

Robust Bayesian Scene Reconstruction With Retrieval-Augmented Priors for Precise Grasping and Planning

Herbert Wright ^{1b}, Weiming Zhi ^{1b}, Martin Matak ^{1b}, Matthew Johnson-Roberson ^{1b}, *Member, IEEE*, and Tucker Hermans ^{1b}, *Senior Member, IEEE*

Abstract—Constructing 3D representations of object geometry is critical for many robotics tasks, particularly manipulation problems. These representations must be built from potentially noisy partial observations. In this work, we focus on the problem of reconstructing a multi-object scene from a single RGBD image using a fixed camera. Traditional scene representation methods generally cannot infer the geometry of unobserved regions of the objects in the image. Attempts have been made to leverage deep learning to train on a dataset of known objects and representations, and then generalize to new observations. However, this can be brittle to noisy real-world observations and objects not contained in the dataset, and do not provide well-calibrated reconstruction confidences. We propose BRRP, a reconstruction method that leverages preexisting mesh datasets to build an informative prior during robust probabilistic reconstruction. We introduce the concept of a retrieval-augmented prior, where we retrieve relevant components of our prior distribution from a database of objects during inference. The resulting prior enables estimation of the geometry of occluded portions of the in-scene objects. Our method produces a distribution over object shape that can be used for reconstruction and measuring uncertainty. We evaluate our method in both simulated scenes and in the real world. We demonstrate the robustness of our method against deep learning-only approaches while being more accurate than a method without an informative prior. Through real-world experiments, we particularly highlight the capability of BRRP to enable successful dexterous manipulation in clutter.

Index Terms—Perception for grasping and manipulation, probabilistic inference.

Received 30 June 2025; accepted 15 October 2025. Date of publication 12 November 2025; date of current version 18 November 2025. This article was recommended for publication by Associate Editor J. Zhu and Editor M. Vincze upon evaluation of the reviewers' comments. This work was supported by NSF under Award 2321852. (*Corresponding author: Herbert Wright.*)

Herbert Wright and Martin Matak are with the Robotics Center and Kahlert School of Computing, University of Utah, Salt Lake City, UT 84112 USA (e-mail: herbiew@seas.upenn.edu).

Weiming Zhi is with the College of Connected Computing, Vanderbilt, TN 37212 USA, and also with the School of Computer Science, University of Sydney, Sydney 2008, Australia.

Matthew Johnson-Roberson is with the College of Connected Computing, Vanderbilt, TN 37212 USA, and also with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

Tucker Hermans is with the Robotics Center and Kahlert School of Computing, University of Utah, Salt Lake City, UT 84112 USA, and also with NVIDIA Corporation, Seattle, WA 98105 USA.

This article has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2025.3632101>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3632101

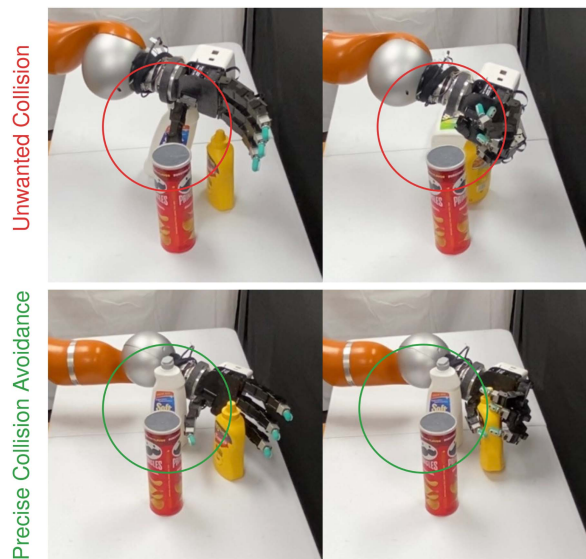


Fig. 1. Poor understanding of geometry can lead to unwanted collisions during robotic grasping (top), whereas robust, accurate reconstructions can enable precise collision avoidance during manipulation (bottom).

I. INTRODUCTION

TO OPERATE autonomously in novel settings, a robot must construct rich internal representations of its environment. These representations need to be particularly fine-grained for robotic manipulation, which often requires closely interacting with and avoiding objects. Consider the task of robotic grasping in clutter; a robot needs to avoid unwanted collisions, while also operating in close proximity to one or more objects to make the necessary contact for the grasp. State of the art grasp synthesis methods often rely on motion planners to reach grasp poses in clutter [1], [2], [3], [4], which need accurate 3D geometry for collision checking. Thus, a lack of geometric understanding can lead to unwanted collisions causing task failure or worse damage to the robot or environment. This is pictured in Fig. 1. Many grasping and motion planning algorithms require explicit 3D representations of the scene's geometry. These representations must be built from observations that are both noisy and, due to occlusion, only contain partial information of the scene. In our case, we focus on the problem of robustly building a 3D representation of multi-object scenes from a single RGBD camera image.

One approach to this problem is to train a neural network to *predict* the full geometry of an object given a partial view. Such approaches are able to use existing mesh datasets to more accurately infer the occluded backside of objects. Unfortunately, these approaches also tend to have a number of problems when used on real-world depth cameras. The presence of unknown objects, significant occlusion, noisy point clouds, or inaccurate segmentation prevents these methods from being reliably deployed in less-structured environments.

Deep learning methods for 3D reconstruction generally lack a sense of uncertainty about the shape of the object. This can be particularly detrimental when an object in the scene is only partially observed and we seek to factor in the geometry of observed and occluded regions. Such uncertainty can enable safer and more robust operation in a range of downstream tasks, such as robot grasping [1], [2], safe motion generation and active learning.

Another common approach to building full 3D representations from a partial view is to build a representation solely from the observed data without considering prior information from mesh datasets. A common example of this is the Gaussian Process Implicit Surface [5] model. A more recent example is V-PRISM [6], which probabilistically maps tabletop scenes without using prior information. These reconstruction methods are more robust to unknown objects because they do not rely on any training distribution. However, this also means that they cannot reconstruct the unobserved backside of objects. Humans have the remarkable ability to infer the geometry of scenes based on prior experience, we seek to imbue robots with the same capability.

In this work, we introduce a novel Bayesian approach for robustly reconstructing multi-object tabletop scenes by leveraging object-level shape priors. We present **B**ayesian **R**econstruction with **R**etrieval-augmented **P**riors (BRRP). BRRP is resilient to many of the pitfalls of learning-based methods while still being able to leverage an *informative prior* to more accurately reconstruct known objects. To further improve efficiency, we introduce the idea of a retrieval-augmented prior, where we retrieve relevant components of our prior distribution based on classification results. We begin with an observed RGBD image with corresponding instance segmentations. Then, we compute an identification result that predicts which objects from our database should be retrieved to use as a prior during reconstruction. We use this prior along with a sampled likelihood to infer a posterior distribution over object shapes. Because we solve for a distribution, we can recover principled uncertainty about each object's shape. In practice, we use a pre-existing foundation model to perform the identification and use registration to make our prior pose invariant. Fig. 2 shows an example scene reconstruction, where one of the objects (the orange juice) is decidedly out-of-distribution and our method is still able to reconstruct it according to the observed point cloud.

We conduct experiments on BRRP in both procedurally generated scenes and in the real world. We quantitatively show that BRRP results in accurate reconstructions and that BRRP is robust to unknown objects in the generated scenes. We qualitatively show that BRRP is robust to noisy real world scenes collected from an RGBD camera, and is able to capture uncertainty within the reconstructions. We also demonstrate downstream improvement with real-world grasping experiments, where the robot is able to more robustly arrive at an accurate pre-grasp pose.

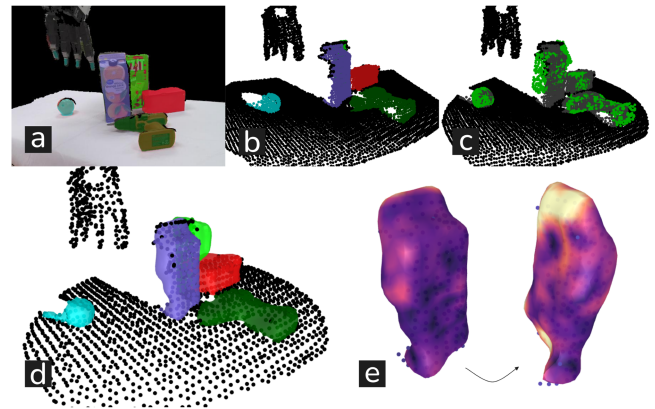


Fig. 2. BRRP (a), (b) takes an input segmented RGBD image, and (c) retrieves objects to use as a prior to (d) reconstruct the scene as well as (e) capture principled *uncertainty* about object shape, in this case, an out-of-distribution orange juice carton.

Our three primary contributions can be summarized as:

- 1) The formulation of retrieval-augmented priors for Bayesian inference.
- 2) A formulation for a prior over Hilbert maps with pose and scale invariance
- 3) A novel, robust *Bayesian* scene reconstruction method that utilizes prior information from existing mesh datasets

We organize our letter as follows. We overview related works in Section II. In Section III, we cover mathematical preliminaries for our method. Section IV introduces retrieval-augmented priors. BRRP is introduced in Section V. Experiments are found in Section VI and a conclusion in Section VII

II. RELATED WORKS

3D Representations. There are many different ways of representing 3D geometry of a scene. In the mapping literature, memory-intensive voxels are used as a representation of the environment. Hilbert maps [7], on the other hand, are a continuous occupancy map that takes the form of a linear function over some hinge point feature space. Hilbert map representations have also been extended to Bayesian Hilbert maps of various forms [8], [9], [10]. Neural implicit functions have also been used to represent continuous 3D geometry [11], [12], [13]. Other representations are built using differentiable rendering and combining multiple views [14], [15]. Foundation models have also been developed for this task in [16] and applied to robotics [17]. We instead focus on the harder problem of reconstruction from only a single-view.

3D Reconstruction with Deep Learning Priors. Many methods have been proposed as ways to leverage deep learning to reconstruct scenes or objects. While some methods aim to predict object shape from RGB data only [18], [19], [20], we instead focus on using depth measurements during reconstruction. DeepSDF [11] is a method to reconstruct an object by running inference-time optimization to recover a latent code for a neural implicit function. In the context of robotics, [21] extends DeepSDF to have uncertainty-awareness. Other work, such as occupancy networks [12] or PointSDF [22] try to directly predict such a latent code without inference-time optimization. Deep learning has also been leveraged to learn kernels, which are used to construct a continuous signed distance function [23], [24].

Language is also used during reconstruction in [25] and [26]. Another work uses a voxel-to-voxel variational autoencoder conditioned on bounding boxes [27]. While these works typically focus on single object scenes, other work focuses on reconstructing multi-object scenes. For example, [28] learns to reconstruct a voxel representation, leveraging different input channels to account for occlusion. Alternatively, work such as [29] aim to reconstruct the scene-wide geometry without bothering to separate each object. Instead, we focus on reconstructing *individual* objects in a manner robust to occlusion. In practice, these deep learning approaches can struggle to reconstruct noisy scenes with multiple, highly occluded, unknown objects on real world depth cameras. These methods can also struggle when segmentation is inaccurate.

3D Reconstruction without Deep Learning Priors. There are also many approaches to perform probabilistic 3D reconstruction without deep learning. Some of such methods for reconstruction use informative prior information by assuming fixed classes of objects, such as 3DP3 [30]. Other methods use an uninformative prior, such as Gaussian process implicit surfaces (GPIS) [5]. While there is an extension of GPIS to a slightly more informative prior [31], the only priors that can be enforced are specifically spherical, ellipsoidal, cylindrical, or planar priors. We instead derive our prior from pre-existing mesh data. V-PRISM [6] is another method that probabilistically *maps* the scene using a multi-class framing.

Using Reconstructions in Manipulation. 3D reconstruction methods have seen extensive use in manipulation. In [22], PointSDF provides collision constraints during grasping. PointSDF is also utilized in [3], where tactile sensors are used along with the reconstruction during grasping. A learning-based voxel representation is used for grasping in [32]. Neural shape completion is also used during the anthropomorphic grasping pipeline proposed in [33]. GPIS is also a common representation for manipulation applications. Some recent work has utilized the uncertainty from GPIS representations during grasp selection [1], [2]. We believe BRRP provides principled uncertainty measurements that can similarly be utilized in downstream manipulation tasks.

III. BACKGROUND

A. Hilbert Maps

A Hilbert map [7] is a *continuous* occupancy map. It represents the environment by a continuous function that is defined by a linear function of a fixed feature transform. Typically this feature transform is induced by a set of *hinge points*, $\{\mathbf{h}_1, \dots, \mathbf{h}_H\} \subset \mathbb{R}^3$ and a translation-invariant kernel $k(d)$. The transform is then defined as:

$$\phi(\mathbf{x}) = [1, k(\mathbf{x} - \mathbf{h}_1), \dots, k(\mathbf{x} - \mathbf{h}_H)]^\top.$$

Typically, a Gaussian kernel is used and hinge points are placed in an evenly-spaced grid. An occupancy map can then be defined by a single weight vector, $\mathbf{w} \in \mathbb{R}^{H+1}$, as such:

$$m(\mathbf{x}) = \sigma(\mathbf{w}^\top \phi(\mathbf{x})).$$

To recover the weight vector corresponding to a given depth observation, *negative sampling* is performed along the unoccupied portions of the depth rays. These negative samples are assigned a label of unoccupied and the points at the end of the ray are labeled as occupied. Then, stochastic gradient descent (SGD)

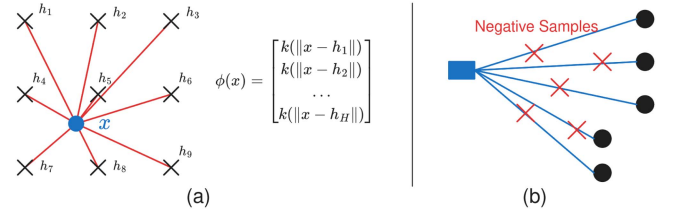


Fig. 3. (a) A *hinge point* feature transform induced by a set of hinge points is used by Hilbert maps [7]; (b) these maps are built by first sampling *negative samples* along the unoccupied portions of the camera ray.

is performed on the binary cross entropy (BCE) of the negative samples and terminal points of the ray. The binary cross entropy measures the *likelihood* of the samples, and is defined as:

$$\text{BCE}(y, \mathbf{w}^\top \phi(\mathbf{x})) = \begin{cases} -\ln[\sigma(\mathbf{w}^\top \phi(\mathbf{x}))], & y = 1 \\ -\ln[1 - \sigma(\mathbf{w}^\top \phi(\mathbf{x}))], & y = 0 \end{cases} \quad (1)$$

Fig. 3 shows an illustration of both the hinge point feature transform and the sampling used for creating Hilbert maps.

Hilbert maps have previously been extended to Bayesian Hilbert maps, where a distribution over weights is modeled as a multivariate Gaussian [8], [9], [10]. There is also a multi-class variant that defines a weight *matrix* with Gaussian rows described in [6]. In this work, we adopt the Hilbert map representation, but model the distribution over weights as a collection of particles. This allows our method the capability to capture irregular, non-Gaussian posterior distributions.

B. Stein Variational Gradient Descent

Stein Variational Gradient Descent (SVGD) [34] is an algorithm for variational inference that closely resembles gradient descent. The general problem of variational inference is to find a distribution $q^* \in \mathcal{Q}$ that is close to some target distribution p . Usually, this takes the form of an optimization problem over the Kullback-Leibler (KL) divergence:

$$q^* = \arg \min_{q \in \mathcal{Q}} \mathbb{KL}(q||p).$$

SVGD aims to iteratively transform q in descent directions of the KL divergence in a d -dimensional reproducing kernel Hilbert space, \mathcal{H}^d . Because this Hilbert space is a space of functions, a descent direction requires deriving the *functional gradient* of our KL divergence objective.

Theorem 1: From [34]. Let $T(\mathbf{x}) = \mathbf{x} + f(\mathbf{x})$, where $f \in \mathcal{H}^d$ and $q_{[T]}$ is the density of random variable $\mathbf{z} = T(\mathbf{x})$ when $\mathbf{x} \sim q$. Then

$$\nabla_f \mathbb{KL}(q_{[T]}||p)|_{f=0} = -g_{q,p}^*,$$

where $g_{q,p}^* = \mathbb{E}_{\mathbf{x} \sim q(\mathbf{x})} [k(\mathbf{x}, \cdot) \nabla_{\mathbf{x}} \ln p(\mathbf{x}) + \nabla_{\mathbf{x}} k(\mathbf{x}, \cdot)]$.

In SVGD, q is approximated by a set of particles $\mathbf{x}_1^{(0)}, \dots, \mathbf{x}_P^{(0)} \sim q(\mathbf{x})$. This can be used to approximate the gradient in Theorem 1 with \hat{g}^* :

$$\hat{g}^*(\mathbf{x}) = \frac{1}{P} \sum_{i=1}^P k(\mathbf{x}_i, \mathbf{x}) \nabla_{\mathbf{x}_i} \ln p(\mathbf{x}_i) + \nabla_{\mathbf{x}_i} k(\mathbf{x}, \cdot). \quad (2)$$

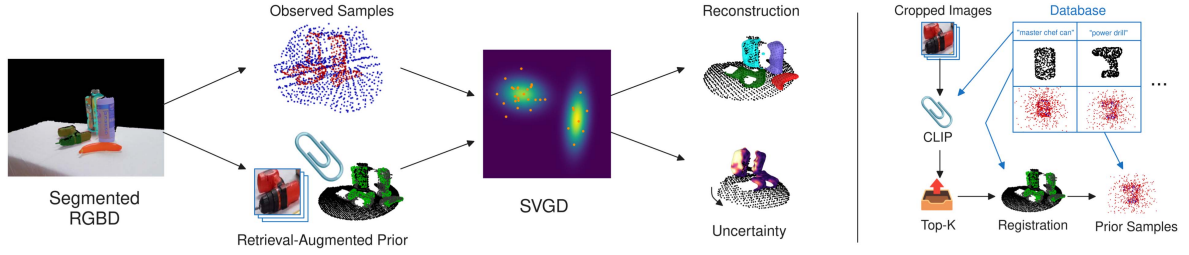


Fig. 4. *Left:* Overview of BRRP method. We begin with a segmented RGBD image and compute both a retrieval-augmented prior (Sections V-A and V-B) as well as a likelihood from negative sampling (Section V-C). Then, we combine both in a Bayesian manner via SVGD (Section V-D). *Right:* An overview of our retrieval-augmented priors for reconstruction. We use CLIP to retrieve the top- k relevant objects in our database (Section V-B), then we perform registration and retrieve the corresponding prior samples (Section V-A).

The particles can then be iteratively updated according to \hat{g}^* in Equation (2) with:

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \epsilon \hat{g}^*(\mathbf{x}_i^{(t)})$$

The result of these iterations is that the set of particles converges to an approximation of the target distribution p . Importantly, Equation (2) only relies on the gradient of the log of p , which means we can perform variational inference to an unnormalized distribution. Such unnormalized distributions are common in many Bayesian inference problems.

IV. RETRIEVAL-AUGMENTED PRIORS

Retrieval-augmented generation [35] was originally introduced in the context of improving language generation. The work has served as inspiration for an approach to affordance-prediction in [36]. In our case, we draw inspiration from retrieval-augmented generation, but we use the retrieved results to improve efficiency in certain *explicit* formulations for prior density functions during Bayesian inference.

To motivate retrieval-augmented priors, consider the problem of Bayesian inference with a mixture model acting as the prior distribution. Given some data, we would like to infer a posterior distribution over hypotheses. Defining a mixture model prior over components $C = \{c_1, \dots, c_{N_c}\}$ gives:

$$P(H|D) \propto P(D|H) \sum_{c \in C} P(H|c). \quad (3)$$

If our prior distribution has a lot of components, it may be inefficient to fully evaluate. This could be a serious problem for algorithms like SVGD, which requires iteratively computing the gradient of both the likelihood and prior. Inspired by [35], the insight behind retrieval-augmented priors is to determine which subset of the prior distribution components to retrieve and use given some detection result R . Conditioning on this detection result, we have a new posterior distribution, $P(H|D, R)$. Making an independence assumption,

$$P(H|D, R) \propto P(D|H) \cdot \mathbb{E}_{c \sim P(c|R)} [P(H|c)].$$

Comparing to Equation (3), the expectation now replaces the true prior. Then, we can use a top- k approximation for the expectation:

$$P(H|D, R) \propto P(D|H) \sum_{c \in \text{topk}} P(H|c) P(c|R) \quad (4)$$

Thus, we only evaluate a subset of the prior components.

V. THE BRRP METHOD

Our method takes a single RGBD image and produces reconstructions for each object in the scene. We treat the problem as a Bayesian inference problem over an observation described by negative samples. We incorporate prior information on the shape of the object by leveraging retrieval-augmented priors introduced in Section IV. We use the pretrained CLIP [37] foundation model to determine which objects to retrieve and define our object-specific priors by a registered set of pre-computed samples from the stored mesh. We then use SVGD to optimize for a set of samples over map weights. We can generate predicted reconstructions by taking the expected occupancy over our weights for a given location. Fig. 4 shows a visual overview of our method.

In Section V-A, we explain how we leverage pre-existing mesh assets to create a prior that is robust to different poses and scales. Then, in Section V-B, we explain how we utilize the retrieval-augmented priors paradigm to retrieve relevant objects in the prior. The specific negative sampling is explained in Section V-C. Then we give the specific SVGD objective used in Section V-D.

A. Negative Samples as Reconstruction Priors

We want to leverage existing mesh assets as our priors during Bayesian reconstruction. We define our prior as a mixture model with different objects serving as each component, c_1, \dots, c_{N_c} . Because there is not a direct way to convert a mesh into a Hilbert map prior, we instead *sample* points $\tilde{\mathbf{x}}_{c,1}, \dots, \tilde{\mathbf{x}}_{c,Q} \in \mathbb{R}^3$ around each object c 's mesh. We refer to these samples as the *prior samples*. We give them labels $\tilde{y}_{c,1}, \dots, \tilde{y}_{c,Q} \in \{-1, 1\}$ determined by whether they are outside or inside the mesh. Then we simply define our prior using this data and a Gaussian prior over weight norm:

$$\hat{P}(\mathbf{w}|c) := P(\{\tilde{y}_{c,i}\} | \{\tilde{\mathbf{x}}_{c,i}\}, \mathbf{w}) P(\mathbf{w}) \quad (5)$$

$$\propto \exp(\lambda \|\mathbf{w}\|^2) \prod_{i=1}^Q \exp(\text{BCE}(\tilde{y}_{c,i}, \mathbf{w}^\top \phi(\tilde{\mathbf{x}}_{c,i}))). \quad (6)$$

In order to enforce pose-invariance for each prior component, we first register a small stored point cloud of the respective object to the observed points and then transform the prior samples to this reference frame. In practice we use RANSAC [38] and the FPFH features from [39] to perform registration. In order to also have scale invariance, we do a linear scan over 10 different scales and

select the the scale that resulted in the most inlier pairs from the registration.

B. Retrieval-Augmented Priors for Hilbert Maps

Because it would be inefficient to register all meshes that are part of the prior mixture model, we propose using the retrieval-augmented prior approach introduced in Section IV. In order to determine which objects to use, we need to compute $P(c|R)$ from Equation (4). In our case, we use CLIP [37] as a zero-shot classifier for our different objects. For each object in our prior, we store a small textual description of the object. These descriptions are then used as classes for CLIP to classify each segmented object. In order to make sure CLIP knows which object we are targeting, we crop the RGB image to fit the predicted segmentation of each object. One could consider these cropped images to be the R in our $P(c|R)$ formulation. We feed the cropped images as input into CLIP. The resulting distribution over objects predicted by CLIP becomes our $P(c|R)$ for each component.

Once we have the probability of each object, we retrieve and register the stored point clouds of the top- k objects. After registration, we retrieve the prior samples corresponding to these objects to define our top- k component prior approximation according to Equation (5).

C. Negative Sampling

We adopt the negative sampling method introduced in [6]. The negative sampling method makes the assumption that all objects are lying on or above a planar surface. We begin by labeling the points segmented to each object as occupied for that object. Next, we perform stratified sampling along each camera ray near each object to recover a set of negatively sampled points, labeled as unoccupied. Then, we use RANSAC over points not segmented to any object to recover the flat surface all objects are resting on. This plane is used to randomly sample points in a sphere underneath each object that are near the object. We also label these points as occupied. Finally, we use grid subsampling from [40] to reduce the number of points and increase uniformity of sampled points. We refer to these points and labels as *observed samples* and denote them as $\{\mathbf{x}_i\}_{i \in [S]}$, $\{y_i\}_{i \in [S]}$. The entire negative sampling process can be easily parallelized for efficient computation.

D. SVGD Reconstruction

Once we have retrieved our prior samples and computed our observed samples, we can perform optimization-based reconstruction with SVGD. Given both sets of samples and our prior definition from Equation (5), we have the following posterior distribution,

$$P(\{y_{c,i}\}|\{\mathbf{x}_{c,i}\}, \mathbf{w})P(\mathbf{w}) \sum_{c \in \text{topk}} P(c|R)P(\{\tilde{y}_{c,i}\}|\{\tilde{\mathbf{x}}_{c,i}\}, \mathbf{w}).$$

Taking the log and applying Equation (6) gives the objective:

$$\ell = \frac{\lambda_1}{S} \sum_{i=1}^S \text{BCE}(y_i, \mathbf{w}^\top \phi(\mathbf{x}_i)) \quad (7)$$

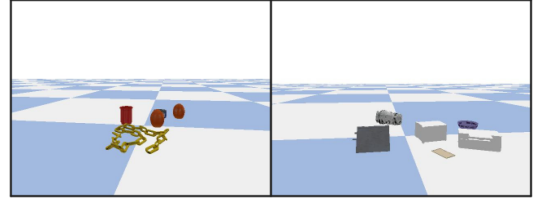


Fig. 5. Sample images of procedurally generated scenes used to evaluate BRRP. **Left:** a YCB scene. **Right:** a ShapeNet scene.

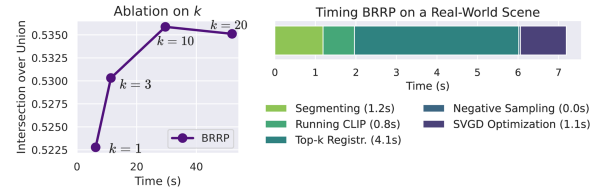


Fig. 6. Results pertaining to the speed of BRRP on an NVIDIA RTX 4070 GPU. **Left:** Ablation on k for YCB scenes highlights the accuracy-speed tradeoff in retrieval-augmented priors. We use $k = 3$ for all other experiments. **Right:** Timing breakdown of BRRP on a real-world scene with 4 objects.

$$+ \frac{\lambda_2}{K} \sum_{c \in \text{topk}} P(c|R) \ln \left[\frac{1}{Q} \sum_{i=1}^Q \exp(\text{BCE}(\tilde{y}_{c,i}, \mathbf{w}^\top \phi(\tilde{\mathbf{x}}_{c,i}))) \right] \quad (8)$$

$$+ \lambda_3 \|\mathbf{w}\|^2, \quad (9)$$

where K is the number of objects retrieved for the prior. For practical reasons, we introduce multipliers to each term as hyperparameters during optimization and use means instead of sums for the first two terms. This objective is used as the log of the target distribution, $\ln P(\mathbf{w})$, in Equation (2), where we also adopt the original median kernel suggested in [34]. We also opt to use SVGD in a stochastic manner, where both the observed samples and query samples are mini-batched.

From a non-probabilistic standpoint, one can interpret Equation (7) as the likelihood of the observed data, Equation (8) as the object shape prior, and Equation (9) as regularization.

VI. EXPERIMENTS

In this section, we aim to experimentally validate the following claims: (1) BRRP is more *robust* than deep learning methods; (2) BRRP is more *accurate* than methods that use uninformative priors; (3) BRRP can capture principled uncertainty. (4) BRRP's robustness improves the downstream success of grasping. First, we provide details on the experiments, such as the baselines and metrics in Section VI-A, then we show then show the results and analyses in Section VI-B.

A. Experimental Details

BRRP Implementation: We use a set of 50 objects from the YCB dataset [41] to act as the prior for our experiments with BRRP. We implement the method in PyTorch. We also display results for the speed of running BRRP on an NVIDIA RTX 4070 GPU in Fig. 6. We use $k = 3$ for all experiments, but note that our method's performance is fairly robust to the k parameter (Fig. 6).

Baselines: We compare our work against two main baselines, V-PRISM [6] and a version of PointSDF [22] that predicts occupancy and is trained on ShapeNet [42] scenes. V-PRISM is a probabilistic mapping method that uses an uninformative prior. This means that it is robust to novel objects, but doesn't accurately reconstruct object backsides. We refer to this baseline as **V-PRISM**. In contrast, PointSDF is a learning-based method. This means it can leverage prior information from mesh datasets to accurately reconstruct the backside, but can suffer in performance under significant distributional shift. We train two versions of PointSDF. One is trained on scenes containing a wide variety of Shapenet objects, and another trained only on scenes containing the 50 YCB objects in our BRRP prior. We refer to this baseline as **PointSDF**, and generally use the ShapeNet version when the training dataset is not mentioned. When reconstructing meshes with PointSDF, a level set of $\tau = 0.3$ is used.

Procedurally Generated Scenes: We use the generated scenes from [6] to evaluate our method. These scenes are constructed with objects from ShapeNet [42], YCB [41], and Objaverse [43] datasets. There are 100 multi-object scenes for each mesh dataset. Each scene contains up to 10 objects. Some meshes in the Objaverse and ShapeNet scenes did not have correctly rendered textures and were instead rendered as plain white objects. Fig. 5 contains two example images of these procedurally generated scenes. We use the ground-truth masks for experiments on procedural scenes. We also conduct an experiment on robustness where we perturb the masks of the ShapeNet scenes by two pixels and evaluate reconstructions.

We evaluate performance on the procedurally generated scenes with two metrics: intersection over union (IoU) and chamfer distance. We refer readers to [6] for further explanation of these metrics.

Real World Scenes: In order to showcase robustness to real-world noise, we evaluate on real world scenes collected with a Kinect depth camera. In order to obtain instance segmentations, we use Grounded SAM [44] along with some depth filters. We include timing of performing this segmentation in the real-world timing breakdown in Fig. 6. We evaluate on these real world scenes qualitatively with images of scene reconstruction and visualizing surface uncertainty.

Grasping Experiment: To understand the downstream impact of our method, we assess its effectiveness on the problem of dexterous grasping from a fixed RGB-D camera. We place an object on the table, segment it, reconstruct its mesh, plan a grasp trajectory, and execute it open-loop. Fig. 9 shows the 10 objects used.

For segmentation, we use Grounded SAM [44] with the prompt *object that can be picked up with one hand*. Meshes are reconstructed with marching cubes on either BRRP or PointSDF reconstructions. We implement a grasp planner [4] which takes a segmented point cloud as input and moves the robot to a pregrasp configuration that is collision-free according to the reconstruction while also avoiding obstacles according to the scene reconstruction. The robot then closes the hand and lifts the arm. An attempt is successful if the object is lifted. We use KUKA LBR4 with Allegro hand in the experiment. We place each of the 10 objects across 5 different poses on the table, resulting in 50 executed grasp trajectories for each method. We select stable poses where objects are reachable by the robot, visible to the camera, and remain stationary, while ensuring diversity in positions and orientations. We report the grasp success rate per method across all 50 trajectories.

TABLE I
INTERSECTION OVER UNION (IoU) ON PROCEDURALLY GENERATED SCENES FROM TWO DIFFERENT MESH DATASETS

Method	ShapeNet Scenes (IoU \uparrow)	YCB Scenes (IoU \uparrow)
V-PRISM [6]	0.3092	0.5003
PointSDF (ShapeNet) [22]	0.3600	0.4601
PointSDF (YCB) [22]	0.2426	0.5088
BRRP (ours)	0.3124	0.5277

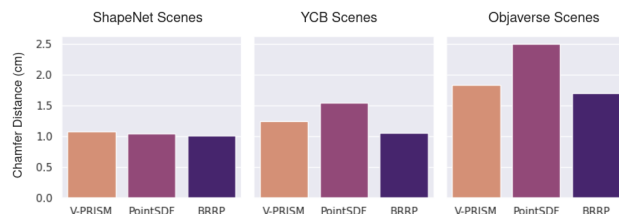


Fig. 7. Chamfer distances (lower is better) for various methods across the procedurally generated scenes. Values are reported in centimeters. BRRP has the lowest chamfer distance on each dataset.

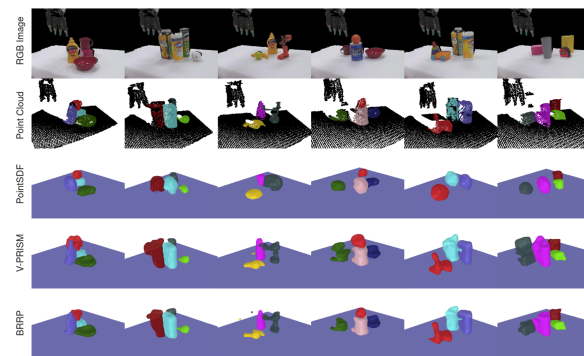


Fig. 8. Qualitative comparison of BRRP and our baselines. PointSDF tends to predict a spherical shape for many non-spherical objects. V-PRISM can sometimes predict occupancy in portions of the scene that are not occupied. Our method is more robust and can more accurately reconstruct the scenes.

B. Results & Discussion

In Table I, we display the IoU results from procedurally generated scenes. The chamfer distances for the procedurally generated scenes is shown in Fig. 7. The qualitative reconstructions on real world scenes can be seen in Fig. 8.

Insight 1: BRRP is more accurate than a method with an uninformative prior.

As showcased in Table I, BRRP outperforms V-PRISM on both sets of procedurally generated scenes. It has the highest IoU improvement from V-PRISM on the YCB scenes. A similar pattern can be seen in the chamfer distances in Fig. 7, where BRRP consistently outperforms V-PRISM, with the biggest gap of 0.19 (cm) occurring on the YCB scenes. This makes sense because BRRP uses a subset of the YCB objects as its prior.

We can see this improvement qualitatively in Fig. 8. While BRRP and V-PRISM are comparable for most objects, there exist certain objects that V-PRISM predicts to occupy a large portion of space that the object *doesn't* occupy. BRRP is able to more accurately reconstruct these objects. The clearest example of this is the dark green object in the right-most scene in Fig. 8. Both of these quantitative and qualitative results suggest BRRP is generally more accurate than V-PRISM. It is the most accurate when evaluated on objects in its prior distribution.

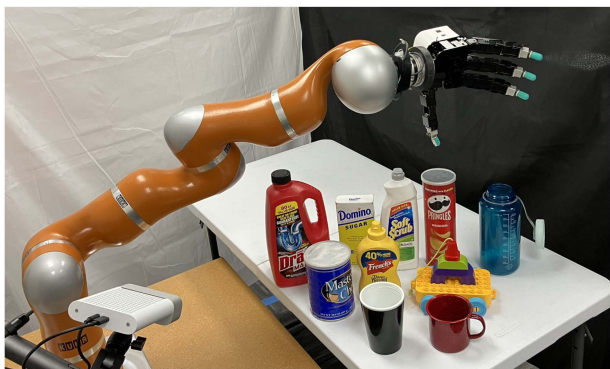


Fig. 9. Objects and the robot used to assess the impact of our method on dexterous grasping.

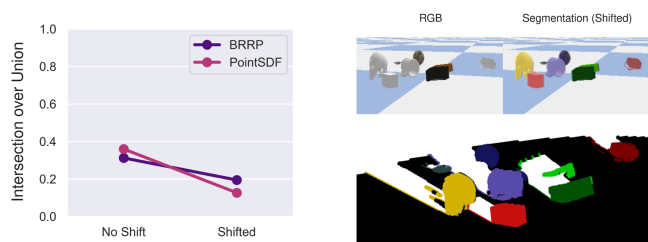


Fig. 10. (left) IoU of BRRP and PointSDF on ShapeNet scenes with and without shifted segmentations. Our method is more robust to segmentation shifts. (right) An example of a scene and the corresponding point cloud with shifted segmentation.

Insight 2: BRRP is more robust than a deep learning method.

While the ShapeNet version of PointSDF outperformed BRRP on the ShapeNet scenes on IoU (Table I), BRRP had a lower Chamfer distance (Fig. 7). BRRP also outperformed the YCB version of PointSDF on both YCB and ShapeNet (Table I). Generally, BRRP performed significantly better than each PointSDF version on mesh datasets the specific version was not trained on. When measuring chamfer distance, BRRP outperformed the ShapeNet version of PointSDF on all datasets as shown in Fig. 7, including the Objaverse dataset, where objects were out-of-distribution for both methods. These results suggest BRRP is more robust to different object distributions than PointSDF. Next, we evaluate robustness to slightly incorrect image segmentations. We take the procedurally generated ShapeNet scenes and shift the segmentation over by 2 pixels. In Fig. 10, we compare the IoU of BRRP and ShapeNet version of PointSDF on the scenes with and without the shift. Our method performs better on the shifted scenes compared to PointSDF.

On the real-world scenes in Fig. 8, BRRP is qualitatively more robust than PointSDF. PointSDF struggles with the noise inherent in real world scenes as well as novel objects. It tends to predict a spherical object on many non-spherical objects in the real world scenes. Our method is better able to reconstruct these scenes, including the objects that are out-of-distribution for its prior. These results suggest BRRP is more robust than PointSDF. Even though by one metric the ShapeNet-trained PointSDF outperforms BRRP on ShapeNet scenes, when the scenes are perturbed, BRRP performs better.

Insight 3: BRRP can capture principled uncertainty about object shape.

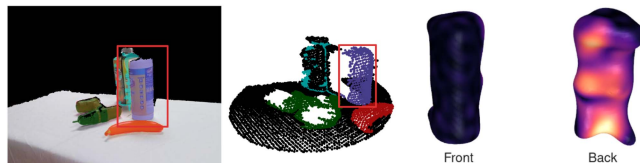


Fig. 11. Visualization of a cylindrical Clorox container surface uncertainty from BRRP. Lighter areas correspond to higher uncertainty about the shape. We observe that the occluded back-side of the container has high uncertainty.

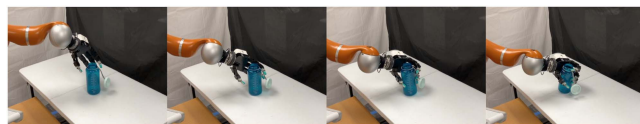


Fig. 12. BRRP builds precise representations, which are critical for downstream dexterous manipulation. Here, the water (out-of-distribution) is particularly large, and the dexterous hand needs to arrive at a pre-grasp pose that is very close to the surface of the bottle, all the while requiring the fingers to avoid colliding.

Fig. 11 shows a qualitative example of uncertainty from BRRP. We measure the uncertainty by taking the variance of logits over weight particles. Our method predicts the highest uncertainty in areas of the surface that are occluded. This suggests that we can utilize surface uncertainty from BRRP in a similar way to how GPIS surface uncertainty is utilized in many grasping applications.

Insight 4: BRRP's robust reconstructions improve the downstream success of dexterous grasping.

The success rate of BRRP and the PointSDF baseline is 64% and 40% respectively on the downstream grasping experiment. The difference in success rates stems from the modeling of the obstacles, consisting of the reconstructed object mesh and the table (fixed). The collision modeling directly impacts how close the robot can get to the object before executing the grasp. This directly affects whether the object will get knocked over or bumped into as the gripper moves to the pre-grasp configuration. BRRP exhibits higher accuracy and robustness in reconstruction, which creates an over 50% increase in grasp success rates. Fig. 12 illustrates an example and highlights the impact of BRRP in significantly improving downstream grasp success. While the difference in BRRP and PointSDF grasp success can be easily attributed to reconstruction quality, we hypothesize the 36% of grasp failures with BRRP stem from a mixture of reconstruction, grasp selection, and motion planning inaccuracies.

Limitations: As shown in Fig. 6, the registration portion of our method grows with the number of objects, which can be resolved by fast learning-based registration methods. While our method is robust to viewpoint, occlusion, and scale; it is still limited by the quality of segmentation, depth accuracy, and its prior database. Advances in segmentation methods will further improve BRRP performance. While BRRP can robustly handle OOD data, the reconstruction and uncertainty remain marginally biased towards the in-distribution dataset, which is common for data-driven approaches. This can be seen in Figs. 2 and 8, and can be mitigated with a larger, more diverse prior database.

VII. CONCLUSION

We introduced the concept of retrieval-augmented priors, where we retrieve relevant components of a prior distribution during Bayesian inference. We also introduced a novel Bayesian method for scene reconstruction that uses *informative priors*. Our method, BRBP, leveraged existing mesh datasets to build its prior and probabilistically reconstructed the scene leveraging these priors in a retrieval-augmented manner. We showed our method was more robust than a deep learning method as well as more accurate than a mapping method. We also showed a qualitative example of recovering principled uncertainty from our method. Finally, we demonstrated that our robust reconstructions improve downstream performance, where BRBP reconstructions enable precise collision avoidance as the robot moves to execute grasps.

REFERENCES

- [1] S. Chen, J. Bohg, and C. K. Liu, "SpringGrasp: An optimization pipeline for robust and compliant dexterous pre-grasp synthesis," in *Robotics: Science and Systems*, 2024.
- [2] C. de Farias, B. Tamadazte, M. Adjigble, R. Stolkin, and N. Marturi, "Task-informed grasping of partially observed objects," *IEEE Robot. Automat. Lett.*, vol. 9, no. 10, pp. 8394–8401, Oct. 2024.
- [3] M. Matak and T. Hermans, "Planning visual-tactile precision grasps via complementary use of vision and touch," *IEEE Robot. Automat. Lett.*, vol. 8, no. 2, pp. 768–775, Feb. 2023.
- [4] M. Matak, M. Devendran K. Shanthi V. Wyk, and T. Hermans, "First plan then evaluate: Use a vectorized motion planner for grasping," 2025, *arXiv:2509.07162*.
- [5] S. Dragiev, M. Toussaint, and M. Gienger, "Gaussian process implicit surfaces for shape estimation and grasping," in *Proc. Int. Conf. Robot. Automat.*, 2011, pp. 2845–2850.
- [6] H. Wright, W. Zhi, M. Johnson-Roberson, and T. Hermans, "V-PRISM: Probabilistic mapping of unknown tabletop scenes," in *Proc. Int. Conf. Intell. Robots Syst.*, 2024, pp. 1078–1085.
- [7] F. Ramos and L. Ott, "Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent," *Int. J. Robot. Res.*, vol. 35, no. 14, pp. 1717–1730, 2016.
- [8] R. Senanayake and F. Ramos, "Bayesian Hilbert maps for dynamic continuous occupancy mapping," in *Proc. 1st Annu. Conf. Robot Learn.*, vol. 78, 2017, pp. 458–471.
- [9] R. Senanayake, A. Tompkins, and F. Ramos, "Automorphing kernels for nonstationarity in mapping unstructured environments," in *Proc. 2nd Conf. Robot Learn.*, vol. 87, 2018, pp. 443–455.
- [10] W. Zhi, L. Ott, R. Senanayake, and F. Ramos, "Continuous occupancy map fusion with fast Bayesian hilbert maps," in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 4111–4117.
- [11] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 165–174.
- [12] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4455–4465.
- [13] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, "Implicit neural representations with periodic activation functions," *Neural Inf. Process. Syst.*, vol. 33, pp. 7462–7473, 2020.
- [14] 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," *Commun. ACM*, vol. 65, no. 1, pp. 99–106, 2021.
- [15] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, 2023, Art. no. 139.
- [16] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "DUST3R: Geometric 3D vision made easy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 20697–20709.
- [17] W. Zhi, H. Tang, T. Zhang, and M. Johnson-Roberson, "Simultaneous geometry and pose estimation of held objects via 3D foundation models," *IEEE Robot. Automat. Lett.*, vol. 9, no. 12, pp. 11818–11825, Dec. 2024.
- [18] J. Xu, W. Cheng, Y. Gao, X. Wang, S. Gao, and Y. Shan, "InstantMesh: Efficient 3D mesh generation from a single image with sparse-view large reconstruction models," 2024, *arXiv:2404.07191*.
- [19] M. Liu et al., "One-2-3-4-5: Any single image to 3D mesh in 45 seconds without per-shape optimization," *Neural Inf. Process. Syst.*, vol. 36, pp. 22226–22246, 2024.
- [20] F. Engelmann, K. Rematas, B. Leibe, and V. Ferrari, "From points to multi-object 3D reconstruction," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4588–4597.
- [21] Z. Liao, J. Yang, J. Qian, A. P. Schoellig, and S. L. Waslander, "Uncertainty-aware 3D object-level mapping with deep shape priors," in *Proc. Int. Conf. Robot. Automat.*, 2024, pp. 4082–4089.
- [22] M. Van der Merwe, Q. Lu, B. Sundaralingam, M. Matak, and T. Hermans, "Learning continuous 3D reconstructions for geometrically aware grasping," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 11516–11522.
- [23] F. Williams et al., "Neural fields as learnable kernels for 3D reconstruction," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 18479–18489.
- [24] J. Huang, Z. Gojcic, M. Atzmon, O. Litany, S. Fidler, and F. Williams, "Neural kernel surface reconstruction," in *Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 4369–4379.
- [25] Y. Kasten, O. Rahamim, and G. Chechik, "Point cloud completion with pretrained text-to-image diffusion models," *Neural Inf. Process. Syst.*, vol. 36, 2024.
- [26] Y.-C. Cheng, H.-Y. Lee, S. Tulyakov, A. G. Schwing, and L.-Y. Gui, "SDFusion: Multimodal 3D shape completion, reconstruction, and generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 4456–4465.
- [27] B. Saund and D. Berenson, "Diverse plausible shape completions from ambiguous depth images," in *Proc. 4th Conf. Robot Learn.*, 2021, pp. 1–12.
- [28] W. Agnew et al., "Amodal 3D reconstruction for robotic manipulation via stability and connectivity," in *Proc. 4th Conf. Robot Learn.*, 2021, pp. 1–11.
- [29] S. Iwase et al., "Zero-shot multi-object scene completion," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 96–113.
- [30] N. Gotthoskar et al., "3DP3: 3D scene perception via probabilistic programming," *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021, pp. 1–13.
- [31] W. Martens, Y. Poffet, P. R. Soria, R. Fitch, and S. Sukkarieh, "Geometric priors for Gaussian process implicit surfaces," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 373–380, Apr. 2017.
- [32] J. Lundell, F. Verdoja, and V. Kyrki, "Robust grasp planning over uncertain shape completions," in *Proc. Int. Conf. Intell. Robots Syst.*, 2019, pp. 1526–1532.
- [33] D. Hidalgo-Carvajal et al., "Anthropomorphic grasping with neural object shape completion," *IEEE Robot. Automat. Lett.*, vol. 8, no. 12, pp. 8034–8041, Dec. 2023.
- [34] Q. Liu and D. Wang, "Stein variational gradient descent: A general purpose Bayesian inference algorithm," *Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 2378–2386.
- [35] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," *Neural Inf. Process. Syst.*, vol. 33, pp. 9459–9474, 2020.
- [36] Y. Kuang et al., "RAM: Retrieval-based affordance transfer for generalizable zero-shot robotic manipulation," in *Proc. of The 8th Conf. Robot Learn.*, 2025, pp. 547–565.
- [37] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.
- [38] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [39] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. Int. Conf. Robot. Robot. Automat.*, 2009, pp. 3212–3217.
- [40] H. Thomas, "Learning new representations for 3D point cloud semantic segmentation," Ph.D. dissertation, Université Paris sciences et lettres, 2019.
- [41] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and model set: Towards common benchmarks for manipulation research," in *Proc. Int. Conf. Adv. Robot. Robot.*, 2015, pp. 510–517.
- [42] A. X. Chang et al., "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.
- [43] M. Deitke et al., "Objaverse: A universe of annotated 3D objects," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 13142–13153.
- [44] T. Ren et al., "Grounded SAM: Assembling open-world models for diverse visual tasks," 2024, *arXiv:2401.14159*.