually increasing the IOU threshold through these three stages. In this work, ResNet-50 was used as the backbone, SAC [5] was applied to the conv3–conv5 layer, and RFP [5] was utilised to the neck.

Table 1. Average object size of given classes and total

| Object | VisDrone | COCO 2017 | Pascal VOC 2012 |
|--------|----------|-----------|----------------|
| Pedestrian | 489.0 | 389.6 | 2129.5 |
| Person | 4437.8 | 1724.2 | 5297.0 |
| Bicycle | 765.0 | 118287 | 3507 |
| Car | 860001 | 7.27 | 2.38 |
| Van | 53.03 | 1464 | 631.9 |

Table 2. Average number of objects per one image in multiple datasets

| Dataset | VisDrone train | COCO 2017 | Pascal VOC 2012 |
|---------|---------------|-----------|----------------|
| # images | 6471 | 118287 | 1464 |
| # objects | 343205 | 860001 | 3507 |
| # Obj/ img | 53.03 | 7.27 | 2.38 |
The SAC [5] structure can be divided into three parts. The first part mainly performs the SAC operation, and the rest two parts insert the global context before and after convolution. In the main part, a switch function is used to fuse convolution results with different atrous rates. In the global context part, the input feature is compressed and passed to the next, similar to SENet [26]. In the DetectorsRS [5], it was proved that the atrous rate tends to be proportional to the size of the objects being trained. Therefore, we can conclude that using SAC is beneficial in aerial images having large variance in the object size.

RFP [5] enables looking and thinking twice by adding a feedback connection to FPN. This means that more robust features can be obtained by constructing a recursive, multiple feature pyramid.

Patch-level division of training set images: One of the solutions for robust detection of small objects is to increase the image resolution of the training dataset. As the input size increases, the detection performance improves as the features of the object can be properly obtained. However, the image size cannot be increased indefinitely since the increase in the image size also increases the required GPU resources for training.

In order to increase effective resolution of objects without affecting the required resources, we propose dividing training set images into some patches. If the ground-truth bounding box is divided into patches, we include its label only in the patch that has most of its bounding box area.

Testing-time techniques: We also apply several testing-time techniques for robust detection of small and dense objects. In general, if two bounding boxes of the same class are largely overlapped, they are merged into one bounding box (generally with the intersection over union (IOU) threshold is 0.5). This merge operation can significantly degrade the detection performance of aerial datasets, as these datasets have a large number of same-class objects densely distributed and overlapped in an image. To resolve this issue, we applied soft-NMS [6] to keep the boxes of the same class overlapped. Accordingly, we increase the number of maximum detections to 500 to cope with soft-NMS that can generate more bounding boxes.

In addition, we apply the multi-scale testing method for higher detection accuracy. We use diverse scales with same aspect ratio since it provides better performance than changing both the scale and aspect ratio.

Experimental results: Experimental setup: For experiments, we use the VisDrone-DET test-dev dataset and toolkit [7], which evaluate the detection results by APIoU=0.50, 0.65, 0.75, AP45°=0.50, AP60°=0.75, and AP90°=0.95 metrics. We use only the VisDrone-DET train [7] dataset for training, and horizontal flipping for data augmentation. We implement our models on the MM Detection [27] with Pytorch. All models are trained for 12 epochs by an SGD optimiser with the momentum of 0.9 and the weight decay of 0.0001. The initial learning rate is set to 0.02 and divided by 10 at epoch 9 and 12.

Effects of training set resolution: Table 3 shows the performance improvement with the resolution increase and patch-level division of training set images. As the input resolution increases from $1360 \times 765$ to $2405 \times 1608$, the test mAP performance gradually improves from 23.85% to 29.98%. Compared to $2405 \times 1608$ images without division, $2 \times 2$ patch-level division into $1600 \times 1050$ provides an 1.32% of mAP increase, as the images are actually enlarged to $3200 \times 2100$.

Effects of testing-time techniques: Table 4 shows mAP performance of the model according to the soft-nms application and max detection size. The table shows that applying soft-NMS and increasing max detection helps improving the detection performance. When the both methods are applied, the mAP performance increases by 1.6%.

Based on the experiments with diverse scales, images of the seven different scales and its horizontal flip images were used for the experiment. Table 5 shows that, as the number of scales used for inference increases, the detection performance enhances. The table indicates that the best performance was obtained with three scales ($3400 \times 2231, 3600 \times 2362, 3800 \times 2494$) with the aspect ratio same as the original dataset.

Final model: The final model was constructed based on the results of the experiments shown in the previous section. At the model level, cascade structure was used with SAC applied in conv3–conv5 and RFP

### Table 3. Performance Evaluation on VisDrone test-dev with various input size and patch-level division

| Dataset | Train size | Test size | Test mAP |
|---------|------------|-----------|----------|
| $1360 \times 765$ | $1360 \times 765$ | 23.85% |
| $1920 \times 1080$ | $1920 \times 1080$ | 28.48% |
| $2080 \times 1170$ | $2240 \times 1260$ | 29.36% |
| $2300 \times 1510$ | $2300 \times 1510$ | 29.52% |
| $2450 \times 1608$ | $2450 \times 1608$ | 29.98% |

### Table 4. Performance evaluation on VisDrone test-dev with applying soft-NMS and changing max detection

| Soft-NMS | Max detection | Train size | Test size | Test mAP |
|----------|---------------|------------|-----------|----------|
| X | 100 | $1600 \times 1050$ | $3600 \times 2362$ | 31.47% |
| O | 100 | $1600 \times 1050$ | $3600 \times 2362$ | 31.63% |
| X | 500 | $1600 \times 1050$ | $3600 \times 2362$ | 32.65% |
| O | 500 | $1600 \times 1050$ | $3600 \times 2362$ | 33.23% |

### Table 5. Performance evaluation on VisDrone test-dev with applying various multi-scale testings

| Train size | Multi-scale testing | Test mAP |
|------------|---------------------|----------|
| $3600 \times 2362$ | 35.6% |
| $2400 \times 1575$ | 35.54% |
| $3600 \times 2362$ | 35.68% |
| $3600 \times 2700$ | 35.74% |
| $3600 \times 2025$ | 36.07% |
| $3600 \times 2362$ | 36.07% |
| $4000 \times 2625$ | 36.24% |
| $3600 \times 2025$ | 36.33% |
| $3600 \times 2700$ | 36.33% |
| $3600 \times 2025$ | 36.07% |
| $3600 \times 2362$ | 36.07% |
| $4000 \times 2625$ | 36.24% |
| $3600 \times 2025$ | 36.33% |
| $3600 \times 2700$ | 36.33% |
| $3600 \times 2025$ | 36.07% |
| $3600 \times 2362$ | 36.07% |
| $4000 \times 2625$ | 36.24% |
| $3600 \times 2025$ | 36.33% |
| $3600 \times 2700$ | 36.33% |
| $3600 \times 2025$ | 36.07% |
as the neck. We use the patch-level division technique, max detection as 500, and soft-nms. For multi-scale testing, we use horizontal flipped images with seven scales as the width resolutions [3000, 3200, 3400, 3600, 3800, 4000, 4200]. As shown in Table 6, we took the first place in the VisDrone-DET 2020 challenge with 34.57% mAP(37.33% on test-dev).

Conclusions: In order to perform efficient detection from aerial images, we conducted dataset analysis and applied various techniques accordingly. Aerial images have many small objects densely distributed, and the size difference between the classes is large. To efficiently detect small and dense objects, patch-level division is performed on the dataset, and RFP and SAC are applied to the baseline cascade structure. When testing, softmax and multi-scale inference techniques are applied to effectively process dense objects. We evaluated the performance of each technique through several experiments and finally obtained the best performance in the VisDrone-DET2020 challenge.

Acknowledgement: This work was partly supported by the KARI Institutional Program (No.1711117560, Research on Aerospace Core Technology based on Artificial Intelligence and Metal 3D Printing Technology), NRF grant (No.2019R1A1A048115, Edge-Host Collaborative Deep Learning for Hyper-Connected Internet-of-Things Environment), and IITP grant (No.IITP-2019-0-00421, Artificial Intelligence Graduate School Program for Sungkyunkwan University) funded by the Ministry of Science and ICT (MSIT) of the Korea government.

© 2021 The Authors. Electronics Letters published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Received: 2 March 2021 Accepted: 16 May 2021 doi: 10.1049/el.12245

References

1 Lin, T.Y., et al.: Microsoft COCO: Common objects in context. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 740–755. Springer, Berlin, Heidelberg (2014)
2 Everingham, M., et al.: The pascal visual object classes challenge: A retrospective. Int. J. Comput. Vis. 111(1), 98–136 (2015)
3 Yang, F., et al.: Clustered object detection in aerial images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 9311–9320. IEEE, Piscataway, NJ (2019)
4 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 (2019)
5 Qiao, S., Chen, L.C., Yuille, A.: Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. arXiv:200602334, (2020)
6 Bodla, N., et al.: Soft-NMS – improving object detection with one line of code. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 5562–5570. IEEE, Piscataway, NJ (2017)
7 Zhu, P., et al.: Vision meets drones: Past, present and future. arXiv:200106303, (2020)
8 Redmon, J., et al.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788. IEEE, Piscataway, NJ (2016)
9 Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525. IEEE, Piscataway, NJ (2017)
10 Redmon, J., Farhadi, A.: YOLOv3: An incremental improvement. arXiv:180402767, (2018)
11 Liu, W., et al.: SSD: Single shot multibox detector. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 1–17. Springer, Berlin, Heidelberg (2016)
12 Ren, S., et al.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99. IEEE, Piscataway, NJ (2015)
13 Liu, S., et al.: Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 936–944. IEEE, Piscataway, NJ (2017)
14 Tan, M., Pang, R., Le, Q.V.: EfficientDet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8759–8768. IEEE, Piscataway, NJ (2018)
15 Ghaissi, G., et al.: NAS-FPN: Learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7029–7038. IEEE, Piscataway, NJ (2019)
16 Xiao, H., et al.: Auto-FPN: Automatic network architecture adaptation for object detection beyond classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6648–6657. IEEE, Piscataway, NJ (2019)
17 Dai, J., et al.: Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 764–773. IEEE, Piscataway, NJ (2017)
18 Zhu, X., et al.: Deformable ConvNets V2: More deformable, better results. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9300–9308. IEEE, Piscataway, NJ (2019)
19 Cao, Y., et al.: GCNet: Non-local networks meet squeeze-excitation networks and beyond. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 1971–1980. IEEE, Piscataway, NJ (2019)
20 Law, H., Deng, J.: CornerNet: Detecting objects as paired keypoints. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 1–17. Springer, Berlin, Heidelberg (2018)
21 Duan, K., et al.: CenterNet: Keypoint triplets for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6568–6577. IEEE, Piscataway, NJ (2019)
22 Dwibedi, D., Misra, I., Hebert, M.: Cut, paste and learn: Surprisingly easy synthesis for instance detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1301–1310. IEEE, Piscataway, NJ (2017)
23 Hong, S., Kang, S., Cho, D.: Patch-level augmentation for object detection in aerial images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 1–8. IEEE, Piscataway, NJ (2019)
24 Shan, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7132–7141. IEEE, Piscataway, NJ (2018)
25 Chen, K., et al.: Mmdetection: Open MmLab detection toolbox and benchmark. arXiv:190607155, (2019)