UDC 004.89

SMALL OBJECT DETECTION IN REMOTE SENSING IMAGES BY EFFICIENTNET-MBCONV AND YOLOV8

Wang HAO, Sergey ABLAMEYKO
(Belarusian State University, Minsk)

Abstract. *Small object detection is a very popular research field in computer vision. In this paper, we use the current good performance YOLOv8 and EfficientNet-MBConv to analyze the detection of small objects and compare them. The results show that EfficientNet-MBConv using YOLOv8 has better precision and box(P) than using other methods.*

Keywords: *Object detection, Small object detection, Multi-Scale Convolution, EfficientNet.*

Introduction. Object detection serves as a vital research area within computer vision, laying the foundation for more intricate visual tasks. It acts as a cornerstone for understanding images and facilitates diverse vision pursuits like segmentation, scene comprehension, object tracking, image description, and event identification.

Tackling small object detection has historically posed a significant challenge in this domain. Small objects are characterized by their diminutive image sizes, needing special attention during detection. The definition of small objects can vary based on absolute or relative size criteria. In the COCO [1] dataset, for example, objects sized below 32 × 32 pixels qualify as small objects in terms of absolute size. On the other hand, relative size criteria, as per the International Society of Optical Engineering, consider an object small if it occupies less than 0.12% of a 256 × 256 pixels image, indicating a minuscule imaging area. When compared to regular-sized objects, small objects tend to occupy a minor portion of the image, featuring lower resolution and less pronounced visual characteristics. The reduced pixel coverage of small objects underscores their challenge, as they possess limited detail for traditional object detection algorithms to effectively detect and delineate.

Object detection methods based on deep learning can be roughly divided into the following three categories, two-stage object detection algorithms RCNN, Faster RCNN, FPN, single-stage object detection algorithm YOLO, SSD, and Transformer-based object detection algorithm DETR. One of the core components of the above algorithm is the convolution layer, and the overall structure of the convolutional neural network includes the following parts: input layer, convolution layer, pooling layer, fully connected layer and output layer. Among them, the function of the convolution layer is to perform a convolution operation on the input image or feature map and output the convolved feature map. The number and size of convolution kernels can be set freely.

The core idea of the convolutional neural network is to extract local features of the input image through convolution operations, and use these features for the next step of processing and classification.

Mingxing Tan et al. [2] proposed a new scaling method that uses neural architecture search to design a new baseline network and scale it up to obtain a family of models called EfficientNets. Mark Sandler et al. [3] described a new mobile architecture, MobileNetV2, that improves the state of the art performance of mobile models on multiple tasks and benchmarks as well as across a spectrum of different model sizes. And inspired by the MobileNet V2 architecture, MBConv (Mobile Inverted Residual Bottleneck Convolution) enhances feature learning with increased efficiency. In this paper, we consider in detail combining the EfficientNet-MBConv with YOLOv8 [4], compare it with other methods and test it. The results show that EfficientNet-MBConv using YOLOv8 has better result.

Methodology. EfficientNet incorporates principles such as efficient scaling, employing a compound scaling method for uniform adjustment of model dimensions, utilizing efficient building blocks like MBConv and depth-wise separable convolutions for enhanced feature representation while minimizing computational complexity, integrating the Swish activation function to boost feature learning capabilities, and offering multiple model variants (B0 to B7) with varying depths and widths to accommodate diverse computational needs and accuracy requirements, ensuring flexibility in model selection.

The structural configuration of the EfficientNet-B0 baseline network is shown in the table 1.

Table 1. – EfficientNet-B0 baseline network – Each row describes a stage *i* with $\hat{L}_i$ layers, with input resolution $\langle \hat{H}_i, \hat{W}_i \rangle$ and output channels $\hat{C}_i$

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels $\hat{C}_i$	#Layers $\hat{L}_i$
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	14×14	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 & Pooling & FC	7×7	1280	1

There are a total of 9 stages in B0. The convolutional layers in the table are followed by BN and Swish activation functions by default. Stage 1 is a 3×3 convolutional layer. For stage 2 to stage 8, MBConv is stacked repeatedly.

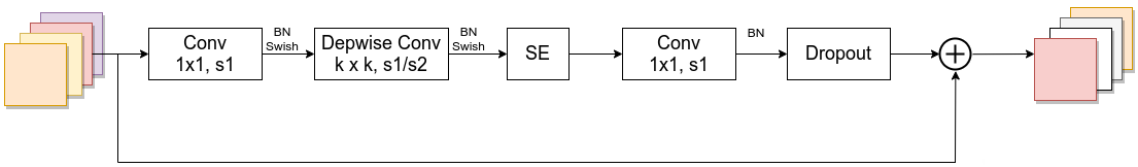


Figure 1. – Structure of Mobile Inverted Residual Bottleneck Convolution module

There are five operations in MBConv module:

(1) Use a 1×1 convolution to increase the dimension, and its output channel is n times the input channel.

(2) Follow by a DW convolution.

(3) Use an attention mechanism to adjust the feature matrix through a SE module.

(4) Perform dimensionality reduction through 1×1 convolution.

(5) Follow a dropout layer.

Results and discussion. In experiments we use DOTA-v2.0 [5] as our dataset, which is a large-scale dataset for object detection in aerial images. It can be used to develop and evaluate object detectors in aerial images. DOTA-v2.0 collects more Google Earth, GF-2 Satellite, and aerial images. There are 16 common categories (planes, helicopters, ships, vehicles, swimming pools, etc.).

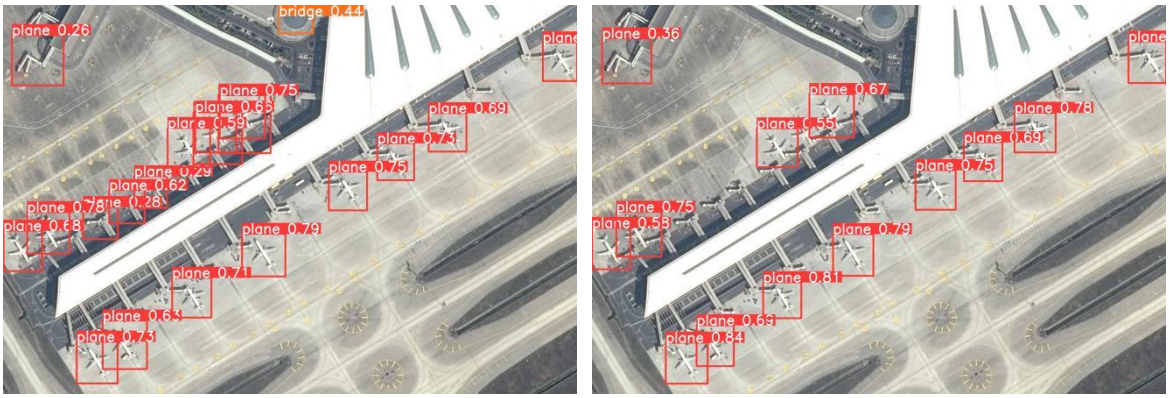
The experiment was completed on the AutoDL platform using Nvidia GeForce RTX 3090 24268MiB. The image size used for training is 640, set batchsize to 4, train for 100 epochs. The experimental results are shown in the table 2.

Table 2. – Comparison results

Detector	precision	box(P)	recall	parameters	mAP50	mAP50-95
YOLOv8n	0.59761	0.567	0.27641	3008768	0.30482	0.17714
YOLOv8_EMBC	0.61641	0.622	0.2522	3352976	0.27447	0.15919
YOLOv8_EMSC	0.57541	0.559	0.27268	2720256	0.29955	0.17388
YOLOv8_EMSCP	0.52958	0.534	0.28195	2874112	0.30034	0.17516
YOLOv5n [6]	0.52834	0.528	0.27819	2506064	0.29409	0.16986
YOLOv5_EMBC	0.58955	0.589	0.24519	3330464	0.25664	0.14662
YOLOv5_EMSC	0.59309	0.604	0.27113	2217552	0.28375	0.16694
YOLOv5_EMSCP	0.50831	0.593	0.26244	2371408	0.28423	0.16529

Experiments show that when using EfficientNet-MBConv for object detection in YOLOv8, the precision and box(P) indicators are the best. In other words, it enhances

the model's ability to accurately locate objects and displays clearer class boundaries, thereby helping to improve the accuracy of object classification.



a *b*

***a* – The result of YOLOv8n; *b* – The result of YOLOv8_EMBC**

Figure 2. – Example of results

In addition, the poor performance of this method in mAP and recall indicators may be because the new method may bring challenges to accurately locate objects and have a negative impact on the above indicators. It is also possible that the new method resulted in higher false negatives, thus lowering the recall rate. These issues will be what we need to explore in next step.

REFERENCES

1. T.-Y. Lin et al., “Microsoft COCO: Common Objects in Context,” in European Conference on Computer Vision, 2014. [Online]. – Available: https://api.semanticscholar.org/CorpusID:14113767.
2. M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” CoRR, vol. abs/1905.11946, 2019, [Online]. – Available: http://arxiv.org/abs/1905.11946.
3. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. -C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018. – P. 4510–4520. – DOI: 10.1109/CVPR.2018.00474.
4. G. Jocher, A. Chaurasia, and J. Qiu, “Ultralytics YOLOv8.” 2023. [Online]. – Available: https://github.com/ultralytics/ultralytics.
5. J. Ding et al., “Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. – P. 1–1. – DOI: 10.1109/TPAMI.2021.3117983.
6. J. Redmon, S. Divvala, R. B. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. null, 2015. – P. 779–788. – DOI: 10.1109/CVPR.2016.91.