Exploring Transformers for Open-world Instance Segmentation

(Supplementary Material)

Appendix A. More Related Work

Open-world Instance Segmentation. This part supplements the related work in main paper. As is pointed out, closed-world models treat the un-annotated objects as background during training and thus can not discover the *novel* objects from backdrop during inference. In order to solve the problem, there have emerged many advanced open-world works [13, 19, 24, 10, 23, 11] recently.

OLN [13] proposes to replace the classification head with localization quality head (*e.g.*, IoU head) to predict the proposal scores. Because it is only trained with positive samples, OLN would not suppress *novel* objects as background. LDET [19] addresses the task from the perspective of synthesizing images without hidden objects as the training source. Specifically, LDET proposes a data augmentation named BackErase, which pastes the annotated objects on a background image sampled from a small region. In this way, objects and background can be clearly distinguished. GGN [24] proposes to solve the problem by exploiting the pseudo ground-truth of learned pairwise affinity. It first uses the classical grouping algorithms [1, 2, 20] to generate pseudo masks from pairwise affinity predictor. Then, Mask-RCNN [8] is trained with the augmented annotations. GOOD [10] exploits the geometric cues such as depth and normals, predicted by the monocular estimators, as the additional training sets. The authors train the OLN-like proposal network for pseudo-labeling novel objects from these training source, which shows significant effectiveness. UDOS [11] combines classical bottom-up grouping with top-down learning framework. It utilizes the affinity-based grouping and refinement modules to gather the part-masks as the robust instance-level segmentations. OpenInst [23] is a concurrent work that uses the query-based detector for open-world instance segmentation.

Appendix B. Architecture

B.1. Contrastive Learning

We provide the pseudo-code of contrastive learning in Algorithm 1. The object center plays the role of query. Positive and negative samples are from the query embeddings for each image. The contrastive learning framework is only used for training and is simply abandoned during inference.

Algorithm 1 Pseudo-code of Contrastive Learning.

```
# transformer: the transformer network
# f_q, f_k: contrastive head for query and key
# queue: store the object embeddings, KxC
# m: momentum
# t: temperature

f_q.params = f_k.params # initialize

# load an image and its targets
for image, targets in loader:

   # get the query predictions
   queries = transformer.forward(image)
   q = f_q.forward(queries) # NxC
   k = f_k.forward(queries) # NxC

   # for each ground-truth object
   for target in targets:
      queue = queue.detach()
      v = mean(queue, dim=0) # object center, 1xC

      # positive and negative selection,
      # according to Eq.(1) in Appendix
      k_pos_id, k_neg_id = SimOTA(queries, target)

      k_pos = k.index_select(k_pos_id) # k1xC
      k_neg = k.index_select(k_neg_id) # (k2-k1)xC

      # positive logits: 1xk1
      l_pos = mm(v, k_pos.transpose(0,1))

      # negative logits: 1x(k2-k1)
      l_neg = mm(v, k_neg.transpose(0,1))

      # logits: 1x[k1+(k2-l1)]
      logits = cat([l_pos, l_neg], dim=1)

      # contrastive loss, Eq.(2) in main paper
      labels = cat([ones(k1), zeros(k2-k1)], dim=0)
      loss = ContrastiveLoss(logits/t, labels)

      # Adam update: transformer and f_k
      loss.backward()
      update(transformer.params)
      update(f_k.params)

      # momentum update: f_q
      f_q.params = m*f_q.params+(1-m)*f_k.params

   # find the best matched queries for ground-truths
   query_ids = BipartiteMatch(queries, targets)

   # update queue
   q_c = q.index_select(query_id)
   enqueue(queue, q_c)
   dequeue(queue)
```

`mm`: matrix multiplication; `cat`: concatenation.

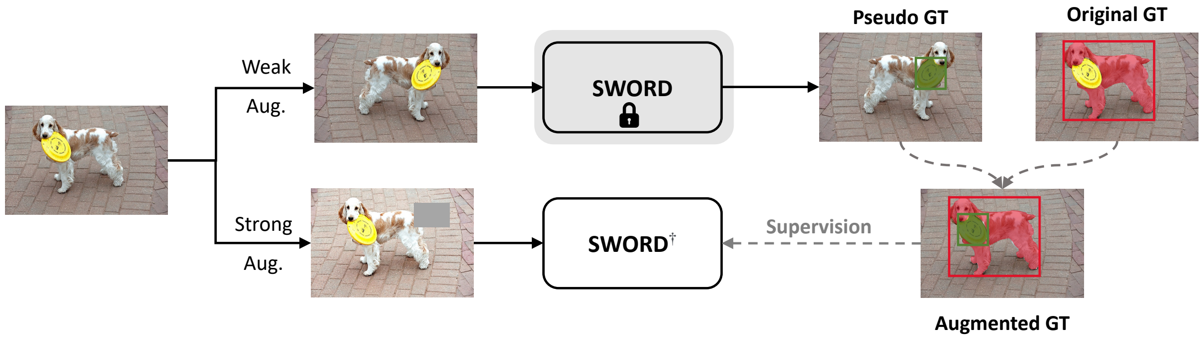


Figure 1: **The pipeline of pseudo ground-truth training**. The pretrained SWORD is first adopted to generate the pseudo boxes/masks. Then the top-scoring predictions are merged with the original annotations. Finally, SWORD† is trained under the supervision of augmented ground-truths. Note that SWORD† uses exactly the same architecture as Deformable-DETR.

Universal Object Queue. The universal object queue $Q = [q_1, q_2, ..., q_K] \in \mathbb{R}^{K \times C}$ stores the object embeddings, where K is the queue size and C is the channel dimension of embeddings. The queue is randomly initialized. In each training iteration, the query embeddings of those predictions best matching the ground-truths are enqueue and the oldest ones are dequeue. Notably, these embeddings are computed by the slowly updated contrastive head f_q to ensure the stability of universal object queue.

Sample Selection. For contrastive learning, we adopt the SimOTA [7, 26] strategy to dynamically select the positive and negative samples according to the matching cost. Given an image, we compute the matching cost between the i-th prediction p_i and the j-th ground-truth g_j as

$$C^{ij} = \lambda_{cls} \cdot C^{ij}_{cls} + \lambda_{L1} C^{ij}_{L1} + \lambda_{giou} C^{ij}_{giou} \qquad (1)$$

where λ_{cls}, λ_{L1} and λ_{giou} are the coefficients. C^{ij}_{cls} is Focal loss [15], and C^{ij}_{box} is a combination of the $\mathcal{L}_1$ loss and generalized IoU loss [18]. For the ground-truth g_j, we sum up the top 10 IoU values to get k_1 and the top 100 IoU values to get k_2. Then, we take the top k_1 predictions with the lowest cost as positive samples. To improve the embedding quality of negative samples, we choose the top k_2 predictions with the lowest cost and exclude the first k_1 ones. The left $k_2 - k_1$ predictions are the hard negatives. We use the regularly updated contrastive head f_k to compute their embeddings and form the positive set $\mathcal{K}^+$ and negative set $\mathcal{K}^-$.

B.2. Pseudo Ground-truth Training

Details. The previous work GGN [24] shows that the pseudo labeling method can greatly boost the performance of Mask-RCNN in open world. Inspired by this work, we also develop an extension model, SWORD†, by exploiting the pseudo ground-truth of SWORD. As shown in Figure 1, we first use SWORD to generate the pseudo boxes/masks. Then the top-scoring predictions are merged with the original annotations to form the augmented ground-truths, which plays the role of supervision to train the SWORD†. Note that SWORD† uses exactly the same architecture as closed-world model Deformable-DETR [29].

In the pseudo labeling process, we empirically find that using the IoU scores of SWORD leads to better learning results. And the merge process directly follows the existing practice [24]. Specifically, we first set the NMS value as 0.3 for SWORD to remove most predictions. Considering that the pseudo labels should focus on covering the *novel* objects, we discard those proposals having the box IoU greater than 0.5 with the annotated objects. Finally, the top-k predictions are kept as pseudo ground-truths.

Data Augmentation. Data augmentation has been demonstrated to play an important role in the self-training [27, 12, 30] and semi-supervised methods [21, 16, 28]. Following [16], we use the random horizontal flip for weak augmentation. And the strong augmentation includes random color jittering, grayscale, Gaussian blur and random cutout [5].

Appendix C. Implementation Details

Model Details. The model configurations mostly follow Deformable-DETR [29]. The Transformer has six encoders and six decoders with the hidden dimension of 256. To ensure a high recall, the object query number of SWORD is set to 2000 when trained on VOC classes and 1000 for all other settings. For contrastive learning, the size of universal object queue is set as 4096 and the exponential moving average (EMA) rate of the momentum contrastive head is 0.999. In the pseudo ground-truth training, SWORD† uses 1000 object queries for all the settings. ResNet-50 [9] is adopted as the backbone otherwise specified.

Training Details. We use the Adam [14] optimizer with a base learning rate of 2×10^{-4} and weight decay of 1×10^{-4} for model training. All the models are trained on 8 GPUs with a batch size of 16. We present two models in this work, SWORD and SWORD†. SWORD is trained for 80k iterations, with the learning rate decaying at the 60k-th iteration. As the VOC classes are partially annotated in COCO

Table 1: **Ablation on strong augmentation in pseudo ground-truth training**. We evaluate the models in COCO to UVO and VOC to non-VOC setups. And the results are reported on the *novel* objects.

Strong Aug.	COCO to UVO						VOC to non-VOC					
	$\mathrm{AP^b}$	$\mathrm{AR^b_{10}}$	$\mathrm{AR^b_{100}}$	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$	$\mathrm{AP^b}$	$\mathrm{AR^b_{10}}$	$\mathrm{AR^b_{100}}$	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$
✗	16.0	22.3	49.5	12.1	20.5	42.3	5.6	21.4	38.8	5.2	19.7	33.8
✓	16.6	22.7	50.0	12.7	20.9	42.8	6.2	22.0	40.0	5.8	20.2	34.9

Table 2: **Ablation on the EMA rate**. The results are based on the COCO to UVO setup.

EMA	Novel			All		
	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$
0.5	8.9	16.3	27.8	16.9	24.4	35.8
0.9	11.3	19.2	37.4	24.3	30.4	47.8
0.99	11.2	19.0	38.5	25.3	30.6	48.9
0.999	12.8	19.4	40.6	28.0	32.4	51.5
0.9999	11.9	18.6	40.7	28.4	32.7	52.0

Table 3: **Ablation on the universal object queue size**. The results are based on the VOC(COCO) to UVO setup.

Size	Novel			All		
	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$	$\mathrm{AP^m}$	$\mathrm{AR^m_{10}}$	$\mathrm{AR^m_{100}}$
256	4.9	12.4	31.4	17.5	23.8	42.1
1024	5.3	13.2	32.9	18.7	24.9	44.0
4096	6.1	13.3	34.9	19.6	25.3	45.2
8192	5.5	12.6	33.9	19.2	24.9	44.8

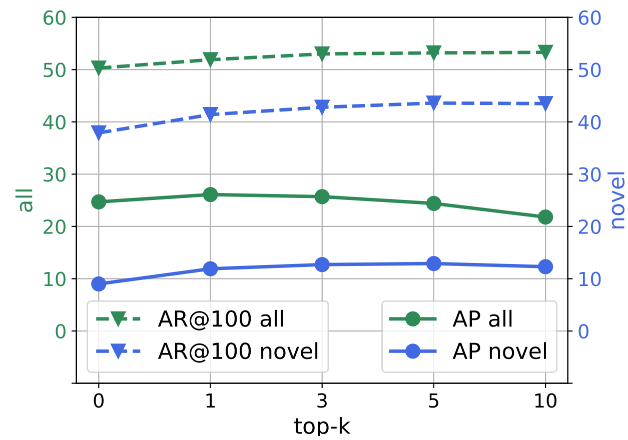


Figure 2: **The effect of top-k in pseudo ground-truth training**. The results are based on mask metrics in COCO to UVO setup.

dataset, the model tends to overfit to the base classes. So we train SWORD from scratch when the training source is VOC. In all other settings, the backbone is initialized with the ImageNet [4] pretrained weights. For SWORD†, backbones always use the ImageNet pretrained weights for initialization. It undergoes 90k iterations of training, with the learning rate reduced by a factor of 10 at the 60k-th and 80k-th iterations. During training, we resize the input images such that the shortest side is at least 480 and at most 800, while the longest side is at most 1333. The loss coefficients are set as $\lambda_{cls} = 2.0$, $\lambda_{cls} = 2.0$, $\lambda_{L1} = 5.0$, $\lambda_{mask} = 2.0$, $\lambda_{dice} = 5.0$ and $\lambda_{iou} = 1.0$, respectively. All the models use the NMS value of 0.7 during inferene.

Appendix D. Additional Experimental Results

We provide additional experimental results to study the critical parameters for our method. The ablation studies are based on the COCO (80 classes) to UVO setup by default.

D.1. Ablation on Contrastive Learning

The Effect of EMA Rate. The momentum update of the contrastive head can improve the consistency of the universal object queue. And a larger EMA rate allows the slower feature change. In Table 2, we present the experimental results with various EMA rate α from 0.5 to 0.9999. As illustrated in the first row, with the EMA rate of 0.5, the model gets relatively low results in both AP and AR metrics. This indicates that the model suffers from the detrimental effect of quick transformation of the object center. And the performance is greatly boosted with the EMA rate increases, *e.g.*, the $\mathrm{AP^b}$ on *all* objects achieves 6.9% gain by increasing α from 0.5 to 0.9. We observe that the performance becomes stable when a larger EMA rate (*e.g.*, $\alpha = 0.999$) is applied.

The Effect of Universal Object Queue Size. In this study, we investigate the impact of the universal object queue size on the VOC(COCO) to UVO setup. Our findings are presented in Table 3. We observe that when the queue size is increased from 256 to 4096, the model achieves a performance gain of 1.2 $\mathrm{AP^m}$ and 2.1 $\mathrm{AP^m}$ for novel and all objects, respectively. This improvement in performance may be attributed to the increased stability of the object center, which ensures that the object center captures the common characteristic of objects. However, we observe a decline in performance with further increases in the queue size, possibly due to the adverse effects of older object features on contrastive learning.

Table 4: **Ablation on the query number.** The results are based on COCO to UVO setup. Our default settings are marked in gray .

Query	Novel			All		
	AP^m	AR^m_{10}	AR^m_{100}	AP^m	AR^m_{10}	AR^m_{100}
Deformable-DETR						
300	8.9	16.1	37.1	24.4	29.8	49.7
1000	9.0	16.7	37.9	24.7	30.1	50.3
2000	8.6	15.8	37.9	24.7	30.0	50.3
SWORD						
300	11.2	18.6	34.4	27.4	32.4	46.3
1000	12.8	19.4	40.6	28.0	32.4	51.5
2000	12.7	19.7	42.7	28.3	32.8	53.0

D.2. Ablation on Pseudo Ground-truth Training

The Effect of Strong Augmentation. To validate the effectiveness of strong augmentation in pseudo ground-truth training, we ablate the experiments in COCO to UVO and VOC to non-VOC settings, respectively. By comparing the two rows in Table 1, it is observed that the model could obtain better performance with the help of strong augmentation. Besides, we observe that the benefit of strong augmentation is more clear in VOC to non-VOC setup than COCO to UVO setup. The reason may attribute to the fact that the annotation density and class number of PASCAL-VOC are more limited, which requires the strong augmentation to generate more diverse training samples.

The Effect of Pseudo Ground-truth Number. The usage of pseudo ground-truth helps the closed-world models discover the *novel* objects. However, it also introduces noisy supervision signals. To study the relationship between the model behavior and the number of pseudo ground-truth, we vary the number of k for selecting the top-scoring predictions and plot the results in Figure 2. Here, we have the critical finding: *More pseudo ground-truths benefit AR while hurting AP*. It can be seen that ARs keep improving with the increase of k, while AP for *all* objects consistently degrades. AP for *novel* objects also starts decreasing when k reaches a large value (*e.g.*, $k = 10$). This is reasonable because more pseudo ground-truths will induce many false positive predictions. The results suggest that the value of top-k should be carefully chosen to achieve the optimal balance between APs and ARs.

D.3. More Ablation Studies

The Effect of Query Number. We study the effect of query number for both Deformable-DETR and proposed SWORD in Table 4. The results show that Deformable-DETR achieves a slight improvement in performance when the object query number is increased from 300 to 1000. However, the performance saturates at a query number of 1000, indicating that 1000 queries represent the upper limit for closed-world models to locate all objects in this open-world setup. In contrast, our proposed SWORD consistently achieves higher average recalls (ARs) as the query number increases. This performance profits can be attributed to the `stop-grad` operation, which prevents the suppression of novel objects and enables the network to discover them more effectively. It is worth noting that we use the same query number for both Deformable-DETR and SWORD in all experiments for fair comparisons.

Table 5: **Ablation on the backbones.** The results are based on COCO to UVO setup.

Backbone	Novel			All		
	AP^m	AR^m_{10}	AR^m_{100}	AP^m	AR^m_{10}	AR^m_{100}
R50	12.8	19.5	40.6	28.0	32.4	51.5
R101	12.6	19.9	41.3	29.5	33.4	52.7
Swin-T	12.2	19.5	40.8	29.4	33.4	52.0
Swin-L	13.5	20.5	41.2	34.3	37.0	54.1

Table 6: **Ablation on the pseudo-label training for different models.** 'w/ PL' represents the model is trained with the pseudo labels generated from the proposed SWORD. 'D-DETR' denotes Deformable-DETR.

Method	w/ PL	VOC to non-VOC			COCO to UVO		
		AP^m	AR^m_{10}	AR^m_{100}	AP^m	AR^m_{10}	AR^m_{100}
D-DETR	-	2.2	10.2	22.7	9.0	16.7	37.4
D-DETR	✓	5.8	20.2	34.9	12.7	20.9	42.8
SWORD	-	4.8	15.7	30.2	12.8	19.4	40.6
SWORD	✓	5.9	20.9	36.2	13.3	21.4	43.5

Do Stronger Backbones Benefit in Open-world? There exists the consensus that stronger backbones [9, 6, 25, 17, 22, 3] could greatly increase the performance under the fully-supervised setup. Of particular interest, we examine with ResNet [9] and Swin-Transformer [17] to study the effect of using strong backbones in open-world scenario. Table 5 illustrates that model consistently performs better with increasing the size of backbones. Interestingly, we also observe that out-of-domain objects gets less benefit from stronger backbone than in-domain objects in the open-world. For example, by switching the backbone from Swin-Tiny to Swin-Large, the model enjoys the significant 4.9% AP^m gain for *all* objects while the advance is marginal for *novel* objects (+1.3% AP^m).

Ablation on the Pseudo Ground-truth Training for Different Models. We conduct the experiments using pseudo labels to train the proposed SWORD and display the results in Table 6. We report the results on novel objects for both cross-category (VOC to non-VOC) and cross-dataset

(COCO to UVO) generalizations. It is observed that the inclusion of pseudo-label training can further enhance the performance of SWORD, which also surpasses the results by using the standard Deformable-DETR for pseudo-label training. This highlights the strong ability of SWORD in discovering novel objects in the open-world scenario, proving the necessity of our designs.

Appendix E. Visualization

We visualize more examples in Figure 3. We demonstrate the superiority of proposed model in diverse scenes.

References

[1] Pablo Arbelaez. Boundary extraction in natural images using ultrametric contour maps. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 182–182. IEEE, 2006. 1

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010. 1

[3] Shoufa Chen, Enze Xie, Chongjian Ge, Ding Liang, and Ping Luo. Cyclemlp: A mlp-like architecture for dense prediction. *arXiv preprint arXiv:2107.10224*, 2021. 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[5] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 2

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[7] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4

[10] Haiwen Huang, Andreas Geiger, and Dan Zhang. Good: Exploring geometric cues for detecting objects in an open world. *arXiv preprint arXiv:2212.11720*, 2022. 1

[11] Tarun Kalluri, Weiyao Wang, Heng Wang, Manmohan Chandraker, Lorenzo Torresani, and Du Tran. Open-world instance segmentation: Top-down learning with bottom-up supervision. *arXiv preprint arXiv:2303.05503*, 2023. 1

[12] Prannay Kaul, Weidi Xie, and Andrew Zisserman. Label, verify, correct: A simple few shot object detection method. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14237–14247, 2022. 2

[13] Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify. *IEEE Robotics and Automation Letters*, 7(2):5453–5460, 2022. 1, 6

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[15] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[16] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. *arXiv preprint arXiv:2102.09480*, 2021. 2

[17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 4

[18] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019. 2

[19] Kuniaki Saito, Ping Hu, Trevor Darrell, and Kate Saenko. Learning to detect every thing in an open world. *arXiv preprint arXiv:2112.01698*, 2021. 1

[20] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000. 1

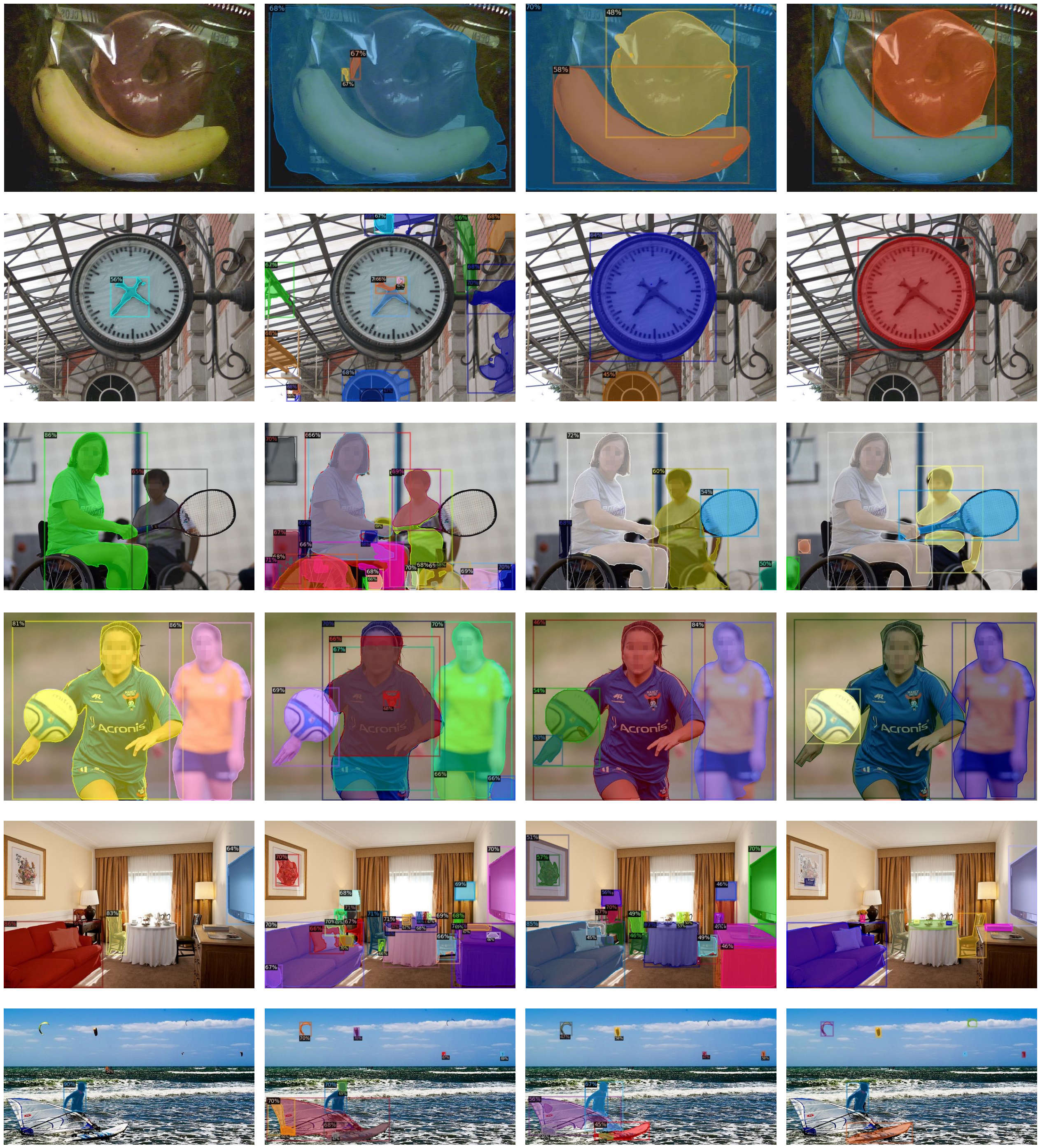
[21] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 2

[22] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021. 4

[23] Cheng Wang, Guoli Wang, Qian Zhang, Peng Guo, Wenyu Liu, and Xinggang Wang. Openinst: A simple query-based method for open-world instance segmentation. *arXiv preprint arXiv:2303.15859*, 2023. 1

[24] Weiyao Wang, Matt Feiszli, Heng Wang, Jitendra Malik, and Du Tran. Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4422–4432, 2022. 1, 2

[25] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao.



(a) Deformable-DETR (b) OLN (c) SWORD† (d) Ground-truth

Figure 3: **Visualization examples in VOC to non-VOC setting**. All the models are trained on the 20 PASCAL-VOC classes of COCO dataset. The score thresholds for visualization are set as 0.45, 0.65 and 0.45 for Deformable-DETR [29], OLN [13] and SWORD†, respectively. It is observed that Deformable-DETR is unable to segment the *novel* objects and OLN produces many false positive predictions. Our model obviously provides the accurate and exhaustive segmentation masks.

Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021. 4

[26] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. *arXiv preprint arXiv:2207.10661*, 2022. 2

[27] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020. 2

[28] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3060–3069, 2021. 2

[29] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2, 6

[30] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33:3833–3845, 2020. 2