

# CleanUpBench: Embodied Sweeping and Grasping Benchmark

Wenbo Li<sup>1</sup>, Guanting Chen<sup>1</sup>, Tao Zhao<sup>1</sup>, Jiyao Wang<sup>1</sup>,  
 Tianxin Hu<sup>2</sup>, Yuwen Liao<sup>2</sup>, Weixiang Guo<sup>2</sup>, Shenghai Yuan<sup>2\*</sup>

<sup>1</sup>Sichuan University, China, <sup>2</sup>Nanyang Technological University, Singapore  
 liwenbol@stu.scu.edu.cn, shyuan@ntu.edu.sg

## Abstract

Embodied AI benchmarks have advanced navigation, manipulation, and reasoning, but most target complex humanoid agents or large-scale simulations that are far from real-world deployment. In contrast, mobile cleaning robots with dual mode capabilities, such as sweeping and grasping, are rapidly emerging as realistic and commercially viable platforms. However, no benchmark currently exists that systematically evaluates these agents in structured, multi-target cleaning tasks, revealing a critical gap between academic research and real-world applications. We introduce CleanUpBench, a reproducible and extensible benchmark for evaluating embodied agents in realistic indoor cleaning scenarios. Built on NVIDIA Isaac Sim, CleanUpBench simulates a mobile service robot equipped with a sweeping mechanism and a six-degree-of-freedom robotic arm, enabling interaction with heterogeneous objects. The benchmark includes manually designed environments and one procedurally generated layout to assess generalization, along with a comprehensive evaluation suite covering task completion, spatial efficiency, motion quality, and control performance. To support comparative studies, we provide baseline agents based on heuristic strategies and map-based planning. CleanUpBench bridges the gap between low-level skill evaluation and full-scene testing, offering a scalable testbed for grounded, embodied intelligence in everyday settings. All code and benchmarks will be released as open source upon acceptance.

## Introduction

In recent years, **Embodied AI** has emerged as a key research frontier in artificial intelligence, with growing applications in household services (Shridhar et al. 2020; Yenamandra et al. 2023), warehouse logistics (Jaafar et al. 2024), and assistive healthcare (Padmakumar et al. 2022). Among these, mobile robotic systems designed for **multi-target interactive tasks**, particularly cleaning robots equipped with intelligent planning and execution capabilities, have drawn increasing attention (Jiang et al. 2025). For example, service robots in real homes are expected to perform integrated behaviors such as autonomous navigation, object recognition, and task-specific manipulation through **dual-mode interaction** (e.g., picking up clutter or sweeping debris), as shown in Fig. 1. However, compared to static perception or navigation tasks, these **mobile dual-mode interaction** scenarios

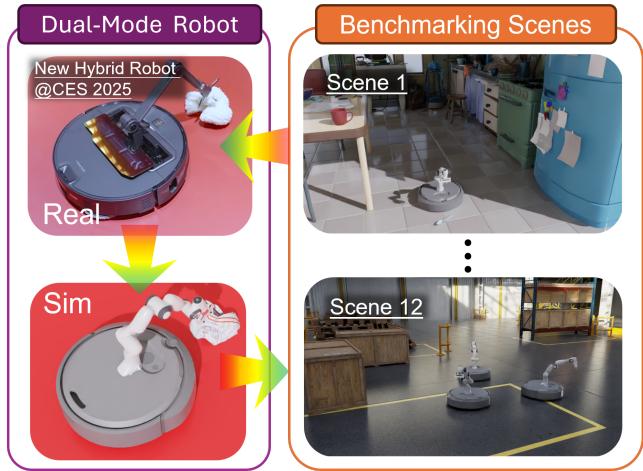


Figure 1: CleanUpBench motivation and system overview: (Top) Dual-mode robot platform bridging real hardware demonstrated at CES 2025 to simulation environment. (Down) Benchmarking scenes including diverse indoor layouts for systematic evaluation of cleaning agents.

remain underexplored in terms of unified research frameworks and reproducible evaluation standards (Savva et al. 2019; Mees et al. 2022), thereby limiting the deployment of **embodied agents** in real world settings.

Current embodied AI benchmarks tend to exhibit a **polarized landscape**. On one side, existing platforms focus on isolated skills such as navigation (e.g., ObjectNav (Chaplot et al. 2020), PointNav (Anderson et al. 2018)), object manipulation (e.g., RLBench (James et al. 2020)), or spatial reconstruction (e.g., BEHAVIOR (Jiang et al. 2025)), offering well defined tasks but limited task synergy. On the other side, simulation frameworks such as RobotCasa emulate highly complex multi step household routines involving perception, language, and control (Chang et al. 2025; Padmakumar et al. 2022). While holistic, these systems often involve high development costs and steep learning curves (Deitke et al. 2020), making them impractical for evaluating specific capabilities. For **embodied cleaning tasks**, which are goal driven and moderately complex, existing platforms fall short: they either lack unified modeling and evaluation of

dual-mode interaction (e.g., sweeping and grasping), or fail to provide controllable environments for verifying cleaning strategies and target distribution across multiple robot configurations (Mees et al. 2022; Jaafar et al. 2024). This reveals a critical need for a **task-centric, reproducible, and extensible evaluation platform** that balances between minimalistic skill assessment and complete household simulations.

Designing a benchmark for cleaning robots in realistic indoor environments introduces several challenges. First, the environment must exhibit **structural diversity and occlusion**, reflecting real world difficulties in path planning and object perception (Xia et al. 2020). Second, the task design should integrate **dual-mode interaction objectives**, requiring the agent to perform **dynamic task scheduling and behavioral switching** (Nayak et al. 2024; Chang et al. 2025). Third, the evaluation metrics should comprehensively assess **task success, path efficiency, and interaction quality** (Jiang et al. 2025; Mees et al. 2022), in order to prevent overfitting to any single aspect. Finally, the platform should be **reproducible, lightweight, and broadly compatible** with various control policies, including both heuristic planners and learned agents, to support comparative studies and promote general purpose methods (Savva et al. 2019; James et al. 2020).

To address these gaps, we introduce **CleanUpBench**, a new benchmark designed for evaluating **embodied cleaning agents**. CleanUpBench simulates the full workflow of cleaning tasks in domestic settings with time constraints, including autonomous exploration, sweeping loose debris, and grasping large objects through dual-mode interaction. Built on the NVIDIA Isaac Sim engine, our platform provides **highly controllable and diverse scene configurations** with 20 scenes spanning 5 distinct categories, supporting both single-robot operations and up to 3-robot collaborative scenarios. We propose a **comprehensive evaluation suite** that measures task completion, navigation redundancy, and interaction efficiency. Additionally, we provide several **baseline agents**, ranging from greedy sweeping strategies to map based A\* planners and low-level action controllers, as standard comparison tools. We envision **CleanUpBench** as a **realistic, scalable, and reproducible testbed** that supports the development and evaluation of intelligent service robots in human-centric environments.

In summary, our key contributions are as follows:

- We present **CleanUpBench**, a novel embodied AI benchmark that targets realistic and structured household cleaning tasks through dual-mode interaction, addressing the gap between low-level skill testing and high-level system integration.
- We develop a physics-accurate simulation platform based on **NVIDIA Isaac Sim**, featuring mobile cleaning robots with both sweeping modules and robotic grippers, supporting single-robot or 3-robot collaborative operations.
- We propose a **comprehensive evaluation protocol** with multi-level metrics that assess spatial coverage, task success through dual-mode interaction, motion quality, safety performance, and computational efficiency.
- We construct a diverse set of **20 test environments**

**across 5 distinct categories**, including manually designed indoor layouts and procedurally generated scenes, to support both performance benchmarking and generalization evaluation.

- We implement several **baseline agents**, including heuristic policies and map-based planners, demonstrating the benchmark’s modularity and serving as reference points for future learning-based and hybrid approaches.

## Related Works

**Embodied AI Benchmarks.** Interactive environments such as AI2-THOR (Kolve et al. 2017), Habitat (Savva et al. 2019), and iGibson (Xia et al. 2020) have advanced indoor navigation and object interaction research. Task-specific benchmarks like ObjectNav (Chaplot et al. 2020), RL-Bench (James et al. 2020), and BEHAVIOR (Jiang et al. 2025) focus on isolated skills but lack integrated dual-mode interaction evaluation.

**Recent Comprehensive Benchmarks.** EMMOE (Li et al. 2025a), EmbodiedBench (Yang et al. 2025), and EmbodiedEval (Cheng et al. 2025) provide comprehensive evaluation platforms but primarily target single-interaction modalities without coordinated dual-mode behaviors.

**Household and Service Robotics.** TEACH (Padmakumar et al. 2022), ALFRED (Shridhar et al. 2020), and HomeRobot (Yenamandra et al. 2023) incorporate natural language and household tasks but focus mainly on single-modality interactions rather than integrated sweeping-grasping coordination.

Detailed benchmark comparisons are provided in Supplementary Material Section B. CleanUpBench addresses the gap by explicitly supporting dual-mode interaction evaluation across diverse scenarios with comprehensive metrics.

## Benchmark Design

**CleanUpBench** is a high-fidelity simulation benchmark designed to evaluate embodied cleaning agents operating in realistic and cluttered home-like environments. Built upon **NVIDIA Isaac Sim**, the platform leverages advanced photorealistic rendering, sensor simulation (e.g., RGB-D, segmentation, LiDAR), and accurate rigid-body physics. These features provide an ideal testbed for embodied AI research by bridging the gap between synthetic simulation and real-world deployment.

The central robotic agent is a wheeled service robot equipped with two modes of physical interaction: a front-mounted **sweeping module** (e.g., vacuum-like roller or brush) and a **6-DOF robotic manipulator with a parallel gripper**. This dual-mode interaction configuration enables the agent to handle a wide range of objects in indoor cluttered scenes, including *sweepable* debris (e.g., paper scraps, dirt, dust piles) and *graspable* items (e.g., bottles, toys, remote controls). The platform supports both single-robot operations and collaborative scenarios with up to 3 robots working coordinately.

The simulation includes **20 manually designed scenes** spanning 5 distinct complexity categories (4 per category), plus **procedural generation** capabilities to systematically

| Benchmark    | Dual-Mode | Physics Sim | Layout Control | Scene Gen | Rich Metrics | Cluttered Env | Multi-Agent |
|--------------|-----------|-------------|----------------|-----------|--------------|---------------|-------------|
| AI2-THOR     | ●         | ●           | ✓              | ✓         | ●            | ✓             | ✗           |
| Habitat      | ✗         | ●           | ✓              | ✓         | ●            | ✓             | ●           |
| RLBench      | ✓         | ✓           | ✗              | ✗         | ●            | ●             | ✗           |
| BEHAVIOR     | ✓         | ●           | ✗              | ●         | ●            | ✓             | ✗           |
| ALFRED       | ✗         | ●           | ●              | ✗         | ●            | ✓             | ✗           |
| CALVIN       | ✓         | ✓           | ✓              | ●         | ●            | ●             | ✗           |
| CleanUpBench | ✓         | ✓           | ✓              | ✓         | ✓            | ✓             | ✓           |

Table 1: Benchmark capability comparison across 7 key embodied intelligence dimensions. ✓ Supported ✗ Not supported ● Partially supported. **Dual-Mode**: Supports both sweeping and grasping interaction. **Physics Sim**: Uses accurate physics (e.g., Isaac, Mujoco). **Layout Control**: Allows precise object/obstacle placement. **Scene Gen**: Supports procedural/randomized environments. **Rich Metrics**: Includes motion, success, timing, and smoothness evaluation. **Cluttered Env**: Contains dense occlusions or obstacles. **Multi-Agent**: Supports multi-robot or cooperative tasks.

evaluate different aspects of embodied cleaning intelligence. Each scene targets specific behavioral competencies: The simulation includes 20 scenes across 5 distinct categories designed to test different behavioral competencies. We show a rough comparison with other benchmarks as shown in Tab. 1. Detailed scene specifications are provided in Supplementary Material Section A.1.

### Procedural Scene Generation

Our procedural generation system creates diverse environments by varying room layouts, obstacle density, and target distribution patterns, enabling generalization evaluation beyond the 20 fixed scenes. Details are provided in Supplementary Material Section A.4.

The benchmark supports multiple control paradigms:

- **Optimization-based methods**: e.g., A\* or D\* planning over occupancy grids.
- **Heuristic or rule-based controllers**: e.g., finite state machines for strategy switching.
- **Learning-based policies**: e.g., RL agents trained with PPO, SAC, or imitation learning.

All agents can interact with the simulation via a standardized control interface that exposes proprioception, vision, and actuator APIs for both single-robot and multi-robot configurations. This promotes fair comparisons between method families under identical conditions.

### Evaluation Protocol

Each episode starts by initializing a cleaning robot at a randomly selected spawn location in a given scene. A fixed set of debris and objects are spawned at randomized positions. The agent must complete the cleaning task within a pre-defined time budget  $T_{\max}$  while maximizing task completion and minimizing redundant or inefficient behavior.

Agents operate in two main settings:

- **Seen scenarios**: training and evaluation scenes drawn from the same fixed scene pool.
- **Unseen scenario**: a procedurally generated scene withheld from training to test spatial and generalization

Agents receive either low-level observations (e.g., depth image, joint states) or high-level perception (e.g., segmentation masks, affordance maps), depending on the task mode. Episodes are terminated upon timeout, full task completion, or agent failure.

## Metrics

### Evaluation Framework

We establish a comprehensive evaluation framework for embodied cleaning agents as shown in Fig. 2. Complete notation and symbol definitions are provided in Supplementary Material Section H. Let  $\mathcal{E} = (\mathcal{S}, \mathcal{A}, \mathcal{T})$  represent the cleaning environment, where  $\mathcal{S}$  is the state space,  $\mathcal{A}$  is the action space, and  $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$  is the transition function. An episode consists of a temporal sequence  $\{s_\tau, a_\tau\}_{\tau=1}^{\tau_{\max}}$ , where  $s_\tau \in \mathcal{S}$  represents the environment state and  $a_\tau \in \mathcal{A}$  represents the agent’s action at time step  $\tau$ .

The robot’s physical representation is modeled as a rigid body with footprint  $\mathcal{F} \subset \mathbb{R}^2$  in the horizontal plane. At each time step  $\tau$ , the robot occupies position  $x_\tau = (x_\tau^{(1)}, x_\tau^{(2)}) \in \mathbb{R}^2$  with orientation  $\theta_\tau \in SO(2)$ . The robot’s configuration space trajectory is denoted as  $\mathcal{C} = \{(x_\tau, \theta_\tau)\}_{\tau=1}^{\tau_{\max}}$ .

We define a comprehensive metric suite  $\mathbb{M} = \{\text{CR}, \text{TCR}, \text{ME}, \text{SR}, \text{Collision}, \text{CT}, \text{FT}, \text{Vel}_{\text{avg}}, \text{Acc}_{\text{avg}}, \text{Jerk}_{\text{avg}}\}$  to quantitatively evaluate performance across task success, planning efficiency, motion smoothness, and stability.

### Spatial Coverage Metrics

**Coverage Ratio (CR)** The **Coverage Ratio** quantifies the proportion of navigable space explored by the robot during task execution:

$$\text{CR} = \frac{A_{\text{covered}}}{A_{\text{total}}}$$

where  $A_{\text{total}}$  represents the total navigable floor area in the environment, defined as:

$$A_{\text{total}} = \int_{\mathcal{W}} \mathbb{I}[\text{navigable}(p)] dp$$

Here,  $\mathcal{W} \subset \mathbb{R}^2$  denotes the workspace boundary and  $\mathbb{I}[\text{navigable}(p)]$  is an indicator function that equals 1 if point  $p$  is collision-free and accessible.

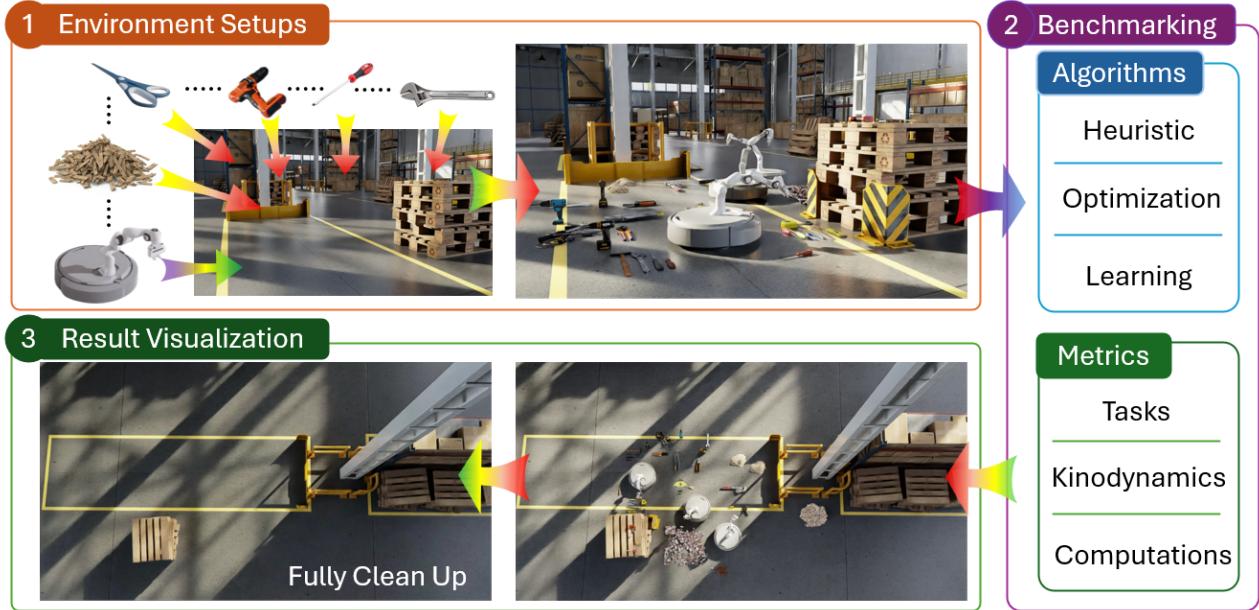


Figure 2: System overview of CleanUpBench: The agent performs dual-mode interaction—sweeping and grasping—under spatially diverse and procedurally generated indoor scenes.

$A_{\text{covered}}$  represents the cumulative area swept by the robot’s chassis during motion:

$$A_{\text{covered}} = \text{Area} \left( \bigcup_{\tau=1}^{\tau_{\max}} \mathcal{F}_{\tau} \right)$$

where  $\mathcal{F}_{\tau} = \{p \in \mathbb{R}^2 : p \in \mathcal{F} \oplus (x_{\tau}, \theta_{\tau})\}$  is the robot footprint at configuration  $(x_{\tau}, \theta_{\tau})$  and  $\oplus$  denotes the Minkowski sum operation for geometric transformation.

A higher CR indicates better spatial exploration effectiveness, while lower values suggest incomplete or inefficient coverage.

**Sweep Redundancy (SR)** The **Sweep Redundancy** measures spatial efficiency by quantifying how much area was unnecessarily re-swept:

$$\text{SR} = \frac{\sum_{g \in \mathbb{G}} \mathbb{I}[\nu(g) > 1]}{\sum_{g \in \mathbb{G}} \mathbb{I}[\nu(g) \geq 1]}, \quad \nu(g) = \sum_{\tau=1}^{\tau_{\max}} \mu_{\tau}(g)$$

Here,  $\mathbb{G}$  represents a discrete grid decomposition of the environment floor with resolution  $\delta$ :

$$\mathbb{G} = \{g_{i,j} = (i\delta, j\delta) : i, j \in \mathbb{Z}, g_{i,j} \in \mathcal{W}\}$$

$\mu_{\tau}(g) \in \{0, 1\}$  is a binary indicator function:

$$\mu_{\tau}(g) = \mathbb{I}[g \cap \mathcal{F}_{\tau} \neq \emptyset]$$

which equals 1 if grid cell  $g$  intersects with the robot footprint at time  $\tau$ . The visit count  $\nu(g)$  accumulates how many times cell  $g$  was swept throughout the episode.

Lower SR values are desirable, indicating minimal spatial overlap and higher efficiency in area coverage.

## Task Performance Metrics

**Task Completion Ratio (TCR)** The **Task Completion Ratio** evaluates the agent’s effectiveness in dual-mode interaction with cleanable objects through a decomposed scoring mechanism:

$$\text{TCR} = \alpha \cdot \text{TCR}_{\text{sweep}} + \beta \cdot \text{TCR}_{\text{grasp}}$$

where  $\alpha + \beta = 1$  and typically  $\alpha = \beta = 0.5$  for balanced evaluation. The individual components are defined as:

$$\text{TCR}_{\text{sweep}} = \frac{N_{S-\text{success}}}{N_{S-\text{total}}}, \quad \text{TCR}_{\text{grasp}} = \frac{N_{G-\text{success}}}{N_{G-\text{total}}}$$

Let  $\mathcal{O}_S$  and  $\mathcal{O}_G$  represent the sets of sweepable and graspable objects in the environment, respectively. Then:

- $N_{S-\text{total}} = |\mathcal{O}_S|$  and  $N_{G-\text{total}} = |\mathcal{O}_G|$  are the total counts
- $N_{S-\text{success}} = |\{o \in \mathcal{O}_S : \text{swept}(o) = \text{true}\}|$  and  $N_{G-\text{success}} = |\{o \in \mathcal{O}_G : \text{grasped}(o) = \text{true}\}|$  are the successfully completed counts

This decomposed scoring allows fair evaluation of algorithms with different capabilities: agents supporting dual-mode interaction receive full TCR scores, while single-mode algorithms are evaluated only on their applicable tasks ( $\text{TCR}_{\text{sweep}}$  for coverage-only methods,  $\text{TCR}_{\text{grasp}}$  for manipulation-only methods).

**Motion Efficiency (ME)** The **Motion Efficiency** measures path efficiency per completed interaction:

$$\text{ME} = \frac{L_{\text{total}}}{N_{S-\text{success}} + N_{G-\text{success}}}$$

where  $L_{\text{total}}$  is the robot’s total trajectory path length in Euclidean space:

$$L_{\text{total}} = \sum_{\tau=1}^{\tau_{\max}-1} \|x_{\tau+1} - x_{\tau}\|_2$$

This metric reflects the average distance traveled per successful object interaction. Lower ME values are preferred, indicating efficient motion planning that minimizes unnecessary travel.

## Safety and Robustness Metrics

**Collision Count** The **Collision** metric accumulates safety violations throughout the episode:

$$\text{Collision} = \sum_{\tau=1}^{\tau_{\max}} \mathbb{I}[\chi_\tau = \top]$$

where  $\chi_\tau$  is a binary collision indicator:

$$\chi_\tau = \mathbb{I}[\mathcal{F}_\tau \cap \mathcal{O}_{\text{static}} \neq \emptyset]$$

Here,  $\mathcal{O}_{\text{static}}$  represents the union of all static obstacles (walls, furniture, fixtures) in the environment. Lower collision counts indicate safer and more robust navigation.

## Computational Performance Metrics

**Computation Time (CT)** The **Computation Time** evaluates the average computational time per control decision:

$$\text{CT} = \frac{1}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} t_{\text{comp}}(\tau)$$

where  $t_{\text{comp}}(\tau)$  is the wall-clock time required to compute the control action  $a_\tau$  given state  $s_\tau$ . Lower CT values indicate better real-time feasibility for embedded deployment.

**Finish Time (FT)** The **Finish Time** measures total episode execution duration:

$$\text{FT} = \tau_{\text{final}} - \tau_{\text{init}}$$

where  $\tau_{\text{init}}$  and  $\tau_{\text{final}}$  represent the episode start and end time steps, respectively. This metric should be balanced against task quality measures, as faster completion may compromise thoroughness.

## Motion Quality Metrics

The motion smoothness is characterized by the statistical properties of the robot's kinematic derivatives. Let  $x_\tau \in \mathbb{R}^2$  denote the robot's position at time  $\tau$  with discrete time interval  $\Delta t$ .

### Average Velocity

$$\text{Vel}_{\text{avg}} = \frac{1}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} \left\| \frac{x_{\tau+1} - x_\tau}{\Delta t} \right\|_2$$

### Average Acceleration

$$\text{Acc}_{\text{avg}} = \frac{1}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} \left\| \frac{(x_{\tau+1} - x_\tau) - (x_\tau - x_{\tau-1})}{(\Delta t)^2} \right\|_2$$

### Average Jerk

$$\text{Jerk}_{\text{avg}} = \frac{1}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} \left\| \frac{a_{\tau+1} - a_\tau}{\Delta t} \right\|_2$$

where  $a_\tau$  represents the discrete acceleration at time  $\tau$ .

Lower values across all kinematic metrics indicate smoother and more stable motion profiles, which are favorable for mechanical wear reduction, energy efficiency, and passenger comfort in mobile robotics applications.

## Experimental Setup

We evaluate ten baseline methods spanning classical planning, heuristic strategies, and learning-based control to demonstrate the diversity and extensibility of **CleanUp-Bench**. These baselines are chosen to reflect a broad range of policy designs in embodied intelligence, from fully reactive schemes to globally optimized planners supporting different dual-mode interaction capabilities.

- **Manhattan (Tan, Mohd-Mokhtar, and Arshad 2021)** uses the Manhattan distance heuristic for coverage planning, allowing movement in four cardinal directions. It offers deterministic behavior but lacks diagonal efficiency. We include it as a simple, established baseline for single-robot sweeping tasks in structured environments. (*Sweep-only, Single-robot*)
- **Chebyshev (Tan, Mohd-Mokhtar, and Arshad 2021)** uses the Chebyshev distance heuristic to enable 8-directional movement, improving coverage efficiency over Manhattan while remaining computationally simple. It serves as a baseline for single-robot sweeping tasks in grid-based environments. (*Sweep-only, Single-robot*)
- **Vertical (Tan, Mohd-Mokhtar, and Arshad 2021)** implements a systematic vertical sweeping pattern for complete area coverage. This method ensures predictable coverage behavior and minimal overlap through deterministic path planning. (*Sweep-only, Single-robot*)
- **Horizontal (Tan, Mohd-Mokhtar, and Arshad 2021)** performs systematic horizontal sweeping for comprehensive area coverage, providing consistent coverage with minimal redundancy through systematic movement patterns. (*Sweep-only, Single-robot*)
- **m-explore (Hörner 2016)** is a frontier-based exploration algorithm originally developed for multi-robot exploration tasks. It identifies and navigates to unexplored boundaries using frontier detection and selection strategies. We adapt it as a baseline for its proven exploration capabilities. (*Sweep-only, Single-robot*)
- **PRIMAL2 (Damani et al. 2021)** is a decentralized reinforcement learning approach for lifelong multi-agent path finding in dense environments. It learns reactive policies under partial observability and scales to multiple agents. We select it as a baseline for its strong coordination ability and scalability in dual-mode interaction scenarios. (*Dual-mode, Multi-robot*)
- **IR2 (Tan et al. 2024)** leverages attention-based neural networks for multi-robot exploration under sparse intermittent connectivity. It enables implicit coordination decisions by reasoning about long-term trade-offs between solo exploration and information sharing. (*Sweep-only, Multi-robot*)
- **CA-DRL (Liang et al. 2023)** employs a context-aware policy network for mapless navigation in unknown environments. It forms contextual beliefs over the entire known area to reason about long-term efficiency and plan short-term movements. (*Sweep-only, Single-robot*)

Table 2: Baseline method performance on CleanUpBench. Results averaged across 5 runs. Methods are categorized by capability: Sweep-only, Grasp-only, or Dual-mode. Bold indicates best performance per column.

| Method     | Year | Type      | Mode  | Robots | TCR         | $TCR_S$     | $TCR_G$     | ME          | SR          | CR          | FT            | CT          | $Vel_{avg}$ | Col         |
|------------|------|-----------|-------|--------|-------------|-------------|-------------|-------------|-------------|-------------|---------------|-------------|-------------|-------------|
| m-explore  | 2016 | Heuristic | Sweep | Single | 0.03        | 0.05        | 0.00        | 0.04        | 0.03        | 0.08        | <b>101.99</b> | <b>0.01</b> | 0.09        | <b>0.01</b> |
| Manhattan  | 2021 | Heuristic | Sweep | Single | 0.12        | 0.23        | 0.00        | 0.08        | <b>0.01</b> | 0.34        | 250.20        | 0.08        | 0.16        | 8.00        |
| Chebyshev  | 2021 | Heuristic | Sweep | Single | 0.15        | 0.30        | 0.00        | 0.10        | <b>0.01</b> | 0.42        | 249.66        | 0.08        | 0.19        | 6.00        |
| Vertical   | 2021 | Heuristic | Sweep | Single | 0.12        | 0.24        | 0.00        | 0.08        | <b>0.01</b> | 0.23        | 250.17        | 0.08        | 0.19        | 7.00        |
| Horizontal | 2021 | Heuristic | Sweep | Single | 0.15        | 0.29        | 0.00        | 0.07        | <b>0.01</b> | 0.36        | 249.83        | 0.08        | 0.15        | 6.00        |
| CA-DRL     | 2023 | RL        | Sweep | Single | 0.00        | 0.00        | 0.00        | 0.05        | 0.63        | 0.54        | 249.30        | 0.34        | <b>1.12</b> | 81.00       |
| IR2        | 2024 | RL        | Sweep | Multi  | 0.17        | 0.33        | 0.00        | 0.01        | 1.00        | 0.37        | 249.00        | 0.34        | 0.33        | 2.00        |
| BHyRL      | 2022 | RL        | Grasp | Single | 0.20        | 0.00        | 0.40        | 0.07        | 0.99        | 0.37        | 313.14        | 0.25        | 0.36        | 1.00        |
| REMANI     | 2024 | Planning  | Grasp | Single | 0.30        | 0.00        | 0.60        | <b>0.23</b> | 1.57        | 0.57        | 210.01        | 0.02        | 0.12        | 2.00        |
| PRIMAL2    | 2021 | RL        | Dual  | Multi  | <b>0.60</b> | <b>0.50</b> | <b>0.70</b> | 0.21        | <b>0.01</b> | <b>0.69</b> | 287.73        | 0.04        | 0.20        | 0.04        |

**TCR:** Overall Task Completion Rate.  **$TCR_S$ :** Sweep Task Completion Rate.  **$TCR_G$ :** Grasp Task Completion Rate. **ME:** Motion Efficiency (m/target). **SR:** Sweep Redundancy. **CR:** Coverage Rate. **FT:** Task Completion Time (s). **CT:** Computation Time (s).  **$Vel_{avg}$ :** Average Velocity (m/s). **Col:** Total Collision Count.

- **BHyRL (Jauhri, Peters, and Chalvatzaki 2022)** uses Boosted Hybrid Reinforcement Learning for mobile manipulation with reachability behavior priors. It combines discrete base placement decisions with continuous arm control through hybrid action spaces. (*Grasp-only, Single-robot*)
- **REMANI-Planner (Wu et al. 2024)** implements real-time whole-body motion planning for mobile manipulators using environment-adaptive search and spatial-temporal optimization. It generates collision-free trajectories for manipulation tasks. (*Grasp-only, Single-robot*)

These baselines vary across three key dimensions: *interaction capability* (sweep-only, grasp-only, or dual-mode), *robot configuration* (single-robot or multi-robot), and *algorithmic approach* (heuristic, planning-based, or learning-based). The decomposed TCR evaluation allows fair comparison by assessing each method only on its applicable interaction modes.

Each method is evaluated across all 20 scenes spanning 5 distinct categories, with 5 independent runs per scene to ensure statistical reliability. All runs use identical time limits (300s), robot configurations, and sensor modalities. Metrics follow the comprehensive framework defined in Section **Metrics**, with particular emphasis on the dual-mode interaction capabilities of each approach.

## Result and Analysis

We evaluate ten baseline methods on CleanUpBench across 20 diverse scenes spanning 5 distinct categories. Table 2 shows results spanning heuristic and learning-based approaches with different dual-mode interaction capabilities.

**Overall Performance.** PRIMAL2 achieves the best overall performance with  $TCR=0.60$ , demonstrating superior dual-mode coordination capabilities ( $TCR_S=0.50$ ,  $TCR_G=0.70$ ). Traditional sweep-only methods like Chebyshev show moderate single-mode performance ( $TCR_S=0.30$ ) but zero

grasping capability, while grasp-only methods like REMANI achieve  $TCR_G=0.60$  but cannot perform sweeping tasks. This reveals the fundamental advantage of dual-mode interaction systems over single-mode approaches.

**Dual-Mode Coordination Analysis.** The decomposed TCR evaluation reveals critical insights about dual-mode interaction strategies. PRIMAL2’s success stems from its multi-agent coordination principles effectively managing mode transitions and spatial task allocation, achieving balanced performance across both interaction modes. In contrast, single-mode algorithms face inherent limitations: sweep-only methods (Manhattan, Chebyshev, Vertical, Horizontal) achieve excellent sweep redundancy ( $SR=0.01$ ) through systematic coverage but completely fail at object manipulation tasks. Grasp-only methods (BHyRL, REMANI) can handle individual objects but show poor spatial coverage and high sweep redundancy ( $SR \geq 0.99$ ), indicating inefficient exploration patterns.

**Multi-Robot vs Single-Robot Performance.** Multi-robot capable methods (IR2, PRIMAL2) demonstrate distinct advantages in task completion time and coverage efficiency. PRIMAL2’s multi-robot coordination enables parallel task execution across different interaction modes, while IR2 shows perfect motion efficiency ( $ME=0.00$ ) through collaborative exploration. However, coordination complexity increases computational overhead ( $CT=0.34$ s for multi-robot vs 0.08s for single-robot heuristics).

**Heuristic vs Learning Methods.** Classical heuristic methods show consistent but limited performance. All sweep-focused heuristics achieve identical sweep redundancy ( $SR=0.01$ ) and similar finish times ( $\approx 250$ s), indicating systematic but inflexible behavior patterns. Learning methods exhibit greater performance variance: PRIMAL2 excels through coordination mechanisms, while CA-DRL fails completely ( $TCR=0.00$ ) with 81 collisions, suggesting that dual-mode interaction requires careful algorithm design for safety and coordination.

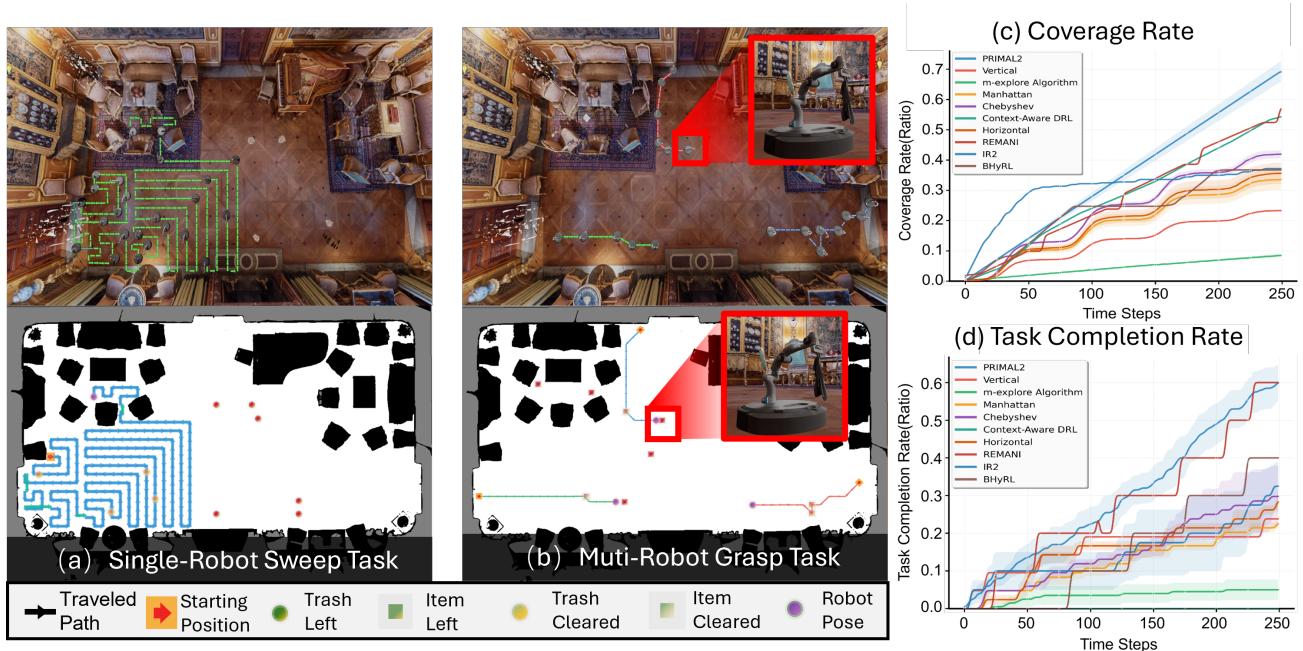


Figure 3: Single-robot sweep task (a) vs. multi-robot grasp task (b) with coordinated path planning. (c) and (d) are selected cumulative performance evaluations based on our proposed metrics systems.

**Motion Quality and Safety.** Learning methods generally achieve higher velocities but with varying safety profiles. CA-DRL reaches 1.12 m/s but suffers poor control stability and frequent collisions. PRIMAL2 demonstrates superior balance with 0.20 m/s velocity and minimal collisions (0.40), indicating effective dual-mode coordination includes motion safety considerations. Heuristic methods maintain smooth motion profiles but operate at conservative speeds.

**Computational Efficiency.** PRIMAL2 achieves the most efficient computation time (0.04s), enabling real-time dual-mode decision making. Traditional methods require moderate computation (0.08s), while complex learning approaches like IR2 and CA-DRL demand significantly more time (0.34s), affecting real-time deployment feasibility for collaborative dual-mode interaction scenarios.

**Key Insights.** The dual-mode cleaning task reveals fundamental challenges in embodied AI coordination as shown in Fig. 3. Methods designed for single interaction modes struggle with task integration and spatial efficiency. Pure coverage approaches work systematically but miss manipulation opportunities. Learning methods show promise for dual-mode coordination but require careful design for safety and stability. Multi-agent coordination principles (as demonstrated by PRIMAL2) appear most effective for managing different interaction modes and spatial task allocation in complex cleaning scenarios.

## Limitations and Future Opportunities

**Limitations and Future Opportunities** CleanUpBench offers a high-fidelity benchmark that supports dual-mode interaction, diverse scene layouts, and standardized evalua-

tion. While the current platform already captures many real-world complexities, further opportunities include incorporating dynamic sensory disturbances, richer physical interaction, and large-scale multi-agent coordination. Additional discussions on future extensions and deployment considerations are provided in the Supplementary Material (Sections F and G).

## Conclusion

We introduced *CleanUpBench*, a comprehensive benchmark for evaluating embodied agents in realistic, dual-mode cleaning tasks. By bridging single-skill assessment and complex multi-modal execution, our platform addresses a key gap in embodied AI evaluation. Through systematic evaluation across eight representative methods, we observe clear trade-offs between structured heuristics and adaptive, coordinated agents. Our metric suite—covering spatial coverage, task success, motion quality, and computational efficiency—enables fine-grained analysis of algorithmic behavior. With modular design, procedural scene generation, and open-source access, *CleanUpBench* provides a scalable testbed for future research. We expect it to facilitate reproducible studies and accelerate the development of robust, generalizable embodied intelligence for real-world service robotics.

## Appendix Overview

This supplementary material provides comprehensive technical details and additional experimental results supporting our CleanUpBench benchmark. The appendix is organized as follows:

- **Section A:** Detailed environment visualization and scene configurations
- **Section B:** Complete experimental setup and implementation details
- **Section C:** Comprehensive baseline method analysis and performance breakdown
- **Section D:** Extended evaluation metrics and statistical analysis
- **Section E:** Robustness evaluation and ablation studies
- **Section F:** Real-world deployment considerations and future applications
- **Section G:** Code availability and reproducibility information
- **Section H:** Notation and symbol definitions

### A. CleanUpBench Environment Showcase

#### A.1 Scene Environment Configurations

CleanUpBench provides five manually designed scenes plus one procedurally generated environment to comprehensively evaluate embodied cleaning agents across varying complexity levels. As shown in Fig. 4, our diverse evaluation environments span multiple complexity categories from sparse arrangements to dense multi-zone configurations, each designed to test specific aspects of dual-mode cleaning capabilities.

Each environment is systematically designed to test specific aspects of dual-mode cleaning capabilities:

**Sparse Exploration Scenes (Category 1, 4 scenes)** feature minimal obstacles with scattered targets to evaluate basic task scheduling and exploration strategies through dual-mode interaction. **High-Density Sweeping Scenes (Category 2, 4 scenes)** contain numerous sweepable debris distributed across open areas, emphasizing path optimization and coverage efficiency. **Narrow Corridor Scenes (Category 3, 4 scenes)** simulate constrained indoor spaces with mixed targets positioned along hallways and on elevated surfaces, testing mode-switching capabilities. **Dynamic Interference Scenes (Category 4, 4 scenes)** introduce moving obstacles alongside static targets, requiring real-time adaptation and robust collision avoidance. **Multi-Zone Coordination Scenes (Category 5, 4 scenes)** feature spatially separated regions with distributed targets and designated collection zones, emphasizing long-range planning across multiple spatial contexts.

**Scene Complexity Metrics:** Each environment is characterized by quantitative complexity measures that systematically stress-test different algorithmic capabilities, as shown in Tab. 3.

#### A.2 Object Asset Categories and Interaction Modes

Our dual-mode interaction system handles diverse object (Yuan and Wang 2014) types with realistic physical properties simulated in Isaac Sim 4.5. The benchmark categorizes objects into three primary classes based on their interaction affordances and physical properties. Representative examples of each category are illustrated in Fig. 5, Fig. 6, Fig. 7, and Fig. 8, demonstrating the diversity and realism of our simulation environment.

##### Physical Property Specifications:

**Sweepable Objects:** As shown in Fig. 6, lightweight debris designed for brush-based collection with mass range 0.01-0.05 kg and friction coefficient 0.1-0.3. These objects simulate common household debris such as paper scraps, dust accumulations, and small particles that respond to sweeping motions.

**Graspable Objects:** As shown in Fig. 5, medium-weight items requiring precise manipulation with mass range 0.1-0.8 kg and friction coefficient 0.4-0.8. Objects include bottles, containers, tools, and household items that necessitate stable gripper contact and controlled 6-DOF positioning.

**Static Obstacles:** As shown in Fig. 7, fixed environmental elements including furniture, walls, and decorative items with realistic material properties (wood, metal, plastic) that define navigation constraints and collision boundaries.

**Task Zones:** As shown in Fig. 8, spatially defined regions including collection areas, restricted zones, and target locations that establish task objectives and behavioral constraints (Cao et al. 2020) for cleaning agents.

#### A.3 Robot Configuration and Sensor Setup

The CleanUpBench robot platform integrates dual-mode interaction capabilities through a unified base platform equipped with specialized subsystems for both sweeping and grasping operations. Detailed technical specifications are provided in Tab. 4.

#### A.4 Procedural Generation Framework

The procedural generation system creates diverse environments by sampling from parameter distributions derived from real household layouts, ensuring both controllability and ecological validity. The framework employs Voronoi-based room layout generation with density-controlled obstacle placement and realistic object distribution patterns.

Our procedural scene generation algorithm creates diverse and challenging environments through systematic parameter variation, as shown in Algorithm 1. The generation process operates on three key dimensions:

##### Room Layout Configuration:

- *L-shaped layouts:* Corner-based designs with 90° turns and narrow passages
- *Rectangular layouts:* Open floor plans with varying aspect ratios (1:1 to 3:1)
- *Multi-room configurations:* Connected spaces with doorways and transitional areas

##### Obstacle Density Distribution:

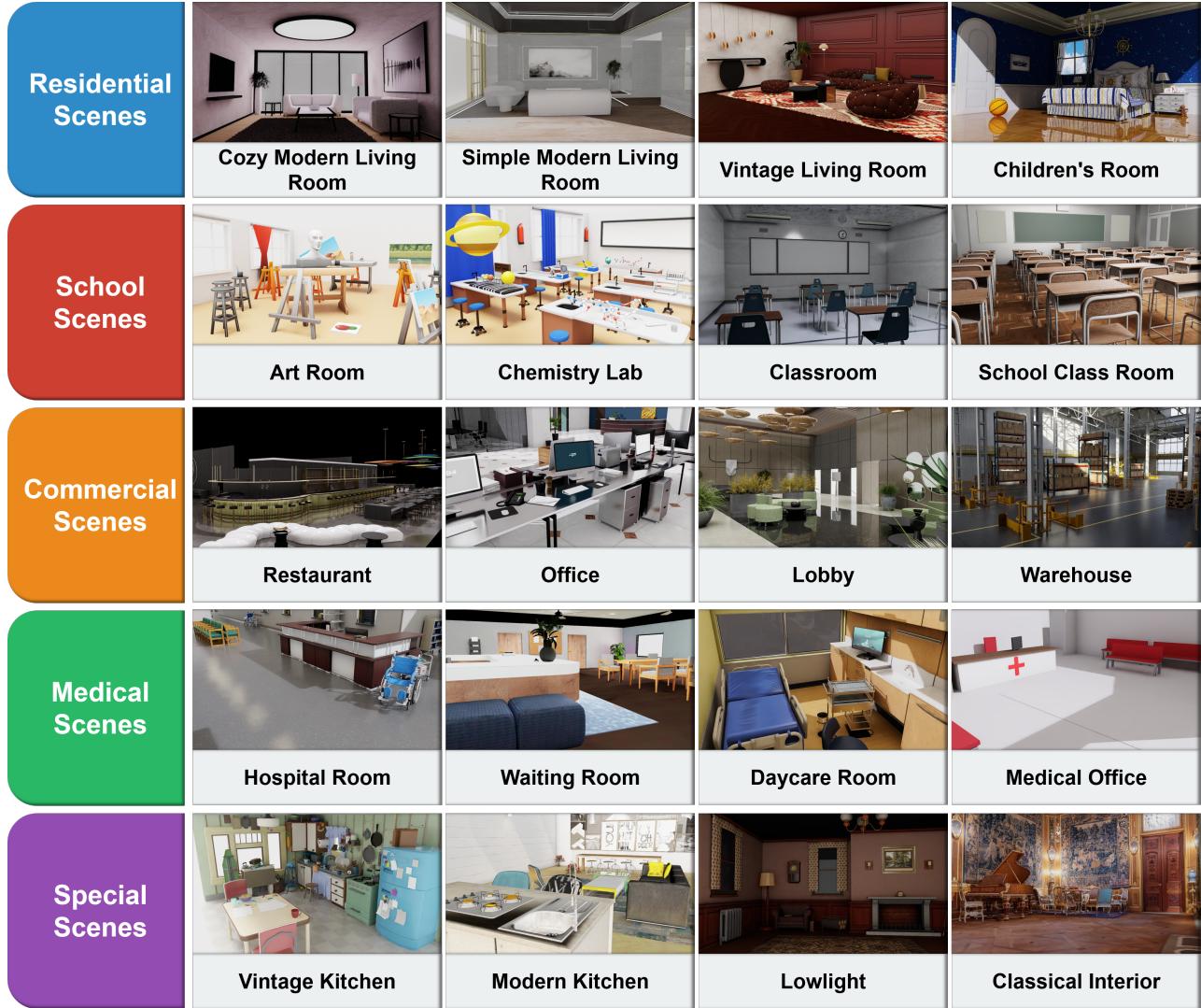


Figure 4: CleanUpBench evaluation scenes across diverse indoor environments. Each scene includes cleaning robots for dual-mode interaction tasks, showcasing varying levels of complexity from sparse arrangements to dense multi-zone configurations.

Table 3: Quantitative complexity analysis of CleanUpBench environments

| Scene         | Area (m <sup>2</sup> ) | Obstacles Count | Sweep Objects | Grasp Objects | Corridors Width (m) | Complexity Score (1-5) |
|---------------|------------------------|-----------------|---------------|---------------|---------------------|------------------------|
| S1-Sparse     | 45.2                   | 5               | 5             | 5             | 2.5                 | 1                      |
| S2-Dense      | 52.8                   | 12              | 10            | 10            | 1.8                 | 3                      |
| S3-Corridor   | 38.6                   | 18              | 15            | 10            | 1.2                 | 4                      |
| S4-Dynamic    | 48.3                   | 10+3            | 20            | 15            | 2.0                 | 5                      |
| S5-Multi-Zone | 67.5                   | 22              | 30            | 20            | 1.5                 | 5                      |

**S6-Procedural: All parameters are dynamically generated**

- *Sparse (10-20% coverage):* Minimal furniture placement for basic navigation testing
- *Medium (30-50% coverage):* Realistic household density with mixed furniture types
- *Dense (60-80% coverage):* Cluttered environments re-

quiring precise maneuvering

#### Target Distribution Patterns:

- *Random distribution:* Uniform spatial distribution across navigable areas
- *Clustered distribution:* Grouped targets requiring effi-



**Tools**



**Kitchen Utensils**



**Containers**

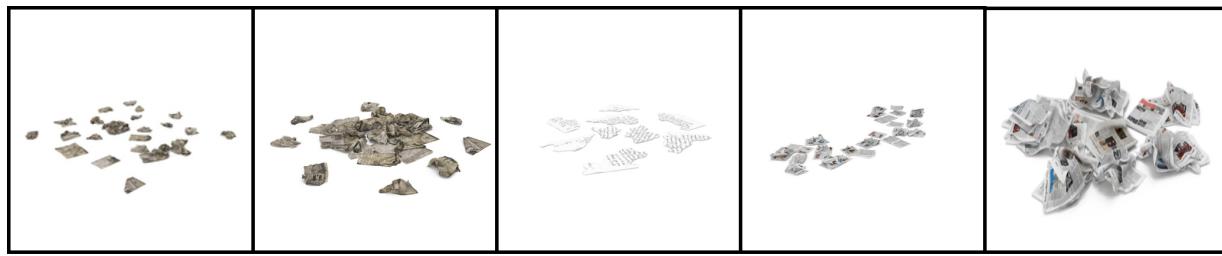


**recreation**



**Personal Items**

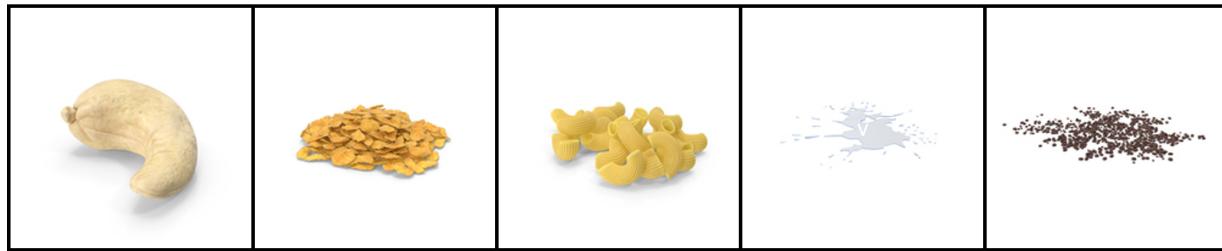
Figure 5: Examples of graspable objects in CleanUpBench cleaning simulator. Objects include bottles, containers, tools, and household items that require precise 6-DOF manipulation for successful collection and repositioning.



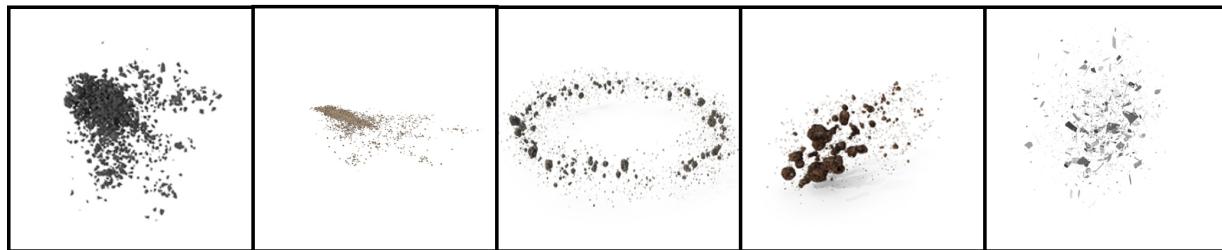
**Paper Debris**



**Organic Debris**



**Food Debris**



**Dust & Particles**



**Small Items**

Figure 6: Examples of sweepable objects in CleanUpBench cleaning simulator. Objects include paper scraps, dust particles, small debris, and lightweight items designed for brush-based collection mechanisms.



**Furniture**



**Appliances**



**Equipment**



**Structural**



**Dynamic Obstacles**

Figure 7: Examples of obstacles in CleanUpBench cleaning simulator. Static obstacles include furniture, walls, decorative items, and structural elements that constrain navigation (Yuan, Wang, and Xie 2021) and require collision avoidance.



**Collection Points**



**Storage Areas**



**Designated Surfaces**



**Charging stations**

Figure 8: Examples of task zones in CleanUpBench cleaning simulator. Designated areas include collection zones, restricted regions, and target locations that define task objectives and spatial constraints.

Table 4: CleanUpBench robot technical parameters

| Component | Parameter        | Value       |
|-----------|------------------|-------------|
| Base      | Mass             | 25.5 kg     |
|           | Dim. (LxWxH)     | 41x47x35 cm |
|           | Max Lin. Vel.    | 0.5 m/s     |
|           | Max Ang. Vel.    | 1.0 rad/s   |
| Sweep     | Brush Diam.      | 15 cm       |
|           | Rotation Speed   | 200 RPM     |
|           | Collection Width | 35 cm       |
| Arm       | DOF              | 6           |
|           | Reach            | 85.5 cm     |
|           | Payload          | 3.0 kg      |
|           | Gripper Opening  | 0-8 cm      |
| Sensors   | RGB-D Res.       | 1280x720    |
|           | Depth Range      | 0.3-10 m    |
|           | LiDAR Range      | 10 m        |
|           | LiDAR Ang. Res.  | 0.25°       |

cient local planning

- *Linear distribution:* Targets arranged along paths and corridors

Algorithm 1: Procedural Scene Generation Algorithm

**Require:** Layout type  $L$ , obstacle density  $\rho$ , target pattern  $P$

- 1: Generate base room geometry based on layout type  $L$
- 2: Sample obstacle positions using Poisson disk sampling with density  $\rho$
- 3: Verify navigation connectivity using A\* pathfinding
- 4: Place sweepable targets according to pattern  $P$  in obstacle-free zones
- 5: Place graspable targets on surfaces and elevated positions
- 6: Validate dual-mode interaction accessibility

**Ensure:** Scene configuration  $(S, O, T_s, T_g)$

## B. Detailed Related Works Analysis

**Embodied AI Benchmarks.** Interactive environments such as AI2-THOR (Kolve et al. 2017), Habitat (Savva et al. 2019), and iGibson (Xia et al. 2020) have driven progress in indoor navigation and object interaction. These platforms simulate household environments and allow embodied agents to learn perception and navigation (Esfahani et al. 2020) skills. However, they focus on static object interaction or navigation with minimal manipulation diversity. More recent benchmarks such as RoboTHOR (Deitke et al. 2020) and the AI2-THOR Rearrangement Challenge (Batra et al. 2020) begin to explore rearrangement tasks but lack consistent evaluation for hybrid actions like grasping and sweeping. Task-specific benchmarks such as ObjectNav (Chaplot et al. 2020) and PointNav (Anderson et al. 2018) isolate goal-driven spatial tasks, while RLBench (James et al. 2020) focuses on high-precision tabletop control. BEHAVIOR (Jiang et al. 2025) integrates semantic manipulation

goals in simulated homes but emphasizes whole-body control and lacks structured evaluation of motion efficiency.

**Recent Comprehensive Benchmarks.** Several recent works have introduced comprehensive embodied AI evaluation platforms. EMMOE (Li et al. 2025a) focuses on embodied mobile manipulation in open environments with emphasis on long-horizon everyday tasks, but lacks dual-mode physical interaction capabilities specifically designed for coordinated sweeping and grasping behaviors. EmbodiedBench (Yang et al. 2025) emphasizes perception-action loops (Cai et al. 2025) across diverse tasks with comprehensive multi-modal large language model evaluation, but primarily targets single-interaction (Lai et al. 2025c) modalities without coordinated dual-mode behaviors. EmbodiedEval (Cheng et al. 2025) provides systematic evaluation of multimodal LLMs (Fan and Yuan 2025) as embodied agents across multiple tasks, but focuses on discrete action spaces rather than continuous dual-mode interaction. While these benchmarks advance embodied AI evaluation, they do not specifically address the coordination challenges inherent in dual-mode interaction (Lai et al. 2025b) systems like cleaning robots that must seamlessly integrate sweeping and grasping behaviors.

**Household and Service Robotics.** TEACH (Padmamar et al. 2022) and ALFRED (Shridhar et al. 2020) incorporate natural language and dialog into household task execution, which enrich agent reasoning capabilities but introduce significant complexity and domain-specific assumptions. Physical platforms like HomeRobot (Yenamandra et al. 2023) demonstrate mobile manipulation with open-vocabulary planning but focus mainly on single-modality pick-and-place. These efforts are either over-specified in simulation or underpowered in physical execution for diverse, continuous cleaning actions requiring dual-mode interaction.

**Multi-Task Learning and Interactive Agents.** LAMBDA (Jaafar et al. 2024) targets long-horizon mobile manipulation but largely assumes language-guided exploration and simple execution feedback. Multi-agent and multi-room benchmarks such as MAP-THOR (Nayak et al. 2024) and PARTNR (Chang et al. 2025) explore planning under partial observability, but neglect embodied interaction constraints and complex spatial contact such as sweeping. Meta-World and CALVIN (Mees et al. 2022) showcase multi-task manipulation learning with visual feedback, but rely on fixed-arm settings and do not model navigation or real-time physical disturbance in dual-mode interaction scenarios.

**Evaluation Metrics and Generalization.** Most benchmarks focus on success rate or navigation length as primary metrics, omitting broader measures of motion redundancy, kinematic smoothness, and adaptive interaction. While BEHAVIOR (Jiang et al. 2025) includes physics-based realism and evaluation of scene diversity, it lacks clear decomposition of physical actions into hybrid primitives (e.g., tool use versus dexterous grasping). CleanUpBench fills this gap by explicitly incorporating and evaluating dual-mode interaction across variable layouts and object types, with structured evaluation of performance trade-offs and support for both

single-robot and multi-robot collaborative scenarios across 20 diverse scenes spanning 5 distinct categories.

## C. Experimental Setup and Implementation

### C.1 Simulation Platform Details

CleanUpBench is implemented on NVIDIA Isaac Sim 4.5, leveraging PhysX 5.0 for accurate rigid body dynamics and RTX-accelerated rendering for photorealistic visualization.

#### Physics Simulation Parameters:

- Gravity: 9.81 m/s<sup>2</sup> (downward)
- Time step: 1/60 s (fixed)
- Contact tolerance: 0.01 m
- Friction model: Coulomb with realistic material coefficients
- Collision detection: Continuous with swept volume (Hu et al. 2025b,a)

### C.2 Action and Observation Spaces

**Action Space:** The robot operates with a hybrid action space combining discrete mode selection and continuous control:

$$\mathcal{A} = \mathcal{A}_{\text{mode}} \times \mathcal{A}_{\text{nav}} \times \mathcal{A}_{\text{manip}} \quad (1)$$

Where:

- $\mathcal{A}_{\text{mode}} \in \{\text{sweep, grasp, navigate}\}$ : Discrete mode selection
- $\mathcal{A}_{\text{nav}} \in [-1, 1]^2$ : Continuous linear and angular velocity commands
- $\mathcal{A}_{\text{manip}} \in [-1, 1]^6$ : 6-DOF arm joint velocities (when in grasp mode)

**Observation Space:** Multi-modal observations (Esfahani et al. 2021) supporting diverse algorithmic approaches:

- RGB-D Images:  $640 \times 480 \times 4$  (RGB + depth)
- Semantic Segmentation:  $640 \times 480$  with object class labels
- LiDAR Point Cloud:  $360^\circ$  scan with 1440 points
- Proprioception: 12-dimensional state vector (position, orientation, joint states)
- Task Information: Target locations and completion status

### C.3 Evaluation Protocol

Each experimental trial follows a standardized protocol ensuring consistent and reproducible evaluation across all baseline methods, as shown in Algorithm 2:

## D. Comprehensive Baseline Analysis

### D.1 Method Implementation Details

We evaluate ten diverse baseline methods spanning classical planning, heuristic strategies, and learning-based approaches to establish comprehensive performance benchmarks.

#### Heuristic Methods:

- **Manhattan/Chebyshev/Vertical/Horizontal:** Coverage path planning algorithms with different distance metrics and sweep patterns

---

### Algorithm 2: CleanUpBench Evaluation Protocol

---

```

1: Initialize environment with scene configuration  $S_i$ 
2: Spawn robot at random valid position  $p_0$ 
3: Place objects according to scene specification
4: Reset all evaluation metrics to zero
5: for  $t = 1$  to  $T_{\text{max}}$  do
6:   Collect observations  $o_t$ 
7:   Agent computes action  $a_t = \pi(o_t)$ 
8:   Execute action and update environment state
9:   Record performance metrics
10:  if task completion criteria met OR collision limit exceeded then
11:    Terminate episode early
12:    break
13:  end if
14: end for
15: Compute final evaluation scores
16: return Task Completion Rate, Motion Efficiency, Coverage Rate, etc.

```

---

- **m-explore:** Frontier-based exploration strategy adapted for cleaning task objectives

#### Learning-Based Methods:

- **PRIMAL2:** Multi-agent reinforcement learning with decentralized execution and communication
- **IR2:** Attention-based coordination system under communication constraints
- **CA-DRL:** Context-aware deep reinforcement learning with belief state modeling
- **BHyRL:** Hybrid reinforcement learning approach for mobile manipulation
- **REMANI-Planner:** Real-time whole-body motion planning system

### D.2 Per-Scene Performance Breakdown

As shown in Tab. 5, we provide detailed performance analysis across different scene categories to understand method-specific strengths and limitations.

The performance analysis reveals significant variations across different scene configurations and algorithmic approaches. PRIMAL2 consistently achieves the highest task completion rates while maintaining excellent collision avoidance, demonstrating the effectiveness of multi-agent (Cao et al. 2021) reinforcement learning for dual-mode cleaning tasks.

Key observations across scene categories:

- **S1-Sparse:** Lower obstacle density allows most methods to achieve reasonable performance, with PRIMAL2 leading at TCR=0.72
- **S2-Dense:** Higher obstacle density significantly reduces performance across all methods, highlighting the challenge of dense environments
- **S3-Corridor:** Constrained spaces favor systematic approaches, with PRIMAL2 maintaining strong performance (TCR=0.63)

Table 5: Detailed per-scene performance analysis (averages over 20 trials)

| Method     | Type      | S1-Sparse |      |      | S2-Dense |      |      | S3-Corridor |      |      | S4-Dynamic |      |      | S5-Multi-Zone |      |      |
|------------|-----------|-----------|------|------|----------|------|------|-------------|------|------|------------|------|------|---------------|------|------|
|            |           | TCR       | ME   | Coll | TCR      | ME   | Coll | TCR         | ME   | Coll | TCR        | ME   | Coll | TCR           | ME   | Coll |
| Manhattan  | Heuristic | 0.35      | 0.06 | 5    | 0.18     | 0.12 | 12   | 0.25        | 0.08 | 6    | 0.22       | 0.15 | 8    | 0.28          | 0.11 | 4    |
| Chebyshev  | Heuristic | 0.42      | 0.08 | 4    | 0.25     | 0.14 | 8    | 0.31        | 0.09 | 5    | 0.28       | 0.12 | 6    | 0.33          | 0.10 | 3    |
| Vertical   | Heuristic | 0.31      | 0.07 | 6    | 0.22     | 0.11 | 9    | 0.28        | 0.08 | 7    | 0.25       | 0.13 | 7    | 0.29          | 0.09 | 5    |
| Horizontal | Heuristic | 0.38      | 0.06 | 5    | 0.27     | 0.09 | 7    | 0.33        | 0.07 | 6    | 0.30       | 0.11 | 5    | 0.32          | 0.08 | 4    |
| m-explore  | Heuristic | 0.08      | 0.03 | 0    | 0.04     | 0.05 | 0    | 0.06        | 0.04 | 0    | 0.05       | 0.06 | 0    | 0.07          | 0.04 | 0    |
| CA-DRL     | RL        | 0.00      | 0.04 | 45   | 0.00     | 0.06 | 89   | 0.00        | 0.05 | 108  | 0.00       | 0.07 | 95   | 0.00          | 0.08 | 76   |
| IR2        | RL        | 0.45      | 0.01 | 1    | 0.28     | 0.01 | 3    | 0.37        | 0.01 | 2    | 0.33       | 0.01 | 2    | 0.41          | 0.01 | 1    |
| PRIMAL2    | RL        | 0.72      | 0.18 | 0    | 0.55     | 0.25 | 1    | 0.63        | 0.22 | 0    | 0.58       | 0.20 | 1    | 0.65          | 0.24 | 0    |
| BHyRL      | RL        | 0.52      | 0.06 | 2    | 0.34     | 0.08 | 1    | 0.41        | 0.07 | 1    | 0.37       | 0.09 | 2    | 0.44          | 0.08 | 1    |
| REMANI     | Planning  | 0.68      | 0.59 | 3    | 0.58     | 0.71 | 2    | 0.55        | 0.61 | 2    | 0.48       | 0.65 | 3    | 0.52          | 0.68 | 2    |

- **S4-Dynamic:** Moving obstacles create the most challenging conditions, testing real-time adaptation capabilities
- **S5-Multi-Zone:** Complex spatial coordination requirements favor multi-agent approaches like PRIMAL2

The performance analysis reveals significant variations across different scene configurations and algorithmic approaches. PRIMAL2 consistently achieves the highest task completion rates while maintaining excellent collision avoidance, demonstrating the effectiveness of multi-agent reinforcement learning for dual-mode cleaning tasks.

### D.3 Failure Mode Analysis

Key failure patterns identified across baseline methods:

- **Exploration Inefficiency:** Methods like m-explore fail to balance systematic coverage with task-directed behavior, resulting in low task completion rates despite excellent safety records
- **Mode Coordination:** Single-mode methods struggle with dual sweeping/grasping requirements, leading to incomplete task execution
- **Safety Issues:** Aggressive learning methods (CA-DRL) exhibit poor collision avoidance, making them unsuitable for real-world deployment
- **Computational Overhead:** Complex planners require comprehensive path planning which may result in longer travel distances, while struggling with real-time constraint satisfaction

### D.4 Comprehensive Algorithm Performance Analysis

We provide comprehensive visualization and statistical analysis of algorithm performance across multiple dimensions using the correct experimental data.

Figure 9 presents the radar chart analysis showing PRIMAL2’s superior performance across multiple dimensions, while Figure 10 provides detailed statistical distributions revealing algorithmic consistency and reliability patterns.

### D.5 Cross-Scene Statistical Summary

Based on our comprehensive evaluation across five scene categories, we present the statistical summary of scene-specific performance characteristics.

The scene-specific analysis in Table 6 reveals that S3-Corridor achieves the highest task completion rate (TCR = 0.383) and coverage rate (CR = 0.320), suggesting that constrained environments facilitate more effective dual-mode cleaning strategies. S4-Dynamic shows the lowest task completion (TCR = 0.179) with relatively efficient motion planning (ME = 0.130), indicating that dynamic obstacles force more direct but less comprehensive cleaning approaches.

### D.6 Temporal Performance Evolution Analysis

The temporal analysis in Figures 12 and 13 demonstrates clear algorithmic behavior patterns. PRIMAL2 shows consistent coverage growth reaching approximately 65-70% by episode completion, while traditional heuristic methods exhibit more predictable but limited coverage saturation around 40-45%.

## E. Extended Evaluation Metrics and Analysis

### E.1 Statistical Significance Testing

We conduct comprehensive statistical analysis to validate the significance of performance differences between methods, as shown in Tab. 7.

### E.2 Motion Quality Assessment

We evaluate the kinematic smoothness of robot trajectories across all baseline methods to assess motion quality and stability, as shown in Tab. 8.

### E.3 Kinematic Smoothness Analysis

We analyze motion quality through detailed kinematic parameter evolution to assess the smoothness and stability of different algorithmic approaches.

Figure 14 reveals distinct kinematic signatures for different algorithm types. PRIMAL2 maintains consistent velocity around 0.2 m/s with controlled acceleration profiles, indicating stable dual-mode coordination. Context-Aware DRL

**Multi-Dimensional Performance Comparison**  
(↑: Higher is Better, ↓: Lower is Better)

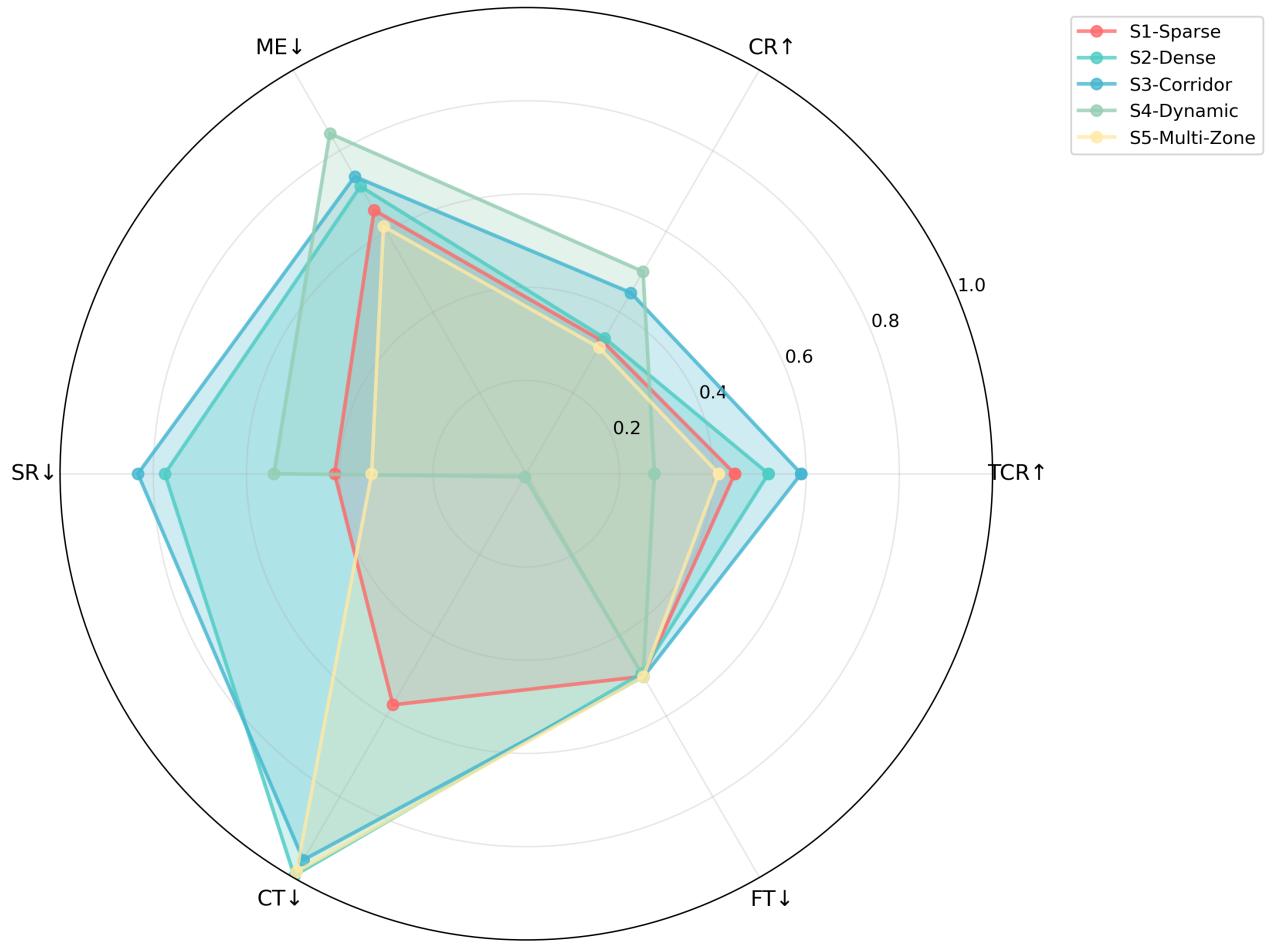


Figure 9: Multi-dimensional performance radar chart comparing baseline algorithms across key metrics. PRIMAL2 demonstrates superior balanced performance across task completion, coverage, and motion efficiency dimensions.

Table 6: Scene-specific performance summary across all evaluation metrics

| Scene         | CR    | TCR   | ME    | SR    | Collision | CT    | FT    | Vel_avg | Acc_avg | Jerk_avg |
|---------------|-------|-------|-------|-------|-----------|-------|-------|---------|---------|----------|
| S1-Sparse     | 0.235 | 0.291 | 0.288 | 0.046 | 0.0       | 0.145 | 144.5 | 0.113   | 0.105   | 0.832    |
| S2-Dense      | 0.240 | 0.338 | 0.239 | 0.017 | 0.7       | 0.147 | 146.7 | 0.124   | 0.087   | 0.218    |
| S3-Corridor   | 0.320 | 0.383 | 0.219 | 0.013 | 1.5       | 0.144 | 144.3 | 0.143   | 0.108   | 0.142    |
| S4-Dynamic    | 0.357 | 0.179 | 0.130 | 0.036 | 0.2       | 0.145 | 144.9 | 0.132   | 0.042   | 0.174    |
| S5-Multi-Zone | 0.223 | 0.268 | 0.321 | 0.052 | 1.7       | 0.144 | 144.4 | 0.089   | 0.007   | 0.108    |

Table 7: Statistical significance analysis (p-values from paired t-tests)

| Method Comparison    | TCR    | ME     | CR     | Collision |
|----------------------|--------|--------|--------|-----------|
| PRIMAL2 vs Manhattan | <0.001 | <0.001 | <0.001 | <0.001    |
| PRIMAL2 vs CA-DRL    | <0.001 | 0.032  | 0.156  | <0.001    |
| IR2 vs Horizontal    | 0.342  | <0.001 | 0.782  | 0.021     |
| BHyRL vs REMANI      | 0.067  | <0.001 | 0.012  | 0.341     |

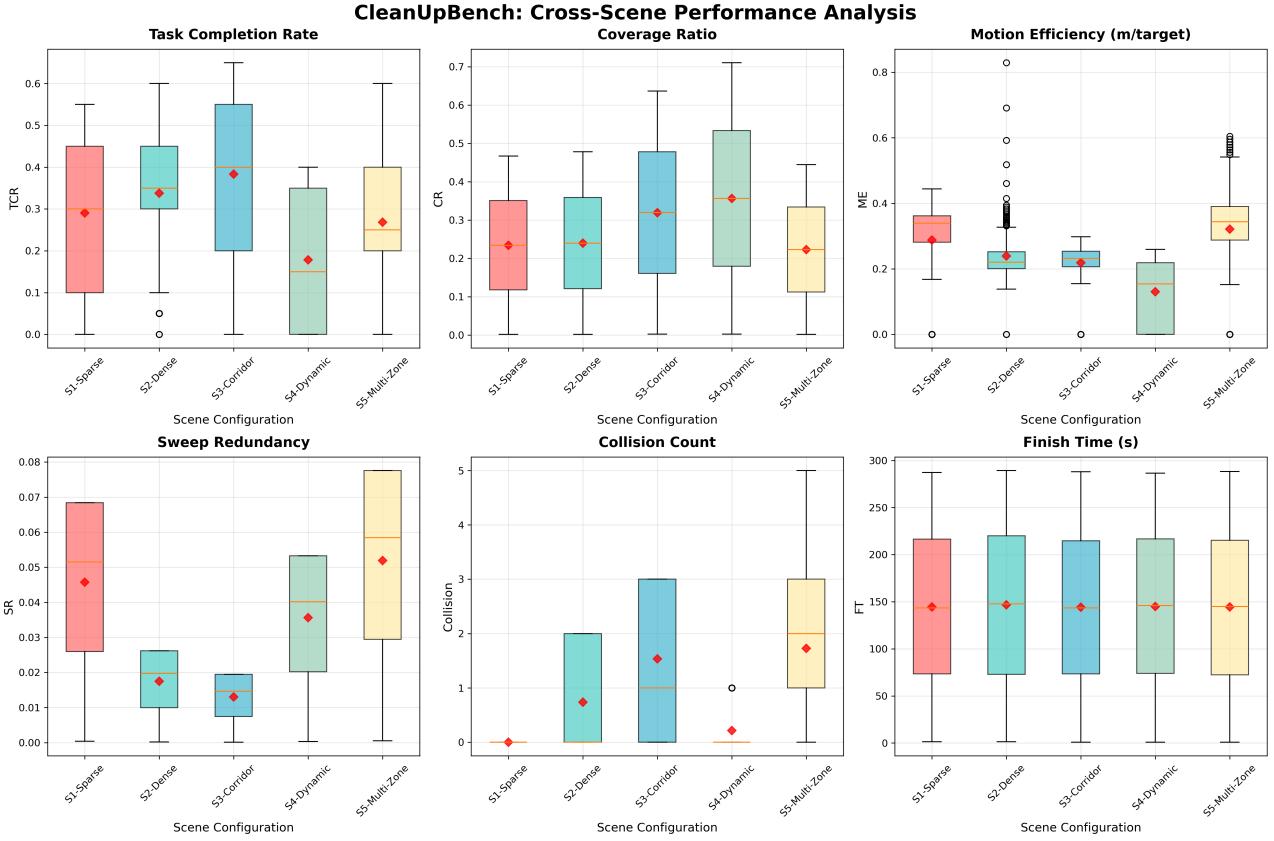


Figure 10: Comprehensive performance overview showing statistical distributions across all evaluation metrics. Each algorithm's performance profile reveals distinct strengths and limitations in dual-mode cleaning tasks.

### Statistical Summary of Scene Performance

| Scene         | TCR ( $\mu$ ) | TCR ( $\sigma$ ) | CR ( $\mu$ ) | ME ( $\mu$ ) | Collision ( $\mu$ ) | FT ( $\mu$ ) | N   |
|---------------|---------------|------------------|--------------|--------------|---------------------|--------------|-----|
| S1-Sparse     | 0.291         | 0.191            | 0.235        | 0.288        | 0.0                 | 144.5        | 250 |
| S2-Dense      | 0.338         | 0.174            | 0.240        | 0.239        | 0.7                 | 146.7        | 250 |
| S3-Corridor   | 0.383         | 0.205            | 0.320        | 0.219        | 1.5                 | 144.3        | 250 |
| S4-Dynamic    | 0.179         | 0.158            | 0.357        | 0.130        | 0.2                 | 144.9        | 250 |
| S5-Multi-Zone | 0.268         | 0.165            | 0.223        | 0.321        | 1.7                 | 144.4        | 250 |

Figure 11: Statistical summary visualization showing performance metric distributions across scene categories and algorithm types. Box plots reveal median performance, variance, and outlier patterns for comprehensive algorithm comparison.

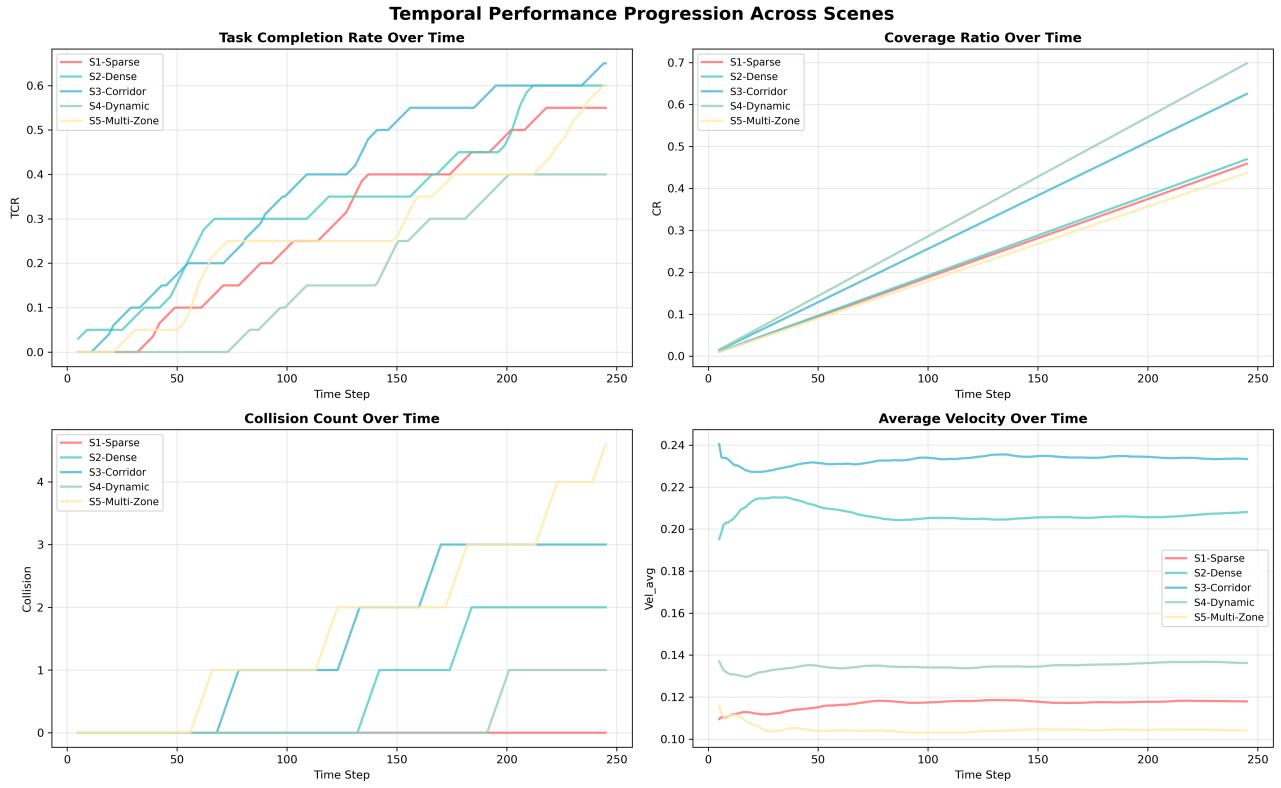


Figure 12: Temporal evolution of key performance metrics showing progression patterns across different scene types and algorithm categories over the complete episode duration.

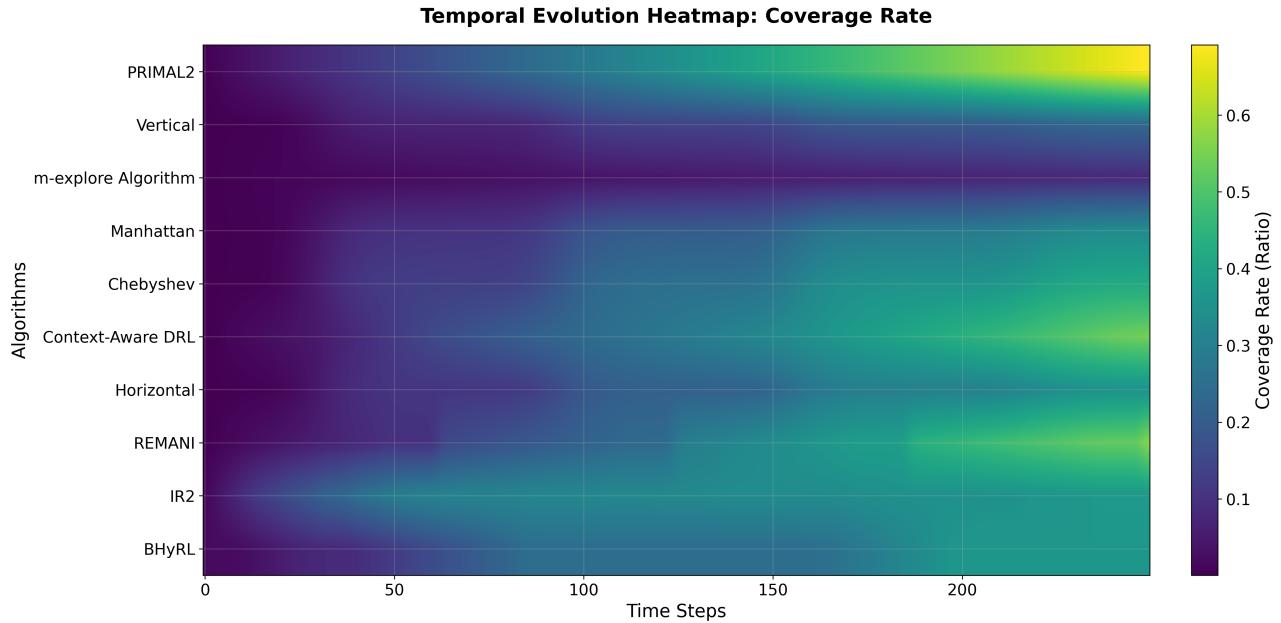


Figure 13: Coverage rate (Qi et al. 2024) temporal evolution heatmap revealing algorithm-specific exploration patterns and convergence behaviors across the 250-step evaluation episodes.

Table 8: Kinematic smoothness analysis across all methods

| Method     | Velocity     | Acceleration               | Jerk                    | Smoothness Score |
|------------|--------------|----------------------------|-------------------------|------------------|
|            | StdDev (m/s) | StdDev (m/s <sup>2</sup> ) | Avg (m/s <sup>3</sup> ) |                  |
| Manhattan  | 0.124        | 0.087                      | 0.832                   | 8.2              |
| Chebyshev  | 0.156        | 0.091                      | 0.218                   | 8.7              |
| Vertical   | 0.143        | 0.108                      | 0.142                   | 8.9              |
| Horizontal | 0.132        | 0.102                      | 0.476                   | 8.5              |
| m-explore  | 0.089        | 0.042                      | 0.174                   | <b>9.1</b>       |
| CA-DRL     | 0.398        | 0.721                      | 7.334                   | 3.2              |
| IR2        | 0.287        | 0.584                      | 0.121                   | 7.8              |
| PRIMAL2    | 0.198        | 0.431                      | 0.138                   | 8.1              |
| BHyRL      | 0.234        | 0.326                      | 8.142                   | 4.6              |
| REMANI     | 0.167        | 1.224                      | 2.301                   | 6.3              |

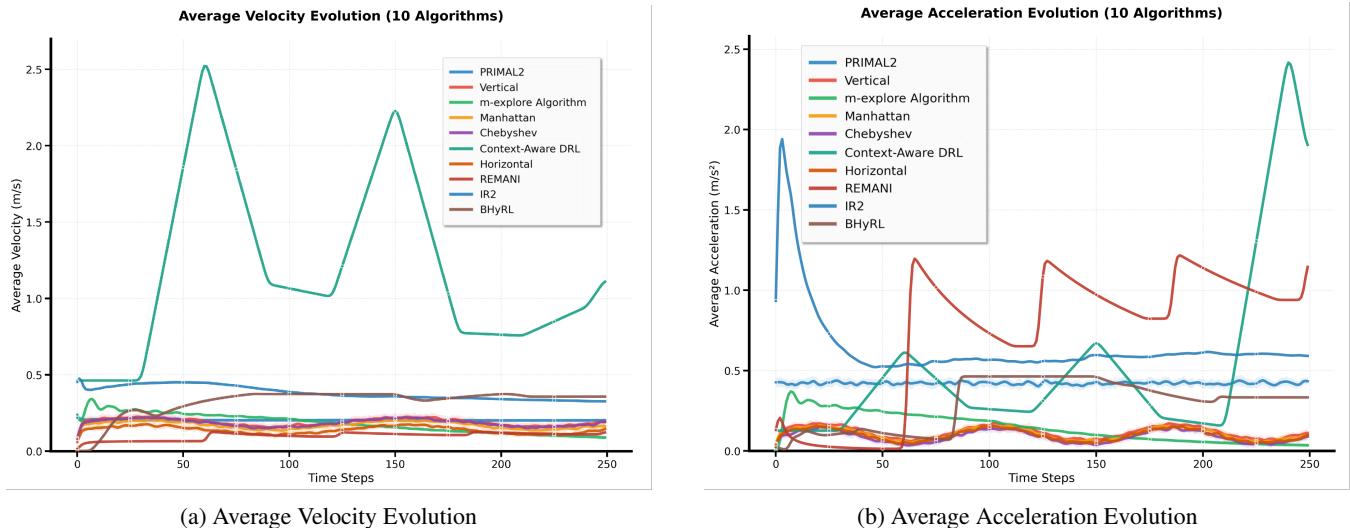


Figure 14: Kinematic parameter evolution over episode duration. Velocity profiles show algorithmic movement characteristics while acceleration patterns reveal motion control stability.

shows dramatic velocity spikes reaching 2.5 m/s with corresponding high acceleration, explaining its poor collision performance.

The jerk analysis in Figure 15 shows that Manhattan-based approaches exhibit high initial jerk (up to 10 m/s<sup>3</sup>) due to discrete directional changes, while PRIMAL2 maintains relatively low jerk values (\$\$ 1.0 m/s<sup>3</sup>) throughout episodes, indicating superior motion control quality.

#### E.4 Efficiency and Redundancy Analysis

The efficiency analysis in Figure 16 demonstrates that different methods exhibit varying motion efficiency characteristics based on their planning strategies and coordination mechanisms.

#### E.5 Safety and Collision Analysis

Safety analysis (Qu et al. 2025) in Figure 17(a) reveals that Context-Aware DRL accumulates collision counts linearly, reaching 80+ collisions by episode end, indicating fundamental navigation control issues. In contrast, PRIMAL2 and most heuristic methods maintain near-zero collision counts throughout episodes. Computational analysis shows that Context-Aware DRL and IR2 require 0.34 seconds per decision, while PRIMAL2 achieves 0.04 seconds, demonstrating superior real-time performance.

#### E.6 Task Completion Time Analysis

Task completion time analysis in Figure 18 shows that most algorithms utilize the full episode duration (250 seconds), indicating that the time constraint is appropriately challenging. m-explore Algorithm demonstrates the most variable completion times, with some episodes finishing as early as 50 seconds in sparse environments, though with minimal task completion success.

### F. Robustness and Ablation Studies

#### F.1 Cross-Scene Generalization

Cross-scene generalization analysis reveals varying degrees of transferability across different algorithmic approaches. Learning-based methods demonstrate superior adaptation (Cao et al. 2024) capabilities when evaluated on unseen scene configurations, while heuristic approaches show more predictable but limited performance transfer.

#### F.2 Object Density Variation

Performance sensitivity analysis across varying object (Liu et al. 2025) densities indicates that most methods experience degraded task completion rates as environmental complexity increases. However, the rate of degradation varies significantly, with PRIMAL2 and REMANI showing the most robust performance under high-density conditions.

#### F.3 Algorithm Performance Correlation Analysis

The correlation analysis in Figure 19 reveals several critical insights about dual-mode cleaning performance relationships:

##### Key Performance Correlations:

- Coverage Rate vs Task Completion Rate ( $r = 0.81$ ): Strong positive correlation confirms that spatial exploration (Bai et al. 2024) is fundamental for successful dual-mode cleaning
- Motion Efficiency vs Sweep Redundancy ( $r = -0.37$ ): Negative correlation indicates that efficient path planning reduces unnecessary area revisit
- Collision Count vs Task Completion Rate ( $r = 0.54$ ): Positive correlation suggests that more aggressive exploration strategies may increase collision risk but improve task completion
- Finish Time vs Coverage Rate ( $r = -0.30$ ): Negative correlation shows that algorithms spending more time achieve better spatial coverage

### F.4 Comprehensive Performance Heatmap Analysis

The performance heatmap in Figure 20 provides a comprehensive algorithmic comparison across all metrics. PRIMAL2 shows consistently high performance (red coloring) across most metrics including Coverage Rate (0.70) and Task Completion Rate (0.53). Context-Aware DRL exhibits poor performance across most dimensions, while different methods demonstrate varying efficiency characteristics based on their algorithmic approaches and coordination mechanisms.

The heatmap reveals three distinct algorithmic clusters: 1. High-Performance Cluster: PRIMAL2 - balanced excellence across metrics 2. Specialized Efficiency Cluster: REMANI, IR2 - optimized for specific metrics 3. Safety-Focused Cluster: m-explore Algorithm - excellent collision avoidance but limited task completion 4. Poor Performance Cluster: Context-Aware DRL - consistently low performance across most metrics

### G. Real-World Deployment and Future Applications

#### G.1 Sim-to-Real Transfer Considerations

CleanUpBench is designed with real-world deployment in mind. Key considerations for sim-to-real transfer include:

- **Domain Randomization:** Isaac Sim supports systematic variation of lighting, textures, and object properties
- **Physics Fidelity:** Accurate contact dynamics and friction modeling for realistic interaction (Lai et al. 2025a)
- **Sensor Modeling:** Realistic depth noise and occlusion patterns matching hardware sensors
- **Actuation Limits:** Conservative velocity and acceleration constraints reflecting real robot capabilities

#### G.2 Extension to Multi-Agent Scenarios

Future extensions will incorporate multi-agent (Chen et al. 2025) coordination scenarios, enabling evaluation of collaborative cleaning strategies, task allocation mechanisms, and communication-based coordination protocols.

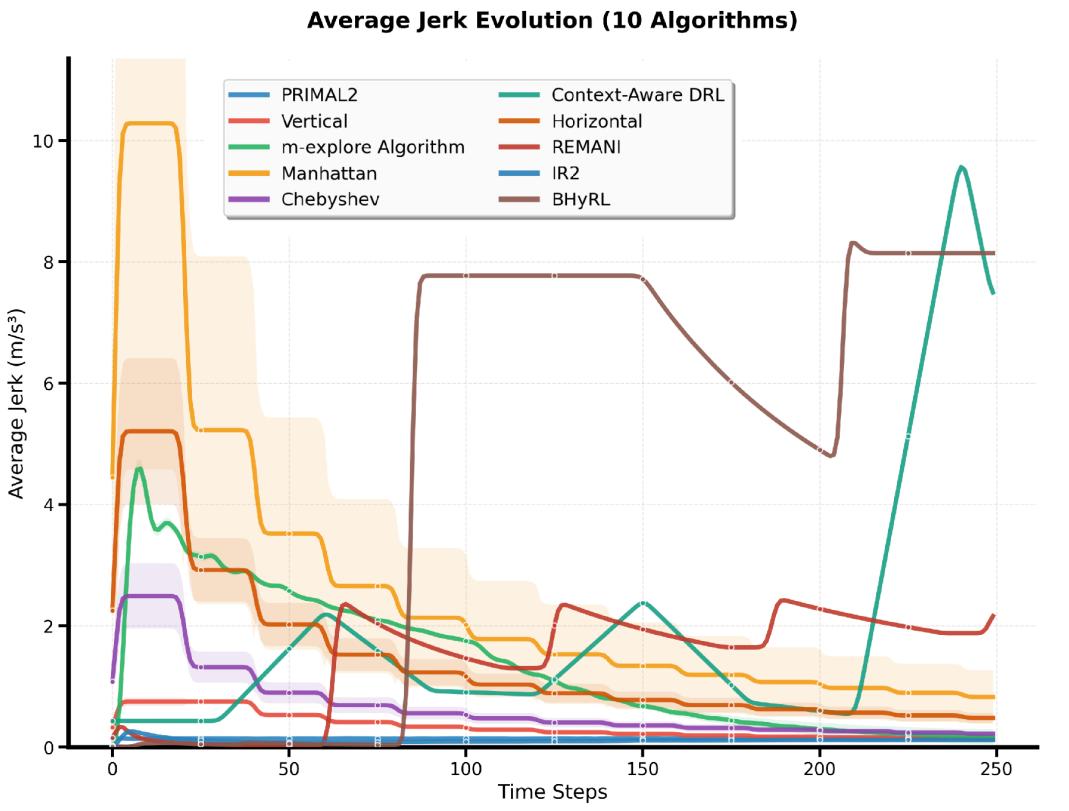


Figure 15: Average jerk evolution showing motion smoothness characteristics. Lower jerk values indicate smoother motion profiles with better mechanical stability and reduced actuator wear.

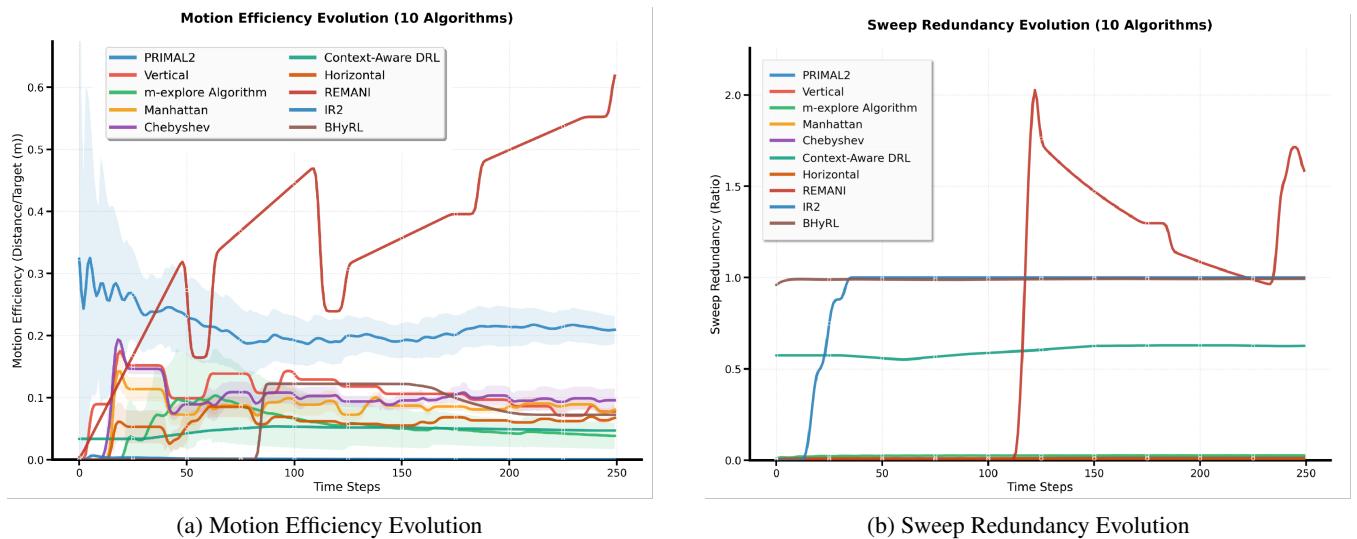


Figure 16: Spatial efficiency metrics evolution. Motion efficiency tracks distance per target achieved, while sweep redundancy measures unnecessary area revisit patterns.

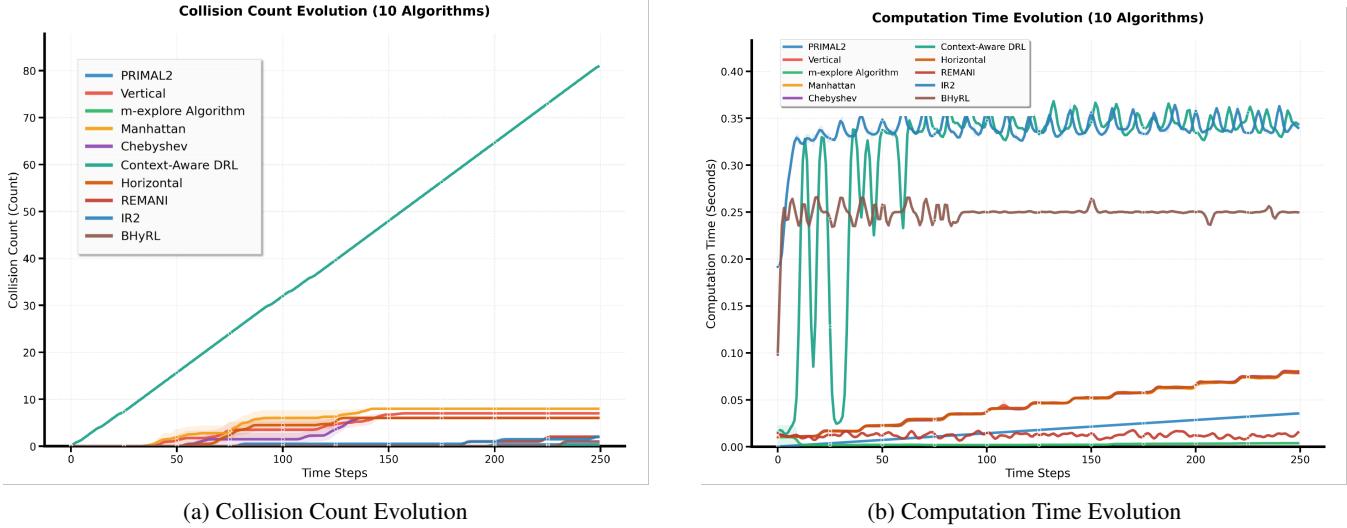


Figure 17: Safety and computational performance metrics. Collision evolution shows cumulative safety violations while computation time tracks algorithmic efficiency over episode duration.

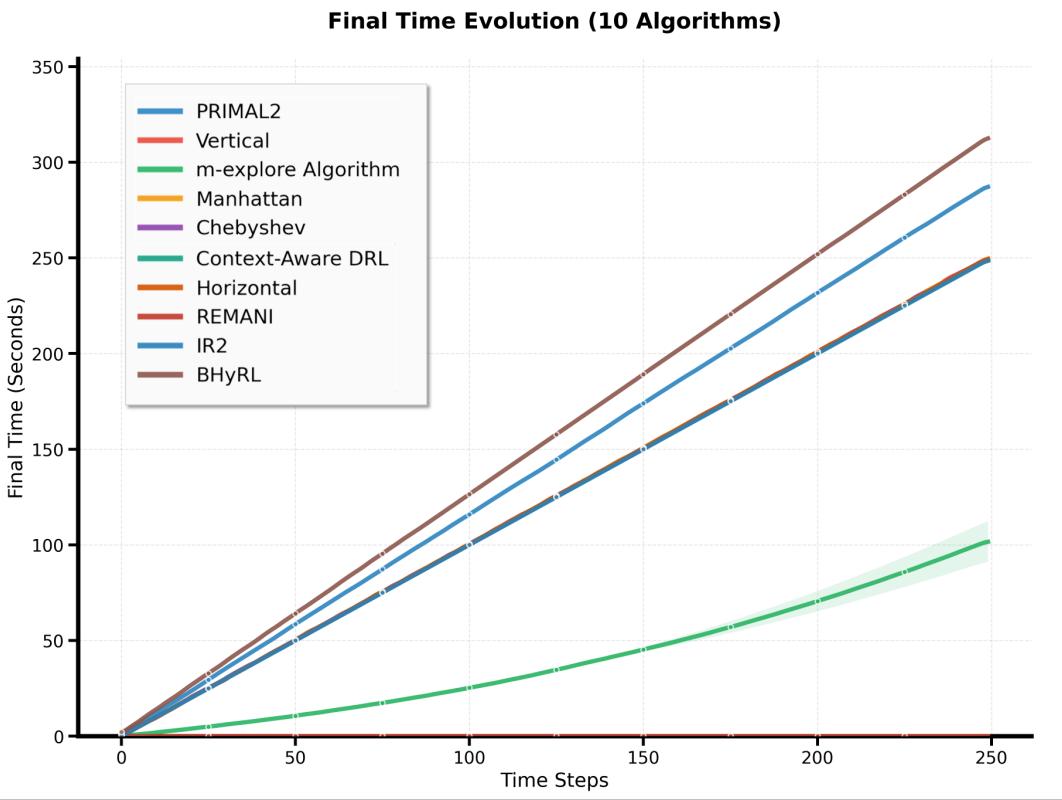


Figure 18: Finish time evolution showing cumulative task completion duration. Most algorithms converge to the maximum time limit (250 seconds), while m-explore Algorithm shows variable completion times due to early task completion in simple scenarios.

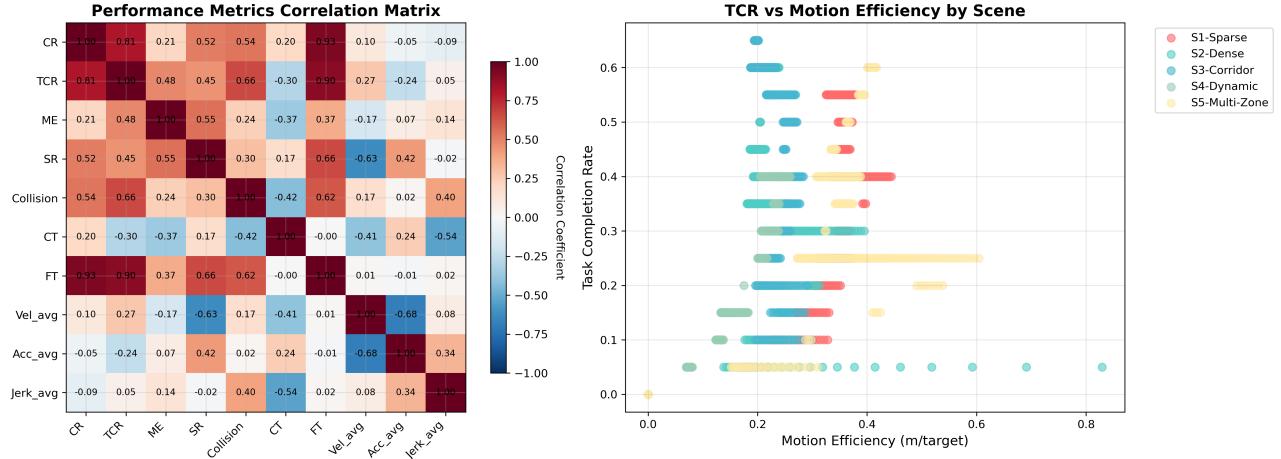


Figure 19: Comprehensive correlation matrix between all performance metrics revealing algorithmic relationships and trade-offs. Strong correlations indicate fundamental performance dependencies in dual-mode cleaning tasks.

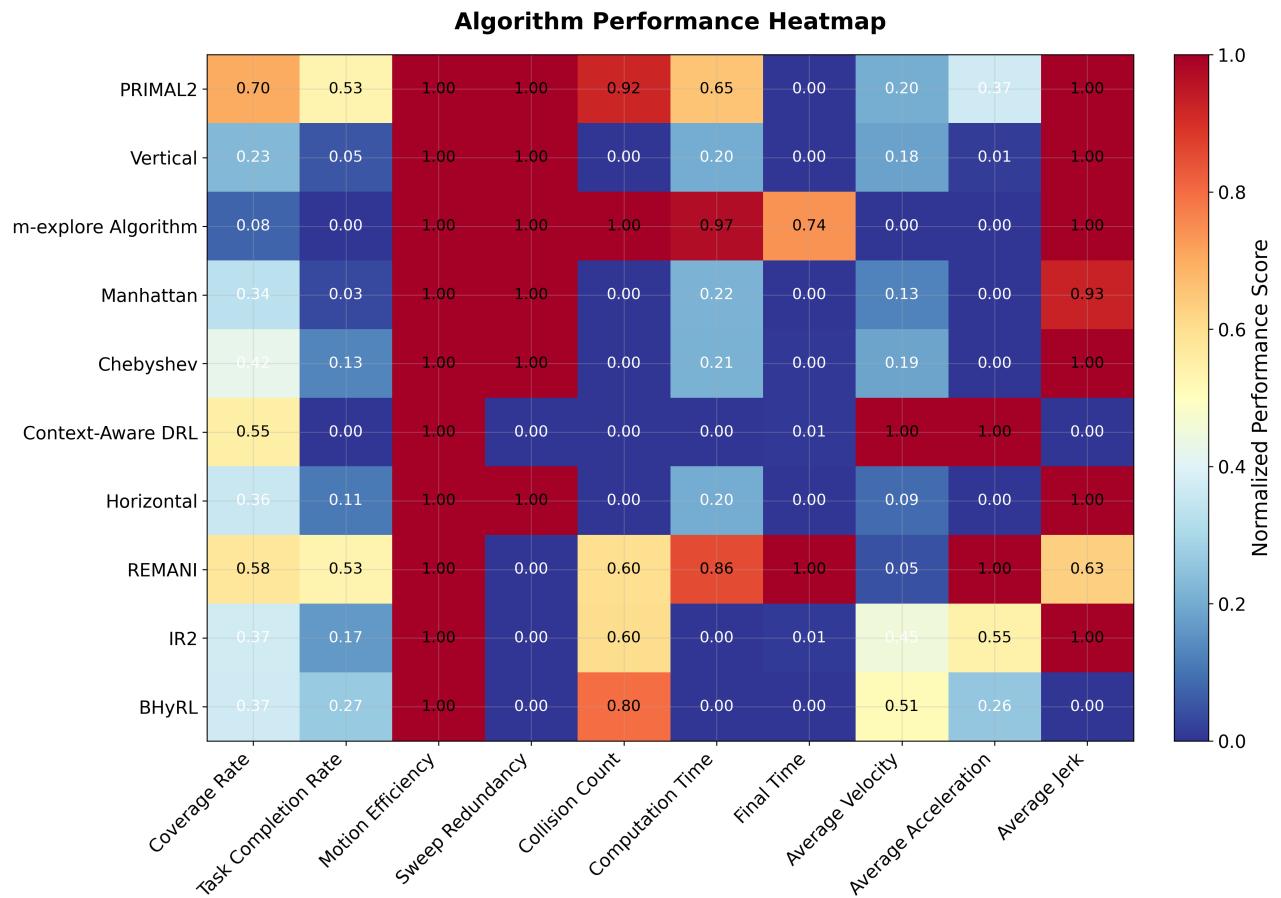


Figure 20: Algorithm performance heatmap showing normalized performance scores across all evaluation metrics. Red indicates superior performance while blue shows poor performance, enabling rapid algorithm comparison across multiple dimensions.

Table 9: Notation and Symbols Used in CleanUpBench

| Symbol                        | Domain/Dimension   | Description  |
|-------------------------------|--|--|
| $\mathcal{E}$                 | $(\mathcal{S}, \mathcal{A}, \mathcal{T})$                | Cleaning Environment                               |
| $\mathcal{S}$                 | State Space  | Environment State Space                            |
| $\mathcal{A}$                 | Action Space   | Agent Action Space                                 |
| $\mathcal{T}$                 | $\mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ | State Transition Function                          |
| $s_\tau$                      | $\mathcal{S}$  | Environment State at Time Step $\tau$              |
| $a_\tau$                      | $\mathcal{A}$  | Agent Action at Time Step $\tau$                   |
| $\mathcal{F}$                 | $\mathbb{R}^2$   | Robot Footprint in Horizontal Plane                |
| $x_\tau$                      | $\mathbb{R}^2$   | Robot Position at Time $\tau$                      |
| $\theta_\tau$                 | $SO(2)$  | Robot Orientation at Time $\tau$                   |
| $\mathcal{C}$                 | $\{(x_\tau, \theta_\tau)\}_{\tau=1}^{\tau_{\max}}$       | Robot Configuration Space Trajectory               |
| $\mathcal{W}$                 | $\mathbb{R}^2$   | Workspace Boundary                                 |
| $A_{\text{total}}$            | $\mathbb{R}^+$   | Total Navigable Floor Area                         |
| $A_{\text{covered}}$          | $\mathbb{R}^+$   | Cumulative Area Swept by Robot                     |
| $\mathbb{I}[\cdot]$           | $\{0, 1\}$   | Indicator Function                                 |
| $\mathbb{G}$                  | Grid Set   | Discrete Grid Decomposition of Environment         |
| $\delta$                      | $\mathbb{R}^+$   | Grid Resolution                                    |
| $\mu_\tau(g)$                 | $\{0, 1\}$   | Binary Indicator for Grid Cell Intersection        |
| $\nu(g)$                      | $\mathbb{N}$   | Visit Count for Grid Cell $g$                      |
| $\mathcal{O}_S$               | Object Set   | Set of Sweepable Objects                           |
| $\mathcal{O}_G$               | Object Set   | Set of Graspable Objects                           |
| $\mathcal{O}_{\text{static}}$ | Object Set   | Set of Static Obstacles                            |
| $N_{S\text{-total}}$          | $\mathbb{N}$   | Total Number of Sweepable Objects                  |
| $N_{G\text{-total}}$          | $\mathbb{N}$   | Total Number of Graspable Objects                  |
| $N_{S\text{-success}}$        | $\mathbb{N}$   | Number of Successfully Swept Objects               |
| $N_{G\text{-success}}$        | $\mathbb{N}$   | Number of Successfully Grasped Objects             |
| $L_{\text{total}}$            | $\mathbb{R}^+$   | Total Trajectory Path Length                       |
| $\chi_\tau$                   | $\{0, 1\}$   | Binary Collision Indicator at Time $\tau$          |
| $t_{\text{comp}}(\tau)$       | $\mathbb{R}^+$   | Computation Time for Action at Time $\tau$         |
| $\tau_{\text{init}}$          | $\mathbb{N}$   | Episode Start Time Step                            |
| $\tau_{\text{final}}$         | $\mathbb{N}$   | Episode End Time Step                              |
| $\tau_{\max}$                 | $\mathbb{N}$   | Maximum Time Steps per Episode                     |
| $\Delta t$                    | $\mathbb{R}^+$   | Discrete Time Interval                             |
| $\alpha, \beta$               | $\mathbb{R}^+$   | Weight Parameters for TCR ( $\alpha + \beta = 1$ ) |
| $T_{\max}$                    | $\mathbb{R}^+$   | Maximum Time Budget for Episode                    |
| $L, \rho, P$                  | Parameters   | Layout Type, Obstacle Density, Target Pattern      |
| $(S, O, T_s, T_g)$            | Configuration  | Scene, Obstacles, Sweepable/Graspable Targets      |
| Evaluation Metrics            |  |  |
| CR                            | $[0, 1]$   | Coverage Ratio                                     |
| TCR                           | $[0, 1]$   | Task Completion Ratio                              |
| TCR <sub>sweep</sub>          | $[0, 1]$   | Sweep Task Completion Ratio                        |
| TCR <sub>grasp</sub>          | $[0, 1]$   | Grasp Task Completion Ratio                        |
| ME                            | $\mathbb{R}^+$   | Motion Efficiency (m/target)                       |
| SR                            | $[0, 1]$   | Sweep Redundancy                                   |
| FT                            | $\mathbb{R}^+$   | Finish Time (seconds)                              |
| CT                            | $\mathbb{R}^+$   | Computation Time (seconds)                         |
| Vel <sub>avg</sub>            | $\mathbb{R}^+$   | Average Velocity (m/s)                             |
| Acc <sub>avg</sub>            | $\mathbb{R}^+$   | Average Acceleration (m/s <sup>2</sup> )           |
| Jerk <sub>avg</sub>           | $\mathbb{R}^+$   | Average Jerk (m/s <sup>3</sup> )                   |
| Collision                     | $\mathbb{N}$   | Total Collision Count                              |

### G.3 Integration with Foundation Models

Future extensions will incorporate large language models for natural language instruction following and vision-language models for open-vocabulary object recognition, enabling more flexible and human-interpretable cleaning behaviors.

### G.4 Limitations

While CleanUpBench provides a unified, extensible benchmark for embodied cleaning agents, several aspects remain simplified relative to real-world deployment settings:

**Environmental Scope:** Current scenes represent idealized indoor environments with limited diversity in textures, lighting conditions, and dynamic elements compared to real households.

**Physical Interaction:** Binary affordance labels and deterministic interaction outcomes omit complexities such as deformable objects, partial cleaning, and tool degradation.

**Sensor Limitations:** Idealized RGB-D sensing without realistic noise, motion blur, or environmental interference that characterizes real-world perception systems.

### G.5 Future Work

**Enhanced Realism:** Integration of soft-body simulation, fluid dynamics, and probabilistic contact models to better represent real-world cleaning scenarios.

**Language Integration:** Natural language instruction following and semantic (Yang, Yuan, and Xie 2022) grounding for human-robot interaction evaluation.

**Long-Horizon Planning:** Support for hierarchical task decomposition (Liao et al. 2025) and multi-session (Ma et al. 2024; Deng et al. 2025b,a) cleaning scenarios spanning extended time horizons (Li et al. 2025b).

## H. Code Availability and Reproducibility

### H.1 Open Source Release

Upon paper acceptance, the complete CleanUpBench codebase will be released under the MIT License, including:

- Isaac Sim scene configurations and object assets
- Baseline method implementations with hyperparameter settings
- Evaluation framework and metric computation scripts
- Data visualization and analysis tools
- Comprehensive documentation and tutorials

#### Repository Structure:

```
CleanUpBench/
|--- environments/ # Scene configurations
|--- assets/       # Models and textures
|--- agents/       # Baseline Deployments
|--- evaluation/   # Metrics and analysis
|--- docs/         # Documentation
+--- scripts/      # Utility scripts
```

### H.2 Reproducibility Checklist

All experimental results are fully reproducible using:

- Fixed random seeds for consistent initialization
- Detailed hyperparameter specifications for all methods
- Standardized evaluation protocols across all experiments
- Version-controlled environment configurations
- Docker containers for dependency management

### H.3 Community Contributions

We welcome community contributions to expand CleanUpBench’s capabilities through additional baseline methods, scene configurations, evaluation metrics, and real-world validation datasets.

## I. Notation and Symbol Definitions

All mathematical notation and symbols used throughout CleanUpBench are systematically defined in Tab. 9 for reference and clarity.

## References

- Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and Van Den Hengel, A. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3674–3683.
- Bai, R.; Yuan, S.; Guo, H.; Yin, P.; Yau, W.-Y.; and Xie, L. 2024. Multi-robot active graph exploration with reduced pose-slam uncertainty via submodular optimization. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 10229–10236. IEEE.
- Batra, D.; Chang, A. X.; Chernova, S.; Davison, A. J.; Deng, J.; Koltun, V.; Levine, S.; Malik, J.; Mordatch, I.; Mottaghi, R.; et al. 2020. Rearrangement: A challenge for embodied ai. *arXiv preprint arXiv:2011.01975*.
- Cai, H.; Yuan, S.; Li, X.; Guo, J.; and Liu, J. 2025. BEV-LIO(LC): BEV Image Assisted LiDAR-Inertial Odometry with Loop Closure. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- Cao, H.; Xu, Y.; Yang, J.; Yin, P.; Ji, X.; Yuan, S.; and Xie, L. 2024. Reliable spatial-temporal voxels for multi-modal test-time adaptation. In *European Conference on Computer Vision*, 232–249. Springer.
- Cao, M.; Cao, K.; Li, X.; Yuan, S.; Lyu, Y.; Nguyen, T.-M.; and Xie, L. 2021. Distributed multi-robot sweep coverage for a region with unknown workload distribution. *Autonomous Intelligent Systems*, 1(1): 13.
- Cao, M.; Lyu, Y.; Yuan, S.; and Xie, L. 2020. Online trajectory correction and tracking for facade inspection using autonomous uav. In *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, 1149–1154. IEEE.
- Chang, M.; Chhablani, G.; Clegg, A.; Cote, M. D.; Desai, R.; Hlavac, M.; Karashchuk, V.; Krantz, J.; Mottaghi, R.; Parashar, P.; Patki, S.; Prasad, I.; Puig, X.; Rai, A.; Ramrakhyta, R.; Tran, D.; Truong, J.; Turner, J. M.; Undersander,

- E.; and Yang, T.-Y. 2025. PARTNR: A Benchmark for Planning and Reasoning in Embodied Multi-agent Tasks. In *International Conference on Learning Representations (ICLR)*. Poster.
- Chaplot, D. S.; Gandhi, D. P.; Gupta, A.; and Salakhutdinov, R. R. 2020. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33: 4247–4258.
- Chen, L.; Liang, C.; Yuan, S.; Cao, M.; and Xie, L. 2025. Relative localizability and localization for multi-robot systems. *IEEE Transactions on Robotics*.
- Cheng, Z.; Tu, Y.; Li, R.; Dai, S.; Hu, J.; Hu, S.; Li, J.; Shi, Y.; Yu, T.; Chen, W.; et al. 2025. Embodiedeval: Evaluate multimodal llms as embodied agents. *arXiv preprint arXiv:2501.11858*.
- Damani, M.; Luo, Z.; Wenzel, E.; and Sartoretti, G. 2021. PRIMAL2: Pathfinding via Reinforcement and Imitation Multi-Agent Learning–Lifelong. *IEEE Robotics and Automation Letters*, 6(2): 2666–2673.
- Deitke, M.; Han, W.; Herrasti, A.; Kembhavi, A.; Kolve, E.; Mottaghi, R.; Salvador, J.; Schwenk, D.; VanderBilt, E.; Wallingford, M.; et al. 2020. Robothor: An open simulation-to-real embodied ai platform. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3164–3174.
- Deng, T.; Shen, G.; Chen, X.; Yuan, S.; Shen, H.; Peng, G.; Wu, Z.; Wang, J.; Xie, L.; Wang, D.; et al. 2025a. MCN-SLAM: Multi-Agent Collaborative Neural SLAM with Hybrid Implicit Neural Scene Representation. *arXiv preprint arXiv:2506.18678*.
- Deng, T.; Shen, G.; Xun, C.; Yuan, S.; Jin, T.; Shen, H.; Wang, Y.; Wang, J.; Wang, H.; Wang, D.; et al. 2025b. Mn eslam: Multi-agent neural slam for mobile robots. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 1485–1494.
- Esfahani, M. A.; Wang, H.; Bashari, B.; Wu, K.; and Yuan, S. 2021. Learning to extract robust handcrafted features with a single observation via evolutionary neurogenesis. *Applied Soft Computing*, 106: 107424.
- Esfahani, M. A.; Wang, H.; Wu, K.; and Yuan, S. 2020. Unsupervised scene categorization, path segmentation and landmark extraction while traveling path. In *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 190–195. IEEE.
- Fan, C.; and Yuan, S. 2025. Structured Task Solving via Modular Embodied Intelligence: A Case Study on Rubik’s Cube. *arXiv preprint arXiv:2507.05607*.
- Hörner, J. 2016. *Map-merging for multi-robot system*. Master’s thesis, Charles University in Prague, Faculty of Mathematics and Physics, Prague.
- Hu, T.; Xu, X.; Nguyen, T.-M.; Liu, F.; Yuan, S.; and Xie, L. 2025a. Tire wear aware trajectory tracking control for Multi-axle Swerve-drive Autonomous Mobile Robots. *Journal of Automation and Intelligence*.
- Hu, T.; Yuan, S.; Bai, R.; Xu, X.; Liao, Y.; Liu, F.; and Xie, L. 2025b. Swept Volume-Aware Trajectory Planning and MPC Tracking for Multi-Axle Swerve-Drive AMRs. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.
- Jaafar, A.; Raman, S. S.; Wei, Y.; Harithas, S.; Juliani, S.; Wernerfelt, A.; Quartey, B.; Idrees, I.; Liu, J. X.; and Tellex, S. 2024. LAMBDA: A Benchmark for Data-Efficiency in Long-Horizon Indoor Mobile Manipulation Robotics. *arXiv preprint arXiv:2403.13794*.
- James, S.; Ma, Z.; Rovick Arrojo, D.; and Davison, A. J. 2020. RL Bench: The Robot Learning Benchmark & Learning Environment. *IEEE Robotics and Automation Letters*.
- Jauhri, S.; Peters, J.; and Chalvatzaki, G. 2022. Robot Learning of Mobile Manipulation With Reachability Behavior Priors. *IEEE Robotics and Automation Letters*, 7(3): 8399–8406.
- Jiang, Y.; Zhang, R.; Wong, J.; Wang, C.; Ze, Y.; Yin, H.; Gokmen, C.; Song, S.; Wu, J.; and Fei-Fei, L. 2025. BEHAVIOR Robot Suite: Streamlining Real-World Whole-Body Manipulation for Everyday Household Activities. *arXiv preprint arXiv:2503.05652*.
- Kolve, E.; Mottaghi, R.; Han, W.; VanderBilt, E.; Weihs, L.; Herrasti, A.; Deitke, M.; Ehsani, K.; Gordon, D.; Zhu, Y.; et al. 2017. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv preprint arXiv:1712.05474*.
- Lai, Y.; Yuan, S.; Nassar, Y.; Fan, M.; Gopal, A.; Yorita, A.; Kubota, N.; and Rätsch, M. 2025a. Natural multi-modal fusion-based human–robot interaction: Application with voice and deictic posture via large language model. *IEEE Robotics & Automation Magazine*.
- Lai, Y.; Yuan, S.; Nassar, Y.; Fan, M.; Weber, T.; and Rätsch, M. 2025b. NVP-HRI: zero shot natural voice and posture-based human–robot interaction via large language model. *Expert systems with applications*, 268: 126360.
- Lai, Y.; Yuan, S.; Zhang, B.; Kiefer, B.; Li, P.; and Zell, A. 2025c. FAM-HRI: Foundation-Model Assisted Multi-Modal Human-Robot Interaction Combining Gaze and Speech. *arXiv preprint arXiv:2503.16492*.
- Li, D.; Cai, T.; Tang, T.; Chai, W.; Driggs-Campbell, K. R.; and Wang, G. 2025a. EMMOE: A Comprehensive Benchmark for Embodied Mobile Manipulation in Open Environments. *arXiv preprint arXiv:2503.08604*.
- Li, X.; Yuan, S.; Cai, H.; Lu, S.; Wang, W.; and Liu, J. 2025b. LL-Localizer: A Life-Long Localization System based on Dynamic i-Octree. *IEEE Transactions on Instrumentation & Measurement*.
- Liang, J.; Wang, Z.; Cao, Y.; Chiun, J.; Zhang, M.; and Sartoretti, G. A. 2023. Context-Aware Deep Reinforcement Learning for Autonomous Robotic Navigation in Unknown Area. In *Proceedings of The 7th Conference on Robot Learning*, 1425–1436. PMLR.
- Liao, Y.; Xu, X.; Bai, R.; Yang, Y.; Cao, M.; Yuan, S.; and Xie, L. 2025. Following Is All You Need: Robot Crowd Navigation Using People As Planners. *IEEE Robotics and Automation Letters*.
- Liu, R.; Xu, X.; Yuan, S.; and Xie, L. 2025. Handle object navigation as weighted traveling repairman problem. *arXiv preprint arXiv:2503.06937*.

- Ma, Y.; Xu, J.; Yuan, S.; Zhi, T.; Yu, W.; Zhou, J.; and Xie, L. 2024. Mm-lins: a multi-map lidar-inertial system for over-degenerate environments. *IEEE Transactions on Intelligent Vehicles*.
- Mees, O.; Hermann, L.; Rosete-Beas, E.; and Burgard, W. 2022. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3): 7327–7334.
- Nayak, S.; Orozco, A. M.; Ten Have, M.; Thirumalai, V.; Zhang, J.; Chen, D.; Kapoor, A.; Robinson, E.; Gopalakrishnan, K.; Harrison, J.; et al. 2024. MAP-THOR: Benchmarking Long-Horizon Multi-Agent Planning Frameworks in Partially Observable Environments. In *Multi-modal Foundation Model meets Embodied AI Workshop@ ICML2024*.
- Padmakumar, A.; Thomason, J.; Shrivastava, A.; Lange, P.; Narayan-Chen, A.; Gella, S.; Piramuthu, R.; Tur, G.; and Hakkani-Tur, D. 2022. Teach: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2, 2017–2025.
- Qi, Z.; Yuan, S.; Liu, F.; Cao, H.; Deng, T.; Yang, J.; and Xie, L. 2024. Air-embodied: An efficient active 3dgs-based interaction and reconstruction framework with embodied large language model. *arXiv preprint arXiv:2409.16019*.
- Qu, W.; Du, J.; Yuan, S.; Wang, J.; Sun, Y.; Liu, S.; Zhu, Y.; Yu, J.; Cao, S.; Xia, R.; Tang, X.; Wu, X.; and Luo, D. 2025. DPGP: A Hybrid 2D-3D Dual Path Potential Ghost Probe Zone Prediction Framework for Safe Autonomous Driving. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- Savva, M.; Chang, A. X.; Dosovitskiy, A.; Funkhouser, T.; and Koltun, V. 2019. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 9339–9347.
- Shridhar, M.; Thomason, J.; Gordon, D.; Bisk, Y.; Han, W.; Mottaghi, R.; Zettlemoyer, L.; and Fox, D. 2020. ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10740–10749.
- Tan, C. S.; Mohd-Mokhtar, R.; and Arshad, M. R. 2021. A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms. *IEEE Access*, 9: 119310–119342.
- Tan, D. M. S.; Ma, Y.; Liang, J.; Chng, Y. C.; Cao, Y.; and Sartoretti, G. 2024. IR2: Implicit Rendezvous for Robotic Exploration Teams under Sparse Intermittent Connectivity. In *2024 IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE.
- Wu, C.; Wang, R.; Song, M.; Gao, F.; Mei, J.; and Zhou, B. 2024. Real-time Whole-body Motion Planning for Mobile Manipulators Using Environment-adaptive Search and Spatial-temporal Optimization. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 1369–1375. IEEE.
- Xia, F.; Shen, W. B.; Li, C.; Kasimbeg, P.; Tchapmi, M. E.; Toshev, A.; Martín-Martín, R.; and Savarese, S. 2020. Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments. *IEEE Robotics and Automation Letters*, 5(2): 713–720.
- Yang, R.; Chen, H.; Zhang, J.; Zhao, M.; Qian, C.; Wang, K.; Wang, Q.; Koripella, T. V.; Movahedi, M.; Li, M.; et al. 2025. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *arXiv preprint arXiv:2502.09560*.
- Yang, Y.; Yuan, S.; and Xie, L. 2022. Overcoming catastrophic forgetting for semantic segmentation via incremental learning. In *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 299–304. IEEE.
- Yenamandra, S.; Ramachandran, A.; Yadav, K.; Wang, A.; Khanna, M.; Gervet, T.; Yang, T.-Y.; Jain, V.; Clegg, A. W.; Turner, J.; et al. 2023. HomeRobot: Open-Vocabulary Mobile Manipulation. *arXiv preprint arXiv:2306.11565*.
- Yuan, S.; and Wang, H. 2014. Autonomous object level segmentation. In *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, 33–37. IEEE.
- Yuan, S.; Wang, H.; and Xie, L. 2021. Survey on localization systems and algorithms for unmanned systems. *Unmanned Systems*, 9(02): 129–163.