Adapting Pre-Trained Vision Models for Novel Instance Detection and Segmentation Appendix

Yangxiao Lu, Jishnu Jaykumar P, Yunhui Guo, Nicholas Ruozzi, Yu Xiang

APPENDIX

I. Training Details

Detection. For detection datasets, the weight adapter is trained with a batch size of 1024, while the CLIP-Adapter is trained with a batch size of 512 to enhance performance. Both adapters are trained for the same number of epochs, as detailed in Table I. To utilize Grounding DINO, a box threshold of 0.15 is set for the high-resolution and YCB-V datasets, and 0.10 for other datasets.

TABLE I

THE TRAINING EPOCHS OF DIFFERENT DETECTION DATASETS.

	High-resolution	RoboTools	LM-O	YCB-V
Both Adapters	40	80	40	40

Segmentation. We combine all instances from seven core datasets of the BOP benchmark. We train both adapters with a batch size of 1344 (32 instances $\times$ 42 templates per instance) for 500 epochs. For Grounding DINO, a box threshold of 0.10 is set for all datasets.

II. More Ablation Study

Image encoder. Given the same object proposals with GS, we evaluate the FFA embeddings from different image encoders on the High-resolution dataset [4]. As illustrated in Table V, DINOv2 exhibits superior performance attributable to its robust visual features. Fig. 1 presents the visual results of various image encoders.

Dinov2 backbones with adapter. Our Weight Adapter is compatible with various backbones of Dinov2. Notably, more powerful backbones, which offer a more effective feature space, enable our adapter to deliver greater improvements. Details of this comparison are provided in Table II.

TABLE II

DETECTION RESULTS USING DIFFERENT VIT BACKBONES OF DINOV2. "REG" INDICATES DINOV2 WITH REGISTERS. THE RESULTS ARE BASED ON ALL TESTING IMAGES OF THE HIGH-RESOLUTION DATASET. **WA DIFF** INDICATES THE IMPROVEMENT ATTRIBUTED TO THE WEIGHT ADAPTER.

Dinov2 backbone	AP	AP50	AP75	**WA Diff**
ViT-S/14	47.5	56.4	51.7	+ 0.9 AP
ViT-S/14 + WA	48.4	57.5	52.9	
ViT-B/14	54.7	65.2	60.1	+ 3.2 AP
ViT-B/14 + WA	57.9	69.0	63.6	
ViT-L/14	56.8	67.7	62.3	+ 3.6 AP
ViT-L/14 + WA	60.4	71.8	66.4	
ViT-L/14 reg	59.3	71.1	65.1	+ 4.6 AP
ViT-L/14 reg +WA	63.9	76.6	70.6	

TABLE III

DETECTION RESULTS ON THE HIGH-RESOLUTION REAL-WORLD DETECTION DATASET USING VARIOUS SAM VARIANTS.

SAM Variant	AP	AP50	AP75	Time (sec)
Mobile SAM (Tiny ViT) [1]	54.5	70.5	62.0	6.80
Mobile SAM (Tiny ViT) [1] + WA	58.2	75.0	66.5	**6.73**
HQ-SAM (ViT-H) [2]	59.8	71.5	65.7	7.38
HQ-SAM (ViT H) [2] + WA	63.8	76.2	70.5	7.31
SAM (ViT-H) (Ours) [3]	59.3	71.1	65.1	6.92
SAM (ViT-H) + WA (Ours) [3]	**63.9**	**76.6**	**70.6**	6.78

TABLE IV

THE DETECTION RESULTS ON ALL IMAGES OF THE HIGH-RESOLUTION DATASET. "PROPOSAL" REFERS TO THE OBJECT PROPOSAL METHOD. "EMBEDDING" DENOTES THE METHOD OF INSTANCE EMBEDDING GENERATION.

Proposal	Embedding	AP	AP50	AP75
SAM	cls token	41.6	49.1	46.0
GS	cls token	54.9	65.4	60.1
GS	FFA	**59.3**	**71.1**	**65.1**

Aggregation. For cls token embeddings, averaging the top k highest (*avg k*) scores yields the best results [8], [9]. For FFA embeddings, the $\max$ aggregation function achieves optimal outcomes. The comparison of these aggregation functions is detailed in Table VI.

SAM variants. We evaluate different SAM variants on the High-resolution real world detection dataset. As illustrated in Table III, SAM and our weight adapter achieves the superior performance. Given the precise bounding boxes, HQ-SAM[2] yields results comparable to those of SAM [3].

Runtime. We compare the efficiency of existing methods for novel instance segmentation, as presented in Table VII. Our approach significantly reduces running time by proposing only high-quality bounding boxes.

III. Unseen Detection of BOP Benchmark

We compare our approach with ZeroPose [10], CNOS [8], and SAM-6D [9] for 2D unseen detection, as illustrated in Table VIII. Our method outperforms the best RGB method by 2.5 AP and competes effectively with the top RGB-D method.

IV. More Qualitative Results

A. Adapter

To facilitate a comparison between the CLIP-Adapter and our Weight Adapter, we present a visual illustration in Figure 2. It is evident that the CLIP-Adapter alters the feature space and spoils the embeddings of non-target objects due to overfitting. In contrast, our Weight Adapter delivers robust embeddings within the original feature space.

TABLE V

THE INSTANCE EMBEDDINGS OF DIFFERENT IMAGE ENCODERS.

Image Encoder	Feature Dimension	AP	AP50	AP75
SAM (Vit-L) [3]	256	0.9	1.1	0.9
SAM (Vit-L) [3] + WA	256	0.7	0.8	0.7
DeiT III [5] (Vit-L)	784	2.9	3.6	3.1
DeiT III [5] (Vit-L) + WA	784	2.8	3.4	3.0
CLIP (Vit-L) [6]	1024	17.9	21.1	18.9
CLIP (Vit-L) [6] + WA	1024	20.4	24.0	21.8
DINOv2 (Vit-L) [7]	1024	56.8	67.7	62.3
DINOv2 (Vit-L) [7] + WA	1024	60.4	71.8	66.4
DINOv2 (Vit-L reg) [7]	1024	59.3	71.1	65.1
DINOv2 (Vit-L reg) [7] + WA	1024	**63.9**	**76.6**	**70.6**

Fig. 1. Visual detection results on the High-resolution dataset using different image encoders.

TABLE VI

COMPARISON OF AGGREGATION FUNCTIONS FOR SEGMENTATION PERFORMANCE. WE REPORT AVERAGE PRECISION (AP). *avg k* REFERS TO AVERAGING THE TOP k SCORES. ALL RESULTS ARE BASED ON OBJECT PROPOSALS FROM GROUNDED SAM (GS).

Embedding	Aggregation	BOP Datasets							
		LM-O	T-LESS	TUD-L	IC-BIN	ITODD	HB	YCB-V	Mean
cls token	*avg k*	41.7	**41.7**	**50.8**	**31.5**	**30**	**58**	**63**	**45.2**
cls token	max	**42**	37.4	45.9	30.2	28.9	54.9	61.5	43.0
FFA	*avg k*	42.5	42	47.4	28.2	27.3	55.1	**61.5**	43.4
FFA	max	**42.9**	**43**	**52**	**30.5**	**28.8**	**56.6**	59.7	**44.8**

TABLE VII

RUNTIME COMPARISONS OF VARIOUS METHODS FOR NOVEL INSTANCE SEGMENTATION.

Method	Proposal	Server	Time (sec)
CNOS [8]	FastSAM	Tesla V100	0.22
CNOS [8]		DeForce RTX 3090	0.23
SAM-6D (RGB) [9]		DeForce RTX 3090	0.25
SAM-6D (RGBD) [9]		DeForce RTX 3090	0.45
CNOS [8]	SAM	Tesla V100	1.84
CNOS [8]		DeForce RTX 3090	2.35
SAM-6D (RGB) [9]		DeForce RTX 3090	2.28
SAM-6D (RGBD) [9]		DeForce RTX 3090	2.80
NIDS-Net w/o adapter (Ours)	GS	RTX A5000	0.49
NIDS-Net + CA (Ours)			0.48
NIDS-Net + WA (Ours)			0.48
NIDS-Net + WA + s_{appe} (Ours)			0.48

B. Detection

We display the visual outcomes of our methodology with the weight adapter on the LMO and YCB-V datasets in Figure 4. The gap between synthetic and real images results in some instances of detection failure. For example, some LM-O instances are not found with our method. The examples of the high-resolution dataset are presented in Fig. 3.

TABLE VIII

UNSEEN INSTANCE DETECTION RESULTS ACROSS THE SEVEN CORE DATASETS OF THE BOP BENCHMARK, WITH ALL RESULTS REPORTED AS AVERAGE PRECISION (AP).

Method	Proposal	BOP Datasets							Mean
		LM-O	T-LESS	TUD-L	IC-BIN	ITODD	HB	YCB-V	
ZeroPose [10]	SAM	36.7	30.0	43.1	22.8	25.0	39.8	41.6	34.1
CNOS [8]	SAM	39.5	33.0	36.8	20.7	31.3	42.3	49.0	36.1
CNOS [8]	FastSAM	43.3	39.5	53.4	22.6	32.5	51.7	56.8	42.8
SAM-6D (RGB) [9]	FastSAM	43.8	41.7	54.6	23.4	37.4	52.3	57.3	44.4
SAM-6D (RGBD) [9]	SAM	**46.6**	43.7	53.7	**26.1**	39.3	53.1	51.9	44.9
SAM-6D (RGBD) [9]	FastSAM	46.3	45.8	57.3	24.5	**41.9**	55.1	58.9	**47.1**
NIDS-Net w/o adapter (Ours)	GS	44.9	42.8	43.4	24.4	34.9	54.8	56.5	43.1
NIDS-Net + WA (Ours)		44.9	48.9	46.0	24.5	36.0	**59.4**	**62.4**	46.0
NIDS-Net + WA + s_{appe} (Ours)		45.7	**49.3**	**48.6**	25.7	37.9	58.7	62.1	46.9

C. Segmentation

We present the visual results of our approach using the weight adapter on the BoP datasets in Figures 5, 6, 7, and 8. These images demonstrate the effectiveness of our approach in cluttered environments. In some cases of T-LESS and IC-BIN datasets, Grounding DINO generates large bounding boxes which include multiple objects, causing under-segmentation. Furthermore, in IC-BIN and HB datasets, some heavily occluded objects with low confidence scores are overlooked by our method.

REFERENCES

[1] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, and C. S. Hong, "Faster segment anything: Towards lightweight sam for mobile applications," *arXiv:2306.14289*, 2023.

[2] L. Ke, M. Ye, M. Danelljan, Y.-W. Tai, C.-K. Tang, F. Yu *et al.*, "Segment anything in high quality," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[3] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[4] Q. Shen, Y. Zhao, N. Kwon, J. Kim, Y. Li, and S. Kong, "A high-resolution dataset for instance detection with multi-view instance capture," in *NeurIPS Datasets and Benchmarks Track*, 2023.

[5] H. Touvron, M. Cord, and H. Jégou, "Deit iii: Revenge of the vit," in *European conference on computer vision*. Springer, 2022, pp. 516–533.

[6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[7] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv:2304.07193*, 2023.

[8] V. N. Nguyen, T. Groueix, G. Ponimatkin, V. Lepetit, and T. Hodan, "Cnos: A strong baseline for cad-based novel object segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 2134–2140.

[9] J. Lin, L. Liu, D. Lu, and K. Jia, "Sam-6d: Segment anything model meets zero-shot 6d object pose estimation," *arXiv:2311.15707*, 2023.

[10] J. Chen, M. Sun, T. Bao, R. Zhao, L. Wu, and Z. He, "3d model-based zero-shot pose estimation pipeline," *arXiv:2305.17934*, 2023.

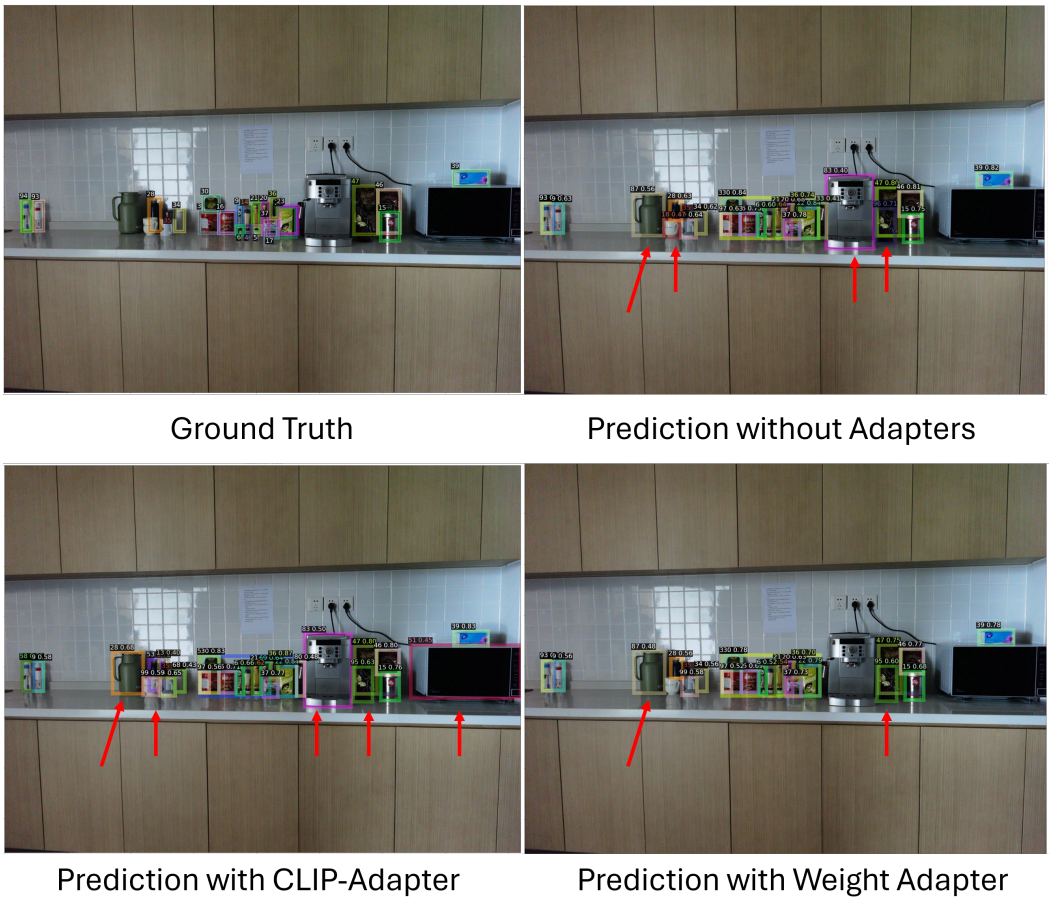

Fig. 2. Comparison of different adapters on a hard scene image from the high-resolution dataset. Red arrows denote non-target objects that are erroneously classified as targets.

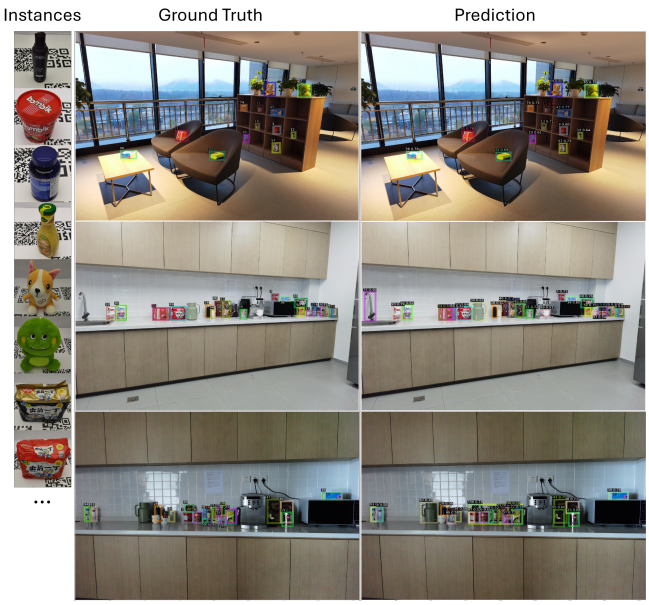

Fig. 3. Visual examples of our results using the Weight Adapter on the high-resolution dataset. Our approach detects specific object instances in cluttered scenes according to their real template images.

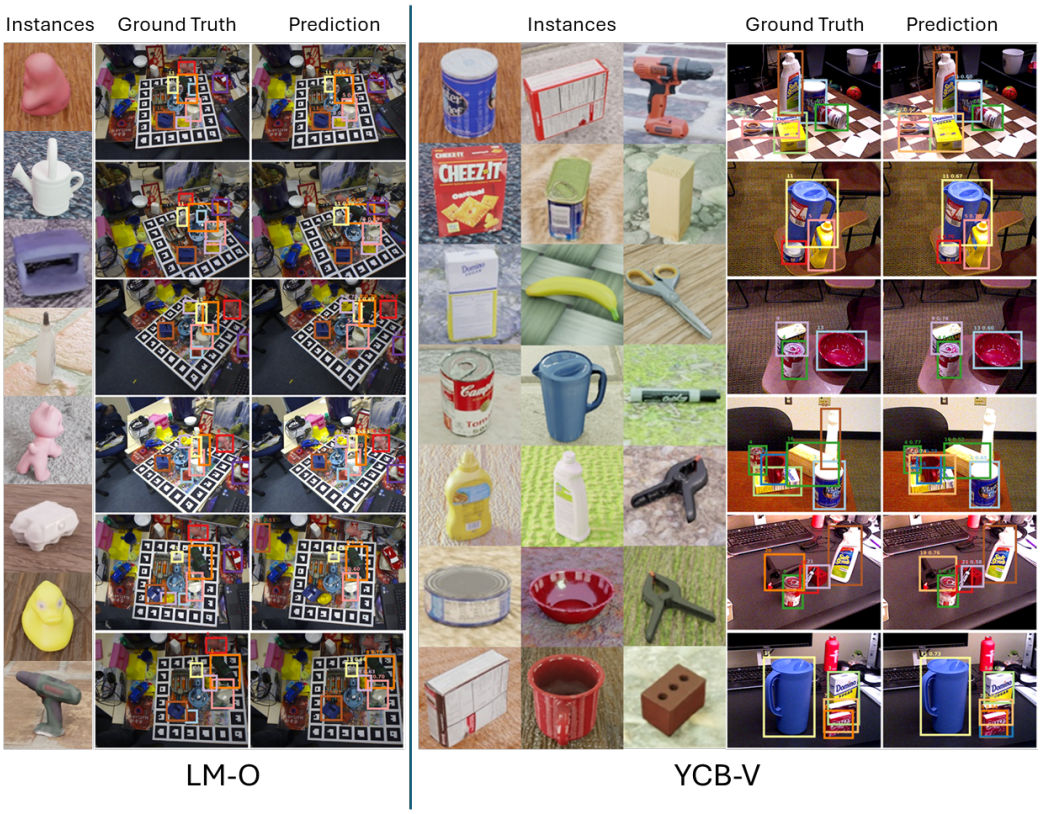

Fig. 4. Visual detection results using the Weight Adapter on the LM-O and YCB-V datasets. Our approach detects object instances according to their synthetic template images.

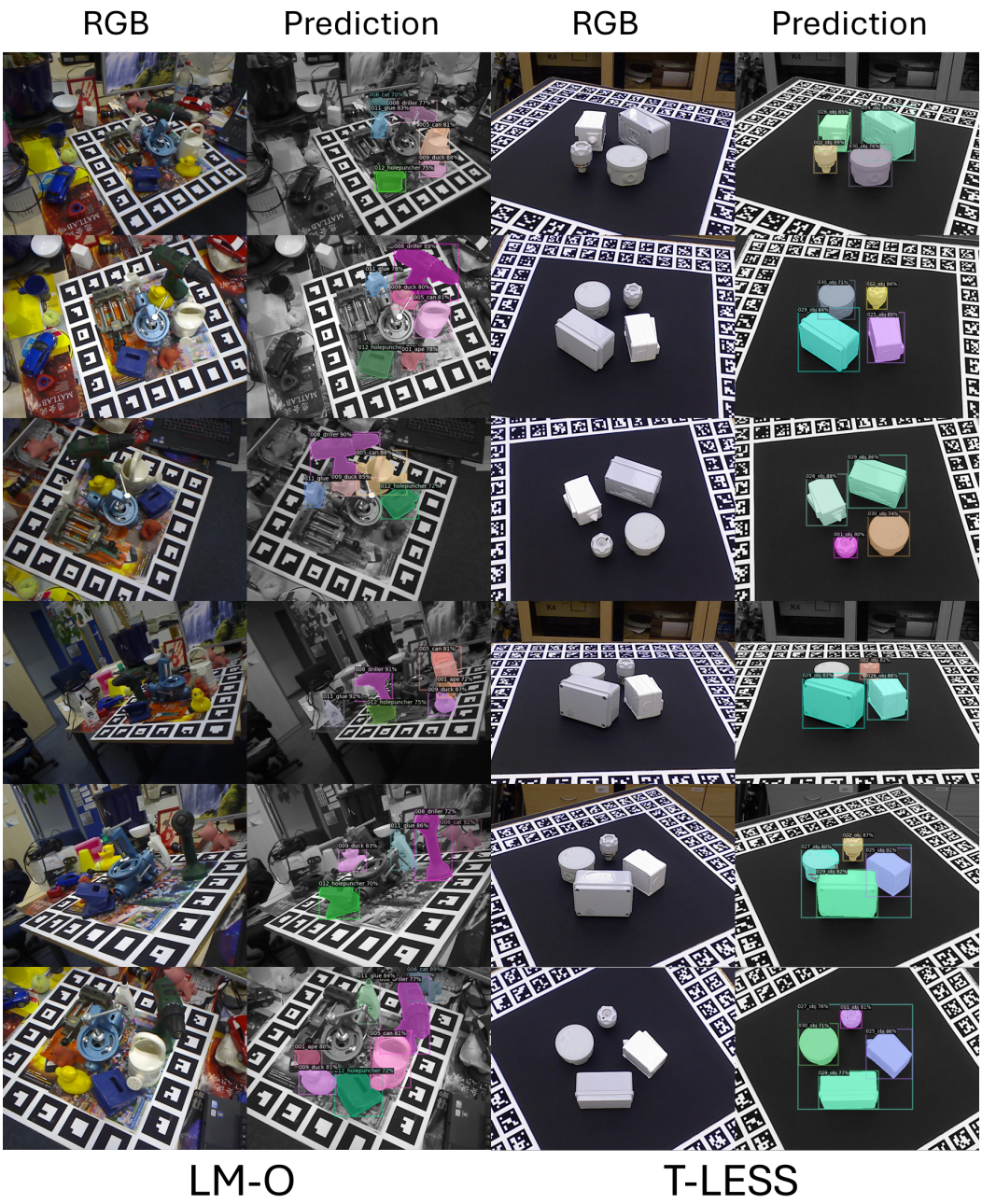

Fig. 5. Qualitative segmentation results on the LM-O and T-Less datasets.

Fig. 6. Qualitative segmentation results on the TUD-L and IC-BIN datasets.

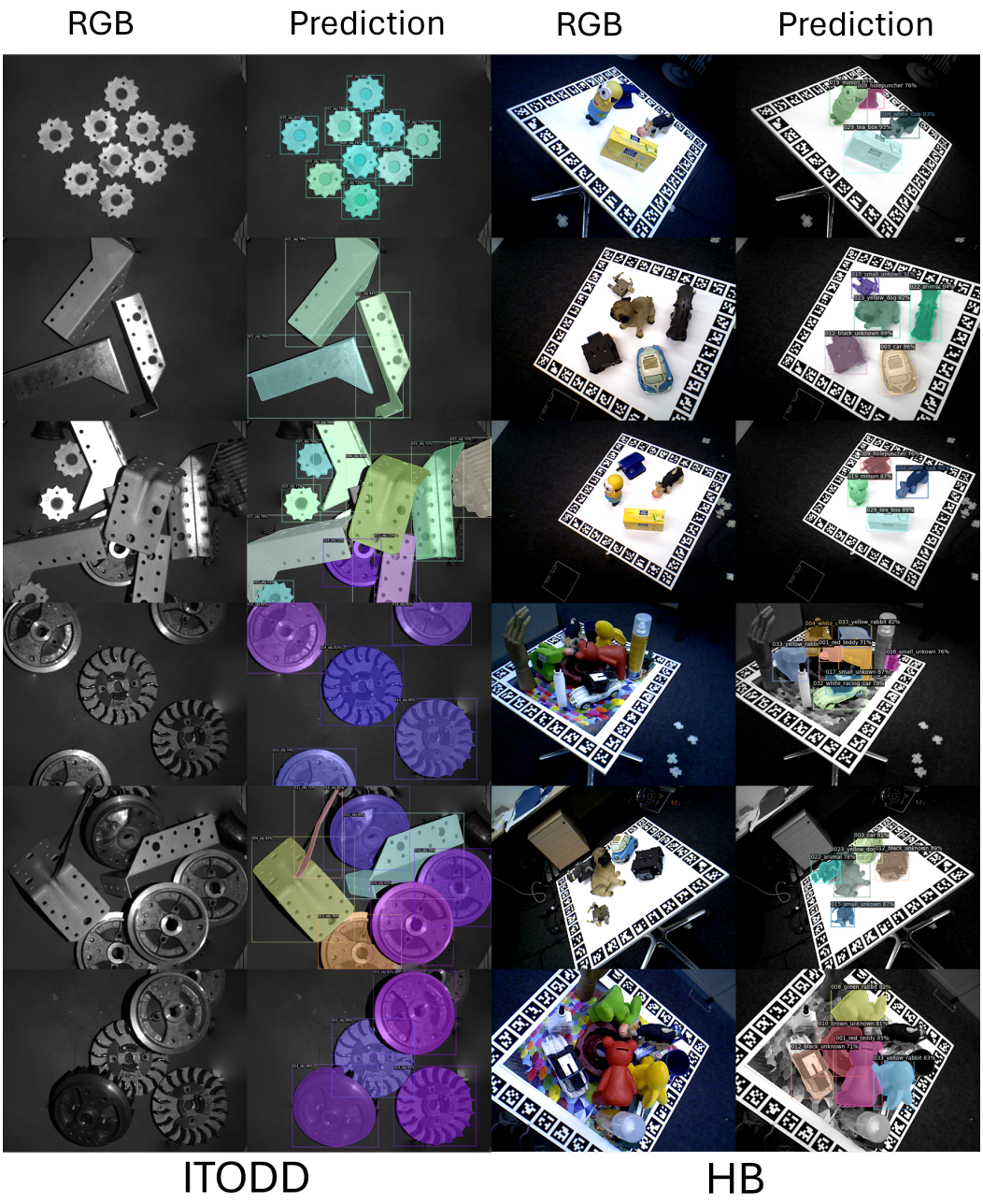

Fig. 7. Qualitative segmentation results on the ITODD and HB datasets

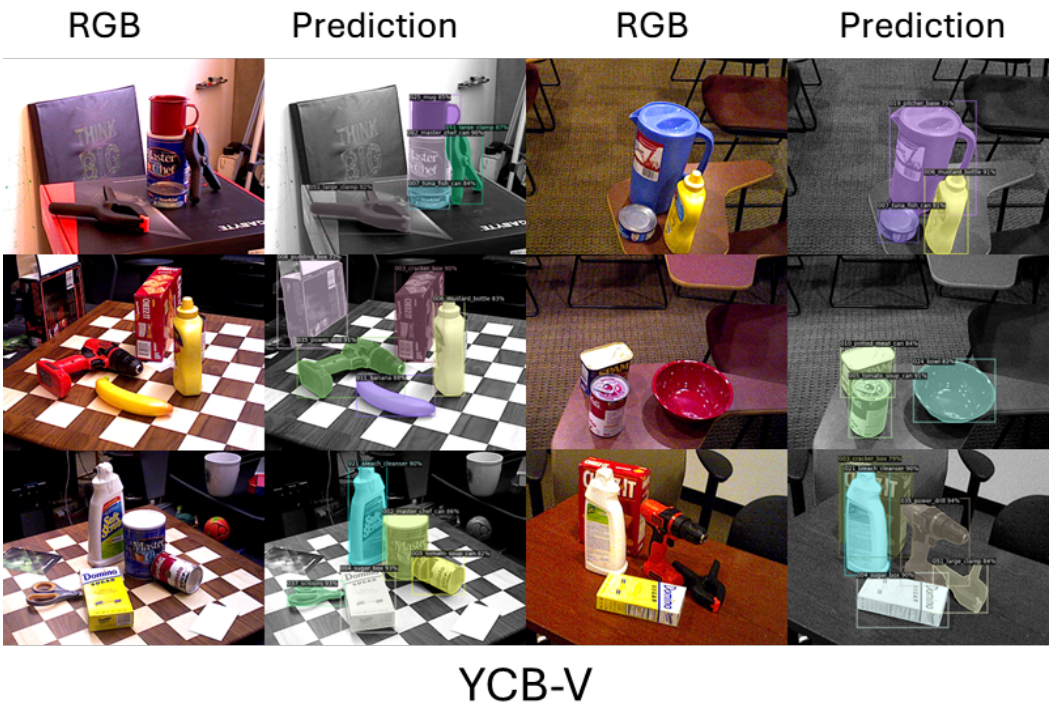

Fig. 8. Qualitative segmentation results on the YCB-V dataset.