RISeg: Robot Interactive Object Segmentation via Body Frame-Invariant Features

Howard H. Qian[1], Yangxiao Lu[2], Kejia Ren[1], Gaotian Wang[1], Ninad Khargonkar[2], Yu Xiang[2], Kaiyu Hang[1]

***Abstract*— In order to successfully perform manipulation tasks in new environments, such as grasping, robots must be proficient in segmenting unseen objects from the background and/or other objects. Previous works perform unseen object instance segmentation (UOIS) by training deep neural networks on large-scale data to learn RGB/RGB-D feature embeddings, where cluttered environments often result in inaccurate segmentations. We build upon these methods and introduce a novel approach to correct inaccurate segmentation, such as undersegmentation, of static image-based UOIS masks by using robot interaction and a designed body frame-invariant feature. We demonstrate that the relative linear and rotational velocities of frames randomly attached to rigid bodies due to robot interactions can be used to identify objects and accumulate corrected object-level segmentation masks. By introducing motion to regions of segmentation uncertainty, we are able to drastically improve segmentation accuracy in an uncertainty-driven manner with minimal, non-disruptive interactions (*ca.* 2-3 per scene). We demonstrate the effectiveness of our proposed interactive perception pipeline in accurately segmenting cluttered scenes by achieving an average object segmentation accuracy rate of 80.7%, an increase of 28.2% when compared with other state-of-the-art UOIS methods.**

I. Introduction

In order to perform autonomous manipulation tasks, robots must be able to robustly perceive and segment unseen objects to gain an understanding of their environment. Thus, competent unseen object instance segmentation (UOIS) is imperative to a robot's manipulation capabilities [1]–[4].

While many state-of-the-art UOIS methods leverage deep neural networks to extract pixel-wise feature representations to perform segmentation, under and over segmentation in cluttered scenes remain a challenge [1], [5]. Because these methods attempt to segment single RGB-D images, only visual features are modeled while some essential physical features, such as how adjacent objects move relatively to one another, are not considered. Interactive perception is an alternative UOIS approach in which robots physically interact with the environment to accumulate information over time [6]. Under interactive perception, we should aim to gather the most sensory data from interactions with as little amount of scene disturbance as possible.

Central to the proposed method is our designed body frame-invariant feature (BFIF). Assuming there are two body frames rigidly attached to an object, we build our system on the insight that, when this object is moving, although the two body frames are rotating and translating differently in space, they will have the same spatial twist as observed by any reference frame fixed to the world [7]. Meanwhile, body frames on different objects that are relatively moving will typically have different spatial twists.

Fig. 1: Interactively segmenting a cluttered scene with minimal, non-disruptive pushes. [Top left] Initial scene and identified robot actions. [Top right] The origins of sampled body frames with matched BFIFs due to scene interactions, where matched body frames share the same color. [Bottom left] Undersegmentation of scene's end configuration by static segmentation model. [Bottom right] Accurate segmentation of scene by RISeg after interactions have been completed.

This work proposes the framework of Robot Interactive Segmentation (RISeg), which leverages active robot-object interactions and the BFIF to improve the performance of UOIS. Rather than learning visual features via data [1], we demonstrate that segmentation of complex, cluttered scenes can be drastically improved by observing object motions and grouping BFIFs throughout robot interactions (see Fig. 1). Singulation of objects at any step of robot interaction is not necessary for our method, which results in fewer pushes (*ca.* 2-3) and less disturbance to environments when compared to prior interactive perception methods [8].

II. Robot Interactive Object Segmentation

A. Framework Overview

Our proposed interactive perception framework makes 2 main contributions in action selection and segmentation mask correction. In Alg. 1, we describe a system in which the scene is observed between interactions to produce more accurate segmentation masks. After each action, a_t, is identified by FindAction$(\cdot)$ and completed by Interact$(\cdot)$, a segmentation mask, $\hat{L}_{t+1}$, is produced by UpdateMask$(\cdot)$ through BFIF analysis. Once the stop condition is met, the final segmentation mask $\hat{L}_{t+1}$ is returned which reflects a more accurate segmentation of the scene's end configuration after all interactions.

[1]Department of Computer Science, Rice University, Houston, TX 77005, USA. HQ, KR, GW, and KH are supported by the US National Science Foundation grant FRR-2133110. [2]Department of Computer Science, University of Texas at Dallas, Richardson, TX 75080, USA. YL, NK and YX are supported by the DARPA Perceptually-enabled Task Guidance (PTG) Program under contract number HR00112220005 and the Sony Research Award Program.

Algorithm 1 RISeg

Input: I_0, StaticSeg$(\cdot)$
Output: $\hat{L}_{t+1}$

1: $t \leftarrow 0$
2: $L_t \leftarrow$ StaticSeg(I_t)
3: $\hat{L}_t \leftarrow L_t$
4: **while** $a_t \leftarrow$ FindAction(I_t) **not** *null* **do** ▷ Alg. 2
5: $\quad I_{t+1} \leftarrow$ Interact(a_t)
6: $\quad L_{t+1} \leftarrow$ StaticSeg(I_{t+1})
7: $\quad \hat{L}_{t+1} \leftarrow$ UpdateMask$(I_t, I_{t+1}, \hat{L}_t, L_{t+1})$ ▷ Alg. 3
8: $\quad t \leftarrow t+1$
9: **return** $\hat{L}_{t+1}$

B. Body Frame-Invariant Feature

The proposed RISeg method is an interactive perception method in which a designed body frame-invariant feature (BFIF) of sampled frames within a scene are grouped with one another based on computed feature similarities. BFIF is based on the spatial twists of body frames attached to various rigid bodies. The key point being that twists of moving body frames on the same rigid body transformed into a fixed space frame will all have the same spatial twist, no matter their relative motion [7].

C. Action Selection

As detailed in Alg. 2, we introduce a heuristic-based approach to finding minimal, non-disruptive robot actions. Given an RGB-D image I_t, static segmentation model MSMFormer$(\cdot)$ [5] returns segmentation mask L_t and uncertainty heatmap U_t. Heatmap U_t gives pixel-wise confidence values for each pixel belonging to an object, where pixels with larger values are more likely to belong to an object. In lines 2 and 3 of Alg. 2, we use heatmap U_t to identify cluster centers via k-means clustering for pixels we are "certain" (superscript c) to be part of an object, $\{C_m^c\}_{m=1}^M$ as well as cluster centers for pixels we are "uncertain" (superscript u) to be part of an object, $\{C_n^u\}_{n=1}^N$.

Algorithm 2 FindAction

Input: I_t
Output: a_t

1: $L_t, U_t \leftarrow$ MSMFormer(I_t)
2: $\{C_m^c\}_{m=1}^M \leftarrow$ KMeans$(U_t^{i,j} \in U_t : \ell_u \leq U_t^{i,j})$
3: $\{C_n^u\}_{n=1}^N \leftarrow$ KMeans$(U_t^{i,j} \in U_t : \ell_l \leq U_t^{i,j} < \ell_u)$
4: $(i^*, j^*) \leftarrow \underset{(i,j)\in\{1,\dots,M\}}{\arg\min} \|C_i^c - C_j^c\|$
$\quad$ s.t. $i \neq j$,
$\quad\quad \|C_i^c - C_j^c\| \leq d_a$,
$\quad\quad \min_{n\in\{1,\dots,N\}} \text{Dist}(C_n^u, \overline{C_i^c C_j^c}\}) \leq d_b$
5: **if** (i^*, j^*) exists **then**
6: $\quad \{P_{i^*}\} \leftarrow$ Boundary$(C_{i^*}^c)$
7: $\quad P^* \leftarrow$ Rand$(\{P_i \in \{P_{i^*}\} : \overline{P_i C_{i^*}^c} \perp \overline{C_{i^*}^c C_{j^*}^c}\})$
8: $\quad a_t \leftarrow (P^*, \overrightarrow{P^* C_{i^*}^c}, d_{push})$
9: $\quad$ **return** a_t
10: **else**
11: $\quad$ **return** *null*

Fig. 2 shows how a specific robot action is selected after obtaining the "certain" and "uncertain" clusters from uncertainty heatmap U_t. In line 4 of Alg. 2, we describe consideration of all pairs (i, j) of cluster centers in $\{C_m^c\}$ where $i \neq j$ and the distance between C_i^c and C_j^c is less than some distance d_a. For each (C_i^c, C_j^c) pair under consideration, we construct a line segment connecting the cluster center pair, and select the pair of interest $(C_{i^*}^c, C_{j^*}^c)$ for which an "uncertain" cluster center C_n^u is closest to. The distance between "uncertain" cluster center C_n^u and line segment $\overline{C_i^c C_j^c}$ must be at most d_b. If no "certain" cluster centers $(C_{i^*}^c, C_{j^*}^c)$ exist to satisfy these constraints, then a *null* action will be returned.

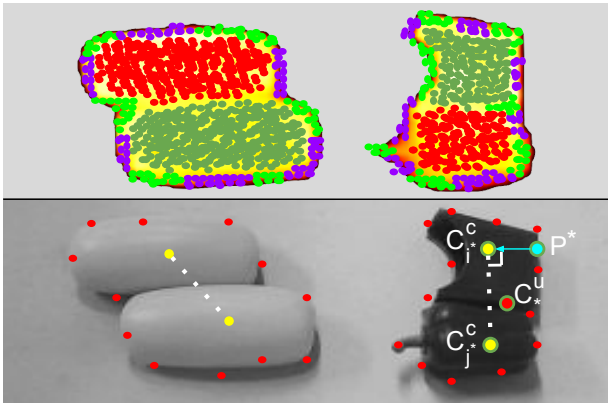

Fig. 2: Visualization of FindAction$(\cdot)$. [Top] "Certain" clusters shown in red and dark green. "Uncertain" clusters shown in purple and light green. [Bottom] "Certain" cluster centers (C_m^c) are shown in yellow. White, dashed line segments connect "certain" cluster centers $(\overline{C_i^c C_j^c})$. "Uncertain" cluster centers (C_n^u) are shown in red. Action a_t, defined by chosen push point P^* and direction $\overrightarrow{P^* C_{i^*}^c}$, is shown in blue. "Uncertain" cluster center C_*^u is used to choose $C_{i^*}^c$ and $C_{j^*}^c$ due to having minimum distance to $\overline{C_{i^*}^c C_{j^*}^c}$.

Algorithm 3 UpdateMask

Input: $I_t, I_{t+1}, \hat{L}_t, L_{t+1}$
Output: $\hat{L}_{t+1}$

1: $O_t \leftarrow$ RAFT(I_t, I_{t+1}) ▷ Optical Flow
2: $\{F_t^i\}, \{F_{t+1}^i\} \leftarrow$ CreateFrames$(\hat{L}_t, O_t)$
3: $\{\mathcal{V}_t^i\} \leftarrow$ CalcBFIFs$(\{F_t^i\}, \{F_{t+1}^i\})$
4: $FG_t \leftarrow$ GroupBFIFs$(\{\mathcal{V}_t^i\}, \hat{L}_t)$
5: $\hat{L}_{t+1} \leftarrow$ CorrectMask$(FG_t, \hat{L}_t, L_{t+1}, O_t)$
6: **return** $\hat{L}_{t+1}$

With a valid $(C_{i^*}^c, C_{j^*}^c)$, action a_t is identified by selecting a push point P^* and direction (see Fig 2). Push point P^* is chosen by first obtaining pixels $\{P_{i^*}\}$ from the cluster boundary of cluster center $C_{i^*}^c$ via Boundary$(\cdot)$. Then, a point P^* that forms a line segment $\overline{P^* C_{i^*}^c}$ perpendicular to line segment $\overline{C_{i^*}^c C_{j^*}^c}$ is chosen at random via line 7 of Alg. 2. Action a_t is now defined as a push from point P^* in direction $\overrightarrow{P^* C_{i^*}^c}$ for short constant distance d_{push}. Once action a_t is executed, and new image I_{t+1} and segmentation mask L_{t+1} are captured.

D. Segmentation Mask Correction

1) Sample Body Frames and Compute BFIFs: Since a main motivation of our method is to improve segmentation through non-disruptive interactions, mask L_{t+1} is likely to have similar segmentation inaccuracies as L_t, such as under segmentation. In Alg. 3, we describe how even without object singulation in I_{t+1}, we are able to produce a more accurate, refined segmentation mask $\hat{L}_{t+1}$ for the current scene state.

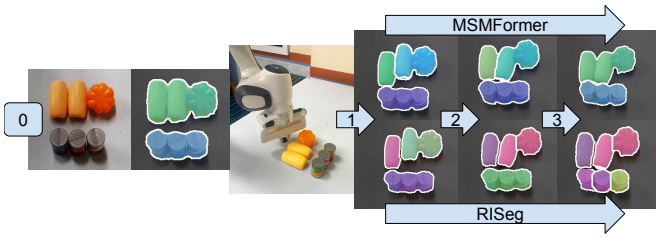

Fig. 3: RISeg and MSMFormer segmentations of a cluttered tabletop scene throughout the interactive perception pipeline. The scene's initial state is shown after label "0". Scene configurations and segmentation masks after push numbers 1, 2, and 3 follow the corresponding arrows. Pushes are minimal and non-disruptive.

To track motions caused by robot interactions, we use an optical flow model RAFT$(\cdot)$ [9]. To compute the BFIFs of objects between scene images I_t and I_{t+1}, we must create body frames attached to rigid bodies in I_t and track their motion through to I_{t+1}, using optical flow O_t. CREATEFRAMES$(\cdot)$ thus creates body frames $\{F_t^i\}$ from I_t and a corresponding set of body frames $\{F_{t+1}^i\}$ from I_{t+1}.

We then compute a set of BFIFs $\{\mathcal{V}_t^i\}$ represented in the space frame $\{s\}$ in CALCBFIFS$(\cdot)$. Remember that BFIFs in $\{\mathcal{V}_t^i\}$ will theoretically be equal if they belong to body frames on the same rigid body. Then, GROUPBFIFS$(\cdot)$ creates groups of body frames FG_t that share similar BFIFs while statistically filtering out noise from optical flow O_t.

2) Segmentation Mask Correction: Once we have identified body frame groups FG_t, we can correct segmentation inaccuracies in L_{t+1}, via line 5 of Alg. 3 CORRECTMASK$(\cdot)$, and return $\hat{L}_{t+1}$. To do so, we first project $\hat{L}_t$ object segmentations onto corresponding objects in $\hat{L}_{t+1}$, and then use the grouped body frames FG_t with similar BFIFs to correct $\hat{L}_{t+1}$.

By using the most recent RISeg segmentation mask $\hat{L}_t$ as an accumulation of previous mask corrections, we first bring the current RISeg mask $\hat{L}_{t+1}$ to the same level of segmentation accuracy as $\hat{L}_t$, which will reflect the information gained from all previous interactions $a_{t-1}, a_{t-2}, \ldots$. Once $\hat{L}_{t+1}$ reflects the segmentation masks of $\hat{L}_t$ by using O_t, we can use the grouped body frames FG_t to correct $\hat{L}_{t+1}$, which will reflect the information gained from interaction a_t.

Each set $fg_i \in FG_t$ represents a group of body frames identified to have the same BFIF. Therefore, each body frame in set fg_i should be segmented as part of the same object with object ID ℓ_i. For each body frame in fg_i, we reassign its corresponding pixel in $\hat{L}_{t+1}$ to ℓ_i, along with similarly moving neighboring points via Breadth First Search.

III. EXPERIMENT AND CONCLUSION

A. Implementation and Dataset

Experiment objects are placed on a flat, white tabletop and come from a set of play food toys for kids due to similarity in shape and color to one another. These objects are particularly difficult to segment in cluttered environments. Because there is no standard interactive perception dataset, we manually evaluate our proposed pipeline by creating 23 tabletop scenes in which 4-6 objects are placed in close proximity to one another, often touching.

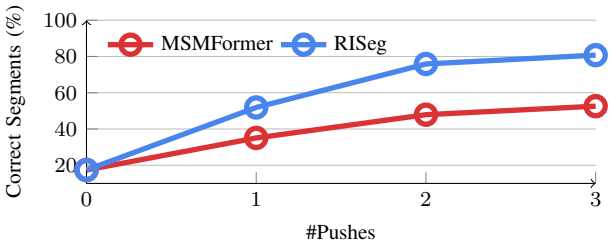

Fig. 4: Percentage of objects correctly segmented as measured by the Overlap F-measure $\geq 75\%$.

Method	Push #	Overlap			Boundary		
		P	R	F	P	R	F
MSMFormer [5]	0	53.7	55.4	52.3	44.6	50.6	40.0
	1	66.6	62.4	64.3	62.1	52.4	56.8
	2	72.8	68.6	70.5	69.0	61.1	64.7
	3	73.2	67.6	70.1	70.0	62.5	65.9
RISeg	0	53.7	55.4	52.3	44.6	50.6	40.0
	1	74.1	69.6	71.6	69.0	61.5	64.9
	2	85.8	81.1	83.3	79.4	76.0	77.6
	3	**88.1**	**79.6**	**83.3**	**82.4**	**77.4**	**79.6**

Table 1. Segmentation results of MSMFormer and RISeg across scene configurations resulting from robot actions.

B. Evaluation Metrics

For each scene, we evaluate the segmentation accuracy at each scene configuration, using precision, recall and F-measure [1], [10]. Fig. 4 shows the percentage of objects segmented with a high accuracy throughout scene configurations, which is the percentage of segmented objects with Overlap F-measure $\geq 75\%$. Fig 3 shows a qualitative comparison of segmentation results between MSMFormer and RISeg.

C. Discussion of Results

In Table I and Fig. 4, we compare segmentation results of our RISeg method with state-of-the-art UOIS model MSMFormer. Push 0 indicates the scene's initial configuration, in which both methods have the same segmentation results because RISeg uses MSMFormer for base segmentation masks. Each push number indicates average segmentation statistics across all scenes after that numbered interaction has been completed, regardless of total number of pushes for each individual scene. With each robot-scene interaction, both methods see object segmentation accuracy increases for all metrics, though to different degrees. On average, MSMFormer object segmentation accuracy increases because some object singulation results from interactions. However, RISeg object segmentation accuracy increases drastically faster and sees a higher peak when compared to MSMFormer because analysis of BFIFs results in robust segmentations even with minimal object displacements and no object singulation. After all robot interactions, RISeg is able to accurately segment 80.7% of objects in the scene's end configuration while MSMFormer is still only able to segment 52.5% of objects. Overlap and Boundary P/R/F metrics also increase with each robot interaction. Overlap precision metrics peak after interaction number 3 is completed, with 88.1% for RISeg and 73.2% for MSMFormer.

REFERENCES

[1] Y. Xiang, C. Xie, A. Mousavian, and D. Fox, "Learning rgb-d feature embeddings for unseen object instance segmentation," in *Conference on Robot Learning*. PMLR, 2021, pp. 461–470.

[2] 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 *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 5085–5092.

[3] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "Unseen object instance segmentation for robotic environments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.

[4] M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, and K. Goldberg, "Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic point clouds," *arXiv preprint arXiv:1809.05825*, vol. 16, 2018.

[5] Y. Lu, Y. Chen, N. Ruozzi, and Y. Xiang, "Mean shift mask transformer for unseen object instance segmentation," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024.

[6] J. Kenney, T. Buckley, and O. Brock, "Interactive segmentation for manipulation in unstructured environments," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2009, pp. 1377–1382.

[7] K. M. Lynch and F. C. Park, *Modern robotics*. Cambridge University Press, 2017.

[8] Y. Lu, N. Khargonkar, Z. Xu, C. Averill, K. Palanisamy, K. Hang, Y. Guo, N. Ruozzi, and Y. Xiang, "Self-supervised unseen object instance segmentation via long-term robot interaction," in *Robotics: Science and Systems*, 2023.

[9] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 402–419.

[10] 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.