SOLUTION TO ICCV 2021 VIPRIORS OBJECT DETECTION CHALLENGE

TECHNICAL REPORT

Linfeng Luo, Yanhong Liu, Fengming Cao
Pingan International Smart City, China
`{luolinfeng619,liuyanhong648, caofengming777}@pasmart.com.cn`

September 28, 2021

ABSTRACT

In this report, we present our solution to ICCV 2021 VIPriors object detection challenge. Our solution is based on the open source object detection toolbox MMDetection, we firstly construct a baseline model using the two-stage detector Cascade R-CNN, coupled with DCN, GCNet and data augmentation method Albu. Secondly, we create a synthetic-10K dataset from the training images using cropping-mosaics. We obtain a pre-trained backbone using the unsupervised contrastive learning technique SimSiam. Furthermore, we apply the self-training policy on the synthetic dataset. We also adopt some test tricks.

Keywords Object Detection · Unsupervised Pre-training · Small Dataset

1 Introduction

The 2021 VIPriors Object Detection Challenge was introduced as a workshop on ICCV conference. The models are required to be trained from scratch without using pre-training or transfer learning. The main objective of the challenge is to detect bike parts for a small DelftBikes dataset [Kayhan et al., 2021]. DelftBikes contains 10,000 bike images with 22 densely annotated parts for each bike. All part locations are annotated, with part states denoted by missing, intact, damaged, or occluded. The evaluation is done on available parts, namely intact, damaged and occluded parts. For this challenge, 8000 of 10000 images are selected for training dataset with annotations. The rest of 2000 images are used for test dataset.

It is rather challenging considering the data-deficient setting, occluded bike parts and noisy annotations. We figure out our solution by conducting extensive experimental comparisons and adopting modern state-of-the art strategies. We firstly construct a baseline model using the two-stage detector Cascade R-CNN, coupled with DCN, GCNet and data augmentation method Albu. Secondly, we create a synthetic-10K dataset by cutting the images from the training set into a top-half image and a bottom-half one, and then randomly joining a top-half one and a bottom-half one to make a new image. We pre-train the backbone network using the unsupervised learning technique Simsiam [Chen and He, 2020]. Furthermore, we train the model on the synthetic dataset with self-training policy. We also adopt some test tricks such as soft-NMS [Bodla et al., 2017].

2 Methods and Results

In this section, we provide an overview of our solution to the challenge and present the experimental results.

2.1 Implementation Details

We carried out all the experiments using the open source object detection toolbox MMDetection 2.0 [Chen et al., 2019] developed by PyTorch. We trained the models on 8 GeForce RTX 2080 Ti GPUs with 11GB memory for each. The

Table 1: Results on DelftBikes validation sub-dataset

Methods	Epochs	Init. learning rate	AP@0.50:0.95
Cascade R-CNN with R50-FPN	12	0.1	25.9
+ DCN	40	0.1	26.5
+ GCNet	40	0.1	27.3
+ Multi-Scale Train/Test, Albu	12	0.1	27.7
+ Pre-trained with SimSiam	24	0.1	28.5
+ Test Trick	24	0.1	30.0
+ Self-training Policy	24	0.2	30.3
+ val sub-dataset	24	0.2	31.5

stochastic gradient optimizer SGD is used with the momentum to be 0.9 and the weight decay to be 0.0001. During the experiments, we adjust the initial learning rate and the total epochs for the best possible performance. The batch size is set to 16.

We split the original training set into a training subset including 6400 images, and a validation subset containing the rest of 1600 images. All the models are trained on the split training subset and evaluated on the validation subset, until the final result was submitted. The experimental results are shown in Table 1.

2.2 Baseline Results

For the baseline system, we select the two-stage detector Cascade R-CNN [Cai and Vasconcelos, 2019] and Feature Pyramid Network (FPN) [Lin et al., 2017]. The moderate backbone network ResNet-50 is used, considering about the lack of data. We move further and add deformable convolution networks (DCN) [Dai et al., 2017] and the global context modeling (GCNet)[Cao et al., 2019]. With the single image scale of (1333,800), it improves the AP@0.50:0.95 from 25.9% to 27.3%.

Finally, we include multi-scale training and multi-scale testing. The images are resized over scales (960, 720), (1280, 960), (1600, 1200), (1920, 1440) for both training and testing. The input data is also augmented with Albu method. It achieves 27.7% AP@0.50:0.95.

2.3 Enhancements

At the second stage, we adopt the unsupervised pre-training technique of *SimSiam* based on contrastive learning [Chen and He, 2020] to enhance the feature extraction of the backbone network. For this purpose, we create a synthetic dataset containing 10K images from the originally provided training dataset with 8000 images, as stated below.

We cut each image from the original training set into two segments. One segment includes those parts on the top half of a bike, while the other one includes the parts on the bottom half of the bike. The horizontal coordinate for cutting is derived from the ground truth box annotations, so that each part is fully included in a segment. We randomly select two segments from the top and bottom sets respectively, which are then joined into an image of size 640x480. Figure 1 shows an example of how to make such a joined image.

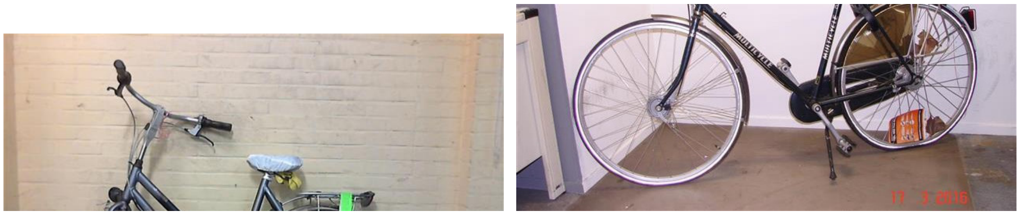
We conduct an unsupervised pre-training of the ResNet-50 network on this synthetic-10K dataset. Using this pre-trained backbone improves the AP from 27.7% to 28.5%.

At the last step, several tricks are applied during the inference phase. We adjust the number of proposals kept per-image before NMS, after NMS and after post-processing for RPN respectively. Soft-NMS is used for RCNN, with the maximum number of detected boxes set to 1000 and the score threshold set to 0.001. Flip is also enabled for the test phase. With these test tricks, we observe a great improvement of 1.5% at AP@0.50:0.95.

2.4 Final Results

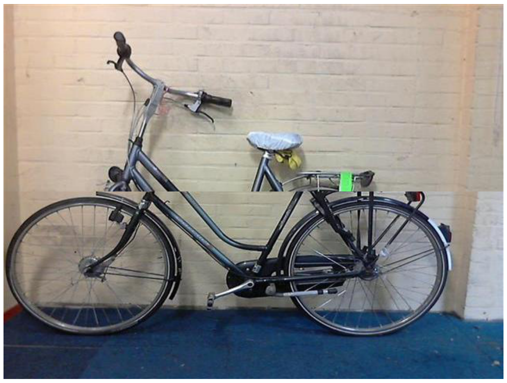
At the final stage, we use the enhanced model trained in the last subsection to label the synthetic-10K dataset created above. We then add the newly labeled 10K images into our training set. With this self-training policy, we achieve 30.3% AP on the validation sub dataset.

Furthermore, we add the validation sub dataset into the training set, i.e. we train on the originally train dataset and the labeled synthetic-10K dataset.



(a) The top half cut from one image

(b) The bottom half cut from the other image



(c) The synthetic image by joining the two halves

Figure 1: Illustration of how to make a synthetic image

The final submitted results on the test dataset achieve 30.11% AP@0.50:0.95, 63.9% AP@0.5, 23.9% AP@0.50, 11.1% AP$_S$, 27.3% AP$_M$, 23.3% AP$_L$.

References

Osman Semih Kayhan, Bart Vredebregt, and Jan C. van Gemert. Hallucination in object detection - A study in visual part verification. *CoRR*, abs/2106.02523, 2021. URL `https://arxiv.org/abs/2106.02523`.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. Soft-nms - improving object detection with one line of code. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5562–5570. IEEE Computer Society, 2017. doi:10.1109/ICCV.2017.593. URL `https://doi.org/10.1109/ICCV.2017.593`.

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1–1, 2019. ISSN 1939-3539. doi:10.1109/tpami.2019.2956516. URL `http://dx.doi.org/10.1109/tpami.2019.2956516`.

Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 936–944. IEEE Computer Society, 2017. doi:10.1109/CVPR.2017.106. URL `https://doi.org/10.1109/CVPR.2017.106`.

Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 2017.

Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. *arXiv preprint arXiv:1904.11492*, 2019.