Hit-Detector: Hierarchical Trinity Architecture Search for Object Detection Supplementary Material

Jianyuan Guo[1,2], Kai Han[2], Yunhe Wang[2], Chao Zhang[1]*, Zhaohui Yang[1]
Han Wu[1], Xinghao Chen[2], Chang Xu[3]
[1] Key Lab of Machine Perception (MOE), Dept. of Machine Intelligence, Peking University.
[2] Noah's Ark Lab, Huawei Technologies. [3] School of Computer Science, Faculty of Engineering, University of Sydney.
jyguo@pku.edu.cn, kai.han@huawei.com, chzhang@cis.pku.edu.cn

In this supplementary material, we list the search space and corresponding sub search spaces for detector trinity in details, and we show the qualitative results of our Hit-Detector compared with other state-of-the-art methods.

1. Search Space

The whole search space consists of $N = 32$ different operation candidates in our experimental setting. We list the candidates bellow:

- ir_k3_d1_e1
- ir_k3_d1_e3
- ir_k3_d1_e6
- ir_k3_d2_e1
- ir_k3_d2_e3
- ir_k3_d2_e6
- ir_k3_d3_e1
- ir_k3_d3_e3
- ir_k3_d3_e6
- ir_k5_d1_e1
- ir_k5_d1_e3
- ir_k5_d1_e6
- ir_k5_d2_e1
- ir_k5_d2_e3
- ir_k5_d2_e6
- ir_k5_d3_e1
- ir_k5_d3_e3
- ir_k5_d3_e6
- ir_k7_d1_e1
- ir_k7_d1_e6
- sep_k3_d1
- sep_k3_d2
- sep_k3_d3
- sep_k5_d1
- sep_k5_d2
- sep_k5_d3
- conv_k3_d1
- conv_k3_d2
- conv_k3_d3
- conv_k5_d1
- conv_k5_d2
- conv_k5_d3

where "ir", "sep" and "conv" indicates the inverted residual block, separable block and convolution block, respectively, "k" indicates the kernel size, "d" indicates the dilation rate, "e" indicates the expansion rate of inverted residual block.

2. Sub search space

In our experiments, we set $N_b = N_n = N_h = 8$, and we list the top-8 operation candidates for **backbone**:

- ir_k3_d1_e3
- ir_k3_d1_e6
- ir_k3_d2_e3
- ir_k5_d1_e3
- ir_k5_d1_e3
- ir_k5_d2_e6
- ir_k5_d3_e6
- ir_k7_d1_e6

*Corresponding author.

The top-8 operation candidates for **neck**:

- conv_k3_d3
- conv_k5_d1
- ir_k3_d2_e1
- ir_k5_d1_e3
- sep_k3_d1
- sep_k3_d3
- sep_k5_d2
- sep_k5_d3

The top-8 operation candidates for **head**:

- ir_k3_d1_e3
- ir_k3_d1_e6
- ir_k3_d2_e6
- ir_k5_d1_e3
- ir_k5_d1_e6
- ir_k7_d1_e6
- conv_k3_d1
- conv_k5_d1

3. Qualitative results

We show the qualitative results of our Hit-Detector. We randomly sample some images from the COCO minival and show detection results with confidence bigger than 0.5. First column is the results of FPN [3], second column is the results of NAS-FPN [2] implemented by ourselves, third column is the results of DetNAS [1], and the fourth column is the results of our Hit-Detector. Different box colors indicate different object categories.

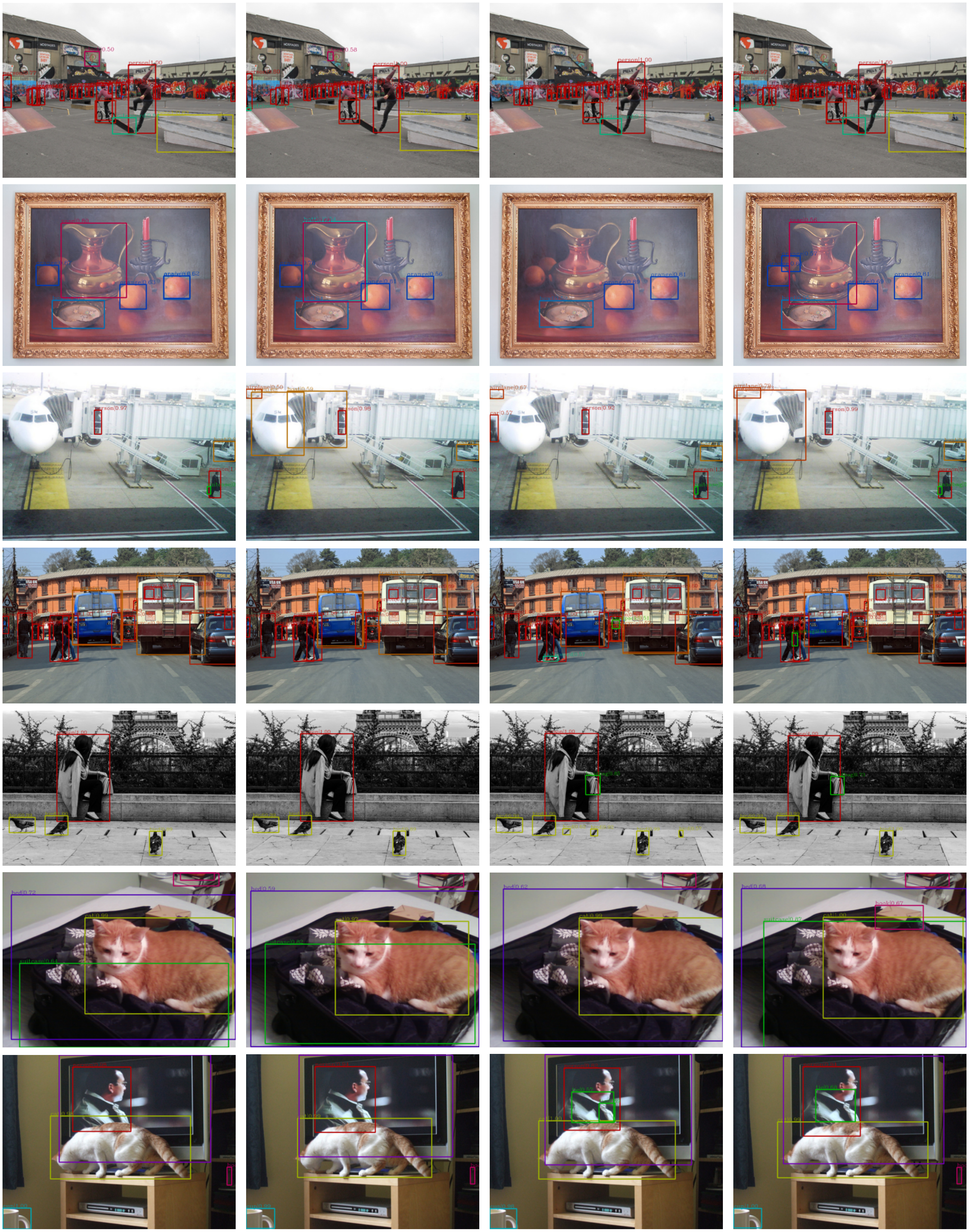
References

[1] Yukang Chen, Tong Yang, Xiangyu Zhang, Gaofeng Meng, Chunhong Pan, and Jian Sun. Detnas: Neural architecture search on object detection. In *arXiv preprint:1903.10979*, 2019. 1, 2, 3

[2] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *CVPR*, 2019. 1, 2, 3

[3] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1, 2, 3

FPN [3] NAS-FPN [2] DetNAS [1] Hit-Detector



FPN [3] NAS-FPN [2] DetNAS [1] Hit-Detector