APPENDIX

A. Second Stage Details

The goal of the second stage is to separate nearby objects or stacked objects. In these cases, when the first-stage model considers multiple objects with similar pixel embeddings as the same object, it could under-segment the objects. To mitigate this issue, we append a second-stage segmentation process as zoom-in refinement as shown in Fig. 1.

In detail, to get an RGB-D Region of Interest (ROI) for each first-stage segment, we pad the segment and resize it to 224×224. Next, each ROI is handled by the second-stage network trained with synthetic ROIs (224×224 pixels) from the Tabletop Object Dataset [1]. The architecture of the second-stage network is exactly the same as the first-stage network. The only difference is that the second-stage network is trained with ROIs with size 224×224 pixels. Training the second-stage network is necessary since reusing the first-stage network could not result in a significant performance increase. The second-stage network outputs some confident mask predictions to provide new segments.

For each ROI, we only keep candidate segments overlapping (thresholded at 0.5) with the original segment from the first stage. By doing this, refined segments may have sharper boundaries or separate merged objects. Finally, the segmentation label map for the whole image is obtained by collecting all the segments from the ROIs and assigning unique IDs to every object segment.

B. More Implementation Details

RGB-D Feature Map. In RGB-D case, the outputs of the ResNets, two feature maps of resolution $1/8$ of the original image size, are first added together and further bilinearly upsampled to generate one feature map of full resolution (480×640 pixels) with embedding dimension $C = 64$. Then, all feature vectors are ℓ_2 normalized to be unit vectors, i.e., projected on a $(C-1)$-dimensional hypersphere. Now we obtain unit RGB-D feature embeddings via Late Fusion Addition in [2]. When training the backbone with RGB-D images, the model is prone to overfit on the Tabletop Object Dataset. To mitigate this issue, we freeze the parameters of this pretrained backbone [2]. The sizes and backbones of MSMFormer models are listed in Table I.

MS Decoder. In the first stage, we use 6 MS decoder layers and 100 object queries, which can be viewed as cluster centers in mean shift clustering. Each decoder layer updates the cluster centers according to the above RGB-D pixel features. In the second stage, we use 8 MS decoder layers. In the hypersphere attention, we set κ as 30 since it emphasizes the points near cluster centers. We only have two classes for unseen object instance segmentation, i.e., background and object. The embedding dimension of object queries is set as 256. Queries can interact with the feature map via decoder layers. The vector dimension of the feature map is increased from 64 to 256 through a convolution layer to be consistent with 256-dim object queries. Once cluster centers are obtained, clustering strategies typically compute the distances of each pixel embedding to the centers and assign the pixels to their corresponding nearest cluster centers.

TABLE I

THE SIZES AND BACKBONES OF MSMFORMER.

Stage	Input	Backbone	#Parameters
1	RGB	ResNet50	39.2M
2	RGB	ResNet50	39.2M
1	RGB-D	ResNet34	52.5M
2	RGB-D	ResNet34	55.7M

Training with the UOAIS-Sim Dataset. We train MSMFormer on UOAIS-Sim dataset [3] ($45,000$ images) for 8 epochs since the dataset has fewer images than the Tabletop Object Dataset [4] ($280,000$ images). Other training settings do not change, such as batch size and learning rate.

C. Additional Results for Ablation Studies

In this section, we show the additional results.

Attention mask. We include attention masks in our mean shift cross-attention. We empirically show that it is useful to boost the performance of the model. For example, we compare the final performance of the second-stage network with and without attention masks, after using the output of the best first-stage model. As is shown in Table II, the usage of attention masks leads to improvements in all metrics.

TABLE II

COMPARISON OF THE SECOND-STAGE NETWORKS WITH OR WITHOUT ATTENTION MASK IN MEAN SHIFT CROSS-ATTENTIONS AFTER USING THE BEST INITIAL LABELS FROM THE FIRST STAGE.

Having Attention Mask	OCID (2390 images)						
	Overlap			Boundary			
	P	R	F	P	R	F	%75
False	91.1	90.1	90.4	86.9	84.3	85.2	83.4
True	**92.5**	**91.0**	**91.5**	**89.4**	**85.9**	**87.3**	**86.0**

ℓ_2 Norm after FFN in MS decoder layers. We use ℓ_2 norm after FFN to output unit vectors as cluster centers on a hypersphere. We compare the final performance of the second-stage network with or without ℓ_2 Norm in MS decoder layers, after using the output of the best first-stage model. As seen in Table III, ℓ_2 norm results in improvements in all metrics.

ℓ_2 Norm after layer Norms in Masked decoder layers vs. MS decoder layers. Simply adding ℓ_2 Norm after layer norms does not improve the performance of masked attention layers. As shown in Table IV, in most cases, incorporating ℓ_2 Norm adversely affects model performance. Empirically, it exhibits dissimilarities in comparison to our MS decoder.

TABLE III

COMPARISON OF THE SECOND-STAGE NETWORKS WITH OR WITHOUT ℓ_2 NORM IN MS DECODER AFTER USING THE BEST INITIAL LABELS FROM THE FIRST STAGE.

Using ℓ_2 Norm	OCID (2390 images)						
	Overlap			Boundary			
	P	R	F	P	R	F	%75
False	92.1	89.9	90.7	88.4	84.8	86.2	84.5
True	**92.5**	**91.0**	**91.5**	**89.4**	**85.9**	**87.3**	**86.0**

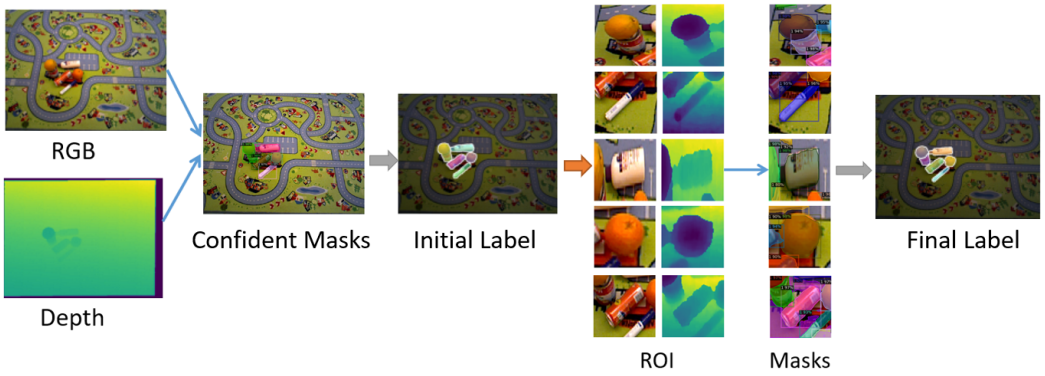

Fig. 1. The two-stage clustering process to refine segmentation labels. For an RGB-D image with size 480×640, the first stage provides approximate segments as initial labels to generate ROIs with size 224×224 (Region of Interest). Each ROI passes through the second-stage network to produce its own masks. The masks from all ROIs are further combined into the final label. In this example, the mask of an orange and a can is successfully split into two object segments in the second stage.

TABLE IV

MASKED IS THE MASKED ATTENTION (I.E. NORMAL ATTENTION WITH ATTENTION MASK). L2 NORM ADDITION IS ADDING L2 NORMALIZATION AFTER EACH LAYER NORM WITH MASKED ATTENTION. MS IS OUR MEAN SHIFT DECODER WITH HYPERSPHERE ATTENTION.

Method	Input	OCID (2390 images)							OSD (111 images)						
		Overlap			Boundary				Overlap			Boundary			
		P	R	F	P	R	F	%75	P	R	F	P	R	F	%75
Masked	RGB	67.2	73.1	67.1	55.9	58.1	54.5	54.3	60.6	60.2	59.5	48.2	41.7	43.3	32.4
L2 Norm Addition	RGB	75.7	51.8	56.3	59.8	43.0	44.8	38.2	53.8	56.1	53.8	29.0	47.2	34.0	26.2
MS (Ours)	RGB	72.9	68.3	67.7	60.5	56.3	55.8	52.9	63.4	64.7	63.6	48.6	47.4	47.0	40.2
Masked	RGBD	88.4	90.0	88.2	85.4	82.4	83.0	78.8	72.4	80.5	76.2	45.6	63.4	52.5	65.3
L2 Norm Addition	RGBD	84.8	79.9	81.3	72.8	73.8	72.2	65.7	82.8	80.3	81.2	57.8	66.2	60.6	70.0
MS (Ours)	RGBD	88.4	90.2	88.5	84.7	83.1	83.0	80.3	79.5	86.4	82.8	53.5	71.0	60.6	79.4

REFERENCES

[1] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "The best of both modes: Separately leveraging rgb and depth for unseen object instance segmentation," in *Conference on robot learning*. PMLR, 2020, pp. 1369–1378.

[2] Y. Xiang, C. Xie, A. Mousavian, and D. Fox, "Learning rgb-d feature embeddings for unseen object instance segmentation," in *Conference on Robot Learning (CoRL)*, 2020.

[3] S. Back, J. Lee, T. Kim, S. Noh, R. Kang, S. Bak, and K. Lee, "Unseen object amodal instance segmentation via hierarchical occlusion modeling," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 5085–5092.

[4] C. Xie, Y. Xiang, Z. Harchaoui, and D. Fox, "Object discovery in videos as foreground motion clustering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9994–10 003.