High-Quality Unknown Object Instance Segmentation via Quadruple Boundary Error Refinement

Seunghyeok Back[1],Sangbeom Lee[2],Kangmin Kim[2],Joosoon Lee[2],Sungho Shin[3],Jemo Maeng[2],Kyoobin Lee[2†]

***Abstract*— Accurate and efficient segmentation of unknown objects in unstructured environments is essential for robotic manipulation. Unknown Object Instance Segmentation (UOIS), which aims to identify all objects in unknown categories and backgrounds, has become a key capability for various robotic tasks. However, existing methods struggle with over-segmentation and under-segmentation, leading to failures in manipulation tasks such as grasping. To address these challenges, we propose QuBER (Quadruple Boundary Error Refinement), a novel error-informed refinement approach for high-quality UOIS. QuBER first estimates quadruple boundary errors—true positive, true negative, false positive, and false negative pixels—at the instance boundaries of the initial segmentation. It then refines the segmentation using an error-guided fusion mechanism, effectively correcting both fine-grained and instance-level segmentation errors. Extensive evaluations on three public benchmarks demonstrate that QuBER outperforms state-of-the-art methods and consistently improves various UOIS methods while maintaining a fast inference time of less than 0.1 seconds. Furthermore, we show that QuBER improves the success rate of grasping target objects in cluttered environments. Code and supplementary materials are available at `https://sites.google.com/view/uois-quber`.**

I. Introduction

The ability to perceive and manipulate unknown objects in cluttered environments is crucial for robotics and embodied AI. At the core of this challenge is Unknown Object Instance Segmentation (UOIS) [1], [2], [3], which enables robots to identify and interact with novel objects in unstructured settings. UOIS aims to detect all object instances in unknown categories and backgrounds and has become fundamental to various robotic manipulation tasks, including grasping, pushing, and rearrangement [4], [5], [6]. The quality of UOIS directly impacts task success, as inaccurate segmentation often leads to failures in subsequent robotic operations.

State-of-the-art UOIS methods [1], [2], [7], [3], [8], [9] directly predict segmentation by leveraging both texture and geometric cues from RGB-D images. Although promising, these methods often struggle in complex, cluttered scenes with significant occlusions, leading to instance-level errors such as over-segmentation (splitting a single object) or under-segmentation (merging multiple objects). Segmentation refinement methods [10], [11], [12], [13], [14] improve local details in boundaries but typically assume instance-level correctness, limiting their ability to rectify major errors. Recent promptable networks, such as Segment Anything Model (SAM) [15], [16], [17], trained on web-scale datasets to achieve strong generalizability, show promise for both initial segmentation and refinement. However, in real-world robotic cluttered scenes, they often over-segment [18], [19] or refine only minor details without accurate human prompts. Additionally, these methods are computationally intensive, making them less suitable for real-time robotic applications.

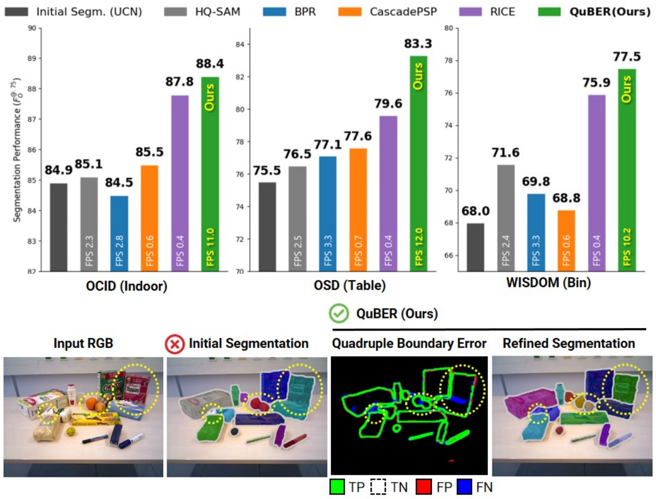


Fig. 1: (Top) Results of the proposed QuBER method for high-quality UOIS across various domains. (Bottom) From the initial segmentation, QuBER performs error-informed refinement by estimating pixel-wise quadruple boundary errors and refining the segmentation based on these error estimates.

In this paper, we propose **Qu**adruple **B**oundary **E**rror **R**efinement (**QuBER**), a novel model for high-quality UOIS using an error-informed refinement strategy (Fig. 1). QuBER estimates quadruple boundary errors - true positive (TP), true negative (TN), false positive (FP), and false negative (FN) pixels at instance boundaries. Our Error Guidance Fusion (EGF) module then incorporates these estimated errors to refine the segmentation accurately. Notably, quadruple boundary error estimation effectively captures both fine-grained and instance-level errors, providing explicit error correction

[1] S. Back is with the Department of AI Machinery, Korea Institute of Machinery & Materials (KIMM), Daejeon 34103, Republic of Korea.

[2] S. Lee, K. Kim, J. Lee, J. Maeng, and K. Lee are with the Department of AI Convergence, Gwangju Institute of Science and Technology (GIST), Gwangju 61005, Republic of Korea.

[3] S. Shin is with the Robotics Lab, Hyundai Motor Company, Uiwang 16082, Republic of Korea.

S. Back and S. Shin were with GIST at the time of the initial submission.

[†] Corresponding author: Kyoobin Lee `kyoobinlee@gist.ac.kr`

information (e.g., FP indicates over-segmented regions to be deleted, FN indicates under-segmented regions to be added). Unlike existing methods that focus solely on local refinements or require computationally expensive operations, our error-informed approach efficiently corrects segmentation errors with a fast inference time (∼0.1s on RTX3090, Intel6248R). Extensive experiments demonstrate that QuBER outperforms state-of-the-art UOIS and refinement methods on three benchmarks by resolving over-segmentation and under-segmentation issues across various initial segmentation methods. We also demonstrate the practical robot application of QuBER by improving target object grasping success.

The main contributions of this study are as follows: (1) We introduce QuBER, an error-informed refinement network for high-quality UOIS. (2) We propose quadruple boundary error estimation to capture and refine both fine-grained and instance-level errors. (3) We propose an Error Guidance Fusion (EGF) module for effectively integrating estimated errors into the refinement process.

II. Related Work

A. Unknown Object Instance Segmentation

Unknown Object Instance Segmentation (UOIS) [1], [2], [3] aims to detect all arbitrary object instances in images with unknown objects and environments, serving as a fundamental perception module for robotic manipulation [4], [5], [6]. Early approaches relied on clustering techniques [20], [21], while recent state-of-the-art methods employ deep networks trained on large-scale synthetic RGB-D data to learn category-agnostic objectness [1], [2], [22], [7], [3], [8], [9]. Prompt-based segmentation models like SAM, trained on web-scale data [15], [16], [17], show promise but suffer from over-segmentation without accurate prompts [18], [19] and incur high computational costs during inference. Despite these advancements, high-quality object segmentation in cluttered environments remains challenging due to common issues like over-segmentation and under-segmentation [2], [7], [8], [9]. Our method addresses these issues by refining existing UOIS outputs, thereby enhancing segmentation quality for robust manipulation in complex scenes.

B. Refining Segmentation

Object segmentation quality directly impacts robotic manipulation tasks like grasping [5]. Conditional random fields [23], [24] refine segmentation by modeling spatial relationships but struggle with semantic information and large error regions. Recent methods [25], [26] focus on fine-grained improvements: Segfix [11] replaces unreliable boundary predictions with inner predictions, Boundary Patch Refinement (BPR) [13] predicts boundary-aligned masks from patched coarse masks, and CascadePSP [12] adopts a cascade strategy for pixel-aligned refinement. However, these methods primarily address fine-grained boundary details without resolving instance-level over-segmentation and under-segmentation errors. SAM [15]-based methods such as HQ-SAM [16] show promise but are computationally expensive and focus mainly on fine details. RICE [14] tackles instance-level refinement through perturbation and sampling but requires about 4 seconds per frame. Test-time adaptation [27] improves domain generalization but needs 20 seconds for new scenarios. In contrast, our approach achieves state-of-the-art performance with fast refinement in less than 0.1 seconds per frame, efficiently addressing fine-grained and instance-level errors.

C. Error Detection for Segmentation and Refinement

Estimating errors in deep segmentation models is crucial for system reliability [28]. Most approaches focus solely on detecting binary errors (true, false), employing techniques such as maximum softmax probability [29], Monte Carlo dropout [30], or error segmentation [31], [32]. Only a few works leverage estimated errors for segmentation refinement. SESV [33] proposed a four-step framework (segmentation, evaluation, refinement, and verification), ERA [34] introduced an error-reversing autoencoder, and failure detection and label correction networks were employed in [35]. However, these methods require secondary networks for error detection and refinement, which incur substantial computational costs, and focus on refining fine-grained details by predicting binary errors in semantic segmentation. In contrast, our approach introduces an error-informed refinement method for instance segmentation that unifies error estimation and refinement within a single network. Additionally, we propose quadruple boundary error estimation (TP, TN, FP, FN) for both fine-grained and instance-level error correction.

III. Quadruple Boundary Error Refinement

In this paper, we propose QuBER (Fig. 3), an error-informed refinement network for high-quality UOIS that addresses fast and accurate refinement of over-segmentation and under-segmentation in complex, cluttered scenes.

A. Error-informed Refinement

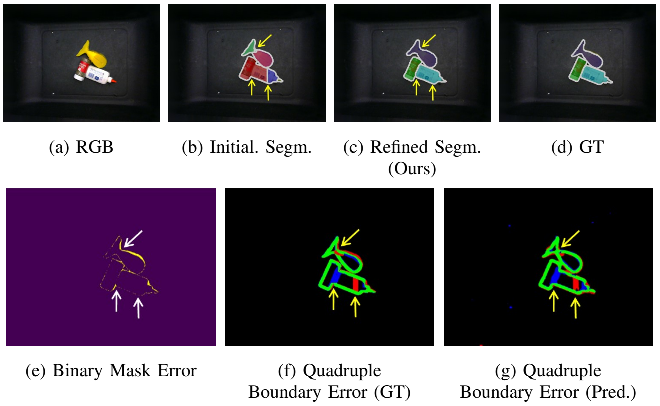
Our goal is to design a segmentation refinement model $\mathcal{G} : (I, M_i) \rightarrow M_r$ that produces refined, high-quality masks M_r for unknown object instances from an RGB-D image I and initial segmentation (IS) masks M_i of arbitrary initial segmentation models. To achieve this, we introduce an error-informed refinement process comprising the following steps:

- **IS feature extractor** $\mathcal{F} : (I, M_i) \rightarrow h$ to obtain initial segmentation features h from the input RGB-D image I and coarse masks M_i.
- **Error estimator** $\mathcal{E} : h \rightarrow \hat{e}_i$ to predict segmentation errors $\hat{e}_i$ in the initial segmentation, providing critical error information to guide the refinement process.
- **Error-informed refiner** $\mathcal{H} : (h, \hat{e}_i) \rightarrow M_r$ to integrate predicted errors $\hat{e}_i$ with IS features to produce refined segmentation M_r, focusing on erroneous regions for accurate and targeted refinement.

The error-informed refinement process is formulated as:

$$\mathcal{G}(I, M_i) = \mathcal{H}(h, \mathcal{E}(h)) = M_r, \text{where } h = \mathcal{F}(I, M_i) \quad (1)$$

For both training and inference, these modules are integrated into a single QuBER network and trained jointly in an end-to-end manner. This unified approach enhances segmentation


(a) RGB

(b) Initial. Segm.

(c) Refined Segm. (Ours)

(d) GT

(e) Binary Mask Error

(f) Quadruple Boundary Error (GT)

(g) Quadruple Boundary Error (Pred.)

Fig. 2: Comparison of binary mask errors (🟪 True, 🟨 False, shown in (e)) and our quadruple boundary errors (🟩 TP, ⬚ TN, 🟥 FP, 🟦 FN, shown in (f) and (g)) to represent segmentation errors in the initial segmentation (b) for error estimation. The proposed quadruple boundary error estimation effectively captures instance-level errors and facilitates precise refined segmentation (c).

by explicitly estimating errors in the initial results, enabling targeted refinements for more accurate corrections. By sharing the IS feature extractor, the architecture reduces computational overhead, ensuring fast and efficient refinement.

B. Quadruple Boundary Error Estimation

We employ quadruple boundary errors as the segmentation error e_i in our error estimator (Fig. 2). Quadruple boundary errors are error pixel maps on instance boundaries with four categories: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). This approach can effectively capture both fine-grained and instance-level errors while providing clear refinement guidance. The computation of quadruple boundary errors involves three steps: First, we obtain the boundaries of all IS and GT instance masks using dilation. Next, we create an instance boundary map by taking the union of these boundaries. Finally, we evaluate pixel-wise TP, TN, FP, and FN errors between the IS and GT boundary maps, resulting in an error map shape of $w \times h \times 4$, where w and h are the width and height, respectively.

This quadruple boundary error estimation (Fig. 2f, and 2g) offers significant advantages over standard binary mask errors (true, false). By focusing on boundary errors, we effectively target the most critical areas for refinement, as object boundaries are where most segmentation errors occur. Additionally, it captures instance-level errors such as under-segmentation and over-segmentation in overlapping instances. Moreover, it provides specific guidance for refinement by clearly indicating accurately segmented (TP, TN), over-segmented (FP), or under-segmented (FN) pixels, leading to effective refinement, as demonstrated in Fig. 2c. In contrast, existing methods using binary mask errors (Fig. 2e) struggle to capture instance-level errors and provide limited guidance for refinement, indicating only correct or incorrect pixels, which possibly leads to ambiguity in the refinement.

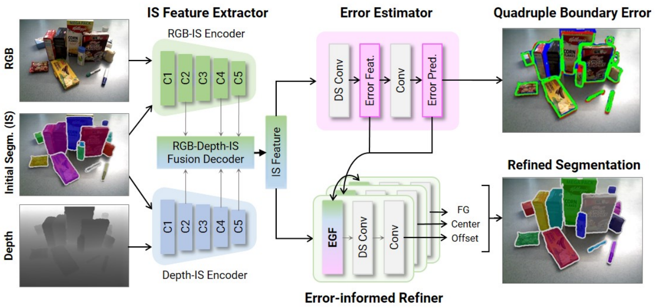
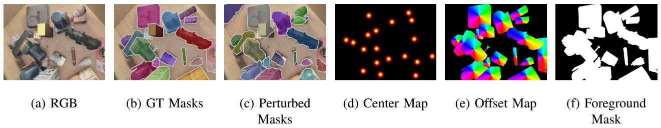


Fig. 3: Overview of QuBER for error-informed refinement.


(a) RGB

(b) GT Masks

(c) Perturbed Masks

(d) Center Map

(e) Offset Map

(f) Foreground Mask

Fig. 4: Examples of (a) RGB images, (b) ground truth (GT) masks, and (c) perturbed masks used during training. (d-f) the instance representations utilized in QuBER.

C. Network Architecture

QuBER comprises three main components (Fig. 3): an IS feature extractor $\mathcal{F}$, an error estimator $\mathcal{E}$, and an error-informed refiner $\mathcal{H}$. We introduce our error-informed refinement scheme with our Error Guidance Fusion (EGF) module, which explicitly incorporates estimated errors into the refinement process. While we implement our approach built on top of Panoptic-DeepLab [36] due to its simple lightweight design, the core principles of our error-informed refinement are not inherently tied to this specific architecture.

IS Representation. We represent the initial segmentation using three maps: 1) a center map (Fig. 4d) indicating the probability of each pixel being an instance center, 2) an offset map (Fig. 4e) with x and y directions from each pixel to its closest instance center, and 3) a binary foreground (FG) mask (Fig. 4f). These maps represent both spatial and relational information of the initial segmentation in a fixed-shape format, regardless of the number of instances.

IS Feature Extractor. We employ two parallel ResNet-50 backbones [37] as RGB-IS and Depth-IS encoders, processing the concatenation of IS with RGB and depth, respectively. The RGB-D-IS fusion decoder combines the RGB-IS and Depth-IS features at residual blocks $C2$, $C3$, and $C5$, using 1×1 and 3×3 convolutions (only 1×1 at $C5$ for efficiency). An atrous spatial pyramid pooling [24] then extracts multi-scale contextual information. Subsequently, the DeepLab decoder [24] produces IS features ($w \times h \times c$, where c is the channel dimension), encoding multi-scale texture and geometry information conditioned on IS.

Error Estimator. We implement a lightweight design for the error estimator to predict quadruple boundary errors in the IS. A single 5×5 depthwise separable (DS) convolution [38] first extracts error features ($w \times h \times c$). These features are then processed through 1×1 convolutions to predict quadruple boundary errors, guiding the subsequent error-informed refiner to focus on crucial error-containing regions.

Error-Informed Refiner: This component consists of separate branches for predicting foreground, center, and offset maps. We introduce a novel Error Guidance Fusion (EGF) module into each branch to integrate the predicted error estimates with the IS features. The EGF modules take error maps, error features, and IS features as inputs, combining them using a 1×1 convolution to reduce channels from $2c+4$ to c. This is followed by feature extraction using three 3×3 convolutions. This dense fusion of estimated quadruple boundary errors enables targeted and effective adjustments. Each branch then predicts its respective map using 5×5 depthwise separable and 1×1 convolutions. Following [36], post-processing groups foreground pixels with their nearest center to form the final, refined instance masks.

D. Implementation Details

Mask Perturbation. To enhance the generalization of our model across arbitrary UOIS models, we avoid using initial segmentations from existing models during training. Instead, we apply diverse perturbations to ground truth masks, simulating both fine-grained and instance-level segmentation errors. For fine-grained errors, we employ random contour subsampling, dilation, and erosion operations [12]. Instance-level errors are simulated through random mask removal and splitting of neighboring masks [14]. We also randomly add false positive instances using graph-based segmentation [20]. This approach generates a wide distribution of potential segmentation errors, enabling more robust training. Fig. 4c illustrates an example of our perturbed masks, demonstrating the diversity and realism of the simulated errors.

Training. QuBER, including its error estimator and error-informed refiner, is trained end-to-end. We employ a combination of loss comprising: standard dice loss for error estimation (L_{err}), cross-entropy loss for foreground segmentation (L_{fg}), MSE loss for center (L_{ctr}) [39], and L1 loss for offset (L_{off}) [40]. The total loss L is computed as follows:

$$L = \lambda_{err} L_{err} + \lambda_{fg} L_{fg} + \lambda_{ctr} L_{ctr} + \lambda_{off} L_{off} \qquad (2)$$

We set $\lambda_{err} = 1$, $\lambda_{fg} = 1$, $\lambda_{ctr} = 200$, and $\lambda_{off} = 0.01$.

Following standard UOIS protocols [8], [7], [14], [3], we train on synthetic data and evaluate on real images without fine-tuning. We use the UOAIS-SIM dataset [8], comprising 50k photorealistic and 100k non-photorealistic synthetic images (Fig. 4a). Training runs for 90k iterations using Adam optimizer [41] with a learning rate of 0.000125 and a batch size of 8. Color [42] and depth [43] augmentations are applied for sim-to-real transfer. We use a 640×480 resolution for both training and testing. Training takes approximately 21 hours on two RTX 3090 GPUs. Following [8], we use a lightweight foreground segmentation model [44] trained on TOD [2] to filter background instances (overlap ratio 0.3). Masks smaller than 500 pixels are removed [2], [3]. We use a ResNet-50 backbone pre-trained on ImageNet [45] and set the center threshold to 0.3 during post-processing.

IV. EXPERIMENTS

A. Comparison with State-of-the-Art Methods

We conducted experiments to (1) assess QuBER's effectiveness in refining segmentations from various UOIS models, (2) compare its performance with state-of-the-art refinement methods, and (3) evaluate its consistency across datasets and initial segmentation qualities. Comprehensive experiments were conducted using diverse initial segmentation models and refinement methods across multiple datasets.

Datasets. We evaluate our model on three widely used UOIS benchmark datasets of diverse real cluttered scenes: OCID [46], OSD [21], and WISDOM [1]. The OCID consists of 2,346 indoor images, including both tabletop and floor scenes, with semi-automated labels. It features an average of 7.5 objects per image, with a maximum of 20 objects. The OSD contains 111 images of tabletop scenes with human-annotated ground truths, having an average of 3.3 objects per image and a maximum of 15 objects. The WISDOM comprises 300 test images of bin scenes with human-annotated ground truths, with an average of 3.8 objects per image, and a maximum of 11 objects. Importantly, the test objects in these datasets do not overlap with those in the training set, allowing for the evaluation on unknown objects.

Metrics. For performance evaluation, we employ standard UOIS metrics [14]: object size normalized (OSN) precision, recall, and F-measure for both overlap (P_O, R_O, F_O) and boundaries (P_B, R_B, F_B). These metrics evaluate segmentation performance over masks and boundaries, accounting fairly for objects of all sizes. Additionally, the percentage of objects with overlap F-measure greater than 0.75 ($F_O^{@.75}$), which evaluates instance-level segmentation accuracy.

Initial Segmentation Models. We employed five state-of-the-art UOIS methods for initial segmentation: Grounded-SAM [17], UOIS-Net-3D [3], UOAIS-Net [8], MSMFormer [9], and UCN [7] (the latter two include zoom refinement), using their official implementations. Grounded-SAM, built on SAM, segments objects from RGB images using a given vocabulary; we used a fixed query prompt (a rigid object) following [19]. The other models are category-agnostic UOIS models trained on RGB-D images. We excluded SAM [15] as an initial segmentation model due to severe over-segmentation without manual prompts ($F_O^{@.75} = 8.5$, OSD).

Segmentation Refinement Baselines. We compared QuBER with the following state-of-the-art models:

- BPR [13]: Crops patches on the boundary of IS masks and refines them using an encoder-decoder network.
- CascadePSP [12]: Refines individual IS masks toward fine-grained masks in a coarse-to-fine manner.
- RICE [14]: An instance-level refinement network that samples instance-level perturbations and selects the best segmentation using a graph neural network.
- SAM [15]: A promptable zero-shot segmentation model trained on web-scale datasets (1 billion masks).
- HQ-SAM [16]: An enhanced SAM for high-quality segmentation, with additional high-quality token training.
- HQ-SAM† [16]: A fine-tuned HQ-SAM for UOIS tasks.

TABLE I: Performance comparison of QuBER and state-of-the-art methods on OCID using various initial segmentation

Method	Time (ms)	FLOPs (G)	Grounded-SAM [17]			UOAIS-Net [8]			UOIS-Net-3D [3]			MSMFormer[9]			UCN[7]		
			F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$
Initial Segm.			65.1	64.9	64.5	66.7	64.4	66.5	77.4	74.2	76.3	81.9	81.3	81.6	84.1	83.0	84.9
+ BPR [13]	351	1722	64.6	60.0	65.2	65.2	61.5	64.8	76.9	70.9	75.6	80.1	74.2	80.8	83.2	77.5	84.5
+ CascadePSP [12]	1735	40242	64.9	63.3	63.9	66.6	64.9	66.5	78.0	76.0	76.8	81.4	80.6	81.9	84.1	83.4	85.5
+ RICE [14]	2349	972	72.1	**70.4**	72.5	69.4	66.6	70.0	82.9	79.1	84.3	84.8	83.4	85.7	86.4	**84.8**	87.8
+ SAM [15]	519	5487	60.2	61.5	59.4	60.2	60.4	59.8	74.5	74.4	73.4	77.3	77.9	77.7	78.7	79.4	79.6
+ HQ-SAM [16]	436	5520	62.6	64.1	61.6	65.6	65.5	65.0	78.2	77.2	77.0	81.2	81.5	81.6	84.1	84.7	85.1
+ HQ-SAM† [16]	426	5519	63.9	65.1	63.0	65.8	65.5	65.3	78.3	77.3	77.1	81.3	81.6	81.8	84.5	85.1	85.6
+ QuBER (Ours)	**91**	**431**	**74.1**	67.4	**73.4**	**77.8**	**75.1**	**76.4**	**84.5**	**81.9**	**85.2**	**86.2**	**83.8**	**87.7**	**86.6**	84.3	**88.4**

TABLE II: Performance comparison on OSD dataset

Method	UOAIS-Net [8]			MSMFormer [9]			UCN [7]		
	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$
Initial Segm.	75.7	68.6	71.5	77.0	63.4	75.7	76.4	64.7	75.5
+ BPR [13]	74.7	67.6	71.3	77.6	69.3	75.9	78.7	69.9	77.1
+ CascadePSP [12]	76.3	72.3	71.6	78.8	72.5	76.2	79.3	72.8	77.6
+ RICE [14]	77.6	69.6	74.5	79.3	64.3	79.2	79.9	67.5	79.6
+ SAM [15]	70.7	69.7	66.5	75.7	74.8	71.7	72.4	72.0	69.8
+ HQ-SAM [16]	75.5	73.8	71.3	78.1	75.6	74.5	79.0	76.8	76.5
+ HQ-SAM† [16]	75.6	73.2	70.5	78.9	75.7	76.2	80.2	77.5	76.9
+ QuBER (Ours)	**81.4**	**74.8**	**78.8**	**81.4**	**73.9**	**79.7**	**83.8**	**76.3**	**83.3**

TABLE III: Performance comparison on the WISDOM

Method	UOAIS-Net [8]			MSMFormer [9]			UCN [7]		
	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$	F_O	F_B	$F_O^{@.75}$
Initial Segm.	72.8	65.7	73.3	75.3	67.5	76.8	67.8	60.0	68.0
+ BPR [13]	78.4	71.0	79.8	77.0	69.4	78.9	69.9	63.2	69.8
+ CascadePSP [12]	78.4	**71.3**	79.0	77.4	70.2	78.7	68.9	61.9	68.8
+ RICE [14]	78.4	70.7	79.5	78.1	69.2	80.2	75.4	66.0	75.9
+ SAM [15]	67.1	63.5	67.7	70.0	66.6	70.9	63.9	60.5	63.9
+ HQ-SAM [16]	74.3	70.1	74.8	77.2	72.6	77.9	71.5	67.0	71.6
+ HQ-SAM† [16]	74.3	70.1	74.9	77.4	**72.8**	78.4	71.8	67.1	72.2
+ QuBER (Ours)	**78.5**	70.9	**80.8**	**79.7**	71.7	**81.9**	**76.4**	**68.5**	**77.5**

TABLE IV: Ablation of error-informed refinement on OCID (PDL: Panoptic-DeepLab)

Method	segm. refine.	error estim.	EGF	F_O	F_B	$F_O^{@.75}$
UCN [7]	✗	✗	✗	84.1	83.0	84.9
PDL [36]	✗	✗	✗	80.4 (-3.7)	71.3 (-11.7)	76.4 (-8.5)
UCN [7] + PDL [36]	✓	✗	✗	82.2 (-1.9)	77.2 (-5.8)	81.0 (-3.9)
UCN [7] + **QuBER**	✓	✓	✗	84.7 (+0.6)	81.3 (-1.7)	85.3 (+0.4)
UCN [7] + **QuBER**	✓	✓	✓	**86.1** **(+2.0)**	**83.7** **(+0.7)**	**87.6** **(+2.6)**

To ensure a fair comparison, we used official implementations and evaluated each method's best setup for optimal performance. For BPR and CascadePSP, we trained them on UOAIS-SIM with RGB-D inputs. Official pre-trained weights were used for RICE, SAM, and HQ-SAM. For SAM, HQ-SAM, and HQ-SAM†, the ViT-H [47] backbone was used with the box and mask prompts from initial segmentation. HQ-SAM† was fine-tuned on UOAIS-SIM using the token learning approach proposed in the original HQ-SAM paper, starting from the official pre-trained SAM weights.

Results. Table I compares the UOIS performance on the OCID (indoor) dataset of QuBER with state-of-the-art segmentation refinement models over various UOIS models. Tables II and III show results on the OSD (tabletop) and WISDOM (bin) datasets. Across these diverse scenarios, QuBER consistently improves various initial segmentation models and achieves superior performance over all refinement methods. In particular, QuBER achieved superior $F_O^{@.75}$ performance, effectively resolving over-segmentation and under-segmentation issues with error-informed refinement.

We evaluated computational efficiency in Table I using UCN [7] as the IS model. We measured average refinement times and FLOPs from model forward to post-processing on an RTX 3090 and an Intel Xeon Gold 6248R. QuBER demonstrated superior efficiency with inference times under 0.1 seconds and the lowest FLOPs, enabling efficient UOIS refinement. While RICE showed more competitive performance than others due to its instance-level error refinement, its inference time was significantly longer than ours.

Fig. 5 shows sample qualitative evaluations, and Fig. 6 provides a comparison with other state-of-the-art refinement methods using initial segmentation of UCN [7]. The results illustrated in these figures demonstrate that QuBER successfully refines both fine-grained and instance-level errors.

B. Ablation Studies

Effect of Error-informed Refinement. We evaluated the impact of our proposed error-informed refinement by comparing QuBER with variants lacking the error estimation and EGF modules (Table IV). We used the original Panoptic-DeepLab [36] as a baseline, evaluating it as both an IS and a refinement model with RGB-D inputs, similar to QuBER's setup. Results show that Panoptic-DeepLab without error estimation and EGF fails to achieve state-of-the-art performance as both an IS and a refinement model, even underperforming UCN. In contrast, QuBER, which incorporates both error estimation and EGF, significantly outperforms the others, demonstrating the effectiveness of our error-informed refinement scheme for high-quality UOIS.

Effect of Quadruple Boundary Error. To assess the importance of our novel quadruple boundary error, we conducted an ablation study on the OCID dataset using UCN as the IS model (Table V). We compared three error estimation approaches: binary boundary error, mask quadruple error, and our proposed quadruple boundary error (first to third rows, respectively). Results clearly demonstrate that our quadruple boundary error yields the best performance, highlighting its effectiveness in capturing complex segmentation errors and guiding precise refinement.

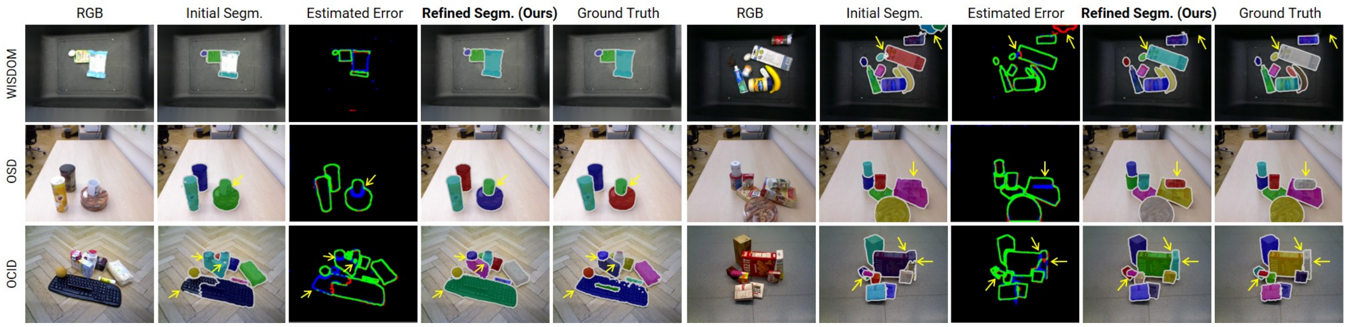


Fig. 5: High-quality UOIS results of QuBER on diverse scenes demonstrating accurate object instance segmentation

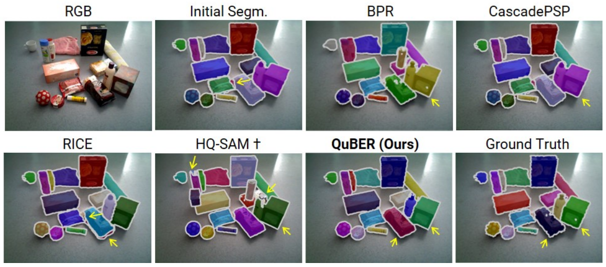


Fig. 6: Comparison of QuBER with state-of-the-art methods

TABLE V: Ablation of quadruple boundary error on OCID

Quadruple Error	Boundary Error	Overlap P_N	R_O	F_O	Boundary P_B	R_B	F_B	$F_O^{@.75}$
✗	✓	88.2	88.5	84.6	84.0	85.3	81.5	85.4
✓	✗	**88.9**	89.3	85.8	85.7	86.4	83.1	87.3
✓	✓	**88.9**	**89.8**	**86.1**	**86.1**	**87.1**	**83.7**	**87.6**

C. *Robot Experiments: Target Object Grasping*

We evaluated the effect of QuBER on the performance of UOIS in a practical robotic task—segmenting and grasping unknown target objects from cluttered bins—a scenario common in warehouse pick-and-place [48]. By integrating QuBER into state-of-the-art UOIS methods (UOAIS-Net [8] and UCN [49]), we aimed to demonstrate its ability to improve segmentation quality and grasping success rates.

Setup. We used a UR5 robotic arm with an Azure Kinect camera mounted for hand-eye coordination and a suction gripper. In each trial, up to 20 objects were randomly placed in a bin (Fig. 7). To simulate real-world identification constraints, we used ten template images per target object for matching. The pipeline involved three steps: 1) performing UOIS on RGB-D images using either UOAIS-Net or UCN, both with and without QuBER refinement; 2) matching the segmented objects to the templates using cosine distance between pre-trained DINOv2 features [50]; and 3) executing a suction grasp on the most planar regions using RANSAC plane fitting [51]. We conducted 100 trials per method, attempting ten grasps for each of ten distinct target objects. To ensure a fair comparison, object placements remained consistent across trials with and without QuBER refinement.

Results. Table VI presents our findings, evaluated using two metrics: segmentation success rate (accurate segmentation and matching without over- or under-segmentation) and grasp success rate (successful target object removal from the bin). QuBER consistently improved both metrics across all experiments. For UCN, QuBER improved overall performance by detecting missing instances. For UOAIS-Net, QuBER effectively resolved over- and under-segmentation issues, enhancing object matching and grasping performance.

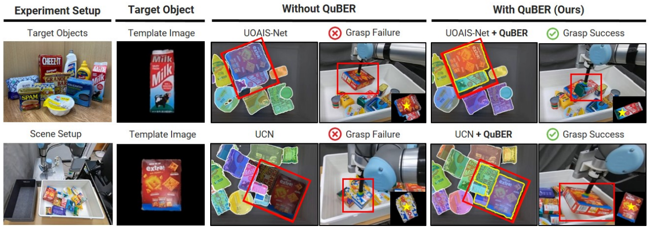


Fig. 7: Target object grasping setup and results. QuBER refines initial UOIS results, leading to the successful grasping of the target object. (Yellow star: grasping point from segmentation. Videos are available in the supplementary.)

TABLE VI: Target object grasping performances

Method	segmentation success rate	grasp success rate
UCN [3]	65%	59%
+ QuBER	**82%** (+17%)	**75%** (+16%)
UOAIS-Net [8]	80%	70%
+ QuBER	**86%** (+6%)	**76%** (+6%)

V. CONCLUSION AND FUTURE WORK

We introduced QuBER, an error-informed refinement method for high-quality UOIS. With quadruple boundary error estimation and EGF module, QuBER achieved state-of-the-art segmentation and enhanced robotic grasping performance. However, estimated errors may still contain inaccuracies, especially in cases of severe occlusion and out-of-distribution. Future work will focus on scaling up datasets and incorporating continual learning to improve robustness.

ACKNOWLEDGMENT

This work was supported by the Technology Innovation Program (Project Number: 00442029) funded by the Ministry of Trade Industry & Energy (MOTIE, Korea), and Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (Project Number: 20008613). This work was partially supported by Artificial intelligence industrial convergence cluster development project funded by the Ministry of Science and ICT (MSIT, Korea) & Gwangju Metropolitan City.

References

[1] 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 data," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7283–7290.

[2] 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.

[3] ——, "Unseen object instance segmentation for robotic environments," *IEEE Trans. on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.

[4] H. Yu and C. Choi, "Self-supervised interactive object segmentation through a singulation-and-grasping approach," in *European Conference on Computer Vision*. Springer, 2022, pp. 621–637.

[5] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 438–13 444.

[6] A. Goyal, A. Mousavian, C. Paxton, Y.-W. Chao, B. Okorn, J. Deng, and D. Fox, "Ifor: Iterative flow minimization for robotic object rearrangement," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 14 787–14 797.

[7] 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.

[8] 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)*, 2022, pp. 5085–5092.

[9] Y. Lu, Y. Chen, N. Ruozzi, and Y. Xiang, "Mean shift mask transformer for unseen object instance segmentation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 2760–2766.

[10] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1925–1934.

[11] Y. Yuan, J. Xie, X. Chen, and J. Wang, "Segfix: Model-agnostic boundary refinement for segmentation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*. Springer, 2020, pp. 489–506.

[12] H. K. Cheng, J. Chung, Y.-W. Tai, and C.-K. Tang, "Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8890–8899.

[13] C. Tang, H. Chen, X. Li, J. Li, Z. Zhang, and X. Hu, "Look closer to segment better: Boundary patch refinement for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 926–13 935.

[14] C. Xie, A. Mousavian, Y. Xiang, and D. Fox, "Rice: Refining instance masks in cluttered environments with graph neural networks," in *Conference on Robot Learning*. PMLR, 2022, pp. 1655–1665.

[15] 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.

[16] 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.

[17] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, *et al.*, "Grounded sam: Assembling open-world models for diverse visual tasks," *arXiv preprint arXiv:2401.14159*, 2024.

[18] H. Zhang, Y. Su, X. Xu, and K. Jia, "Improving the generalization of segmentation foundation model under distribution shift via weakly supervised adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 23 385–23 395.

[19] X. Fang, L. P. Kaelbling, and T. Lozano-Pérez, "Embodied uncertainty-aware object segmentation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 2639–2646.

[20] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International journal of computer vision*, vol. 59, pp. 167–181, 2004.

[21] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze, "Segmentation of unknown objects in indoor environments," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 4791–4796.

[22] S. Back, J. Kim, R. Kang, S. Choi, and K. Lee, "Segmenting unseen industrial components in a heavy clutter using rgb-d fusion and synthetic data," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 828–832.

[23] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," *Advances in neural information processing systems*, vol. 24, 2011.

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. on Patt. Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2017.

[25] A. Kirillov, Y. Wu, K. He, and R. Girshick, "Pointrend: Image segmentation as rendering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9799–9808.

[26] X. Shen, J. Yang, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang, "Dct-mask: Discrete cosine transform mask representation for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8720–8729.

[27] L. Zhang, S. Zhang, X. Yang, H. Qiao, and Z. Liu, "Unseen object instance segmentation with fully test-time rgb-d embeddings adaptation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 4945–4952.

[28] Q. M. Rahman, P. Corke, and F. Dayoub, "Run-time monitoring of machine learning for robotic perception: A survey of emerging trends," *IEEE Access*, vol. 9, pp. 20 067–20 075, 2021.

[29] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016.

[30] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[31] Q. M. Rahman, N. Sünderhauf, P. Corke, and F. Dayoub, "Fsnet: A failure detection framework for semantic segmentation," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 3030–3037, 2022.

[32] B. Sun, J. Xing, H. Blum, R. Siegwart, and C. Cadena, "See yourself in others: Attending multiple tasks for own failure detection," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 8409–8416.

[33] Y. Xie, J. Zhang, H. Lu, C. Shen, and Y. Xia, "Sesv: Accurate medical image segmentation by predicting and correcting errors," *IEEE Trans. on Med. Imag.*, vol. 40, no. 1, pp. 286–296, 2020.

[34] C. B. Kuhn, M. Hofbauer, G. Petrovic, and E. Steinbach, "Reverse error modeling for improved semantic segmentation," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 106–110.

[35] S. Ali, F. Dayoub, and A. K. Pandey, "Learning from learned network: An introspective model for arthroscopic scene segmentation," in *Proceedings of International Conference on Information and Communication Technology for Development*. Springer, 2023, pp. 393–406.

[36] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen, "Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 475–12 485.

[37] 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*, 2016, pp. 770–778.

[38] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv:1704.04861*, 2017.

[39] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," *Advances in neural information processing systems*, vol. 27, 2014.

[40] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 269–286.

[41] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[42] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

[43] S. Zakharov, B. Planche, Z. Wu, A. Hutter, H. Kosch, and S. Ilic, “Keep it unreal: Bridging the realism gap for 2.5 d recognition with geometry priors only,” in *2018 International Conference on 3D Vision (3DV)*. IEEE, 2018, pp. 1–11.

[44] M. Shi, J. Shen, Q. Yi, J. Weng, Z. Huang, A. Luo, and Y. Zhou, “Lmffnet: a well-balanced lightweight network for fast and accurate semantic segmentation,” *IEEE Trans. on Neural Net. Learn. Sys.*, 2022.

[45] 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*. Ieee, 2009, pp. 248–255.

[46] M. Suchi, T. Patten, D. Fischinger, and M. Vincze, “Easylabel: A semi-automatic pixel-wise object annotation tool for creating robotic rgb-d datasets,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6678–6684.

[47] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[48] C. Mitash, F. Wang, S. Lu, V. Terhuja, T. Garaas, F. Polido, and M. Nambi, “Armbench: An object-centric benchmark dataset for robotic manipulation,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9132–9139.

[49] Y. Xia, Y. Zhang, F. Liu, W. Shen, and A. L. Yuille, “Synthesize then compare: Detecting failures and anomalies for semantic segmentation,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 2020, pp. 145–161.

[50] 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 preprint arXiv:2304.07193*, 2023.

[51] A. Gouda, A. Ghanem, and C. Reining, “Dopose-6d dataset for object segmentation and 6d pose estimation,” in *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2022, pp. 477–483.