# Metric Semantic Manipulation-Enhanced Mapping via Belief Prediction Models

Nils Dengler[1*] Joao Marcos Correia Marques[2*] Jesper Mücke[1] Tobias Zaenker[1]
Shenlong Wang[2] Kris Hauser[2] Maren Bennewitz[1]

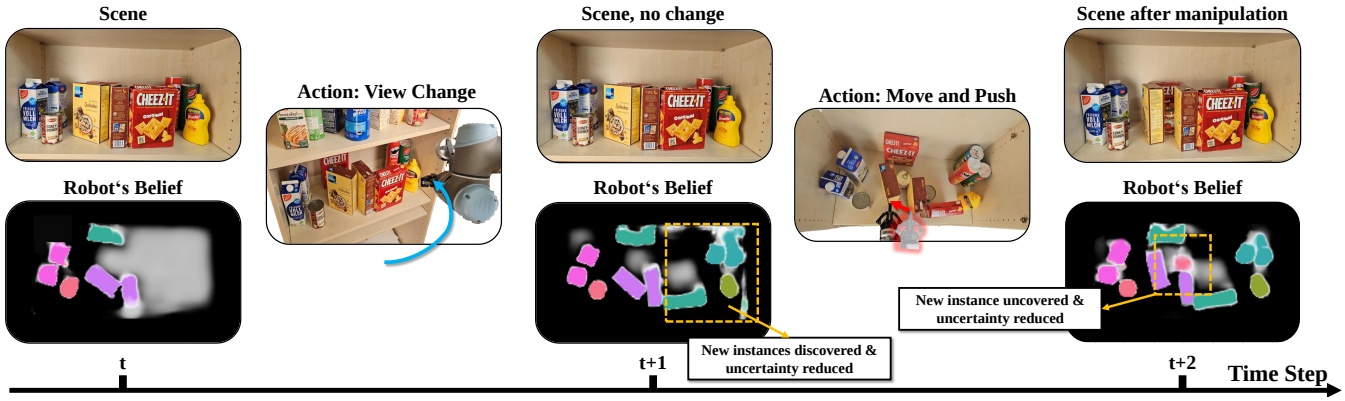



Fig. 1: Example scenario with occlusions in a confined shelf environment. Given a current partial map of the environment (belief $t$), our planner decides whether gathering another observation or manipulating the scene would be best to reduce map uncertainty. In this example, first a viewpoint action would increase environmental knowledge, followed by a push to unveil the hidden can behind the two boxes at time $t + 2$.

***Abstract*— Searching for objects in cluttered environments requires selecting efficient viewpoints and manipulation actions to resolve occlusions and reduce uncertainty about object locations, shapes, and categories. We address the problem of manipulation-enhanced semantic mapping, where a robot efficiently identifies all objects in a cluttered shelf. Although Partially Observable Markov Decision Processes (POMDPs) are standard for decision-making under uncertainty, representing unstructured interactive worlds remains challenging in this formalism. To overcome this, we introduce a novel POMDP framework that summarizes beliefs using a metric-semantic grid map and leverages neural networks for efficient belief updates, simultaneously reasoning about object geometries, locations, categories, occlusions, and manipulation physics. To ensure efficient exploration via information gain maximization, we propose to use Calibrated Neural-Accelerated Belief Updates (CNABUs), providing confidence-calibrated predictions that generalize to novel scenarios. Our experiments demonstrate improved map completeness and accuracy over existing methods, successfully transferring to real-world cluttered shelves in a zero-shot manner.**

## I. Introduction

Active sensing has long been studied in robotics for tasks such as exploring an unknown environment [1], complete 3D object model acquisition [2], and searching for an unobserved target object [3], [4]. To build complete maps as efficiently as possible, Next Best View (NBV) planning [5] is often employed to reduce the uncertainty about the map as quickly as possible. Although NBV planning handles static scenes in which the robot simply moves the camera passively through free space, there are many applications, such as household and warehouse robotics, in which robots may need to manipulate the environment in order to gain better viewpoints [6], [7]. We refer to this problem as *manipulation-enhanced mapping* (MEM). MEM offers two significant new challenges beyond standard NBV problems. First, in order to decide when and where to manipulate objects, the robot should reason about how object movement may affect previously occluded regions. Second, it must anticipate the impact of manipulations on observed objects and possibly partially-observed or unobserved objects. For example, pushing boxes in a grocery shelf backward will move them simultaneously until the furthest, occluded box hits a wall.

In this work, we formulate the MEM problem as a Partially Observable Markov Decision Process (POMDP) in the belief space of semantic maps. By maintaining map-space beliefs, our approach is applicable to unstructured cluttered environments with an arbitrary number of objects. The POMDP computes the next best viewpoint or manipulation action that maximizes the agent's expected information gain over a short horizon (Fig. 1). Our approach leverages neural network methods for map-space belief propagation, which have been shown in the object goal navigation literature to drastically improve map completion rates and offer better guidance for object search by improving the reasoning about objects beyond directly observed space [3], [4]. The key challenge in belief propagation with manipulation actions is that they often reduce certainty when the object's dynamics are unknown or the robot interacts with unobserved objects. To address this challenge, we introduce the Calibrated Neural-Accelerated

∗ These authors contributed equally to this work.;
1. Humanoid Robots Lab, University of Bonn, Germany
2. University of Illinois at Urbana-Champaign, IL, USA.
This work has been supported by the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

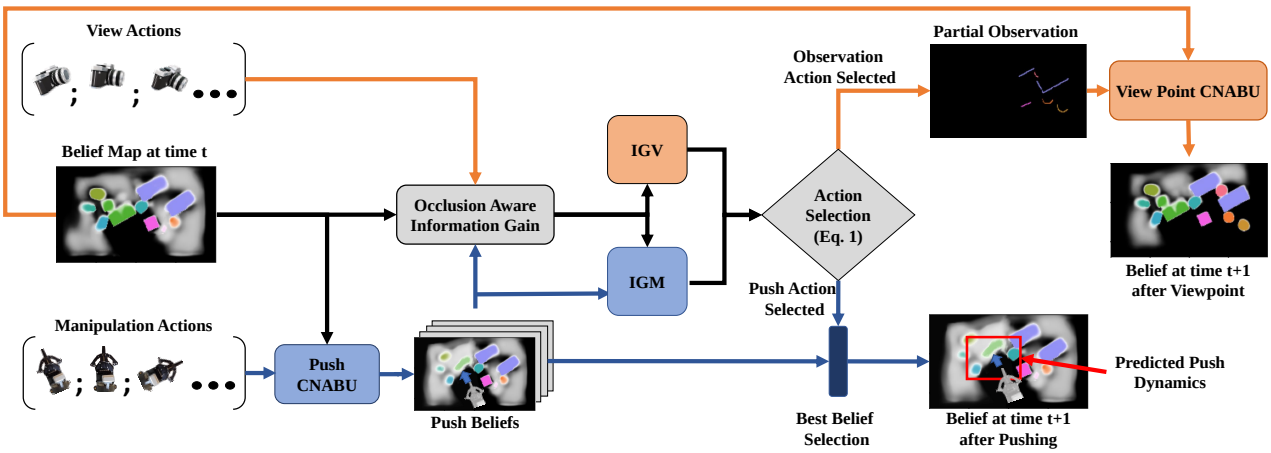



Fig. 2: From a prior map belief, our pipeline predicts the potential map belief resulting from a series of potential pushes. It then weighs the information gain from taking two consecutive independent views given the current belief (orange arrows) or taking a single observation given any of the predicted beliefs after pushing (blue arrows), selecting the path of highest cumulative information gain and taking its respective first action - either taking the next best view or executing the best push.

Belief Update (CNABU) technique to learn unified belief propagation models for both viewpoint and manipulation actions. Confidence calibration [8] is especially important for belief propagation because overconfidence in either object dynamics or map prediction would result in ineffective exploration and/or early agent termination. Therefore, we leverage evidential deep learning to obtain better calibrated CNABU networks [9].

## II. Methods

### A. Overview

In this work, we consider a confined environment with movable objects of varying sizes and orientations, where some objects may be unobservable from any viewpoint due to occlusions. We aim to determine the most informative sequence of actions for a robot, within a given action budget, that minimizes the difference between the robot's internal map belief and the true environment configuration using a similarity metric, such as IoU. A robotic arm, equipped with a wrist-mounted RGB-D camera and a gripper, aims to build an accurate map of the current workspace configuration $C_W$ [10] after a sequence of actions, which can be either taking an RGB-D image or performing a manipulation (i.e., a push) to move objects and reveal occluded areas.

Let $\Phi^t$ represent the robot's internal environment map at time $t$. When manipulating the environment, it causes a transition on the workspace configuration space from $c^t \mapsto c^{t+1} \in C_W$ according to the environment's dynamics. Further, whenever the robot takes an action, it updates its internal environment representation according to its belief update, $\Phi^t \rightarrow \Phi^{t+1}$. However, traditional POMDP updates are impractical due to the high dimensionality of the belief space [11]. We propose using uncertainty-aware evidential deep learning [12] to predict a factorized belief distribution that aligns with plausible configurations while maintaining compactness.

### B. Solving the POMDP

We solve the POMDP using a $k$-step receding horizon greedy planner (Fig. 2) and approximating the reward function with Volumetric Information Gain (VIG) [13].

To perform an observation action, the robot chooses from $v^i \in \mathbb{V}$ possible views in a fixed array of camera positions $\mathbb{V}$ to which the robot can move. Furthermore, let $\theta_t \in \Theta_t$ be a sampled manipulation action from a set of feasible actions. We propose to use a two-step greedy receding horizon policy search strategy, where we only consider two possible kinds of action sequences: taking two observation actions $(v_t, v_{t+1})$ or performing a manipulation action followed by an observation $(\theta_t, v_{t+1})$. This is because $(\theta_t, \theta_{t+1})$ would result in no observation and therefore no information gain and the information gain of $(v_t, \theta_{t+1})$ is smaller than the VIG by any $(v_t, v_{t+1})$, $v_{t+1} \neq v_t$, given that no observation is obtained from a manipulation action. These action sequence branches can, thus, be culled from the policy search tree. Unlike other methods for solving POMDPs with high-dimensional state spaces, like POMCP [14], our proposed solution does not require drawing state samples from the current belief to estimate history rewards. Instead, we leverage confidence calibrated neural networks (CNABUs) to directly estimate the mean per-voxel occupancy and semantics conditioned on the action at a given time and marginalized over all states. We can then combine the submodularity of the information gain metric and the efficiency of visual information gain heuristics on independent-cell voxel grids [13] to estimate the expected reward from each action sequence. We next detail how to train these CNABUs.

### C. Approximating Belief Dynamics with Neural Networks

We propose recursively estimating the belief update after an observation action $o_t$ using a deep posterior network, $\sigma_o(\Phi_{t-1}, o_t)$, which we call Calibrated-Neural Accelerated Belief Update (**CNABU**) network to create an implicit Monte-Carlo estimate of the POMDP belief update by $\Phi_t = \sigma_o(\Phi_{t-1}, o_t)$. Similarly, the update after a manipulation action $a_t$ is learned via an action-specific CNABU, $\sigma_m(\Phi_{t-1}, a_t)$.

Given that evidential posterior networks are shown to

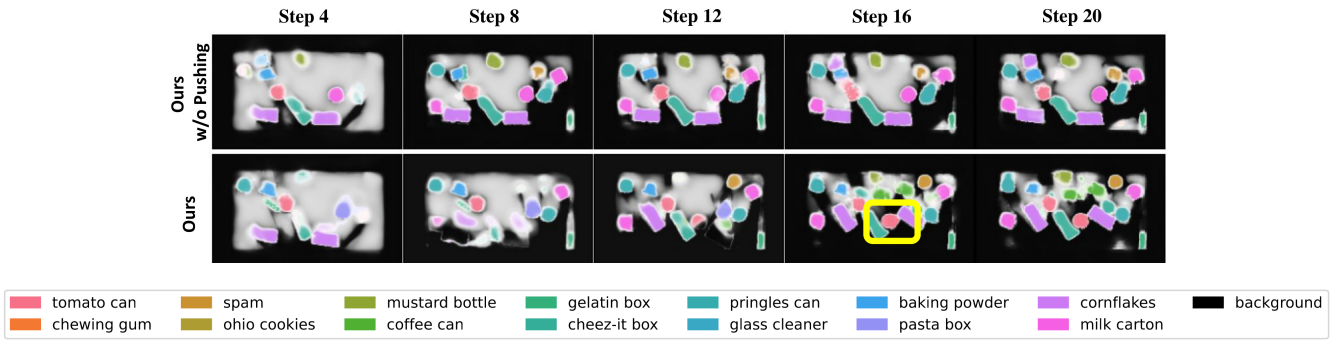



Fig. 3: Real-world experiment results show that VPP and Random baselines struggle with occlusions, while our method explores effectively through manipulations. In Step 16, we highlight a revealed object in yellow.

handle uncertainty more effectively [12], $\sigma_o$ is designed to produce evidential outputs: $\boldsymbol{\alpha^S} \in \mathbb{R}^{H\times W\times N_{classes}}$ and $\boldsymbol{\alpha^O} \in \mathbb{R}^{H\times W\times D\times 2}$. These correspond to a grid of Dirichlet distribution parameters over a 2D map of the environment (with shape $H \times W$ and $N_{classes}$) and a dense 3D grid of Beta distribution parameters, one for each voxel in the environment (with shape $2 \times H \times W \times D$). Let $Dir(\cdot)$ and $Beta(\cdot)$ denote the Dirichlet and Beta distributions. Therefore, the semantic and occupancy beliefs used to solve the POMDP are defined as $\Phi^S = \mathbb{E}[Dir(\boldsymbol{\alpha^S})]$ and $\Phi^O = \mathbb{E}[Beta(\boldsymbol{\alpha^O})]$.

The manipulation CNABU $\sigma_m$ is defined similarly to the viewpoint CNABU $\sigma_o$, except it takes as an input the parametrization of action $a_t$, which we call $\zeta_t$ and it has an auxiliary output, which predicts a Beta distribution over a voxel grid modeling the probability of a given voxel being changed in $\Phi^{GT}$ after the manipulation is executed.

### D. Using CNABUS for Solving the POMDP

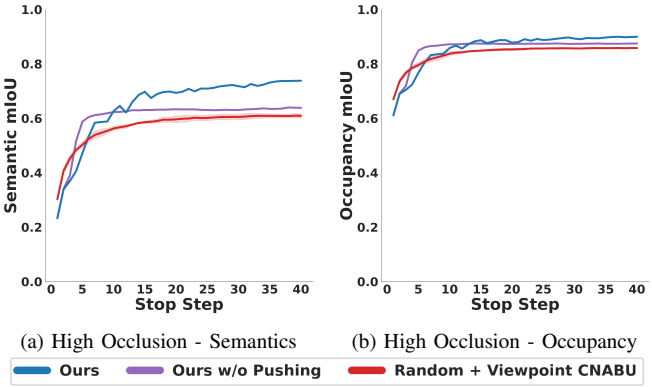
Using the VIG [13]for information gain estimates, the term $IGV_t = IG(v_t^*, v_{t+1}^*|\Phi_t^O)$ denotes the highest information gain obtained from two viewpoints, given map representation $\Phi_t^O$, while $IGM_t = IG(v_{\theta_t}^*|\tilde{\Phi}_{t+1}^{\theta_t^*})$ represents the best information gain from a pushing action followed by a viewpoint execution, given the posterior map representation after executing the push action $\tilde{\Phi}_{t+1}^{\theta_t^*}$. Lastly, $Reg_t = \Delta H(\Phi_t, \tilde{\Phi}_{t+1}^{\theta_t^*})$ captures the entropy difference between the current semantic map and the map after the best push action. Our policy decides the action $a_t$ through:

$$a_t = \begin{cases} v_t^* & \text{if } IGV_t > IGM_t + \gamma Reg_t \\ \theta_t^* & \text{otherwise} \end{cases} \quad (1)$$

Here, $\gamma$ balances VIG and entropy increase due to manipulation, with $\Delta H$ regularizing manipulation actions to limit large unnecessary disturbances to the scene. Belief updates depend on the action: $\Phi^{t+1} = \tilde{\Phi}_{t+1}^{\theta_t^*}$ for manipulation or $\Phi^{t+1} = \sigma_o(\Phi^t, o^t)$ for observation, with $o_t \sim Z(v_t^*|c_w^t)$.

## III. Experimental Results

For experimental evaluation, we set up a shelf scene with a UR5 arm for observation and action execution in PyBullet [15]. The robot is equipped with a Robotiq parallel-jaw gripper and an realsense L515 RGB-D camera for




(a) High Occlusion - Semantics (b) High Occlusion - Occupancy

Fig. 4: Simulation results in MEM task.

observations. To sample realistic object configurations, a total of 14 different object categories from the YCB dataset are used and sampled in a shelf board of size $(0.8\times0.4\times0.4)m$.

### A. Simulation Experiments

Our simulation experiments consider high occlusion scenarios for manipulation-enhanced mapping. We generate 25 high occlusion scenarios by hand to be challengingly crowded and with many objects occluded. Furthermore, we us a fixed set of 300 viewpoints in front of the shelf for $\mathbb{V}$.

The robot always begins with a naive uniform prior over the environment. We measure the individual methods' success in both metric and semantic mIoU against the ground truth map at time $t$, with a 40 step budget.

Quantitative results are shown in Fig. 4. We observe that belief prediction is a powerful approach, leading to excellent scene coverage even without pushing. On highly occluded scenarios, pushing is required to make progress after the visible surfaces are observed. Our method uses pushing to achieve significant higher mIoUs. Note its IoU growth is slower early on, because pushing does not provide information until a viewpoint step is taken in the next action.

### B. Hardware Experiments

We performed 10 real-world experimental runs on a UR5. All results are collected in a zero-shot fashion, i.e., no fine-tuning on real data was performed. We set the budget to 20 steps and sampled a fixed set of 75 reachable camera poses in front of the shelf for $\mathbb{V}$.

TABLE I: Comparing our method in 10 trials to the strongest baselines in zero-shot transfer to real-world shelves.

| Policy | Correctly Found ↑ | Missclassified But Found↑ | Not Found↓ | Hallucinated↓ |
|---|---|---|---|---|
| **Random + CNABU** | 72 | 47 | 52 | 11 |
| **Ours w/o Pushing** | 81 | 38 | 52 | **6** |
| **Ours** | **85** | **52** | **35** | 7 |

We handcrafted 10 challenging scenes, each with an average of 18 objects from the YCB dataset [16], where pushing is required to reveal other objects. We collect the ground truths by removing the top of the shelf at the end of each episode to manually score the final maps. We score each scenario according to the status of **all of the objects** present in the map. Each object in the map is classified in four categories: 1) **Correctly Found** if the majority of the object is correctly represented in the map with the right class; 2) **Misclassified But Found** if the majority of the object is present in the occupancy map but is mislabeled; 3) **Not Found** - if the majority of the object is absent from the occupancy map and 4) **Hallucinated** if an object that is not present in the scene is present in the map. For each model, we report the total quantity of each detection at time step 20 summed over all 10 trials.

Results in Tab. I show that with zero-shot transfer from sim-to-real, the proposed method still manages to retain its edge over the compared baselines. Note that all methods compared use calibrated belief prediction. However, little difference is seen between viewpoint planning (Ours w/o Pushing) and random viewpoint choices. However, Ours w/o Pushing classified more objects correctly with fewer hallucinated objects. We expect that this is due to a domain gap caused by camera noise from the realsense L515 leading to some strong artifacting in the depth images and the inaccuracies of the open-set semantic segmentation pipeline. However, we can see that our methods (both with pushing and without pushing) greatly reduce the number of hallucinations and improve the number of correctly identified objects. Further, our complete pipeline reveals 39% of the objects that were previously unseen by the non-interactive baselines, performance consistent with the simulation experiments, despite the significant sim-to-real gap, particularly in segmentation performance, leading to many of the newfound objects being incorrectly classified.

In Fig. 3 we show a qualitative result of our agent after efficient viewpoint and push selection. The qualitative results show that with zero-shot transfer from sim-to-real, our proposed method generates a good representation of the scene. Our method (both with pushing and without pushing) is able to identify the majority of objects in the scene, from a total of 16. Further, our complete pipeline reveals several objects that were previously unseen, e.g., a tomato can as highlighted in yellow.

## IV. Conclusion

In this paper, we presented a POMDP-inspired policy solver, that decides between different action types to generate an uncertainty-aware map-apace dynamics model as belief. Furthermore, our pipeline considers all action types to be equally effective and decides according to the best informative outcome. Our results show the qualitative performance of our system in terms of occupancy and semantics map accuracy and demonstrate that our agent is able to reason about map dynamics and impact of actions to the scene.

## References


[1] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots," *Autonomous Robots*, vol. 40, no. 6, pp. 1059–1078, 2016.

[2] M. Krainin, B. Curless, and D. Fox, "Autonomous generation of complete 3d object models using next best view manipulation planning," in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 5031–5037.

[3] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, and K. Daniilidis, "Learning to Map for Active Semantic Goal Navigation," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=swrMQttr6wN

[4] A. J. Zhai and S. Wang, "PEANUT: Predicting and Navigating to Unseen Targets," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10 2023, pp. 10 926–10 935. [Online]. Available: https://ieeexplore.ieee.org/document/10378364

[5] R. Zeng, Y. Wen, W. Zhao, and Y.-J. Liu, "View planning in robot active vision: A survey of systems, algorithms, and applications," *Computational Visual Media*, 2020. [Online]. Available: https://link.springer.com/article/10.1007/s41095-020-0179-3

[6] N. Dengler, S. Pan, V. Kalagaturu, R. Menon, M. Dawood, and M. Bennewitz, "Viewpoint Push Planning for Mapping of Unknown Confined Spaces," 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10341809

[7] T. Pitcher, J. Förster, and J. J. Chung, "Reinforcement learning for active search and grasp in clutter," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10801366

[8] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *ICML*, vol. 70, 2017, pp. 1321–1330. [Online]. Available: https://proceedings.mlr.press/v70/guo17a.html

[9] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential Deep Learning to Quantify Classification Uncertainty," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

[10] J. Chase Kew, B. Ichter, M. Bandari, T.-W. E. Lee, and A. Faust, "Neural Collision Clearance Estimator for Batched Motion Planning," in *Algorithmic Foundations of Robotics XIV*, S. M. LaValle, M. Lin, T. Ojala, D. Shell, and J. Yu, Eds. Cham: Springer International Publishing, 2021, pp. 73–89.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1, pp. 99–134, 1998. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S000437029800023X

[12] D. Ulmer, C. Hardmeier, and J. Frellsen, "Prior and Posterior Networks: A Survey on Evidential Deep Learning Methods For Uncertainty Estimation," *PMLR*, 2023.

[13] J. Delmerico, S. Isler, R. Sabzevari, and D. Scaramuzza, "A comparison of volumetric information gain metrics for active 3D object reconstruction," *Autonomous Robots*, vol. 42, no. 2, pp. 197–208, 2018. [Online]. Available: https://doi.org/10.1007/s10514-017-9634-0

[14] D. Silver and J. Veness, "Monte-carlo planning in large pomdps," in *Advances in Neural Information Processing Systems*, J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, Eds., vol. 23. Curran Associates, Inc., 2010.

[15] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," http://pybullet.org, 2016–2021.

[16] 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 *2015 International Conference on Advanced Robotics (ICAR)*, 2015, pp. 510–517.