Exemplar-Based Open-Set Panoptic Segmentation Network
Supplementary Materials

Jaedong Hwang[1*] Seoung Wug Oh[2] Joon-Young Lee[2] Bohyung Han[1]
[1]Seoul National University
[2]Adobe Research
[1]{jd730, bhhan}@snu.ac.kr, [2]{seoh, jolee}@adobe.com

1. Details

1.1. Implementation

The implementation is based on Panoptic FPN [3]. The backbone is ResNet-50-FPN [2, 4]. We utilize Pytorch [5] distributed and Detectron2 [6].

1.2. Training

For baseline models, we employ ImageNet pre-trained ResNet-50-FPN [2, 4]. The initial learning rate is 0.04 with a linear warm-up [1] and reduced by 0.1 at 30K and 40K step. The total training step is 45K (namely $1\times$ schedule). The weight decay is 0.0001 and momentum is 0.9. Semantic segmentation head trains *void* label as a new class. EOPSN is fine-tuned from the baseline with Eq. (2) (*suppression*) maintaining the learning rate 0.0004 for 30K steps. All other hyper-parameters follow those in Detectron2 [6]. All models are trained in 8 Titan V100 GPUs using synchronized SGD, with a mini-batch size of 4 images per GPU.

For exemplar-based learning, we utilize middle or large sized bounding boxes (*i.e.*, the area is larger than 32^2) to reduce noise. We sample at most 20 object proposals in every mini-batch and generate 128 clusters in every 200 step. For finding high-quality clusters, only top 10% clusters in terms of the average cosine similarity between the centroids and their elements are used. The objectness score threshold for selecting high-quality clusters is starting from 0.9 and it is linearly increased to 0.99 depending on the number of found *unknown* classes. Cosine distance threshold for finding coupled elements in a cluster and mining new exemplars is 0.15, 0.025, respectively and it is slightly decreased to 0.01.

2. Hyper-parameters

We test the proposed model with various hyper-parameters: the number of clusters, clustering interval, bounding box size. All experiments in this section are conducted with $K = 10\%$. While comparing one hyper-parameter, all other parameters are fixed as described in Section 1.2.

Table 1: Sensitivity analysis about the number of clusters on COCO *val* set where K is 10%.

The number of clusters	Known			Unknown		
	PQ	SQ	RQ	PQ	SQ	RQ
64	37.3	76.2	45.8	12.9	76.5	16.8
128 (ours)	37.7	76.8	46.3	**17.9**	76.8	**23.3**
256	37.2	76.8	45.7	12.6	**78.8**	16.0

The number of clusters Table 1 presents that the number of clusters affects the performance. This might be due to that the number of noise in each cluster is increased in the small number of clusters setting and the intra-class variability in each cluster is decreased in the large number of clusters setting. However, all are still higher than the baseline ($8.5, 73.2, 11.6$ for PQ, SQ, and RQ, respectively).

Clustering Interval Table 2 shows that short clustering interval decreases RQ and PQ. This is because the number of object proposals and the number of used images for clustering are decreased, which leads the clustering stage to be more vulnerable to the noise and be hard to find *unknown* classes with exemplars correctly.

Size of Object Proposals Table 3 presents that using large and medium proposals achieves the best performance on COCO *val* set where K is 10%. On the other hand, using small sized object proposals dramatically degrades recognition performance since it increases noise during clustering, which hinders finding correct *unknown* class.

*This work was done during an internship at Adobe Research.

Table 2: Sensitivity analysis about clustering intervals on COCO *val* set where K is 10%.

Clustering interval	Known			Unknown		
	PQ	SQ	RQ	PQ	SQ	RQ
100	37.6	77.2	46.3	8.2	**77.5**	10.6
200 (ours)	37.7	76.8	46.3	**17.9**	76.8	**23.3**
400	37.7	77.5	46.3	14.6	76.4	19.1

Table 3: Effectivity of size of proposals on COCO *val* set where K is 10%.

Proposal size	Known			Unknown		
	PQ	SQ	RQ	PQ	SQ	RQ
Large	37.7	77.5	46.4	13.5	**78.1**	17.3
Medium	37.7	77.5	46.4	12.5	74.8	16.7
Small	37.6	76.7	46.3	0.3	64.1	0.4
Large + Medium	37.7	76.8	46.3	**17.9**	76.8	**23.3**
Large + Medium + Small	37.8	77.1	46.6	6.9	69.8	9.9

3. Qualitative Results

Figure 1 shows open-set panoptic segmentation results from EOPSN on COCO *val* set with $K = 10\%$. Instances in the *unknown* class are denoted by orange color.

References

[1] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. 1

[3] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 1

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

[5] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 1

[6] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 1

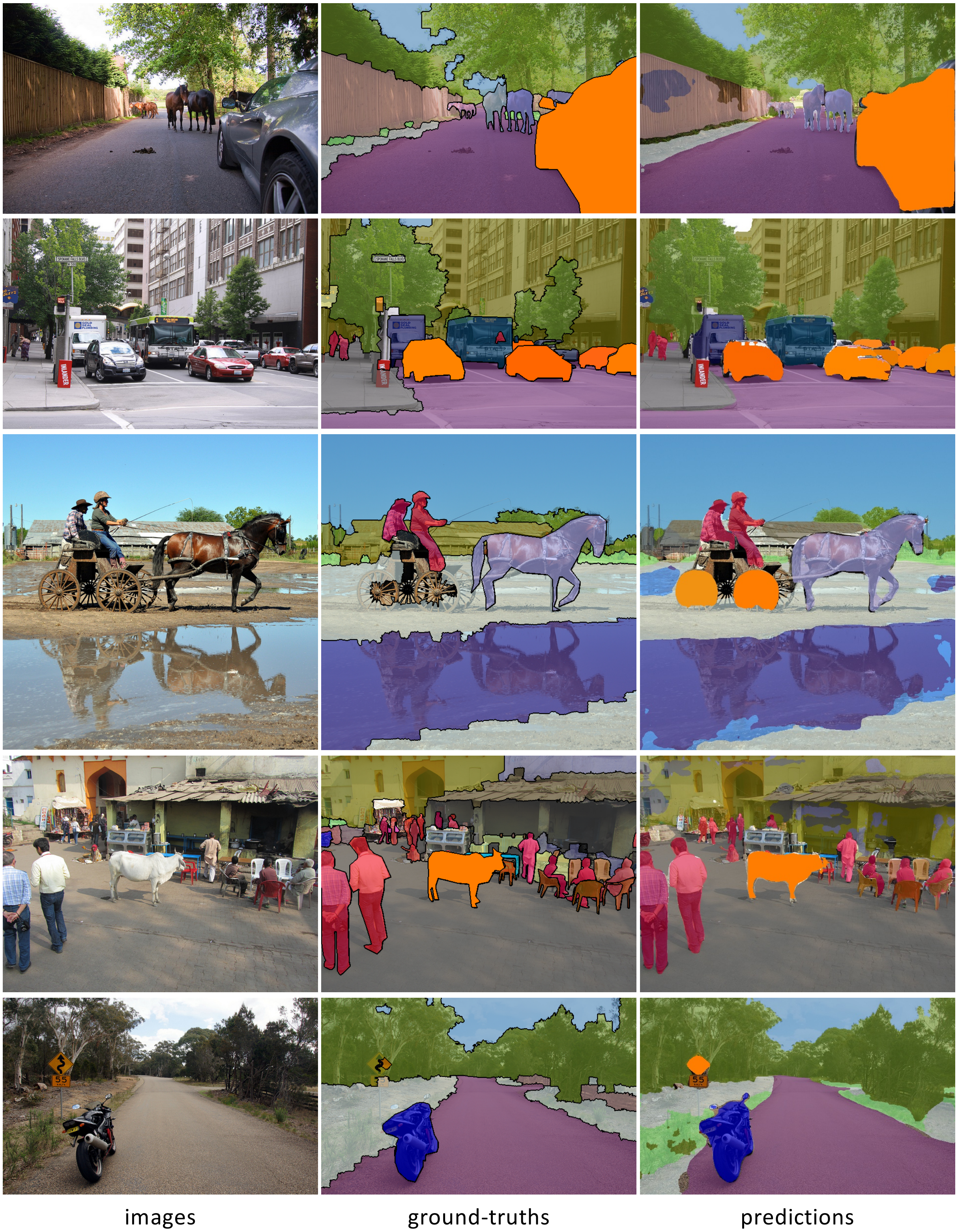

Figure 1: Qualitative results on COCO *val* set with $K = 10\%$. Each column denotes images, ground-truths and predictions of EOPSN, respectively. Instances in the *unknown* class are denoted by orange color.