IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

# HEAPGrasp: Hand-Eye Active Perception to Grasp Objects with Diverse Optical Properties

Ginga Kennis[1] and Shogo Arai[2]

***Abstract*—Autonomous robotic handling requires accurate 3-D scene measurement followed by grasp planning. Conventional systems struggle with transparent or specular objects. Additionally, in hand–eye setups, moving through multiple viewpoints increases handling execution time. In this paper, we propose HEAPGrasp—Hand-Eye Active Perception to Grasp objects with diverse optical properties. To measure such objects, we focus on the ability to segment objects regardless of their optical properties in RGB images. We employ Shape from Silhouette based on the segmented images for 3-D measurement. To shorten the time required for multi-view capture with a hand-eye camera, we plan its trajectory using a cost function that balances 3-D measurement accuracy against its trajectory length. Real-robot experiments achieve a 96.0% grasp success rate on transparent, specular, and opaque objects, while reducing the hand-eye camera's trajectory length by 52% and handling execution time by 19% relative to a baseline that circles around the scene for 3-D measurement.**

***Index Terms*—Grasping, Perception for Grasping and Manipulation, Transparent Objects, Hand-eye Camera**

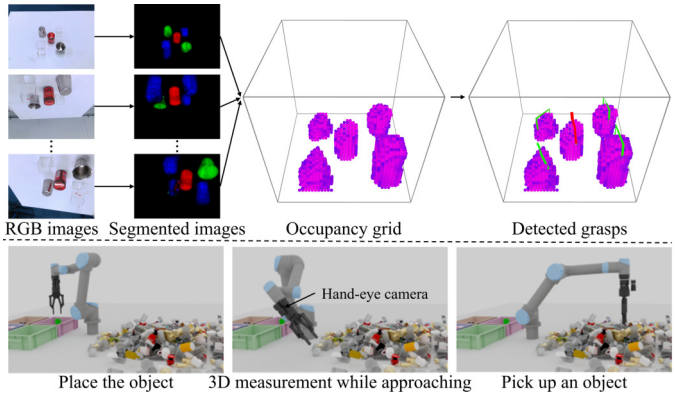



Fig. 1. Overview of HEAPGrasp. We segment multi-view RGB images and perform 3-D measurement via Shape from Silhouette. The hand-eye camera trajectory is generated using a cost function that balances measurement accuracy and its trajectory length to reduce multi-view capture time.

## I. Introduction

ROBOT handling—grasping objects and transporting them to target locations—constitutes a large share of robotic applications and is one of the most critical operations. Demands range from automotive parts and logistics packages to food ingredients and restaurant dishes.

Autonomous handling requires accurate 3-D measurement of the scene followed by grasp planning. However, the difficulty of 3-D measurement depends significantly on the object's optical properties. Measuring opaque objects governed by diffuse reflection is relatively easy, whereas measurement becomes increasingly challenging as an object's transparency increases. Measuring specular objects is also challenging. High-power laser-scanning sensors [1] can measure specular surfaces, but their high price prevents them from being widely adopted.

To address these issues, we propose HEAPGrasp-Hand-Eye Active Perception to Grasp objects with diverse optical properties. An overview of the proposed method is shown in Fig. 1.

Manuscript received: August 9, 2025; Revised November 13, 2025; Accepted December 19, 2025.

This paper was recommended for publication by Editor Julia Borras Sol upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by JSPS KAKENHI Grant Number JP24K07379 and by JKA and its promotion funds from KEIRIN RACE under Grant Number 2022M-263.

[1]Ginga Kennis is with the Graduate School of Science and Technology, Department of Mechanical and Aerospace Engineering, Tokyo University of Science, Chiba, Japan kennis.ginga@arai-lab.org

[2]Shogo Arai is with the Faculty of Science and Technology, Department of Mechanical and Aerospace Engineering, Tokyo University of Science, Chiba, Japan arai.shogo@rs.tus.ac.jp

Digital Object Identifier (DOI): see top of this page.



To measure objects with diverse optical properties, we focus on the ability to segment objects regardless of their optical properties in RGB images. The hand-eye camera captures multi-view RGB images, and semantic segmentation extracts object silhouettes. We then employ Shape from Silhouette (SfS) [2] using the segmented images for 3-D measurement. In SfS, increasing the diversity of viewpoints improves 3-D measurement accuracy, leading to a higher grasp success rate. However, moving the hand-eye camera to multiple viewpoints is time-consuming. Therefore, there is a trade-off between 3-D measurement accuracy and the time required for multi-view capture. To resolve this trade-off, we generate the hand-eye camera's trajectory using a cost function that balances 3-D measurement accuracy against its trajectory length. While the hand-eye camera moves along this trajectory, it continuously captures the scene and updates the 3-D measurement result.

We validate the proposed method on a real robotic system. Results show that the proposed method achieves a grasp success rate of 96.0% for objects with diverse optical properties. Additionally, it reduces the hand-eye camera's trajectory by 52% and the handling execution time by 19% compared to a baseline that circles around the scene for 3-D measurement.

In summary, the contributions of this work are as follows:

- A 3-D measurement and grasping method for objects with diverse optical properties using segmentation and Shape from Silhouette,
- An active perception planning for a hand-eye camera to resolve the trade-off between 3-D measurement accuracy and the time required for multi-view capture,
- Experimental results showing 96.0% grasp success rate

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**

on objects with diverse optical properties, $52\%$ reduction in the hand-eye camera's trajectory, and $19\%$ decrease in handling execution time compared to a baseline that circles around the scene for 3-D measurement.

This paper uses the following notations. Let $\mathbb{Z}_+$ denote the set of positive integers, $\mathbb{R}$ the set of real numbers, and $\mathbb{G}$ the set of grasp candidates. For a vector $a \in \mathbb{R}^n$, Euclidean norm is denoted by $\|a\|$. All poses are represented as $p := [t^\top, q^\top]^\top \in \mathbb{R}^7$, where $t := [x, y, z]^\top \in \mathbb{R}^3$ is position and $q := [q_x, q_y, q_z, q_w]^\top \in \mathbb{R}^4$ is orientation in quaternion.

## II. Related Work

Object handling is often classified by sensing modality into tactile-based approaches [3], image-based methods [4], and 3-D measurement–based techniques [6]. Here, we focus on image-based and 3-D measurement–based techniques.

### A. Handling Objects with Diverse Optical Properties

Prior work on handling opaque objects typically relies on depth images [4], [5], [6]. Dex-Net [4] and TossingBot [5] use a single top-down depth image, and VGN [6] fuses multi-view depth images into a TSDF [7]. Such depth-based methods cannot handle transparent or specular objects.

For transparent objects, earlier studies used model-based refraction analysis [8], [9], while recent work applies deep learning. ClearGrasp [10] predicts normals, boundaries, and masks of transparent objects and restores depth via refinement [11]. However, because these methods are specialized for the optical characteristics of transparent objects or trained on datasets containing only transparent objects, they do not generalize to objects with diverse optical properties.

More recent studies fully exploit deep learning to handle objects with diverse optical properties [12], [13]. ASGrasp [12] predicts visible and occluded depth from RGB and IR stereo images, and GraspNeRF [13] performs end-to-end 3-D reconstruction and grasp planning using Generalizable NeRF [14]. By training on datasets that include objects with diverse optical properties, these models generalize to transparent, specular, and opaque objects.

### B. Active Perception to Reduce Handling Execution Time

Accurate 3-D measurement benefits from capturing the scene from multiple viewpoints. However, obtaining multi-view images by moving a hand-eye camera increases the handling execution time. To address this issue, recent studies have explored active perception using hand-eye cameras to reduce multi-view acquisition time [15], [17]. Evo-NeRF [15] performs 3-D measurement from multi-view RGB images using implicit NeRF representation [16], executing a full capture trajectory for the first grasp and updating the NeRF with shorter camera trajectories for subsequent grasps. However, NeRF optimization in Evo-NeRF still requires several seconds, which limits its applicability to real-time viewpoint optimization. In contrast, Breyer et al. [17] propose a closed-loop next-best-view strategy to grasp partially occluded objects efficiently. The method performs explicit TSDF integration for 3-D measurement, as in VGN [6], which can be executed in tens of milliseconds. It then computes the next viewpoint based on information gain over the TSDF volume, enabling real-time viewpoint optimization.

### C. Positioning of This Work

Although significant progress has been made in both directions (Sections II-A and II-B), no existing method achieves both simultaneously. We propose a method that enables grasping objects with diverse optical properties using a monocular camera, while reducing the handling execution time through active perception. To enable real-time viewpoint optimization while measuring such objects, we adopt a hybrid approach combining deep-learning-based segmentation with explicit Shape from Silhouette (SfS) measurement.

## III. Problem Formulation

We consider the problem of performing a pick and place task of $n_{\mathrm{ob}}$ objects $O_1, O_2, \ldots, O_{n_{\mathrm{ob}}}$ in space $\mathbb{L} \in \mathbb{W}$ using a hand-eye robot system with a monocular camera, where $\mathbb{W}$ denotes the robot's workspace. The true 3-D shape of the scene is defined as $S \in \mathbb{S}$. We change the pose of the hand-eye camera $p_{\mathrm{cam}} \in \mathbb{R}^7$ and capture the scene $N$ times. A 3-D measurement algorithm $\mathcal{R}$ processes the captured images and outputs a 3-D measurement result $\hat{S}$. Subsequently, a grasp planning module $\mathcal{G}$ takes $\hat{S}$ as input and computes a set of grasp candidates. Finally, the robot system picks up the object with the optimal grasp and places it at the desired pose $p_{\mathrm{place}} \in \mathbb{R}^7$.

First, by using the images $\mathbf{I} := \{I^1, I^2, \ldots, I^N\}$, the 3-D measurement result can be expressed as

$$\hat{S} = \mathcal{R}(S, \mathbf{I}). \tag{1}$$

We assume that the images $\mathbf{I}$ depend only on the poses of the hand-eye camera $\mathbf{P}_{\mathrm{cam}} := \{p^1_{\mathrm{cam}}, p^2_{\mathrm{cam}}, \ldots, p^N_{\mathrm{cam}}\}$ when they are captured. Therefore, the 3-D measurement result can be rewritten as $\hat{S} = \mathcal{R}(S, \mathbf{P}_{\mathrm{cam}})$. For convenience, we sometimes write this as $\hat{S}(\mathbf{P}_{\mathrm{cam}})$.

Then, the grasping module $\mathcal{G}$ computes the grasp candidates $\mathbf{G} := \{g^1, g^2, \ldots, g^M\}$ and their associated grasp scores $\mathbf{S}_{\mathrm{grasp}} := \{s^1_{\mathrm{grasp}}, s^2_{\mathrm{grasp}}, \ldots, s^M_{\mathrm{grasp}}\}$ from the 3-D measurement result, which can be expressed as

$$\mathcal{G}(\hat{S}) = (\mathcal{G}_{\mathrm{cd}}(\hat{S}), \mathcal{G}_{\mathrm{score}}(\hat{S})) : \mathbb{S} \to \mathbb{G}^M \times \mathbb{R}^M. \tag{2}$$

To simplify notation, we expand the expression of the grasp planning module $\mathcal{G}$ as follows

$$\mathcal{G} = (\mathcal{G}_{\mathrm{cd}}, \mathcal{G}_{\mathrm{score}}, \mathcal{G}^*_{\mathrm{cd}}, \mathcal{G}^*_{\mathrm{ob}}) : \mathbb{S} \to \mathbb{G}^M \times \mathbb{R}^M \times \mathbb{G} \times \mathbb{Z}_+, \tag{3}$$

where $\mathcal{G}^*_{\mathrm{cd}}$ represents the optimal grasp and $\mathcal{G}^*_{\mathrm{ob}}$ denotes the label of the object that $\mathcal{G}^*_{\mathrm{cd}}$ targets. Here, we have omitted the input $\hat{S}$. Accordingly, the target object of the optimal grasp $\mathcal{G}^*_{\mathrm{cd}}$ is denoted by $O_{\mathcal{G}^*_{\mathrm{ob}}}$.

The grasp success rate of the target object $O_{\mathcal{G}^*_{\mathrm{ob}}}$ depends on the grasp planning module and the true 3-D shape of the scene, which can be expressed as

$$s_{\mathrm{rate}} = s_{\mathrm{rate}}(\mathcal{G}(\hat{S}), S). \tag{4}$$

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**

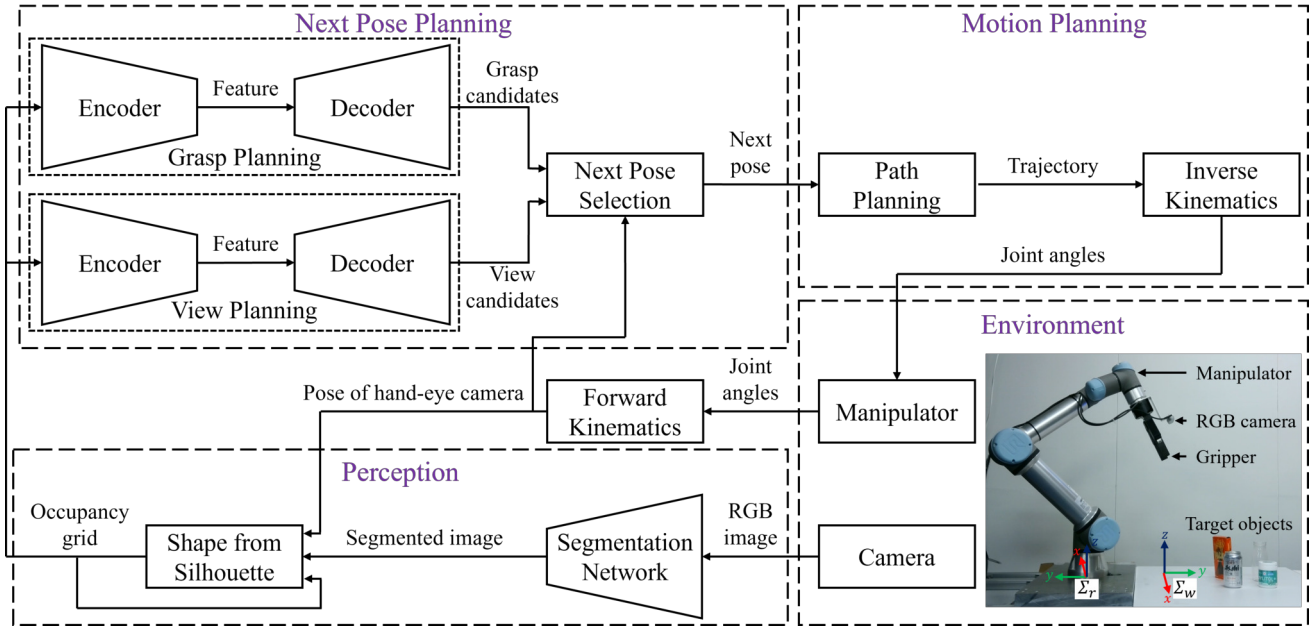



Fig. 2. Block diagram of the proposed method. In the Perception block, semantic segmentation is applied to RGB images captured by the hand-eye camera, and Shape from Silhouette updates the 3-D measurement based on the segmented image, the hand-eye camera pose, and the previous 3-D measurement. The Next Pose Planning block uses the 3-D measurement at each view waypoint for grasp and view planning, selecting the next view waypoint that improves measurement accuracy while approaching the target grasp. The Motion Planning block generates a trajectory between consecutive view waypoints and moves the camera accordingly. This process repeats until the camera reaches the final view waypoint, after which the final grasp is executed.

Given the initial pose of the hand-eye camera $p^0_{\mathrm{cam}} \in \mathbb{R}^7$, the problem of minimizing the total time to complete a pick and place task of all objects while ensuring the grasp success rate is above a threshold $\epsilon_{\mathrm{rate}} \in \mathbb{R}$ can be written as

$$\min_{\mathbf{P}_{\mathrm{cam},i}} \sum_{i=1}^{n_{\mathrm{ob}}} T(p^0_{\mathrm{cam}}, \mathbf{P}_{\mathrm{cam},i}, \mathcal{G}^*_{\mathrm{cd}}(\hat{S}(\mathbf{P}_{\mathrm{cam},i})), p_{\mathrm{place}}) \tag{5}$$

$$\text{s.t. } s_{\mathrm{rate}}(\mathcal{G}(\hat{S}(\mathbf{P}_{\mathrm{cam},i})), S_i) \geq \epsilon_{\mathrm{rate}}. \tag{6}$$

Here, $\mathbf{P}_{\mathrm{cam},i} := \{p^1_{\mathrm{cam},i}, p^2_{\mathrm{cam},i}, \ldots, p^{N_i}_{\mathrm{cam},i}\}$ denotes the hand–eye camera poses at which images in phase $i \in \{1, 2, \ldots, n_{\mathrm{ob}}\}$[1] are captured, where $p^n_{\mathrm{cam},i}$ is the pose at which the $n$-th image is captured, and $N_i$ is the total number of images in phase $i$. $S_i$ is the true 3-D shape of the scene in phase $i$.

Under the following assumptions:

- Task execution time $T$ is determined mainly by the length of the hand-eye camera's trajectory,
- Grasp success rate increases with 3-D measurement accuracy,

the optimization problem we consider can be written as

$$\min_{\mathbf{P}_{\mathrm{cam},i}} \sum_{i=1}^{n_{\mathrm{ob}}} \alpha d_{\mathrm{space}}(p^0_{\mathrm{cam}}, \mathbf{P}_{\mathrm{cam},i}, \mathcal{G}^*_{\mathrm{cd}}(\hat{S}(\mathbf{P}_{\mathrm{cam},i})), p_{\mathrm{place}}) + \beta d_{\mathrm{measure}}(\hat{S}(\mathbf{P}_{\mathrm{cam},i}), S_i). \tag{7}$$

Here, $d_{\mathrm{space}}$ represents the hand-eye camera's trajectory length as it moves from the initial pose $p^0_{\mathrm{cam}}$ through viewpoints $\mathbf{P}_{\mathrm{cam},i}$, optimal grasp $\mathcal{G}^*_{\mathrm{cd}}(\hat{S}(\mathbf{P}_{\mathrm{cam},i}))$ to the place pose $p_{\mathrm{place}}$.

[1]A phase begins with the camera at its initial pose and ends when the object is placed at its target location.

$d_{\mathrm{measure}}$ is the error between the true 3-D shape of the scene $S_i$ and the 3-D measurement result $\hat{S}(\mathbf{P}_{\mathrm{cam},i})$.

Finally, to reduce computational time, we do not optimize the entire set of viewpoints $\mathbf{P}_{\mathrm{cam},i}$. Instead, we optimize only a small subset of $L \ll N_i$ view waypoints $\bar{\mathbf{P}}_{\mathrm{cam},i} := \{\bar{p}^1_{\mathrm{cam},i}, \ldots, \bar{p}^L_{\mathrm{cam},i}\}$ and interpolate between consecutive view waypoints to generate the full set of viewpoints

$$\mathbf{P}_{\mathrm{cam},i} = (f(p^0_{\mathrm{cam}}, \bar{p}^1_{\mathrm{cam},i}), f(\bar{p}^1_{\mathrm{cam},i}, \bar{p}^2_{\mathrm{cam},i}), \ldots, f(\bar{p}^{L-1}_{\mathrm{cam},i}, \bar{p}^L_{\mathrm{cam},i})). \tag{8}$$

Here, $f(p_A, p_B)$ is an interpolation function that generates a sequence of poses connecting poses $p_A \in \mathbb{R}^7$ and $p_B \in \mathbb{R}^7$.

## IV. Method

This section focuses on the $i$-th phase, and we drop the subscript $i$. The block diagram of the proposed method is shown in Fig. 2. Let $\Sigma_r$ and $\Sigma_w$ denote the robot base and workspace frames, respectively. We assume the relative pose between $\Sigma_r$ and $\Sigma_w$ is given. Unless stated otherwise, all poses are expressed in the workspace frame $\Sigma_w$.

### A. Perception

When the hand-eye camera arrives at the $n$-th viewpoint ($n = 1, 2, \ldots, N$), denoted by $p^n_{\mathrm{cam}}$, it captures an RGB image $I^n_{\mathrm{rgb}}$. We apply semantic segmentation on $I^n_{\mathrm{rgb}}$ to extract object silhouettes and generate a segmented image $I^n_{\mathrm{seg}}$. Given the hand-eye camera pose $p^n_{\mathrm{cam}}$, the segmented image $I^n_{\mathrm{seg}}$, and the 3-D measurement result up to the $n-1$-th viewpoint $\hat{S}^{n-1} := \hat{S}(p^1_{\mathrm{cam}}, p^2_{\mathrm{cam}}, \ldots, p^{n-1}_{\mathrm{cam}})$, we perform Shape from Silhouette (SfS) [2] to update the 3-D measurement result.

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**

We define scene $\mathbb{S}$ as a cubic volume of side length $l$. The cube is discretized into $N_x$, $N_y$, and $N_z$ voxels along the $x$-, $y$-, and $z$-axes, respectively, and the 3-D measurement result at the $n$-th viewpoint is represented as an occupancy grid

$$\hat{S}^n_{0-1} \in [0,1]^{N_x \times N_y \times N_z}. \tag{9}$$

For each voxel $a \in \{1,\dots,N_x\}$, $b \in \{1,\dots,N_y\}$, $c \in \{1,\dots,N_z\}$, we compute the occupancy as

$$\hat{S}^n_{0-1,a,b,c} := \begin{cases} \dfrac{|\mathbf{I}^n_{\mathrm{seg},a,b,c}|}{|\mathbf{I}^n_{\mathrm{rgb},a,b,c}|}, & \text{when } |\mathbf{I}^n_{\mathrm{rgb},a,b,c}| > 0 \\ 0, & \text{when } |\mathbf{I}^n_{\mathrm{rgb},a,b,c}| = 0 \end{cases} \tag{10}$$

where $|\mathbf{I}^n_{\mathrm{seg},a,b,c}|$ is the number of images up to the $n$-th viewpoint in which the voxel is segmented, and $|\mathbf{I}^n_{\mathrm{rgb},a,b,c}|$ is the number of images up to the $n$-th viewpoint in which the voxel projects inside the image. Hereafter, let $\hat{S}^n_{\geq\lambda}$ be referred to as the high-confidence occupancy grid, the occupancy grid with values below $\lambda \in [0,1]$ set to zero[2].

### *B. Next Pose Planning*

When the hand-eye camera reaches the $\ell(n)$-th view waypoint ($\ell(n) = 1, 2, \dots\ L$), denoted by $\bar{p}^{\ell(n)}_{\mathrm{cam}}$, we perform grasp planning, view planning, and next pose selection to compute the $\ell(n)+1$-th view waypoint $\bar{p}^{\ell(n)+1}_{\mathrm{cam}}$. Here, $\ell(n)$ denotes the index of the view waypoint that the hand-eye camera passed just before capturing the $n$-th image. The Next Pose Planning block is executed only for those indices $n$ that satisfy $\ell(n)-\ell(n-1) = 1$, i.e., immediately after the hand-eye camera arrives at a view waypoint. We repeat this process until the hand-eye camera reaches the final view waypoint $\bar{p}^{\ell(N)}_{\mathrm{cam}}$. After reaching the final view waypoint, we perform grasp planning and next pose selection once more to compute the optimal grasp $\mathcal{G}^*_{\mathrm{cd}}(\hat{S}^N)$.

**Grasp Planning**: We input the high-confidence occupancy grid $\hat{S}^n_{\geq\lambda}$ into an encoder–decoder 3-D CNN that predicts the grasp orientation $\mathbf{R}^{\ell(n)}_{\mathrm{grasp}} \in \mathbb{R}^{N_x \times N_y \times N_z \times 4}$, the gripper width $\mathbf{W}^{\ell(n)}_{\mathrm{grasp}} \in \mathbb{R}^{N_x \times N_y \times N_z \times 1}$, and the grasp score $\mathbf{S}^{\ell(n)}_{\mathrm{grasp}} \in [0,1]^{N_x \times N_y \times N_z \times 1}$ at every voxel. Let $\mathbf{T}_{\mathrm{grasp}} \in \mathbb{R}^{N_x \times N_y \times N_z \times 3}$ denote the voxel center positions. Using the grasp poses $\mathbf{P}^{\ell(n)}_{\mathrm{grasp}} := [\mathbf{T}_{\mathrm{grasp}}, \mathbf{R}^{\ell(n)}_{\mathrm{grasp}}] \in \mathbb{R}^{N_x \times N_y \times N_z \times 7}$ and the gripper width $\mathbf{W}^{\ell(n)}_{\mathrm{grasp}}$, we define the grasp candidates computed at the $\ell(n)$-th view waypoint as

$$\mathbf{G}^{\ell(n)} := [\mathbf{P}^{\ell(n)}_{\mathrm{grasp}}, \mathbf{W}^{\ell(n)}_{\mathrm{grasp}}] \in \mathbb{R}^{N_x \times N_y \times N_z \times 8}. \tag{11}$$

**View Planning (VP)**: We predefine a spherical coordinate system of radius $r$, whose origin is at the bottom center of the scene. By uniformly dividing the azimuthal angle $\phi \in [0, 2\pi)$ into $U$ intervals and the polar angle $\theta \in [0, \theta_{\max})$ into $V$ intervals, we obtain $UV$ view waypoint candidates

$$\bar{\mathbf{P}}_{\mathrm{cam,cd}} := [\bar{p}^{u,v}_{\mathrm{cam}}]_{u=1,2,\dots,U;\, v=1,2,\dots,V} \in \mathbb{R}^{U \times V \times 7}, \tag{12}$$

where candidates are oriented toward the center of the scene.

We first input the occupancy grid $\hat{S}^n_{0-1}$ into a 3-D CNN encoder to produce a 3-D feature map. Next, we compress this 3-D feature map along the $z$-axis (height) to obtain a 2-D feature map. We then apply a sequence of 2-D convolution, ReLU activation, and upsampling multiple times to the 2-D feature map. Finally, we once again apply a 2-D convolution followed by a sigmoid activation to output the view scores

$$\mathbf{S}^{\ell(n)+1}_{\mathrm{cam}} := [s^{\ell(n)+1,u,v}_{\mathrm{cam}}]_{u=1,2,\dots,U;\, v=1,2,\dots,V} \in [0,1]^{U \times V}. \tag{13}$$

Each view score $s^{\ell(n)+1,u,v}_{\mathrm{cam}}$ corresponds to the view waypoint candidate $\bar{p}^{u,v}_{\mathrm{cam}}$, whose spherical coordinates are given by $\phi := 2\pi(u-1)/U$ and $\theta := \theta_{\max}(v-1)/V$.

In SfS, smaller intersections of visual hulls indicate less uncertainty in the 3-D measurement result and thus higher measurement accuracy. Based on this property, the view score $s^{\ell(n)+1,u,v}_{\mathrm{cam}}$ is interpreted as the expected reduction in occupied voxels and is defined as

$$s^{\ell(n)+1,u,v}_{\mathrm{cam}} := 1 - \frac{\#\hat{S}_{\geq\lambda}(\mathbf{P}^{\ell(n)}_{\mathrm{cam}}, f(\bar{p}^{\ell(n)}_{\mathrm{cam}}, \bar{p}^{u,v}_{\mathrm{cam}}))}{\#\hat{S}_{\geq\lambda}(\mathbf{P}^{\ell(n)}_{\mathrm{cam}})}. \tag{14}$$

Here, $\mathbf{P}^{\ell(n)}_{\mathrm{cam}}$ denotes the viewpoints up to the $\ell(n)$-th view waypoint, $\#\hat{S}_{\geq\lambda}(\mathbf{P}^{\ell(n)}_{\mathrm{cam}})$ and $\#\hat{S}_{\geq\lambda}(\mathbf{P}^{\ell(n)}_{\mathrm{cam}}, f(\bar{p}^{\ell(n)}_{\mathrm{cam}}, \bar{p}^{u,v}_{\mathrm{cam}}))$ represent the number of non-zero voxels in the current high-confidence occupancy grid and in the high-confidence occupancy grid obtained by hypothetically moving the hand-eye camera to the view waypoint candidate $\bar{p}^{u,v}_{\mathrm{cam}}$, respectively.

**Next Pose Selection**: We first smooth the grasp scores $\mathbf{S}^{\ell(n)}_{\mathrm{grasp}}$ with a 3-D Gaussian filter. We then remove grasp candidates whose scores fall below a threshold $\epsilon \in \mathbb{R}$. Finally, we apply non-maximum suppression to retain candidates with the highest score among their neighbors. The grasp poses retained after this process are defined as

$$\mathbf{P}^{\ell(n)}_{\mathrm{grasp},\epsilon} := \{p^{\ell(n),a,b,c}_{\mathrm{grasp}} \mid s^{\ell(n),a,b,c}_{\mathrm{grasp}} \geq \epsilon\}. \tag{15}$$

Here, $p^{\ell(n),a,b,c}_{\mathrm{grasp}}$ denotes the grasp pose and $s^{\ell(n),a,b,c}_{\mathrm{grasp}}$ denotes the associated grasp score at voxel indices $a$, $b$, $c$. We select the target grasp pose at the $\ell(n)$-th view waypoint as

$$p^{\ell(n)}_{\mathrm{grasp}} := \arg\min_{p \in \mathbf{P}^{\ell(n)}_{\mathrm{grasp},\epsilon}} \mathrm{dist}(p^{\ell(n)-1}_{\mathrm{grasp}}, p) \tag{16}$$

thereby ensuring that the target grasp pose changes minimally between consecutive view waypoints. Here, $\mathrm{dist}(p_a, p_b) \in \mathbb{R}$ is the distance between two poses $p_a := [t_a^\top, q_a^\top]^\top, p_b := [t_b^\top, q_b^\top]^\top \in \mathbb{R}^7$, computed as

$$\mathrm{dist}(p_a, p_b) := \|t_a - t_b\| + \gamma_1 \min(\|q_a + q_b\|,\ \|q_a - q_b\|), \tag{17}$$

where $t_a, t_b \in \mathbb{R}^3$ denote the positions, and $q_a, q_b \in \mathbb{R}^4$ denotes the orientation in quaternion. $\gamma_1$ is a hyperparameter.

Finally, We select the $\ell(n)+1$-th view waypoint as $\bar{p}^{\ell(n)+1}_{\mathrm{cam}} := \bar{p}^{u^*,v^*}_{\mathrm{cam}}$ where

$$(u^*, v^*) := \arg\max_{1 \leq u \leq U,\ 1 \leq v \leq V} s^{\ell(n)+1,u,v}_{\mathrm{cam}} + \gamma_2\, h(u,v). \tag{18}$$

Here, $h(u,v)$ is the rate of reduction in distance to the target grasp pose, defined as

$$h(u,v) := 1 - \frac{\mathrm{dist}(\bar{p}^{u,v}_{\mathrm{cam}},\ p^{\ell(n)}_{\mathrm{grasp}})}{\mathrm{dist}(\bar{p}^{\ell(n)}_{\mathrm{cam}},\ p^{\ell(n)}_{\mathrm{grasp}})}, \tag{19}$$

where $\gamma_2$ is a hyperparameter that balances 3-D measurement accuracy and distance reduction to the target grasp pose.

[2]We set $\lambda$ close to 1 to retain only voxels with high occupancy.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

## V. Training

For semantic segmentation, we use DeepLabv3+ [18] with a ResNet-50 [19] backbone pretrained on ImageNet [20]. We fine-tune the model on 427 real-world images captured by our research group to segment images into four classes: transparent objects, specular objects, opaque objects, and background. We evaluate the segmentation performance on 40 validation images and obtain IoU values of 0.93, 0.95, and 0.94 for transparent, specular, and opaque objects, respectively, with a mIoU of 0.94. These results demonstrate robust segmentation performance regardless of the optical properties of the objects.

We adopt the Volumetric Grasping Network (VGN) [6] as the grasp planning model. Unlike the original VGN, our method represents its input as a high-confidence occupancy grid. Following the same dataset generation scheme as VGN, we use the PyBullet physics engine to generate approximately $2.0 \times 10^6$ grasp candidates.

The encoder in the view planning model shares the same architecture as that in the grasp planning model. We use the PyBullet physics engine to generate the view planning dataset, with the same objects used in the grasp planning dataset, as described in Algorithm 1. In this algorithm, SfS (lines 12 and 17) maps hand-eye camera poses and segmented images to an occupancy grid. After generating the dataset in this manner and removing invalid samples, we obtain approximately $5.5 \times 10^6$ view scores.

**Algorithm 1** View Planning Dataset Generation Scheme

1: **for** $i = 1$ **to** $N_{\mathrm{scene}}$ **do**
2: $n_{\mathrm{ob}} \leftarrow \mathrm{UniformRandom}(1, N_{\mathrm{ob}})$
3: Randomly place $n_{\mathrm{ob}}$ objects in the scene
4: $k \leftarrow \mathrm{UniformRandom}(0, L-1)$
5: **if** $k = 0$ **then**
6: $\mathbf{P}^k_{\mathrm{cam}} := \left(p^0_{\mathrm{cam}}\right)$
7: **else**
8: $\{\bar{p}^1_{\mathrm{cam}}, \ldots, \bar{p}^k_{\mathrm{cam}}\} \leftarrow$ Randomly select $k$ view waypoints from $\bar{\mathbf{P}}_{\mathrm{cam,cd}}$
9: $\mathbf{P}^k_{\mathrm{cam}} := (f(p^0_{\mathrm{cam}}, \bar{p}^1_{\mathrm{cam}}), \ldots, f(\bar{p}^{k-1}_{\mathrm{cam}}, \bar{p}^k_{\mathrm{cam}}))$
10: **end if**
11: $\mathbf{I}^k_{\mathrm{seg}} \leftarrow \mathrm{GetSegmentedImage}(\mathbf{P}^k_{\mathrm{cam}})$
12: $\hat{S}(\mathbf{P}^k_{\mathrm{cam}}) = \mathrm{SfS}(\mathbf{P}^k_{\mathrm{cam}}, \mathbf{I}^k_{\mathrm{seg}})$
13: **for** $(u, v)$ **in** $\{1, \ldots, U\} \times \{1, \ldots, V\}$ **do**
14: $\bar{p}^{u,v}_{\mathrm{cam}} \leftarrow \mathrm{GetPoseFromIndex}(\bar{\mathbf{P}}_{\mathrm{cam,cd}},\ u,\ v)$
15: $\mathbf{P}^{k+1,u,v}_{\mathrm{cam}} := \left(\mathbf{P}^k_{\mathrm{cam}}, f(\bar{p}^k_{\mathrm{cam}}, \bar{p}^{u,v}_{\mathrm{cam}})\right)$
16: $\mathbf{I}^{k+1,u,v}_{\mathrm{seg}} \leftarrow \mathrm{GetSegmentedImage}(\mathbf{P}^{k+1,u,v}_{\mathrm{cam}})$
17: $\hat{S}(\mathbf{P}^{k+1,u,v}_{\mathrm{cam}}) = \mathrm{SfS}(\mathbf{P}^{k+1,u,v}_{\mathrm{cam}}, \mathbf{I}^{k+1,u,v}_{\mathrm{seg}})$
18: $s^{k+1,u,v}_{\mathrm{cam}} := 1 - \dfrac{\#\hat{S}_{\geq\lambda}(\mathbf{P}^{k+1,u,v}_{\mathrm{cam}})}{\#\hat{S}_{\geq\lambda}(\mathbf{P}^k_{\mathrm{cam}})}$
19: **end for**
20: **end for**

## VI. Experiments

### A. Experimental Setup

We conduct experiments in both simulation and real-world environments. The simulation experiments are performed in the same environment as VGN [6], and do not model the optical properties of objects. Therefore, both depth and segmentation images are assumed to be ideal. This simulation aims to compare the algorithms of each method under ideal conditions, while the robustness to optical properties is verified in real-robot experiments.

The real-robot system consists of a Universal Robots A/S UR5e, an Intel Corporation RealSense Depth Camera D415, and a ROBOTIQ Inc. 2F-140 Adaptive Gripper. Note that depth images are used only by the baseline method. Fig. 3 shows the objects used in the experiments.

### B. Baseline Methods

We compare the proposed method (**HEAPGrasp** and **HEAPGrasp w/o VP**) with baseline methods (**VGN**, **GraspNeRF**):

- **VGN** [6]: We predefine $L$ view waypoints evenly spaced around the scene. As the hand-eye camera moves from its initial pose through each view waypoint, it captures depth images (640 × 480 pixels at 30 fps) and integrates them into a TSDF. After reaching the final view waypoint, we execute the top-scoring grasp.
- **GraspNeRF [13]**: We uniformly sample six view waypoints on a hemisphere centered in the workspace. All viewpoints point toward the workspace center and are represented in spherical coordinates with radius $r = 0.5\mathrm{m}$, polar angle $\theta = \pi/6$, and azimuthal angle $\phi \in U(0, 2\pi)$, where $U(a, b)$ denotes a uniform distribution over the interval $[a, b]$. After capturing six RGB images (640 × 360 pixels) from these view waypoints, we execute the grasp with the highest grasp score.
- **HEAPGrasp w/o VP**: As in VGN, we use the same $L$ predefined view waypoints and trajectory to capture RGB images (320 × 240 pixels at 30 fps) and perform 3-D measurement with the proposed perception module. After reaching the final view waypoint, we input the high-confidence occupancy grid to the proposed grasp planning module and execute the top-scoring grasp.

The parameters used in the experiments are listed in Table I. In HEAPGrasp, when computing the first view waypoint $\bar{p}^1_{\mathrm{cam}}$ at the initial pose $p^0_{\mathrm{cam}}$, the high uncertainty in the 3-D measurement result makes grasp candidates unreliable. To address this, we set the weighting parameter $\gamma_2$ to 0.0 for the first view waypoint calculation. After reaching $\bar{p}^1_{\mathrm{cam}}$, we set $\gamma_2$ to 0.35 for subsequent view waypoint calculations.

### C. Experimental Protocol

We evaluate each method across 20 scenes, each containing 5 objects. The 20 scenes are evenly divided into 4 groups of 5 scenes: transparent objects only, opaque objects only, specular objects only, and mixed scenes with all three categories. For each scene, we execute the handling task until one of the following conditions is met: all objects have been successfully picked and placed, two consecutive grasp attempts fail, or no grasp candidates are detected.

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**


Fig. 3. Objects used in the experiments. Some appear in the segmentation dataset (Seen), while others do not (Unseen).

TABLE I
PARAMETERS USED IN THE EXPERIMENTS

| Description | Symbol | Value |
|---|---|---|
| Number of objects | $N_{\mathrm{ob}}$ | 5 |
| Side length of the cubic volume | $l$ | 0.30 m |
| Voxel grid resolution | $N_x, N_y, N_z$ | 40, 40, 40 |
| Radius | $r$ | 0.33 m |
| Maximum polar angle | $\theta_{\max}$ | $\pi/5$ rad |
| Azimuth and polar divisions | $U, V$ | 40, 20 |
| Number of view waypoints | $L$ | 3 |
| Occupancy threshold | $\lambda$ | 0.9 |
| Grasp score threshold | $\epsilon$ | 0.8 |
| Weight in (17) | $\gamma_1$ | 0.2 |
| Weight in (18) | $\gamma_2$ | 0.0, 0.35 |

### D. Evaluation Metrics

We evaluate each method using the following metrics:

- **Grasp success rate (%)**: the number of successful grasps divided by the total number of grasp attempts.
- **Trajectory length** (m): the average length of the hand-eye camera's trajectory per phase[3].
- **Shortest trajectory ratio (%)**: the average ratio of the hand-eye camera's trajectory length to the Euclidean distance between the initial position and the grasp position per phase.
- **Execution time** (s): the average execution time from the start to the end of each phase.

### E. Results

Table II summarizes the results in the simulation. Although HEAPGrasp w/o VP uses only RGB images, it achieves a grasp success rate of 92.7%, which is higher than that of VGN (87.1%) relying on depth images. Furthermore, HEAPGrasp shortens the trajectory length by 50% and the shortest trajectory ratio by 55% compared with HEAPGrasp w/o VP, while maintaining an 86.4% success rate.

Table III summarizes the real-robot experimental results for VGN, GraspNeRF, HEAPGrasp w/o VP, and HEAPGrasp. VGN achieves a grasp success rate of 88.5% on opaque objects, but this rate falls to 72.0% for specular and 53.8% for transparent objects. GraspNeRF also achieves a high success rate of 91.7% on opaque objects, but this also drops to 52.2% for specular and 68.2% for transparent objects. By contrast, both HEAPGrasp w/o VP and HEAPGrasp achieve grasp success rates above 92.6% across all categories. This gap in grasp success rates is mainly attributable to differences in the 3-D measurement results. Fig. 4 compares the 3-D measurement results for the same scene. Since VGN relies on depth images, its 3-D measurements are incomplete for transparent objects and contain many outliers for specular objects. Although GraspNeRF produces more accurate 3-D measurements for transparent and specular objects, some regions remain incomplete or contain outliers. Compared with VGN and GraspNeRF, HEAPGrasp w/o VP produces cleaner 3-D measurements. Furthermore, Table III shows that HEAPGrasp w/o VP and HEAPGrasp exhibit strong generalization capability to unseen objects, achieving grasp success rates of 98.0% and 96.0%, respectively.

Table III shows that GraspNeRF requires the longest trajectory at 2.33m, with a shortest trajectory ratio of 432% since it requires six viewpoints around the scene. HEAPGrasp w/o VP records a shorter trajectory of 2.03m and a shortest trajectory ratio of 323%. In contrast, HEAPGrasp further reduces the trajectory length by 52% to 0.97m and the shortest trajectory ratio by 53% to 152% compared to HEAPGrasp w/o VP. Fig. 5 visualizes the trajectories in the $x$–$y$ plane, and Fig. 6 shows the robot handling objects, both demonstrating that HEAPGrasp generates a shorter trajectory. Finally, HEAPGrasp reduces the handling execution time by 19% compared

[3]Here, a phase ends when the object is grasped.

TABLE II
RESULTS OF THE SIMULATION EXPERIMENTS

| Method | Grasp success rate (%) | Trajectory length (m) | Shortest trajectory ratio (%) |
|---|---|---|---|
| VGN | 87.1 (433/497) | $1.87 \pm 0.09$ | $325 \pm 35$ |
| HEAPGrasp w/o VP | **92.7 (458/494)** | $1.87 \pm 0.08$ | $337 \pm 32$ |
| HEAPGrasp | 86.4 (432/500) | $\mathbf{0.94 \pm 0.12}$ | $\mathbf{151 \pm 27}$ |

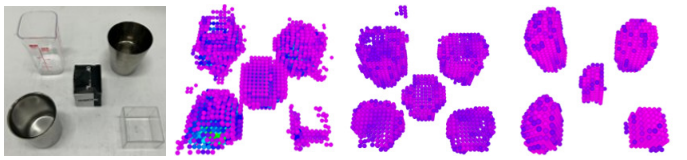



Fig. 4. Comparison of 3-D measurement results for the same scene.

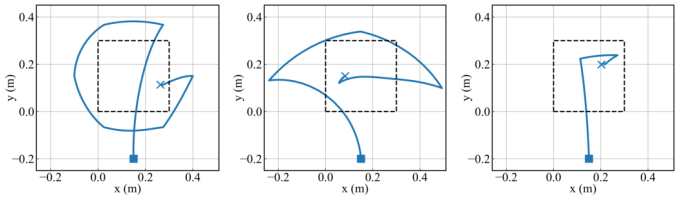



Fig. 5. Comparison of the hand-eye camera's trajectories on the $x$–$y$ plane. The dashed line represents the scene boundary, the square represents the initial position, and the cross represents the grasp position.

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**

TABLE III
RESULTS OF THE REAL ROBOT EXPERIMENTS

| Method | Scene | Grasp success rate (%) | | | Trajectory length (m) | Shortest trajectory ratio (%) | Execution time (s) |
|---|---|---|---|---|---|---|---|
| | | Seen | Unseen | Overall | | | |
| VGN [6] | Opaque | – | – | 88.5 (23/26) | 1.95 ± 0.10 | 340 ± 40 | 10.06 ± 0.35 |
| | Specular | – | – | 72.0 (18/25) | 1.97 ± 0.09 | 343 ± 37 | 10.00 ± 0.38 |
| | Transparent | – | – | 53.8 (14/26) | 1.99 ± 0.11 | 366 ± 30 | 10.12 ± 0.31 |
| | Mixed | – | – | 74.1 (20/27) | 1.99 ± 0.13 | 349 ± 46 | 10.13 ± 0.38 |
| | Overall | – | – | 72.1 (75/104) | 1.98 ± 0.11 | 348 ± 41 | 10.07 ± 0.36 |
| GraspNeRF [13] | Opaque | – | – | 91.7 (22/24) | 2.31 ± 0.06 | 419 ± 48 | 18.50 ± 0.62 |
| | Specular | – | – | 52.2 (12/23) | 2.36 ± 0.10 | 439 ± 33 | 19.00 ± 0.64 |
| | Transparent | – | – | 68.2 (15/22) | 2.36 ± 0.11 | 446 ± 50 | 19.11 ± 0.99 |
| | Mixed | – | – | 73.1 (19/26) | 2.30 ± 0.06 | 433 ± 41 | 18.79 ± 0.43 |
| | Overall | – | – | 71.6 (68/95) | 2.33 ± 0.09 | 432 ± 45 | 18.80 ± 0.72 |
| HEAPGrasp w/o VP | Opaque | **92.9 (13/14)** | **100.0 (12/12)** | **96.2 (25/26)** | 2.03 ± 0.08 | 315 ± 27 | 9.98 ± 0.37 |
| | Specular | **100.0 (13/13)** | **100.0 (12/12)** | **100.0 (25/25)** | 2.03 ± 0.07 | 322 ± 25 | 9.88 ± 0.25 |
| | Transparent | **100.0 (13/13)** | 91.7 (11/12) | **96.0 (24/25)** | 2.03 ± 0.07 | 332 ± 34 | 9.85 ± 0.18 |
| | Mixed | **100.0 (11/11)** | **100.0 (14/14)** | **100.0 (25/25)** | 2.03 ± 0.08 | 323 ± 25 | 9.94 ± 0.34 |
| | Overall | **98.0 (50/51)** | **98.0 (49/50)** | **98.0 (99/101)** | 2.03 ± 0.07 | 323 ± 29 | 9.91 ± 0.30 |
| HEAPGrasp | Opaque | **92.9 (13/14)** | 92.3 (12/13) | 92.6 (25/27) | **0.98 ± 0.13** | **153 ± 32** | **8.23 ± 1.10** |
| | Specular | **100.0 (13/13)** | **100.0 (12/12)** | **100.0 (25/25)** | **0.95 ± 0.11** | **151 ± 27** | **7.90 ± 0.81** |
| | Transparent | 92.3 (12/13) | **100.0 (10/10)** | 95.7 (22/23) | **0.92 ± 0.08** | **146 ± 23** | **7.83 ± 0.53** |
| | Mixed | **100.0 (11/11)** | 93.3 (14/15) | 96.2 (25/26) | **1.01 ± 0.17** | **159 ± 33** | **8.07 ± 0.97** |
| | Overall | 96.1 (49/51) | 96.0 (48/50) | 96.0 (97/101) | **0.97 ± 0.13** | **152 ± 29** | **8.01 ± 0.90** |

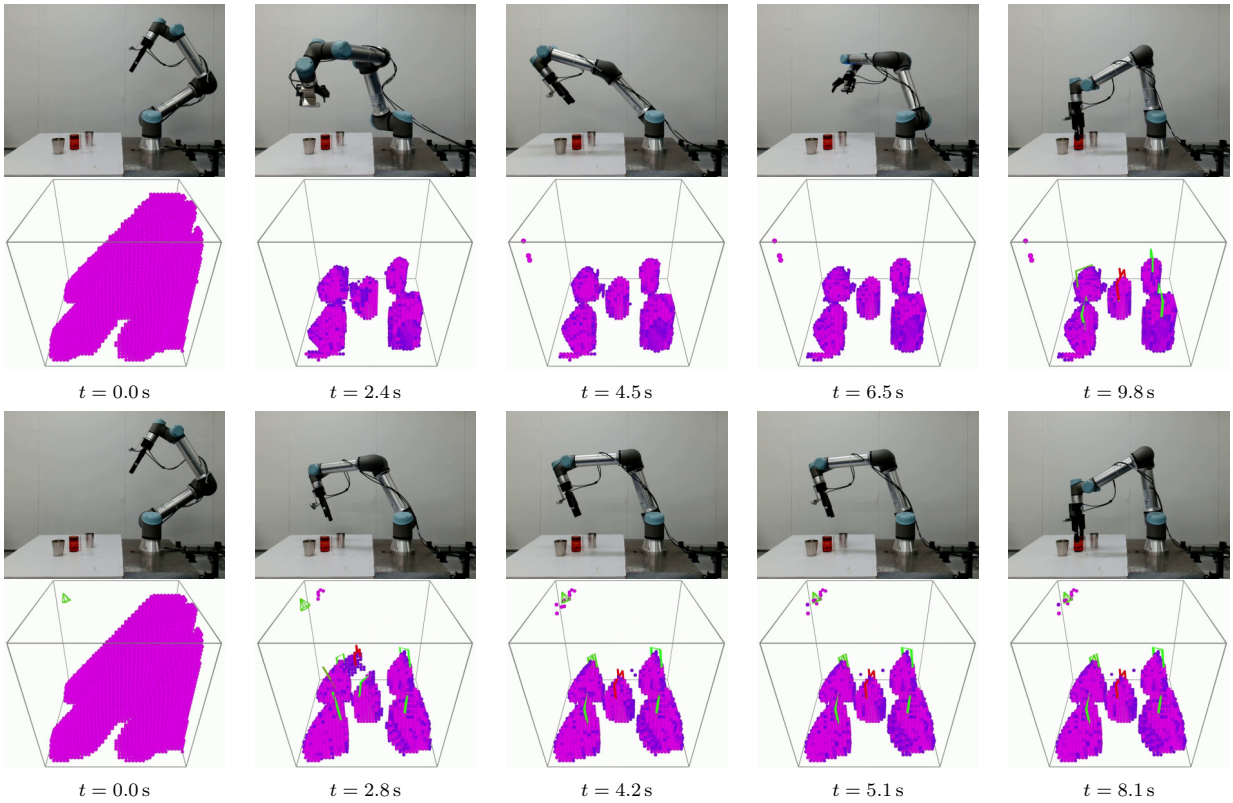



Fig. 6. Qualitative results of HEAPGrasp w/o VP (top) and HEAPGrasp (bottom). Each time step shows the manipulator during execution and the 3-D measurement result with grasp candidates in green, the target grasp in red, and view waypoints.

with HEAPGrasp w/o VP, achieving an execution time of 8.01s.

### *F. Ablation Studies and Discussion*

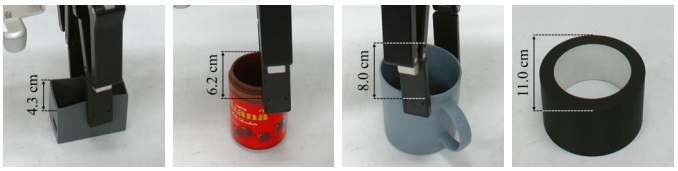
We first analyze the effect of object geometry on grasp performance, particularly the limitation of SfS in measuring concave objects. As shown in Fig. 7, the grasp-planning model generates grasp candidates when (1) the candidate lies on the visual hull with high grasp stability and (2) its opening width is smaller than the maximum opening width of the gripper used during grasp-planning model training.

Next, we analyze the effect of scene complexity by comparing HEAPGrasp w/o VP and HEAPGrasp (w/ VP) in cluttered scenes. As shown in Table IV, their success rates are 95% and 80%, decreasing by 3% and 16% compared with Table III. This drop mainly results from the degradation of

**IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.**

TABLE IV
RESULTS OF THE REAL-ROBOT EXPERIMENTS IN CLUTTERED SCENES WITH MIXED OPTICAL PROPERTIES

| Number of objects | Grasp success rate (%) | | Trajectory length (m) | | Shortest trajectory ratio (%) | | Execution time (s) | |
|---|---|---|---|---|---|---|---|---|
| | w/o VP | w/ VP | w/o VP | w/ VP | w/o VP | w/ VP | w/o VP | w/ VP |
| 6 | 80 (4/5) | 80 (4/5) | 2.15 ± 0.10 | 1.02 ± 0.08 | 308 ± 27 | 139 ± 13 | 9.99 ± 0.23 | 8.36 ± 0.72 |
| 8 | 100 (5/5) | 100 (5/5) | 2.13 ± 0.14 | 1.06 ± 0.11 | 281 ± 18 | 138 ± 15 | 9.92 ± 0.13 | 8.24 ± 0.40 |
| 10 | 100 (5/5) | 80 (4/5) | 2.07 ± 0.13 | 0.93 ± 0.11 | 281 ± 14 | 123 ± 13 | 9.84 ± 0.14 | 7.58 ± 0.48 |
| 12 | 100 (5/5) | 60 (3/5) | 2.07 ± 0.10 | 1.12 ± 0.19 | 292 ± 39 | 148 ± 23 | 9.86 ± 0.22 | 8.36 ± 0.95 |
| Overall | 95 (19/20) | 80 (16/20) | 2.10 ± 0.13 | 1.03 ± 0.14 | 290 ± 28 | 137 ± 18 | 9.90 ± 0.19 | 8.13 ± 0.71 |




(a) (b) (c) (d)

Fig. 7. Grasps for concave objects. The gripper's maximum opening width during grasp-planning model training was 8 cm. Therefore, grasp candidates were generated for objects (a)–(c) but not for (d).

TABLE V
EFFECT OF SEGMENTATION NOISE ON MEASUREMENT AND GRASPING

| Noise level | MSE | IoU | Successfully grasped objects (%) |
|---|---|---|---|
| 0.01 | 0.02 | 0.99 | 90.2 (451/500) |
| 0.05 | 0.12 | 0.92 | 92.8 (464/500) |
| 0.10 | 0.93 | 0.44 | 92.8 (464/500) |
| 0.15 | 1.63 | 0.03 | 68.6 (343/500) |

measurement accuracy in cluttered scenes, yet HEAPGrasp still achieves 80% success by grasping objects near scene boundaries, as SfS reconstructs the visual hull.

Finally, we examine the effect of segmentation accuracy on 3-D measurement and grasping performance. We evaluate HEAPGrasp w/o VP by varying the noise level—the probability that each pixel in a binary mask is randomly flipped—in simulation, and evaluate the mean squared error (MSE) and intersection over union (IoU) between the high-confidence occupancy grids obtained from noisy and noise-free masks, as well as the grasp success rate. As shown in Table V, the IoU remains 0.44 even with 10% noise, while the success rate stays high at 92.8%, demonstrating robustness to segmentation errors.

## VII. CONCLUSION

This paper proposes HEAPGrasp—Hand-Eye Active Perception to Grasp objects with diverse optical properties. HEAPGrasp segments objects in multi-view RGB images and measures their shape via Shape from Silhouette. The view planning module reduces multi-view capture time by optimizing a cost function balancing 3-D measurement accuracy and hand-eye camera's trajectory length. Experiments show that HEAPGrasp achieves a 96.0% grasp success rate, while reducing the hand-eye camera trajectory length by 52% and handling execution time by 19% compared to a baseline that circles around the scene for 3-D measurement.

## REFERENCES

[1] Photoneo s.r.o., "PhoXi 3D Scanner," Accessed: Aug. 9, 2025. [Online]. Available: https://www.photoneo.com/phoxi-3d-scanner/

[2] A. Laurentini, "The Visual Hull Concept for Silhouette-Based Image Understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 2, pp. 150–162, Feb. 1994.

[3] C.-T. Wen, S. Arai, J. Kinugawa, and K. Kosuge, "Tactile Servoing Based Pressure Distribution Control of a Manipulator Using a Convolutional Neural Network," *IEEE Access*, vol. 9, pp. 117132–117139, 2021.

[4] J. Mahler et al., "Learning ambidextrous robot grasping policies," *Sci. Robot.*, vol. 4, no. 26, 2019, Art. no. eaau4984.

[5] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser, "TossingBot: Learning to Throw Arbitrary Objects With Residual Physics," *IEEE Trans. Robot.*, vol. 36, no. 4, pp. 1307–1319, Aug. 2020.

[6] M. Breyer, J. J. Chung, L. Ott, R. Siegwart, and J. Nieto, "Volumetric Grasping Network: Real-time 6 DoF Grasp Detection in Clutter," in *Proc. Conf. Robot. Learn.*, 2021, pp. 1602–1611.

[7] B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn.*, 1996, pp. 303–312.

[8] Y. Qian, M. Gong, and Y.-H. Yang, "3D Reconstruction of Transparent Objects with Position–Normal Consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4369–4377.

[9] B. Wu, Y. Zhou, Y. Qian, M. Gong, and H. Huang, "Full 3D Reconstruction of Transparent Objects," *ACM Trans. Graph.*, vol. 37, no. 4, Aug. 2018, Art. no. 103.

[10] S. Sajjan et al., "ClearGrasp: 3D Shape Estimation of Transparent Objects for Manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 3634–3642.

[11] Y. Zhang and T. Funkhouser, "Deep Depth Completion of a Single RGB-D Image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 175–185.

[12] J. Shi et al., "ASGrasp: Generalizable Transparent Object Reconstruction and 6-DoF Grasp Detection from RGB-D Active Stereo Camera," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 5441–5447.

[13] Q. Dai, Y. Zhu, Y. Geng, C. Ruan, J. Zhang, and H. Wang, "GraspNeRF: Multiview-based 6-DoF Grasp Detection for Transparent and Specular Objects Using Generalizable NeRF," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2023, pp. 1757–1763.

[14] Y. Liu et al., "Neural Rays for Occlusion-aware Image-based Rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7814–7823.

[15] J. Kerr et al., "Evo-NeRF: Evolving NeRF for Sequential Robot Grasping of Transparent Objects," in *Proc. Conf. Robot. Learn.*, 2023, pp. 353–367.

[16] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 405–421.

[17] M. Breyer, L. Ott, R. Siegwart, and J. J. Chung, "Closed-Loop Next-Best-View Planning for Target-Driven Grasping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2022, pp. 1411–1416.

[18] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.