MoMa-Kitchen: A 100K+ Benchmark for Affordance-Grounded Last-Mile Navigation in Mobile Manipulation

Pingrui Zhang[1,2*] Xianqiang Gao[2,3*] Yuhan Wu[3] Kehui Liu[2,4]
Dong Wang[2] Zhigang Wang[2] Bin Zhao[2,4] Yan Ding[2†] Xuelong Li[5†]
[1]Fudan University [2]Shanghai AI Laboratory [3]University of Science and Technology of China
[4]Northwestern Polytechnical University [5]TeleAI, China Telecom Corp Ltd
{zhangpingrui, dingyan}@pjlab.org.cn, xuelong_li@ieee.org

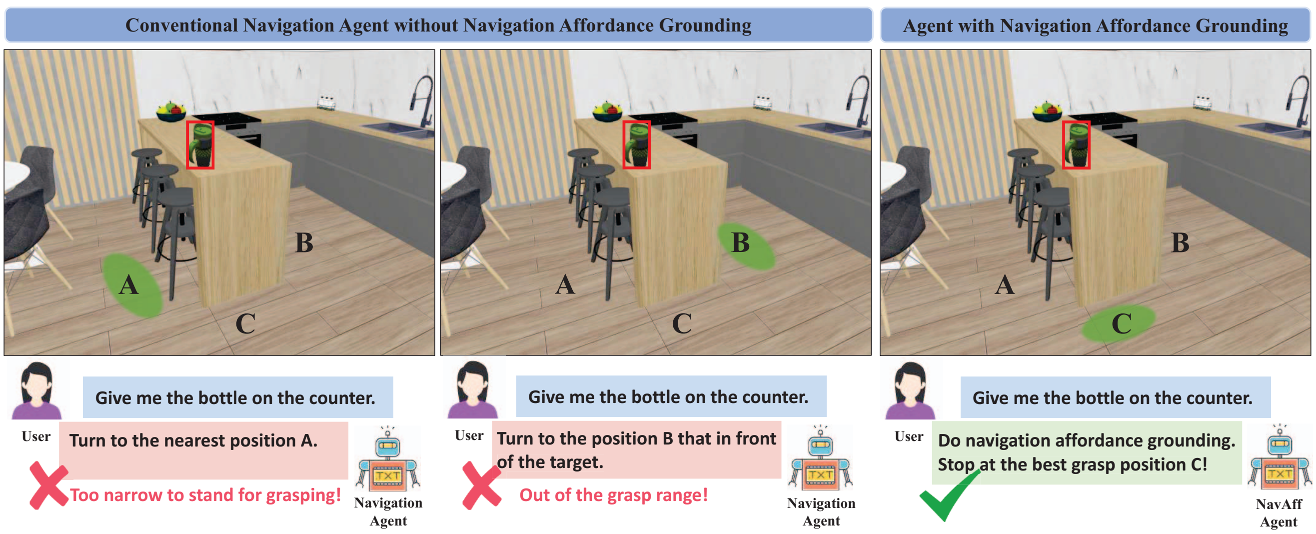


Figure 1. Conventional navigation methods typically prioritize reaching a target location but do not account for constraints affecting manipulation feasibility. *Left*: Position **A** prioritizes proximity but is obstructed by chairs, preventing stable execution. *Middle*: Position **B** places the robot in a spacious and stable area for operation but beyond its effective reach. *Right*: Our approach, leveraging navigation affordance grounding, identifies Position **C** as the optimal stance, ensuring both reachability and task feasibility.

Abstract

In mobile manipulation, navigation and manipulation are often treated as separate problems, resulting in a significant gap between merely approaching an object and engaging with it effectively. Many navigation approaches primarily define success by proximity to the target, often overlooking the necessity for optimal positioning that facilitates subsequent manipulation. To address this, we introduce MoMa-Kitchen, a benchmark dataset comprising over 100k samples that provide training data for models to learn optimal final navigation positions for seamless transition to manipulation. Our dataset includes affordance-grounded floor labels collected from diverse kitchen environments, in which robotic mobile manipulators of different models attempt to grasp target objects amidst clutter. Using a fully automated pipeline, we simulate diverse real-world scenarios and generate affordance labels for optimal manipulation positions. Visual data are collected from RGB-D inputs captured by a first-person view camera mounted on the robotic arm, ensuring consistency in viewpoint during data collection. We also develop a lightweight baseline model, NavAff, for navigation affordance grounding that demonstrates promising performance on the MoMa-Kitchen benchmark. Our approach enables models to learn affordance-based final positioning that accommodates different arm types and platform heights, thereby paving the way for more robust and generalizable integration of navigation and manipulation in embodied AI. Project page: https://momakitchen.github.io/.

1. Introduction

Most existing navigation algorithms define success *in terms of* reaching a location near the target [40]. However, in household environments, navigation is an intermediate step preceding task-specific manipulation. As a result, such navigation strategies are inadequate for mobile manipulation tasks that requires precise end-effector positioning. In practice, navigation and manipulation are tightly integrated. For instance, when tasked with retrieving an object from a kitchen counter, a robot typically navigates toward the target using conventional policies before attempting manipulation. While existing navigation algorithms reliably guide robots across rooms, they often fail in proximity to the target. The robot may stop beyond the manipulator's reachable workspace or be obstructed by spatial constraints, rendering manipulation infeasible (as shown in Fig. 1). Additionally, obstacles such as furniture, bins, or containers in cluttered environments are seldom accounted for in existing navigation policies, further complicating the selection of feasible grasping positions. As a result, reliance solely on these navigation algorithms, without incorporating the demands of manipulation, limits their efficacy in addressing complex tasks in household settings.

This limitation highlights the disconnect between object localization (*navigation*) and physical interaction (*manipulation*) in mobile robotics. In particular, optimizing final positioning in the "last mile" remains a fundamental challenge, yet existing datasets and benchmarks provide limited supervision for this aspect. While recent efforts have employed large language models (LLMs) to assist in selecting optimal navigation positions, these approaches fall short when transitioning to the manipulation phase [37, 79, 81]. Training-free LLMs struggle to accurately predict the requirements of robotic arm interactions. Additionally, they cannot dynamically adjust positioning strategies based on different robotic arm models or base morphology [64, 80].

To bridge this gap, we introduce **MoMa-Kitchen**, a large-scale benchmark with over 100k samples designed to train models for affordance-grounded final positioning in mobile manipulation tasks. Our dataset comprises 127,343 episodes spanning 569 diverse kitchen scenes, where each episode involves predicting floor affordances that enable a robot to approach a target while avoiding collisions with obstacles. We collect large-scale floor affordance data by creating kitchen scenes with various layouts—in which target objects are either randomly placed or selected from common appliances (e.g., refrigerators and cabinets) and obstacles are strategically positioned to generate distinct scenarios. In the simulator, various mobile manipulators attempt to grasp target objects from multiple positions, and success rates are recorded to obtain ground truth affordance labels for each floor position. The collected visual data, consisting of RGB-D and point cloud inputs from a first-person view camera mounted on the robotic arm, along with robot-related information, are subsequently used to train our lightweight baseline model, **NavAff**, which identifies the optimal floor affordance region for subsequent manipulation. To enhance generalization, we employ various robotic arms (e.g., Flexiv and Franka) and varied mobile bases during grasping, enabling the model to learn that floor affordance predictions depend on arm height and operational range. This approach is intended to develop a model that can be used on heterogeneous devices [52]. In summary, our contributions are as follows:

- We propose MoMa-Kitchen, the first large-scale dataset with over 100k samples that bridges the gap between navigation and manipulation in mobile manipulation tasks by enabling models to optimize final positioning near target objects.
- We develop a fully automated data collection pipeline – including scene generation, affordance labeling, and object placement – to simulate diverse real-world scenarios and enhance model generalizability.
- We design a lightweight baseline model, NavAff that employs RGB-D and point cloud inputs for navigation affordance grounding, achieving promising results on the MoMa-Kitchen benchmark.

2. Dataset Generation

Task Definition. MoMa-Kitchen focuses on determining feasible final navigation positions that enable successful manipulation in cluttered environments. Given RGB-D inputs from a first-person camera and robot-specific parameters (e.g., arm reach, base height), the goal is to produce an affordance map over the floor. This map indicates where the robot can position itself to reliably manipulate the target while accounting for obstacles, bridging navigation and manipulation within a unified pipeline.

MoMa-Kitchen includes diverse kitchen environments with multiple types of robotic mobile manipulators, first-person view visual data, and navigation-specific affordance ground truth. As illustrated in Fig. 2, we first construct large-scale kitchen environments that contain different targets, obstacles, and furniture. Mobile manipulators are then placed in these environments to obtain navigation affordance labels through robotic manipulation. For each labeled scene, we collect first-person view visual data from multiple randomly sampled robot viewpoints, including RGB-D images of the scene. The position of each viewpoint is also recorded to generate and transform both the global and floor point clouds. Additionally, floor-level affordance ground truth is collected near the target.

2.1. Scene Setup

To construct diverse kitchen environments for affordance annotation, we employ BestMan [66], a PyBullet-based

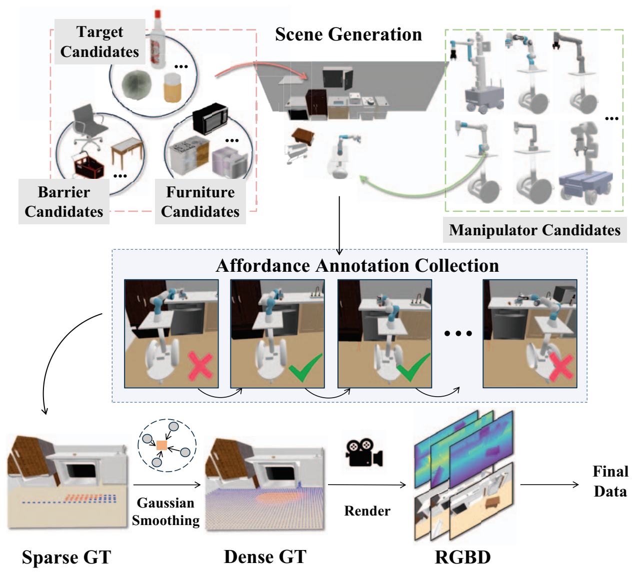


Figure 2. **Overview of scene setup and data generation pipeline.** Each scene features unique base furniture and layout, with randomly placed obstacles surrounding the target object to enhance scene complexity. Discrete navigation affordance values are collected by moving the mobile manipulator and interacting with the target objects in the scene. View transformation and Gaussian interpolation are then applied to generate a dense affordance map, along with corresponding RGBD data.

simulation platform that integrates assets from PartNet-Mobility [7, 33, 63]. The generation process begins with a rectangular kitchen layout, where common kitchen furniture and appliances (e.g., sinks, cabinets, dishwashers, and fridges) are procedurally arranged along one wall. To ensure scene diversity, we randomize both the placement order of object categories and the specific instance selection within each category. We further augment each scene with both rigid and articulated objects as manipulation targets, and introduce additional obstacles around them to increase scene complexity. Once the scene assets are ready, we randomly select a manipulator from various types and place it into the scene to collect navigation affordance labels.

2.2. Visual Data Collection

To comprehensively capture the target and its surroundings from the robot's perspective, we sample ten distinct camera poses $[\mathbf{R_c}|\mathbf{T_c}] \in \mathbb{R}^{4\times4}$ around each target. These viewpoints are strategically selected to cover the target object, its surrounding environment, and the floor. For viewpoint selection, we alternate the positions to the left and right of the target object, gradually increasing the distance from it. After placing the robot at the selected position, we check whether the target object is within the observation range. If not, we randomly generate a new position. This process continues until the target object is within the line of sight, at which point we stop and record the viewpoint. For each camera pose, we collect RGB images $\mathbf{I}$, depth maps $\mathbf{D}$, and floor point clouds $\mathbf{P}_{\text{floor}} \in \mathbb{R}^{n\times3}$, where n denotes the number of points in the point cloud.

2.3. Affordance Labeling

We gather mobile manipulators with various robotic arms to collect floor-level navigation affordance ground truth data for target objects. For each target object, we define the affordance sampling area $\mathscr{A}$ as a semicircular region on the floor, centered at the target's position with a radius equal to the maximum reach of the robotic arm. At each sampling position $\mathbf{p} \in \mathscr{A}$, we place the robot base and initialize the manipulator configuration, then align the end-effector orientation $\mathbf{R} \in \mathrm{SO}(3)$ with the surface normal of either the target object (for rigid objects) or its functional link (for articulated objects). Here, $\mathrm{SO}(3)$ denotes the Special Orthogonal Group, representing the set of all rotation matrices that describe rotations in three-dimensional space.

For each configuration $(\mathbf{p},\mathbf{R})$, we attempt a manipulation and record a binary affordance outcome $v_p \in 0,1$ at position $\mathbf{p}$. Success is determined by the outcome of the manipulation attempt: for robots equipped with two-finger parallel grippers (e.g., Panda, Flexiv, and xArm6), success is defined by a successful grasp of the target object; for the robot employing a suction-based end-effector (e.g., UR5e), success is determined by a valid suction-and-move action. To associate these affordance values with the previously collected floor point cloud $\mathbf{P}_{\text{floor}}$, we first transform v_p from the world coordinate system to the robot base frame. This transformation yields v_{pb} through:

$$v_{pb} = \mathbf{T}_{bc}\mathbf{T}_{cw}v_p, \tag{1}$$

where $\mathbf{T}_{bc}$ and $\mathbf{T}_{cw} \in \mathbb{R}^{4\times4}$ are the transformations from the camera to the base and from the world to the camera, respectively. We then match each transformed affordance value v_{pb} to its nearest neighbor in $\mathbf{P}_{\text{floor}}$. This association process can be formally expressed as:

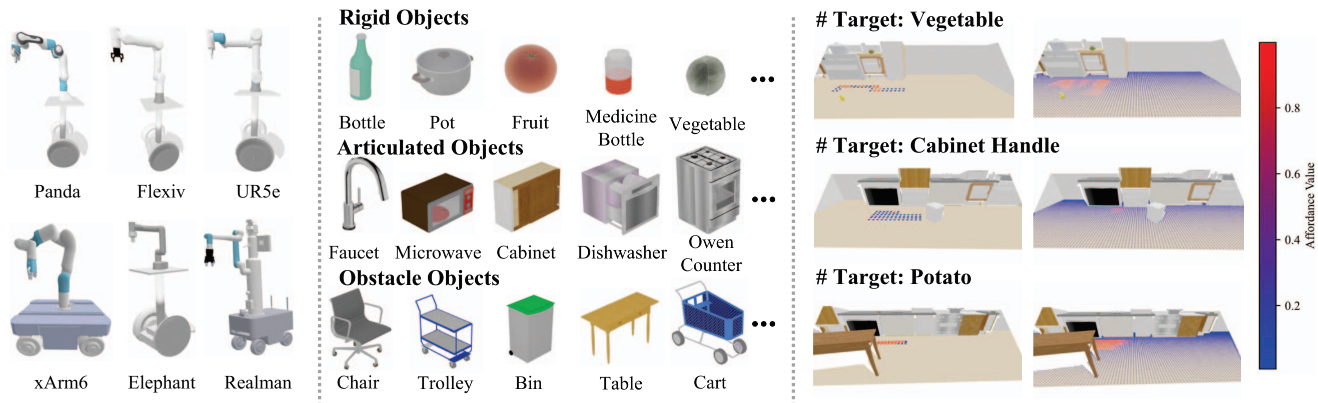
$$\mathbf{V}_{\text{aff}}(p_j) = \begin{cases} v_{pb} & \text{if } \min_{p\in\mathscr{A}} \|p-p_j\| < \theta, \forall p_j \in \mathbf{P}_{\text{floor}}, \\ 0 & \text{otherwise}, \end{cases} \tag{2}$$

where θ is a distance threshold. This yields sparse affordance values $\mathbf{V}_{\text{aff}} \in \mathbb{R}^{n\times1}$. To produce dense and continuous floor-level affordance maps, we employ Gaussian interpolation with k-nearest neighbors. This process produces interpolated affordance values $\hat{\mathbf{V}}_{\text{aff}} = \{\hat{v}_i \in [0,1] \mid i = 1,\ldots,n\}$ through the following formulation:

$$\hat{v}_i = \frac{\sum_{j=1}^{k} w_{ij}v_j}{\sum_{j=1}^{k} w_{ij}} \tag{3}$$

where the Gaussian weights w_{ij} are computed as follows.

$$w_{ij} = \exp\left(-\frac{\|p_i-p_j\|^2}{2\sigma^2}\right), \quad \forall i,j \tag{4}$$



(a) Visualization of Robot Arms **(b) Category Overview** **(c) Visualization of Affordance Map**

Figure 3. **(a) Robot arms used in MoMa-Kitchen.** The end-effectors of the Panda, Flexiv, Elephant, Realman and xArm6 robots are grippers, while the end-effector of the UR5e is a suction cup. **(b) Object categories utilized in MoMa-Kitchen.** Each category consists of multiple object instances. Rigid and articulated objects serve as manipulation targets, while obstacle objects are strategically placed around the target to enhance scene complexity. **(c) Examples of affordance maps in MoMa-Kitchen.** Discrete affordance values are first collected (left) by moving the mobile manipulator and allowing it to interact with the target. Gaussian interpolation is then applied to obtain a smooth affordance map (right).

Here, $\|p_i - p_j\|$ is the Euclidean distance between the interpolation target p_i and the sparse point p_j, σ controls the width of the Gaussian kernel, and s_j is the affordance value at p_j. This weighting scheme ensures that points closer to p_i have a larger influence, resulting in a smooth, continuous affordance map that captures the spatial distribution of potential manipulation outcomes.

2.4. Generated Dataset Statistics

As summarized in Table 1, MoMa-Kitchen spans 569 kitchen scenes, each with procedural variations in objects and obstacles. These configurations yield over 127k episodes, each capturing an RGB-D view and the resulting affordance map. Six robot arms (Flexiv, Panda, UR5e, xArm6, Realman, Elephant) are deployed to ensure diverse reachability constraints and end-effector types. (see Fig. 3(a)) The dataset covers 137 kitchen-relevant assets (65 rigid, 48 articulated, and 24 obstacle types), providing a broad spectrum of layouts and manipulation targets. (see Fig. 3(b)) (Details in Supp.) During data collection, each configuration is labeled via discrete success/failure outcomes for different robot placements. These discrete labels are then interpolated to produce dense affordance maps (see Fig. 3(c)). Collectively, MoMa-Kitchen provides large-scale supervision for end-to-end training of navigation-to-manipulation models that generalize across varying hardware and scene complexity.

3. Baseline Model

In this section, we introduce our baseline model, NavAff, which is designed for optimal manipulation positioning and obstacle interaction in complex mobile manipulation tasks. As shown in Fig. 4, Our baseline method consists of two main components: Visual Alignment Module (**VAM**) and Navigation Affordance Grounding Module (**NAG**). This two-part structure is designed to process visual and spatial information sequentially, facilitating effective affordance prediction for mobile manipulation tasks.

Table 1. **Dataset Split Statistics.**

Split	Scenes	Configurations	Episodes
Train	456	11,408	102,687
Test (Unseen Scenes)	113	2,747	24,656

3.1. Visual Alignment

VAM extracts and projects features of target object $\mathbf{T}$ and obstacles $\{\mathbf{O}_1, \mathbf{O}_2, \ldots, \mathbf{O}_n\}$ onto the global point cloud $\mathbf{P}_{global} \in \mathbb{R}^{3 \times N_{global}}$, to ensure accurate 3D representation of them. As illustrated in Fig. 4(a), VAM processes an image $\mathbf{I} \in \mathbb{R}^{3 \times H \times W}$ captured by the robot's camera along with its corresponding depth $\mathbf{D} \in \mathbb{R}^{H \times W}$, extracting target and obstacle masks using BestMan[66], where H, W denotes the height and width of the camera observation. The global point cloud $\mathbf{P}_{global}$ is then generated via inverse projection. To enhance feature representation, we leverage PointNet++ [34], which supports additional feature channels in the point cloud input. Pixels from the target mask are projected into the 3D space to form a *target channel*, with values at target locations set to 1, while obstacle mask pixels create an *obstacle channel*, assigned related values of -1 [18]. This process enriches the global point cloud by

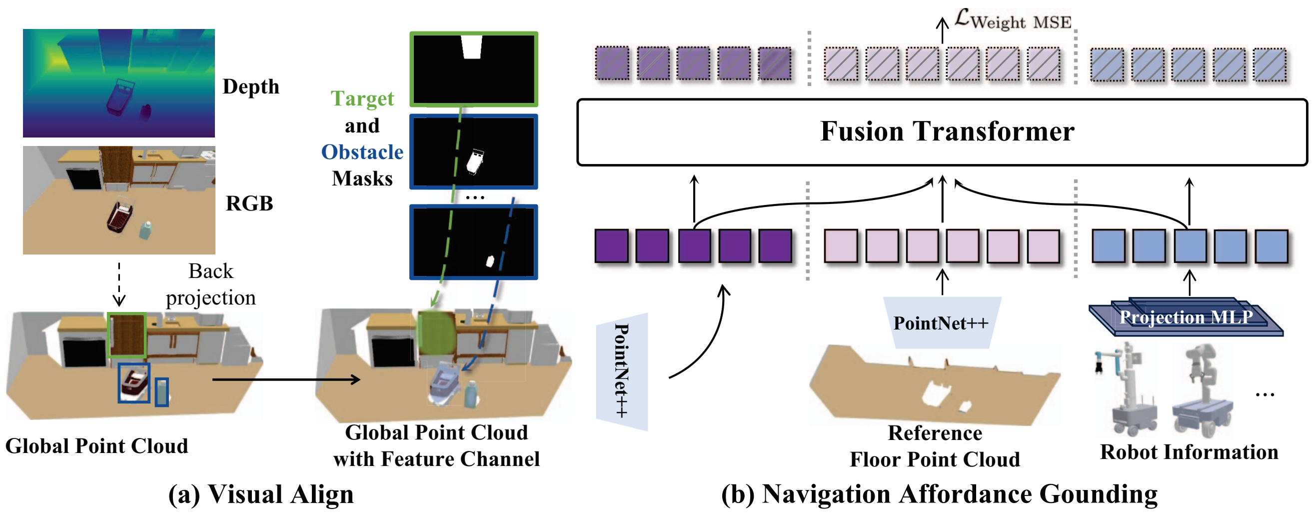


Figure 4. **NavAff Baseline. (a) Visual Alignment Module:** Projects object masks to align 2D visual features with 3D spatial representations. **(b) Navigation Affordance Grounding Module:** Fuses global point cloud, floor point cloud, and robot-specific features to predict navigation affordance maps.

embedding 2D visual cues (e.g., target and obstacle features) into the 3D spatial structure. The resulting feature-enhanced point cloud $\bar{\mathbf{P}}_{global}$ serves as input for the next stage, aligning visual perception with spatial information for improved downstream processing.

3.2. Navigation Affordance Grounding

NAG uses the robot's relevant information and the visually aligned global point cloud to interact with the reference floor, generating a robot-specific affordance prediction for optimal positioning. As illustrated in Fig. 4(b), NAG takes as input the feature-enhanced global point cloud $\bar{\mathbf{P}}_{global}$ from VAM, the floor point cloud $\mathbf{P}_{floor}$, and the robot-related information $\mathbf{R}_I$. To extract features from the point cloud data, PointNet++ processes $\bar{\mathbf{P}}_{global}$ and $\mathbf{P}_{floor}$, yielding feature representations $\mathbf{F}_{pcg}$ and $\mathbf{F}_{pcf}$, respectively. These features are then tokenized using a multi-layer perceptron (MLP) in preparation for multi-head cross-attention [48]:

$$\bar{\mathbf{F}}_{pcg} = \text{Tokenizer}(\mathbf{F}_{pcg}), \tag{5}$$

$$\bar{\mathbf{F}}_{pcf} = \text{Tokenizer}(\mathbf{F}_{pcf}), \tag{6}$$

where $\bar{\mathbf{F}}_{pcg} \in \mathbb{R}^{n_{global} \times dim}$ and $\bar{\mathbf{F}}_{pcf} \in \mathbb{R}^{n_{floor} \times dim}$.

For the mobile manipulator, $\mathbf{R}_I$ encodes the base platform height and the operational radius of the robotic arm. This information is also processed by an MLP to produce the robot-specific tokens $\mathbf{F}_r$, which are then concatenated with $\bar{\mathbf{F}}_{pcg}$, forming a combined key and value for cross-attention fusion with $\bar{\mathbf{F}}_{pcf}$. The process is defined as follows:

$$\mathbf{F}_r = \text{Tokenizer}(\text{MLP}(\mathbf{R}_I)) \tag{7}$$

$$\mathbf{F}_{out} = \text{MCA}(\bar{\mathbf{F}}_{pcf}\mathbf{W}_q, [\bar{\mathbf{F}}_{pcg}, \mathbf{F}_r]\mathbf{W}_k, [\bar{\mathbf{F}}_{pcg}, \mathbf{F}_r]\mathbf{W}_v) \tag{8}$$

where $\mathbf{F}_r \in \mathbb{R}^{n_{robot} \times dim}$, $\mathbf{F}_{out} \in \mathbb{R}^{n_{floor} \times dim}$. Finally, $\mathbf{F}_{out}$ is passed through a decoder f_d to predict the navigation affordance grounding $\mathbf{P}_{out}$.

3.3. Loss Function

Following standard practices in affordance grounding, we utilize the mean squared error (MSE) loss as the primary objective function to align the model's predictions $\mathbf{P}_{out}$ with the ground truth affordance labels $\mathbf{P}_{label}$. In our experiments, we observed that zero-valued elements constitute a large proportion of the ground truth floor affordance labels, leading to an imbalance between zero-valued elements and those with non-zero values. To address this issue, we apply a weighted mask to perform a weighted average on the MSE loss (Weighted MSE). The Weighted MSE loss is computed by adjusting the weight for zero-valued elements in the ground truth. Specifically, for elements in the ground truth that are zero, we apply a weight of λ with a probability of 0.5, while other elements are assigned a weight of 1. This ensures the model places adequate emphasis on non-zero affordance areas. The formula for the Weighted MSE loss is as follows:

$$\mathcal{L}_{\text{Weight MSE}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{W}_i \cdot (\mathbf{P}_{out,i} - \mathbf{P}_{label,i})^2 \tag{9}$$

where

$$\mathbf{W}_i = \begin{cases} \lambda & \text{if } \mathbf{P}_{label,i} = 0 \text{ and with prob. } 0.5 \\ 1 & \text{otherwise.} \end{cases} \tag{10}$$

Table 2. **Main results.** Quantitative evaluation of navigation affordance grounding performance of NavAff.

Method	RMSE ↓	logMSE ↓	PCC ↑	SIM ↑
PointNet++ [34]	0.164	0.0142	0.565	0.589
VoteNet [35]	0.167	0.0143	0.543	0.570
H3DNet [76]	0.174	0.0156	0.503	0.522
NavAff	**0.147**	**0.0115**	**0.680**	**0.696**

Here, N represents the total number of elements, $\mathbf{W}_i$ is the weight applied to each element based on the corresponding floor ground truth, and $\lambda \in (0, 1)$ is a hyperparameter that assigns a reduced weight to samples with a zero ground truth value, applied with a probability of 0.5.

4. Experiments

In this section, we benchmark NavAff on the MoMa-Kitchen. We introduce baseline results for NavAff, conduct a comprehensive performance analysis across both simulated and real-world environments, and identify emerging challenges in complex mobile manipulation scenarios that are uniquely highlighted through our benchmark.

4.1. Experimental Settings

Given that this is a newly proposed task, no existing methods provide a direct basis for comparison. To establish an evaluation, we adapt three well-established models from point cloud learning: PointNet++[34], VoteNet [35], and H3DNet [76]. Specifically, PointNet++ serves as a foundational backbone model widely used in point cloud processing tasks; VoteNet and H3DNet, both originally designed for 3D object detection, are suited for adaptation to navigation affordance grounding on our benchmark. By leveraging these models, we aim to establish strong baselines and gain insights into how existing point cloud techniques perform when applied to this new challenge.

For evaluation, we employ standard and diversity metrics tailored for the MoMa-Kitchen benchmark. Each metric provides a distinct perspective on navigation affordance grounding performance: Root Mean Squared Error (**RMSE**), facilitating understanding the magnitude of prediction error; Logarithmic Mean Squared Error (**logMSE**) focuses more on relative differences rather than absolute differences; Pearson Correlation Coefficient (**PCC**) helping to gauge the model's ability to maintain consistent patterns with the ground truth data across various affordance regions; Cosine Similarity (**SIM**) compares the structural or shape similarity between predicted and ground truth.

All the above experiments are trained on a single NVIDIA A100 GPU with a batch size of 64, using the Adam optimizer with a learning rate of 8e-4. Further experimental details are available in the Appendix.

Table 3. **Manipulation Success Rate (MSR)**.

MSR	Random	H3DNet	VoteNet	PointNet++	NavAff
`Top1`	0.080	0.54	0.56	0.60	**0.72**
`Top5`	0.046	0.47	0.53	0.58	**0.66**

4.2. Quantitative Analysis

4.2.1. Main Results

Tab. 2 reports the metrics of the proposed method **NavAff** compared with other methods on the **MoMa-Kitchen** benchmark. The results highlight that NavAff achieves superior performance in navigation affordance grounding, outperforming transferred methods across all evaluated metrics. Specifically, NavAff achieves an RMSE of 0.147 and a logMSE of 0.0115, demonstrating a marked improvement in prediction accuracy compared to the second-best method, PointNet++. In terms of correlation and similarity measures, NavAff achieves the highest scores with a PCC of 0.680 and SIM of 0.696, highlighting its robustness in capturing affordance patterns even in cluttered environments. Notably, VoteNet and H3DNet fall short of NavAff's accuracy and consistency across the various metrics, showcasing the effectiveness of our approach.

For further analysis, while VoteNet and H3DNet are effective in point cloud detection and proposal classification tasks, they are less suited to the fine-grained requirements of navigation affordance grounding, resulting in slightly lower performance. All methods are trained for the same number of epochs, but H3DNet converges more slowly due to its larger parameter count. Consequently, within the same training duration, H3DNet demonstrates the weakest performance. These findings suggest that a lightweight model capable of fine-grained feature prediction may achieve superior results on our MoMa-Kitchen benchmark.

4.2.2. Manipulation Success Rate

To more intuitively demonstrate the improvement of our method and benchmark its performance in mobile manipulation, we introduce the manipulation success rate (**MSR**) (See Tab. 3). The `Top1` **MSR** refers to moving the robot to the location with the highest affordance score and recording the MSR in the test scenes. The `Top5` **MSR** refers to recording the average MSR across the top 5 locations.

As quantitatively shown in Tab. 3 using the **MSR** metric, our navigation affordance prediction paradigm shows significant practical advantages. A comparison between the "Random" baseline and other methods reveals that approaches trained on MoMa-Kitchen benchmark significantly outperform those relying on randomly sampled points within the robot's operational radius, thereby validating the effectiveness of our benchmark. Among all the trained methods, NavAff sets a new state-of-the-art performance, achieving an impressive `Top1` **MSR** of 72%

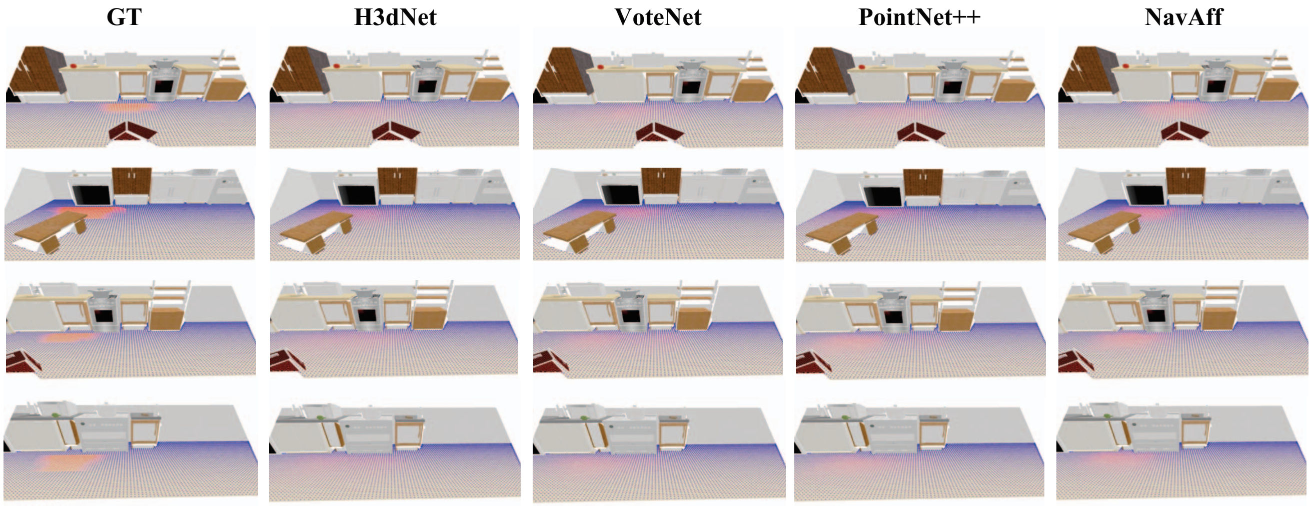


Figure 5. **Qualitative comparison of navigation affordance between all methods and ground truth.** Blue to red regions indicate affordance values ranging from 0 to 1, while void areas represent obstacle-occupied spaces.

and `Top5` **MSR** of 66%. Notably, when expanding the candidate point selection from Top1 to Top5, the model demonstrates strong generalization with minimal performance degradation, suggesting its robust adaptability to varying operational conditions.

Additionally, by comparing the **MSR** values of all models, we observe a strong positive correlation between MSR and the accuracy of navigation affordance prediction (see Tab. 2 and Tab. 3), indicating that accurate navigation affordance prediction indeed contributes to higher manipulation success rates in mobile manipulation tasks.

4.3. Qualitative Results

We show the qualitative results of NavAff and compared methods for navigation affordance grounding in Fig. 5. As illustrated, our model successfully predicts floor-level affordance maps that closely align with the ground truth patterns. The visualization clearly reveals distinct void regions in both predicted and ground truth maps, which correspond to areas occupied by obstacles, effectively demonstrating our model's ability to recognize spatial constraints. Compared to other methods, NavAff exhibits a more accurate representation of these void regions, indicating its superior capability in handling complex spatial relationships and navigating cluttered environments.

4.4. Ablation Study

To understand the impact of each module in our proposed NavAff model, we conduct an ablation study on the MoMa-Kitchen benchmark. Tab. 4 shows the results of this study, where we assess the model variations by excluding robot information, the Visual Alignment Module (VAM), and the global point cloud. Below, we analyze the role and impact of each component in detail.

Table 4. **Ablation study results of NavAff.**

	RMSE ↓	**logMSE** ↓	**PCC** ↑	**SIM** ↑
NavAff	0.147	0.0115	0.680	0.696
w/o robot information	0.148	0.0115	0.670	0.688
w/o VAM	0.165	0.0140	0.562	0.589
w/o global point cloud	0.168	0.0144	0.534	0.568

- ***w/o* robot information.** Removing robot-specific parameters (*e.g.*, base height and arm reach) leads to moderate performance degradation, particularly in RMSE and logMSE. The limited impact may be attributed to the simplified representation of robot information in NavAff, which only considers two parameters. For instance, only xArm6 has a unique base height, while other platforms share identical base configurations.
- ***w/o* VAM.** Excluding the Visual Alignment Module significantly reduces performance, particularly in PCC and SIM metrics, underscoring the importance of integrating 2D visual cues for accurate navigation affordance prediction. This demonstrates that aligning visual data with spatial information is essential for accurate and robust navigation affordance grounding.
- ***w/o* global point cloud.** Without the global point cloud, the model's performance drops across all metrics, with the largest degradation in PCC. This indicates that global spatial context is crucial for capturing the layout of objects and obstacles in the scene.

4.5. Real World Application

As shown in Fig. 6, we validate our method by conducting experiments in a real-world setting. A D435i camera is mounted on the head of a mobile manipulator to capture data. Using Grounded-Sam [39], we obtain open-vocabulary object masks, while Depth Anything v2 [68]

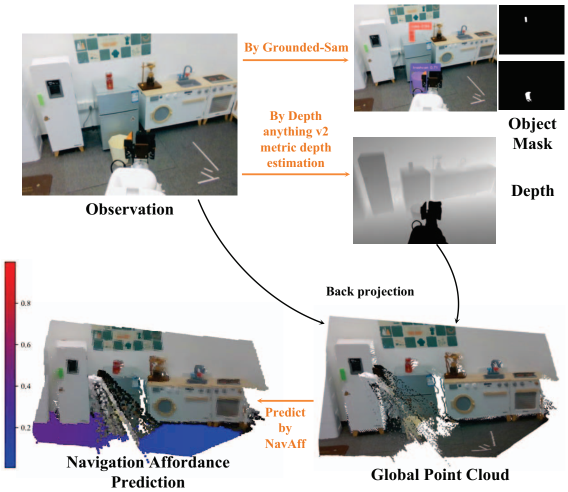


Figure 6. **Pipeline of real-world application with simulator trained NavAff.** The D435i camera mounted on the mobile manipulator captures image, and our model predicts optimal navigation affordances on the floor based on this data.

provided the depth images. We generate the global point cloud through back projection and segmented the object's local point cloud based on the mask. Then, we apply our model—trained on simulation data—to predict navigation affordances on the floor.

The prediction results, shown in Fig. 6, demonstrate that our model performs well in real-world scenarios, validating the strong generalization capability of our benchmark. This success highlights the robustness of our method, which was specifically designed to minimize the visual gap between simulation and reality. By addressing this challenge, our model can effectively adapt to real-world settings, showing that our approach is not only effective in controlled environments but also highly reliable in real-world situations.

5. Related Work

Mobile Manipulation. Such tasks have been extensively studied by researchers in both simulated and real-world settings [14, 15, 32, 41, 42, 59, 60, 70]. More recently, approaches that integrate visual and linguistic information have emerged as promising methods for achieving unified reasoning in mobile manipulation [29, 37, 50, 69, 78]. However, most existing methods still lack the versatility required to seamlessly combine coarse and fine motions for both navigation and manipulation. Prior mobile manipulation approaches often assume obstacle-free or easily navigable environments. While interactive navigation techniques [62] attempt to tackle scenarios in which obstacles must be moved or manipulated—such as shifting boxes or pressing buttons—they typically rely solely on geometric reasoning [24, 43, 47, 53, 55, 72]. In contrast, our proposed benchmark is specifically designed to address these limitations by integrating both semantic and geometric reasoning for optimal positioning and manipulation. This benchmark provides a robust and cohesive foundation for advancing mobile manipulation tasks in complex environments.

Embodied Dataset. In recent years, large-scale embodied datasets have become essential for advancing both navigation and manipulation tasks in robotics. On navigation, several datasets have been proposed to tackle challenges in object-goal navigation [6, 21, 38, 61], rearrangement [4, 31, 57], vision-language navigation [3, 22, 23, 36], and question answering [9, 10, 58, 71, 75]. In contrast, manipulation-focused works leverage large-scale proprioceptive and visual data (e.g., RGB images or 3D point clouds) to simulate real-world dexterous hand operations [28, 30, 49, 54, 65], while other datasets provide object affordance labels to facilitate interaction learning [5, 8, 13, 16, 19, 44, 67, 73]. In parallel to simulation-based approaches, several real-world datasets have emerged to address the gap between simulated and physical environments. Like FastUMI [77] propose a scalable hardware-independent robotic manipulation data collection system. More recently, research has increasingly focused on integrating manipulation capabilities with navigation for everyday tasks. Recent frameworks like BEHAVIOR Robot Suite [20] enable whole-body manipulation with bimanual coordination in household environments, while AgiBot World Colosseo [1] offers over one million trajectories across 217 tasks. Moreover, some approaches use vision-language models or procedural methods to autonomously generate scalable language annotations in simulated environments [2, 11, 12, 17, 25–27, 45, 46, 51, 56, 74]. Despite these advances, navigation datasets often offer rich spatial information yet lack guidance for optimal positioning during subsequent manipulation, whereas manipulation datasets—despite providing valuable interaction data—do not fully capture the complexities of achieving optimal grasping positions via navigation.

6. Conclusion

Our work addresses the "last mile" challenge in mobile manipulation by introducing MoMa-Kitchen, the first large-scale dataset (127k+ episodes across 569 kitchen scenes) featuring comprehensive, high-quality ground truth affordance maps to guide optimal positioning for manipulation tasks. We resolve the navigation-manipulation disconnect through automated cross-platform data collection and a unified framework compatible with diverse robotic systems, ensuring generalizability. Experimental results demonstrate that our baseline model NavAff achieves robust affordance prediction, validating the approach's efficacy. This dataset and methodology not only advance integrated navigation-manipulation systems for real-world deployment.

Acknowledgment. This work is supported by the Shanghai AI Laboratory, the National Natural Science Foundation of China (62376222), and Young Elite Scientists Sponsorship Program by CAST (2023QNRC001).

References

[1] AgiBot-World-Contributors, Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xu Huang, Shu Jiang, Yuxin Jiang, Cheng Jing, Hongyang Li, Jialun Li, Chiming Liu, Yi Liu, Yuxiang Lu, Jianlan Luo, Ping Luo, Yao Mu, Yuehan Niu, Yixuan Pan, Jiangmiao Pang, Yu Qiao, Guanghui Ren, Cheng-Xing Ruan, Jiaqi Shan, Yongjian Shen, Chengshi Shi, Mi Shi, Modi Shi, Chonghao Sima, Jianheng Song, Huijie Wang, Wenhao Wang, Dafeng Wei, Chengen Xie, Guo-Liang Xu, Junchi Yan, Cunbiao Yang, Lei Yang, Shukai Yang, Maoqing Yao, Jia Zeng, Chi Zhang, Qingli Zhang, Bin Zhao, Chengyu Zhao, Jiaqi Zhao, and Jianchao Zhu. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. 2025. 8

[2] Michael Ahn, Debidatta Dwibedi, Chelsea Finn, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Karol Hausman, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, et al. Autort: Embodied foundation models for large scale orchestration of robotic agents. *arXiv preprint arXiv:2401.12963*, 2024. 8

[3] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018. 8

[4] Dhruv Batra, Angel X Chang, Sonia Chernova, Andrew J Davison, Jia Deng, Vladlen Koltun, Sergey Levine, Jitendra Malik, Igor Mordatch, Roozbeh Mottaghi, et al. Rearrangement: A challenge for embodied ai. *arXiv preprint arXiv:2011.01975*, 2020. 8

[5] Samarth Brahmbhatt, Chengcheng Tang, Christopher D Twigg, Charles C Kemp, and James Hays. Contactpose: A dataset of grasps with object contact and hand pose. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 361–378. Springer, 2020. 8

[6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 8

[7] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3

[8] Enric Corona, Albert Pumarola, Guillem Alenya, Francesc Moreno-Noguer, and Grégory Rogez. Ganhand: Predicting human grasp affordances in multi-object scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5031–5041, 2020. 8

[9] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–10, 2018. 8

[10] Abhishek Das, Federico Carnevale, Hamza Merzic, Laura Rimell, Rosalia Schneider, Josh Abramson, Alden Hung, Arun Ahuja, Stephen Clark, Gregory Wayne, et al. Probing emergent semantics in predictive agents via question answering. *arXiv preprint arXiv:2006.01016*, 2020. 8

[11] Jiafei Duan, Wentao Yuan, Wilbert Pumacay, Yi Ru Wang, Kiana Ehsani, Dieter Fox, and Ranjay Krishna. Manipulate-anything: Automating real-world robots using vision-language models. *arXiv preprint arXiv:2406.18915*, 2024. 8

[12] Kiana Ehsani, Tanmay Gupta, Rose Hendrix, Jordi Salvador, Luca Weihs, Kuo-Hao Zeng, Kunal Pratap Singh, Yejin Kim, Winson Han, Alvaro Herrasti, et al. Spoc: Imitating shortest paths in simulation enables effective navigation and manipulation in the real world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16238–16250, 2024. 8

[13] Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J Black, and Otmar Hilliges. Arctic: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12954, 2023. 8

[14] Zipeng Fu, Xuxin Cheng, and Deepak Pathak. Deep whole-body control: learning a unified policy for manipulation and locomotion. In *Conference on Robot Learning*, pages 138–149. PMLR, 2023. 8

[15] Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024. 8

[16] Xianqiang Gao, Pingrui Zhang, Delin Qu, Dong Wang, Zhigang Wang, Yan Ding, Bin Zhao, and Xuelong Li. Learning 2d invariant affordance knowledge for 3d affordance grounding. *arXiv preprint arXiv:2408.13024*, 2024. 8

[17] Huy Ha, Pete Florence, and Shuran Song. Scaling up and distilling down: Language-guided robot skill acquisition. In *Conference on Robot Learning*, pages 3766–3777. PMLR, 2023. 8

[18] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *ArXiv*, abs/2307.05973, 2023. 4

[19] Juntao Jian, Xiuping Liu, Manyi Li, Ruizhen Hu, and Jian Liu. Affordpose: A large-scale dataset of hand-object interactions with affordance-driven hand pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14713–14724, 2023. 8

[20] Yunfan Jiang, Ruohan Zhang, Josiah Wong, Chen Wang, Yanjie Ze, Hang Yin, Cem Gokmen, Shuran Song, Jiajun

Wu, and Fei-Fei Li. Behavior robot suite: Streamlining real-world whole-body manipulation for everyday household activities. 2025. 8

[21] Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X Chang, and Manolis Savva. Habitat synthetic scenes dataset (hssd-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16384–16393, 2024. 8

[22] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, 2020. 8

[23] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020. 8

[24] Chengshu Li, Fei Xia, Roberto Martin-Martin, and Silvio Savarese. Hrl4in: Hierarchical reinforcement learning for interactive navigation with mobile manipulators. In *Conference on Robot Learning*, pages 603–616. PMLR, 2020. 8

[25] Xuelong Li, Stephen Lin, Shuicheng Yan, and Dong Xu. Discriminant locally linear embedding with high-order tensor data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(2):342–352, 2008. 8

[26] Xuelong Li, Stephen J Maybank, Shuicheng Yan, Dacheng Tao, and Dong Xu. Gait components and their application to gender recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2): 145–155, 2008.

[27] Xuelong Li, Han Zhang, Rong Wang, and Feiping Nie. Multiview clustering: A scalable and parameter-free bipartite graph fusion method. *IEEE transactions on pattern analysis and machine intelligence*, 44(1):330–344, 2020. 8

[28] Xunsong Li, Pengzhan Sun, Yangcen Liu, Lixin Duan, and Wen Li. Simultaneous detection and interaction reasoning for object-centric action recognition. *ArXiv*, abs/2404.11903, 2024. 8

[29] Peiqi Liu, Yaswanth Orru, Jay Vakil, Chris Paxton, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Ok-robot: What really matters in integrating open-knowledge models for robotics. *arXiv preprint arXiv:2401.12202*, 2024. 8

[30] Yangcen Liu, Ziyi Liu, Yuanhao Zhai, Wen Li, David Doerman, and Junsong Yuan. Stat: Towards generalizable temporal action localization. *arXiv preprint arXiv:2404.13311*, 2024. 8

[31] Karan Mirakhor, Sourav Ghosh, Dipanjan Das, and Brojeshwar Bhowmick. Task planning for visual room rearrangement under partial observability. In *The Twelfth International Conference on Learning Representations*, 2024. 8

[32] Mayank Mittal, David Hoeller, Farbod Farshidian, Marco Hutter, and Animesh Garg. Articulated object interaction in unknown scenes with whole-body mobile manipulation. In *2022 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1647–1654. IEEE, 2022. 8

[33] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[34] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 4, 6

[35] Charles R. Qi, Or Litany, Kaiming He, and Leonidas J. Guibas. Deep hough voting for 3d object detection in point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 6

[36] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9982–9991, 2020. 8

[37] Dicong Qiu, Wenzong Ma, Zhenfu Pan, Hui Xiong, and Junwei Liang. Open-vocabulary mobile manipulation in unseen dynamic environments with 3d semantic maps. *arXiv preprint arXiv:2406.18115*, 2024. 2, 8

[38] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. *arXiv preprint arXiv:2109.08238*, 2021. 8

[39] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks. *ArXiv*, abs/2401.14159, 2024. 7

[40] Beichen Shao, Nieqing Cao, Yan Ding, Xingchen Wang, Fuqiang Gu, and Chao Chen. Moma-pos: An efficient object-kinematic-aware base placement optimization framework for mobile manipulation. *arXiv preprint arXiv:2403.19940*, 2024. 2

[41] Haoming Song, Delin Qu, Yuanqi Yao, Qizhi Chen, Qi Lv, Yiwen Tang, Modi Shi, Guanghui Ren, Maoqing Yao, Bin Zhao, et al. Hume: Introducing system-2 thinking in visual-language-action model. *arXiv preprint arXiv:2505.21432*, 2025. 8

[42] Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martín-Martín, Fei Xia, Kent Elliott Vainio, Zheng Lian, Cem Gokmen, Shyamal Buch, Karen Liu, et al. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. In *Conference on robot learning*, pages 477–490. PMLR, 2022. 8

[43] Mike Stilman and James J Kuffner. Navigation among movable obstacles: Real-time reasoning in complex environments. *International Journal of Humanoid Robotics*, 2(04): 479–503, 2005. 8

[44] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasp-

ing of objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 581–600. Springer, 2020. 8

[45] Yiwen Tang, Ivan Tang, Eric Zhang, and Ray Gu. Point-peft: Parameter-efficient fine-tuning for 3d pre-trained models. *ArXiv*, abs/2310.03059, 2023. 8

[46] Yiwen Tang, Jiaming Liu, Dong Wang, Zhigang Wang, Shanghang Zhang, Bin Zhao, and Xuelong Li. Any2point: Empowering any-modality large models for efficient 3d understanding. *ArXiv*, abs/2404.07989, 2024. 8

[47] Jur Van Den Berg, Sachin Patil, Jason Sewall, Dinesh Manocha, and Ming Lin. Interactive navigation of multiple agents in crowded environments. In *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, pages 139–147, 2008. 8

[48] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 5

[49] Weikang Wan, Haoran Geng, Yun Liu, Zikang Shan, Yaodong Yang, Li Yi, and He Wang. Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3891–3902, 2023. 8

[50] Huaxiaoyue Wang, Kushal Kedia, Juntao Ren, Rahma Abdullah, Atiksh Bhardwaj, Angela Chao, Kelly Y Chen, Nathaniel Chin, Prithwish Dan, Xinyi Fan, et al. Mosaic: A modular system for assistive and interactive cooking. *arXiv preprint arXiv:2402.18796*, 2024. 8

[51] Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. Gensim: Generating robotic simulation tasks via large language models. In *Arxiv*, 2023. 8

[52] Lirui Wang, Xinlei Chen, Jialiang Zhao, and Kaiming He. Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers. *arXiv preprint arXiv:2409.20537*, 2024. 2

[53] Maozhen Wang, Rui Luo, Aykut Özgün Önol, and Taşkin Padir. Affordance-based mobile robot navigation among movable obstacles. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2734–2740. IEEE, 2020. 8

[54] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11359–11366. IEEE, 2023. 8

[55] Xiaohan Wang, Yuehu Liu, Xinhang Song, Beibei Wang, and Shuqiang Jiang. Camp: Causal multi-policy planning for interactive navigation in multi-room scenes. *Advances in Neural Information Processing Systems*, 36, 2024. 8

[56] Yufei Wang, Zhou Xian, Feng Chen, Tsun-Hsuan Wang, Yian Wang, Katerina Fragkiadaki, Zackory Erickson, David Held, and Chuang Gan. Robogen: Towards unleashing infinite data for automated robot learning via generative simulation. *arXiv preprint arXiv:2311.01455*, 2023. 8

[57] Luca Weihs, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. Visual room rearrangement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5922–5931, 2021. 8

[58] Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied question answering in photorealistic environments with point cloud perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6659–6668, 2019. 8

[59] Josiah Wong, Albert Tung, Andrey Kurenkov, Ajay Mandlekar, Li Fei-Fei, Silvio Savarese, and Roberto Martín-Martín. Error-aware imitation learning from teleoperation data for mobile manipulation. In *Conference on Robot Learning*, pages 1367–1378. PMLR, 2022. 8

[60] Bohan Wu, Roberto Martin-Martin, and Li Fei-Fei. M-ember: Tackling long-horizon mobile manipulation via factorized domain transfer. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11690–11697. IEEE, 2023. 8

[61] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018. 8

[62] Fei Xia, William B Shen, Chengshu Li, Priya Kasimbeg, Micael Edmond Tchapmi, Alexander Toshev, Roberto Martín-Martín, and Silvio Savarese. Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments. *IEEE Robotics and Automation Letters*, 5(2):713–720, 2020. 8

[63] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. SAPIEN: A simulated part-based interactive environment. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[64] Jiannan Xiang, Tianhua Tao, Yi Gu, Tianmin Shu, Zirui Wang, Zichao Yang, and Zhiting Hu. Language models meet world models: Embodied experiences enhance language models. In *Advances in Neural Information Processing Systems*, pages 75392–75412. Curran Associates, Inc., 2023. 2

[65] Yinzhen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, et al. Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4737–4746, 2023. 8

[66] Kui Yang, Nieqing Cao, Yan Ding, and Chao Chen. Bestman: A modular mobile manipulator platform for embodied ai with unified simulation-hardware apis. 2024. 2, 4

[67] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. Oakink: A large-scale knowledge repository for understanding hand-object interaction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20953–20962, 2022. 8

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *ArXiv*, abs/2406.09414, 2024. 7

[69] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander William Clegg, John Turner, et al. Homerobot: Open-vocabulary mobile manipulation. *arXiv preprint arXiv:2306.11565*, 2023. 8

[70] Naoki Yokoyama, Alex Clegg, Joanne Truong, Eric Undersander, Tsung-Yen Yang, Sergio Arnaud, Sehoon Ha, Dhruv Batra, and Akshara Rai. Asc: Adaptive skill coordination for robotic mobile manipulation. *IEEE Robotics and Automation Letters*, 9(1):779–786, 2023. 8

[71] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L Berg, and Dhruv Batra. Multi-target embodied question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6309–6318, 2019. 8

[72] Kuo-Hao Zeng, Luca Weihs, Ali Farhadi, and Roozbeh Mottaghi. Pushing it out of the way: Interactive visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9868–9877, 2021. 8

[73] Xinyu Zhan, Lixin Yang, Yifei Zhao, Kangrui Mao, Hanlin Xu, Zenan Lin, Kailin Li, and Cewu Lu. Oakink2: A dataset of bimanual hands-object manipulation in complex task completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 445–456, 2024. 8

[74] Pingrui Zhang, Yifei Su, Pengyuan Wu, Dong An, Li Zhang, Zhigang Wang, Dong Wang, Yan Ding, Bin Zhao, and Xuelong Li. Cross from left to right brain: Adaptive text dreamer for vision-and-language navigation. *arXiv preprint arXiv:2505.20897*, 2025. 8

[75] Yu Zhang, Kehai Chen, Xuefeng Bai, Zhao Kang, Quanjiang Guo, and Min Zhang. Question-guided knowledge graph rescoring and injection for knowledge graph question answering. In *Conference on Empirical Methods in Natural Language Processing*, 2024. 8

[76] Zaiwei Zhang, Bo Sun, Haitao Yang, and Qi-Xing Huang. H3dnet: 3d object detection using hybrid geometric primitives. In *European Conference on Computer Vision*, 2020. 6

[77] Zhaxizhuoma, Kehui Liu, Chuyue Guan, Zhongjie Jia, Ziniu Wu, Xin Liu, Tianyu Wang, Shuai Liang, Pengan Chen, Pingrui Zhang, Haoming Song, Delin Qu, Dong Wang, Zhigang Wang, Nieqing Cao, Yan Ding, Bin Zhao, and Xuelong Li. Fastumi: A scalable and hardware-independent universal manipulation interface with dataset. 2025. 8

[78] Peiyuan Zhi, Zhiyuan Zhang, Muzhi Han, Zeyu Zhang, Zhitian Li, Ziyuan Jiao, Baoxiong Jia, and Siyuan Huang. Closed-loop open-vocabulary mobile manipulation with gpt-4v. *arXiv preprint arXiv:2404.10220*, 2024. 8

[79] Peiyuan Zhi, Zhiyuan Zhang, Muzhi Han, Zeyu Zhang, Zhitian Li, Ziyuan Jiao, Baoxiong Jia, and Siyuan Huang. Closed-loop open-vocabulary mobile manipulation with gpt-4v, 2024. 2

[80] Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. *arXiv preprint arXiv:2407.12366*, 2024. 2

[81] Jun Zhu, Zihao Du, Haotian Xu, Fengbo Lan, Zilong Zheng, Bo Ma, Shengjie Wang, and Tao Zhang. Navi2gaze: Leveraging foundation models for navigation and target gazing. *arXiv preprint arXiv:2407.09053*, 2024. 2