A Appendix

In this appendix, we provide the architecture of the foreground prediction branch (in Figure 6) and detailed experimental settings first. Then some annotations in UVO dataset are visualized in Figure 7 to show the challenges of open world instance segmentation. Finally, additional visualization results of proposed TOIS are shown in Figure 8.

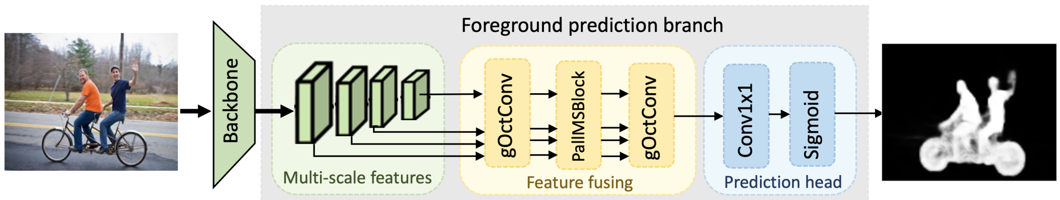

Figure 6: **Architecture of foreground prediction branch**. Multi-scale features extracted from backbone are fed into the feature fusing module to exchange and fuse the multi-scale information. Then a fused feature is sent to the prediction head to predict the final foreground map. Considering the efficiency, we follow [27] to introduce the gOctConv [27] and PallMSBlock [27] to perform feature fusing.

A.1 Detailed experimental settings

Implementation details For feature extracting, we obtain the multi-scale features through a sequential backbone network [21, 22], and FPN [28]. The multi-scale features contain D-dimensional feature maps with resolutions of 1/4, 1/8, 1/16, and 1/32. In the pixel decoder module, six MSDeformAttn layers are employed, while the transformer decoder have three layers with 100 queries by default.

In the fully-supervised setting, the total loss L_f can be formulated as: $L_f = \alpha L_m + \beta L_p + \gamma L_c + \omega L_o$.

In COCO$\rightarrow$UVO evaluation, we set the weight α of mask loss (L_m) to 5.0, the weight β of foreground loss (L_p) to 2.0, the weight γ of cross-task consistency loss (L_c) to 2.0 and the weight ω of objectness loss (L_o) to 2.0.

In UVO$\rightarrow$UVO evaluation, we set the weight α of mask loss (L_m) to 5.0, the weight β of foreground loss (L_p) to 1.0, the weight γ of cross-task consistency loss (L_c) to 1.0 and the weight ω of objectness loss (L_o) to 2.0.

In Cityscapes$\rightarrow$Mapillary evaluation, we set the weight α of mask loss (L_m) to 4.0, the weight β of foreground loss (L_p) to 2.0, the weight γ of cross-task consistency loss (L_c) to 2.0 and the weight ω of objectness loss (L_o) to 2.0.

In COCO(VOC)$\rightarrow$COCO(noneVOC) evaluation, we apply the same hyper-parameter setting as that in COCO$\rightarrow$UVO evaluation for convenience. Perhaps fine-tuning these hyper-prameters can lead to better performance.

Training settings Specifically, AdamW [29] optimizer and the step learning rate schedule are applied to optimize our model. An initial learning rate of 0.0001 and a weight decay of 0.05 are utilized for all backbones. We set a learning rate multiplier of the backbone to 0.1 and we decay the learning rate at 0.9 and 0.95 fractions of the total number of training steps by a factor of 10. For data augmentation, we use the large-scale jittering (LSJ) augmentation with a random scale sampled from range 0.1 to 2.0 followed by a fixed size crop to 1024×1024 on COCO dataset and 640×640 on UVO dataset. Besides, a Cutout [30] strategy that randomly cuts out a region of size [1/8·w, 1/8·h] to [1/3·w, 1/3·h] is introduced during training. On COCO dataset, we train our models for 38×10^4 iterations with a batch size of 16, while on UVO dataset, we train our models for 12×10^4 iterations with the same batch size.

TOIS training process with pseudo-labeling on COCO dataset

Algorithm 1: TOIS training process with pseudo-labeling

Data: Image dataset
Result: Proposed TOIS Model M_u
initialization the student model M_u, and teacher model M_t=M_u.copy();
while *Image* $i \notin \varnothing$ **do**
  read image i and corresponding groundtruth gt_i;
  extract backbone feature X_i;
  pred_masks ← M_t.predictor(X_i);
  pseudo_proposals← filter_masks_with_confidence(pred_masks, confidence_threshold);
  pseudo labels ← filter_masks_with_IOU(pseudo_proposals, IOU_threshold);
  training labels ← merge(gt_i, pseudo labels);
  aug_data← Cutout(X_i, training labels);
  M_u ← M_u.training(aug_data);
  M_t ← M_t.EMA_update(M_t,M_u)
end

A.2 Visualization of annotations and our results on UVO dataset

Unlike in closed-world instance segmentation, where the object categories have been clearly defined, instance definition in OWIS is much more ambiguous and harder for annotators to follow. Inevitably, the instance annotation could become inconsistent across images, as shown in Figure 7. Our method is motivated by this observation that the instance annotation in the existing datasets is very noisy. Our solution to this issue is to introduce a self-correcting mechanism to combat erroneous annotations, which provides additional guidance to both prediction tasks when the noisy annotations fail to provide correct supervision. The visualization results in Figure 8 demonstrate that our proposed TOIS can segment many novel objects that have not been unseen in the training set.

References

[1] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[2] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] W. Wang, M. Feiszli, H. Wang, and D. Tran, "Unidentified video objects: A benchmark for dense, open-world segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10776–10785, 2021.

[4] W. Wang, M. Feiszli, H. Wang, J. Malik, and D. Tran, "Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity," *CVPR*, 2022.

[5] K. Saito, P. Hu, T. Darrell, and K. Saenko, "Learning to detect every thing in an open world," *arXiv preprint arXiv:2112.01698*, 2021.

[6] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," 2022.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

[8] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "Blendmask: Top-down meets bottom-up for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8573–8581, 2020.

[9] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3150–3158, 2016.

[10] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9157–9166, 2019.

Figure 7: Visualizations of UVO annotations. It is notable that the same class of object may be labeled as an instance or as background in different images. (as shown in the area highlighted by the ellipse). This inconsistency of annotations pose a great challenge to the algorithms.

Figure 8: Visualizations results of our proposed TOIS in UVO dataset. TOIS can discover many novel objects, as shown in regions in red boxes.

[11] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "Solo: Segmenting objects by locations," in *European Conference on Computer Vision*, pp. 649–665, Springer, 2020.

[12] Y. Fang, S. Yang, X. Wang, Y. Li, C. Fang, Y. Shan, B. Feng, and W. Liu, "Instances as queries," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6910–6919, 2021.

[13] B. Dong, F. Zeng, T. Wang, X. Zhang, and Y. Wei, "Solq: Segmenting objects by learning queries," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[14] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, pp. 213–229, Springer, 2020.

[15] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[16] Y. Du, W. Guo, Y. Xiao, and V. Lepetit, "1st place solution for the uvo challenge on video-based open-world segmentation 2021," *arXiv preprint arXiv:2110.11661*, 2021.

[17] T. Vu, H. Jang, T. X. Pham, and C. Yoo, "Cascade rpn: Delving into high-quality region proposal network with adaptive convolution," *Advances in neural information processing systems*, vol. 32, 2019.

[18] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[19] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, pp. 565–571, IEEE, 2016.

[20] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," *https://github.com/facebookresearch/detectron2*, 2019.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[22] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[24] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[25] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of the IEEE international conference on computer vision*, pp. 4990–4999, 2017.

[26] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, "Solov2: Dynamic and fast instance segmentation," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[27] S.-H. Gao, Y.-Q. Tan, M.-M. Cheng, C. Lu, Y. Chen, and S. Yan, "Highly efficient salient object detection with 100k parameters," in *European Conference on Computer Vision*, pp. 702–721, Springer, 2020.

[28] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

[29] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[30] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.