# ManiTaskGen: A Comprehensive Task and Benchmark Generator for Vision-Language Models in Long-Horizon Embodied Planning



#### Anonymous ACL submission

Figure 1: **Framework Overview.** ManiTaskGen is a universal tool that, given information from any interactive scene, can generate a comprehensive set of plausible long-horizon manipulation tasks and automatically construct benchmarks to evaluate perception, reasoning and planning capabilities of Vision-Language Models.

#### Abstract

Long-horizon manipulation task planning (e.g., object rearrangement) using vision-language models (VLMs) is a critical research direction in embodied AI. Although numerous recent works have proposed specific algorithms and models, their evaluations typically rely on manually selected scenes and a limited set of annotated tasks. We contend that such evaluation methods are neither comprehensive nor fair, and require significant manual annotation. In this paper, we introduce an automated method for task generation and benchmark construction: given any interactive scene, our approach can generate a comprehensive set of plausible long-horizon manipulation tasks and automatically build a benchmark for evaluating visionlanguage planning models. Moreover, by applying our method to off-the-shelf interactive scenes in simulators, we provide a thorough evaluation and analysis of the performance of existing VLMs on these long-horizon planning tasks. We will open-source our code, offering a universal tool for generating tasks and benchmarks to evaluate VLMs for long-horizon embodied planning.

#### 1 Introduction

Long-horizon manipulation task planning—such as complex object rearrangement—is a fundamental challenge in embodied AI (Durante et al., 2024; Szot et al., 2022; Weihs et al., 2021; Gan et al., 2021). To effectively execute such tasks, embodied agents must understand the underlying logic, decompose tasks into sequential subtasks like navigation, object grasping, and placement (Szot et al., 2021; Anderson et al., 2018). Additionally, they need to perceive the scene layout to navigate, and accurately interpret object configurations and spatial relationships to ensure precise execution of pick-and-place operations (Gu et al., 2022; Liu et al., 2024), demanding comprehensive perceptual and reasoning abilities. 026

027

028

031

032

034

037

038

041

043

044

045

046

In recent years, various approaches have been proposed to tackle these challenges, including skillchaining methods (Gu et al., 2022; Lee et al., 2021) and large model-based planning (Liang et al., 2023; Song et al., 2023). The use of vision-language models (VLMs) for task planning (Yang et al., 2024; Zhang et al., 2023; Skreta et al., 2024) has emerged as a particularly promising direction, especially for open-world applications. Despite significant methodological advancements, the evaluation of these approaches still suffers from a lack of comprehensive benchmarks. Most existing evaluations (Liu et al., 2024; Szot et al., 2021; Gu et al., 2022) rely on manually selected scenes and a limited set of predefined tasks, which introduce two major limitations: (1) Limited Task Diversity: Since tasks are manually designed, their quantity and diversity are inherently limited, resulting in an incomplete assessment of model capabilities across diverse conditions. (2) Limited Scene **Coverage**: Scene selection and task annotation are time-consuming and labor-intensive. Consequently, studies often focus on specific scenes of interest, leading to inconsistent evaluation protocols and hindering fair comparisons across different works.

051

061

062

064

093

097

101

To address the first limitation, we begin by posing the question: Given a mobile robot equipped with robust low-level pick-and-place skills, how many possible tasks could it theoretically execute in an interactive scene? Intuitively, the answer seems infinite. However, as illustrated in Fig. 2, we characterize the robot's execution loop within the scene and consequently posit that all possible tasks can be logically categorized into two types: processbased tasks and outcome-based tasks. Building on this insight, we introduce an automated task generation method that ensures logical comprehensiveness. Specifically, we construct a scene graph that encodes object spatial relationships and free-space information, allowing us to systematically generate all feasible process-based tasks. Additionally, we curate a template dataset, ManiTaskOT-1K, and employ a VLM-based voting mechanism to generate outcome-based tasks tailored to any given scene.

Building on this task generation method, we further develop an automated benchmarking framework. For any interactive scene, this framework enables an agent to execute abstracted interactions—such as navigation and pick-and-place actions—to complete the tasks, followed by an automated evaluation of task performance. This sceneagnostic benchmark generation method can be applied to a wide range of interactive scenes, including both simulated environments and real-world scenes (provided the objects are interactive and their information—such as pose, bounding box, or interior layers—is available), thereby addressing the second limitation mentioned above.

To validate our approach, we apply it to ReplicaCAD (Szot et al., 2021), a dataset of interactive simulated environments, to generate a task set and automatically construct benchmarks. Our experimental results demonstrate the validity and diversity of the generated tasks. Additionally, by testing existing VLMs on our benchmark, we identify their limitations in handling long-horizon manipulation tasks, highlighting key failure cases and providing insights for future model improvements. 102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

In summary, this paper includes following contributions: (1) We propose ManiTaskGen, an automated framework for generating logically comprehensive long-horizon manipulation tasks and constructing benchmarks. (2) We apply ManiTaskGen to ReplicaCAD, generating a large-scale benchmark and conducting an in-depth evaluation that uncovers the limitations of current VLMs in longhorizon task planning. (3) We provide the community with a universal tool for task and benchmark generation, enabling standardized evaluation of vision-language planning models across diverse interactive environments.

# 2 Related Work

**Benchmarks for Embodied Long-horizon Plan**ning. In recent years, numerous benchmarks have emerged to evaluate embodied agents in longhorizon task planning (Li et al., 2023; Szot et al., 2021; Shridhar et al., 2020a,b), including benchmarks specific for VLM-based agents (Liu et al., 2024; Yang et al., 2025). However, these benchmarks typically rely on manually designed and annotated tasks and are limited to a small number of predefined scenes. In contrast, our method can be applied to any given interactive scene, generating plausible and comprehensive tasks that cover a wide range of scenarios. Furthermore, it automatically constructs benchmarks for evaluation, significantly expanding the diversity of test scenarios and tasks.

**Task Generation for Embodied Agents.** Recent research has explored task and environment generation for training Reinforcement Learning (RL) agents(Fang et al., 2020; Cobbe et al., 2020; Raileanu and Rocktäschel, 2020), as well as augmenting diverse tabletop manipulation tasks(Wang et al., 2023) and instruction-finetuning LLM-based agents (Hu et al., 2024). In this work, we focus on scene-level task generation for long-horizon manipulation tasks. Additionally, we combined task generation with autumated benchmark construction to evaluate VLM-based embodied agents.



Figure 2: **Execution Loop.** This loop provides a comprehensive depiction of how an embodied agent executes a manipulation task within a scene. Based on this framework, we categorize all tasks into two distinct types: process-based tasks and outcome-based tasks.

# **3** Comprehensive Task Generation

#### 3.1 Premise

152

153

154

155

156

158

159

161

162

163

166

167

168

169

170

171

172

173

174

175

176

178

179

180

182

185

187

191

Considering a mobile robot with comprehensive pick-and-place capabilities—able to grasp any movable object and place it on any surface that can accommodate it—we seek to answer the following fundamental question: *How many long-horizon manipulation tasks can the robot execute within a given scene?* 

To systematically address the question, we formalize the problem as follows. Let  $\mathcal{O}$  =  $\{o_1, o_2, o_3, \dots\}$  denote the set of all objects in the scene, where each  $o_i$  may include the object's raw information such as mesh model or extracted information like its bounding box and interior layers. We define the scene status as  $S = \{s_0, s_1, s_2, \dots\},\$ where  $s_i$  represents the state of object  $o_i$ , including its position  $p_{o_i}$  and rotation  $r_{o_i}$ . Next, we define a set of atomic actions as:  $\mathcal{A} = \{a_0, a_1, a_2, \dots\}$ . An atomic action  $a \in \mathcal{A}$  corresponds to transitioning an object from its current state  $s_i$  to a new state  $s'_i$ . Given the agent's capabilities are constrained to pick-and-place actions, changes in S are restricted to object positions. Thus, the set A consists of actions of the form: "move object o from p to p'" where p' is a valid placement position for o.

This formalization underpins the execution loop depicted in Fig. 2. At each step, given the current scene state S, the robot selects an action afrom A and executes it. The execution modifies S, updating A accordingly, and the process iterates. This loop encapsulates how the robot iteratively executes pick-and-place tasks in a given scene. Following this execution model, we posit that all pickand-place task instructions into two fundamental types: **process-based tasks** and **outcome-based tasks**.

Process-based tasks explicitly specify one or a sequence of atomic actions drawn from A. For instance:

"I need the robot to pick up object $o_i$ from	
location $p_{\alpha_i}$ and place it at $p'_{\alpha_i}$ .	

192

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

210

211

212

213

Note that, in natural language, the same command can be expressed in various ways while keeping the same meaning. In addition, atomic actions sampled from sequential execution loops can be chained.

In contrast, Outcome-based tasks define a desired final scene status S without specifying the intermediate actions. For example:

"I need the robot to clear the tabletop."

Here the tabletop may initially contain multiple objects  $\{o_1, o_2, o_3, ...\} \subseteq O$ .

Building on this conceptual categorization, we propose a systematic task generation methodology leveraging the iterative execution loop. Specifically, we first construct a 3D scene graph encoding object spatial relationships and placement constraints (Sec. 3.2). This enables us to derive a complete set of atomic actions  $\mathcal{A}$  and establish a comprehensive execution loop. In Sec. 3.3, we detail how we generate both process-based and outcome-based tasks by utilizing the execution loop model.

Alg	orithm 1 Construct 3D Scene Graph Tree
1:	<b>Input:</b> object set $\mathcal{O}$ , scene state $\mathcal{S}$ .
2:	<b>Output:</b> 3D Scene graph $\mathbb{S} = (\mathcal{V}, \mathcal{E})$
3:	Definition:
4:	$\mathcal{V} = \{v_i\}$ is the vertex set of scene graph tree;
5:	$\mathcal{E} = \{e_i\}$ is the edge set of scene graph tree;
6:	<b>function</b> INIT_SCENE_GRAPH_TREE( $\mathcal{O}, \mathcal{S}$ )
7:	for $o_i$ in $\mathcal{O}$ do
8:	$v_i \leftarrow \texttt{init\_vertex}(o_i, s_i)$
9:	$v_i$ .surface $\leftarrow o_i$ .get_surface()
10:	end for
11:	$\mathcal{E} \leftarrow cal\_contact(\mathcal{V})$
12:	$\mathbb{S}_{init} \leftarrow \texttt{build\_tree}(\mathcal{V}, \mathcal{E})$
13:	return $\mathbb{S}_{init}$
14:	end function
15:	<b>function</b> CAL_FREE_SPACE(v)
16:	for each $p$ in v.surface do
17:	$V_p \gets \texttt{p.get_objects()}$
18:	for each $v_c$ on $p$ do
19:	$v_c$ .cal_freespace( $p,V_p-v_c$ )
20:	end for
21:	end for
22:	end function
23:	$\mathbb{S} \leftarrow \texttt{INIT\_SCENE\_GRAPH\_TREE}(\mathcal{O}, \mathcal{S})$
24:	for $v$ in $\mathcal{V}$ do
25:	$CAL_FREE_SPACE(v)$
26:	end for



Figure 3: **Visualizations of receptacle segmentation.** By calculating the relative poses of objects, we partition receptacles into grids based on objects or empty surfaces and compute their sizes, thereby enabling efficient retrieval of all receptacles within the scene.

#### 3.2 Receptacle-Aware 3D Scene Graph

214

215

216

217

218

219

221

226

235

236

240

241

243

245

246

247

249

Given a set of objects  $\mathcal{O}$  and the scene status  $\mathcal{S}$ , our goal is to automatically construct a 3D scene graph  $\mathbb{S}$ . This graph encodes object locations and indexes all available receptacles (free spaces) along with their sizes. The algorithm is outlined in Alg. 1.

We first build an object spatial distribution tree based on each object's position, rotation, bounding box, and interior layer information (in any). The root node represents *Ground*, while all other nodes correspond to scene objects. Parent-child relationships are determined by spatial containment—an object is assigned as a child of another if it rests on one of its outer or interior surfaces. For solid objects, the default supporting surface is their top surface, while for multi-level receptacles (e.g., cabinets, shelves), we record the specific surface where each child object is placed.

Once the tree structure is established, we compute relative positions and distances between objects sharing the same supporting surface. We segment the space around each object's bounding box into eight directional free space regions, with boundaries extending until they encounter another object. For unoccupied surfaces, we default to a 3×3 grid segmentation. We visualize our segmentation of free spaces in Fig. 3. This representation enables efficient retrieval of any receptacle in the scene by querying an object or surface. Notably, a single receptacle may be indexed by multiple objects, such as an intermediate plane between adjacent objects. Additionally, adjacent receptacles can be merged into a larger receptacle if necessary. In the end, we obtain a 3D scene graph S that encodes the location, structure, and relationships of every object and every receptacle within the scene.

#### 3.3 Generating Tasks

Based on the scene graph  $\mathbb{S}$ , next we derive a complete set of atomic actions  $\mathcal{A} = \{a_0, a_1, a_2, ...\}$ . Each atomic action *a* takes the form of *"move object o from*  $p_o$  to  $p'_o$ ", where  $p'_o$  must be a valid placement location for *o*. We extract all feasible  $p'_o$  values from  $\mathbb{S}$  for each object. Using  $\mathcal{A}$  and  $\mathbb{S}$ , we then generate both process-based and outcomebased tasks.

251

252

253

254

255

257

258

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

285

286

288

289

290

291

292

293

295

296

297

298

300

Process-based Tasks. As defined in Sec. 3.1, process-based tasks describe the execution of one or more actions from the atomic action set A within a single or multiple execution loops. A task can be as simple as executing a single action a sampled from  $\mathcal{A}$ . Alternatively, if executing *a* updates the scene state S—resulting in a new atomic action set  $\mathcal{A}'$ —and another action a' is subsequently selected from  $\mathcal{A}'$ , then the sequence (a + a') also forms a longer process-based task. Here, "+" may represent various logical connectors, such as THEN, and OR, corresponding to "execute a followed by a'." "execute a or a'." respectively. By iteratively sampling atomic actions across one or multiple execution loops, we generate a comprehensive set of process-based tasks, with task complexity controlled by limiting the number of loops.

To enhance task diversity, we describe the destination p' in each sampled atomic action "move object o from p to p'" using four distinct strategies based on scene graph information: (1) Move to a surface by specifying its name; (2) Move to a location around a specific object on a surface; (3) Move to a position in a relative direction of a specific object on a surface; and (4) Move to a location between two objects on a surface. Additionally, we offer an option to leverage a large language model (LLM) to rephrase these descriptions while preserving their original semantics, further increasing linguistic variation.

**Outcome-based Tasks**. Unlike process-based tasks, generating outcome-based tasks is more challenging as it requires an abstract description of the scene state S (Sec. 3.1). A straightforward approach would be to input the scene graph S into a vision-language model (VLM) or large language model (LLM) to predict possible scene status changes. However, due to the limitations and instability of large models in understanding complex 3D scene graphs, this method often generates impractical tasks, leading to a long-tail distribution that reduces overall utility. We further discuss this

issue in Sec. 5.1.

301

314

315

316

317

319

320

321

322

325

326

327

330

331

336

337

341

342

344

345

347

348

To address this, we introduce ManiTaskOT-1K, a manually curated outcome-based task template 303 dataset. We first collected scene images from both real-world environments and simulators, then leveraged Amazon Mechanical Turk (Amazon, 2024) to gather human-written instructions describing 307 high-level scene transformations. From these, we extracted 1,000 structured templates, forming the ManiTaskOT-1K dataset. Details are provided in 310 the Appendix, and examples of representative tem-311 plates include: 312

> "Create a tidy arrangement on [OB-JECT0]."

"Disorganize the top of [OBJECT0] to make it messy."

"Sort all [SUB-OBJECTS00] on [OB-JECT0] by material type."

Given a scene, we instantiate ManiTaskOT-1K templates by populating them with scene-specific objects. To ensure task feasibility, we further employ multiple VLMs to vote on each generated instruction, filtering out non-executable tasks and refining the final set of outcome-based tasks.

#### 4 Automated Benchmark Construction

In Sec. 3, we introduced a method for generating a comprehensive set of tasks based on scene information. However, task generation alone is insufficient—we also need to test these tasks. Given the quantity and diversity of the generated tasks, manually constructing testing environments becomes impractical. To address this, we propose an automated benchmark construction framework: Given any scene, we first generate all tasks and classify them by difficulty (Sec.4.1). Next, leveraging the scene graph, we partition the scene space to compute reachable positions and enable abstracted manipulation operations for each object, forming an automated test flow to evaluate agent performance (Sec.4.2).

Due to the lack of robust low-level manipulation algorithms capable of reliably executing primitive skills (e.g., picking and placing objects) (Suomalainen et al., 2022), our benchmark primarily focuses on high-level vision-language planning, with low-level manipulations abstracted and assumed to be executed in a 'magic' manner when running testflows. If more advanced low-level manipulation skills become available in the future, our method can seamlessly integrate with these improvements.

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

371

372

373

374

375

376

377

378

379

381

383

384

385

386

387

390

391

392

393

394

395

#### 4.1 Task Difficulty Levels

We classify the generated process-based and outcome-based tasks into four levels, based on their perception and planning difficulty:

- Level 1: Single-step pick-and-place tasks. These are process-based tasks sampled from a single execution loop, such as "move object o from p to p'". Additionally, the object involved is unique on its supporting surface.
- Level 2: Similar to Level 1, but the target object is not uniquely identifiable without additional context. For instance, a Level 2 task may require to move a teacup from a table containing multiple teacups. In this case, the agent must disambiguate the correct object using additional task-provided descriptions or images.
- Level 3: Multi-step pick-and-place tasks, generated by sampling from multiple execution loops and concatenating atomic actions using logical connector "THEN".
- Level 4: All outcome-based tasks, representing the highest difficulty level due to their complex and abstracted task descriptions.

#### 4.2 Benchmark Test Flow

We present a visualization of our benchmark test flow in Fig. 4, which illustrates a sequence of interactions within a simulated environment.

Action Space. Building on the scene graph constructed in Sec. 3.2, we first automatically compute and label free spaces surrounding each ground object as walkable areas for the agent. These areas are incorporated into the agent's action space as go\_to\_location at every time step. Since a ground object may have multiple walkable areas around it, we provide an additional action change\_view to allow the agent to switch between different walkable areas when needed. When the agent reaches a ground object, it receives rendered observation images with tagged objects, allowing it to select a pick action and choose the tag for any movable object. Next, the agent can place the object in hand into a free space on a platform that can accommodate it. We provide two placement options: (i) place\_r: placing the object in a randomly



Figure 4: Visualization of the automated test flow for a Level 3 task example. The agent is equipped with abstracted navigation (go\_to, change\_view), grasping (pick) and placing (place\_s, place\_r) skills. Blue marks indicate the walkable locations involved in the task.

selected suitable location on the platform, or (ii) place\_s: placing the object by selecting a tagged free space attached to a specified object chosen by the agent. When the agent invokes call\_end action, the benchmark automatically verifies whether the task was successfully completed. Additionally, we set a time step limit for each test episode to prevent indefinite execution.

Evaluation. For Level 1, 2, and 3 tasks (all processbased tasks), the expected scene graph is precisely defined, enabling direct success verification by comparing the initial and final scene graphs. Additionally, our benchmark provides intermediate 409 performance metrics for each task by tracking the number of completed subtasks-such as navigating to the correct object or picking up the correct object-during a testing episode. For Level 4 tasks, considering that these tasks involve abstract descriptions of scene changes (e.g., making a desk "tidy"), it is challenging to define an unbiased and 415 precise success-state scene graph. Possible evaluation methods include human verification or leveraging VLMs to vote on whether the final scene status satisfies the task requirements.

#### 5 **Experiments**

397

400

401 402

403

404

405

406 407

408

410

411

412

413

414

416

417

418

419

420

421

422

We apply ManiTaskGen to the ReplicaCAD dataset (Szot et al., 2021), a collection of interactive simulated environments. The simulator we use is SAPIEN (Xiang et al., 2020). The ReplicaCAD dataset contains 111 distinct scenes, including 6 FRL\_apartment scenes and 105 variation scenes with different object layouts derived from these apartments. All objects in these scenes are fully interactive and come with mesh models. We use the 6 FRL\_apartment scenes to generate tasks and automatically construct benchmarks, referred to as ManiTaskGen-FRL. In total, we generate 199,109 task instructions, including 45,041 Level 1 tasks, 56,680 Level 2 tasks, 60,000 Level 3 tasks (we constrain the execution loop to be executed for two rounds, and progessively generated 10,000 tasks for each scene), and 37,388 Level 4 tasks. For Level 4 tasks, we employ three VLMs (GPT-40 (Achiam et al., 2024), Gemini-1.5-pro (Team et al., 2024), Claude-3.5-sonnet (Anthropic, 2024)) to vote on whether a task-generated using ManiTaskOT-1K templates-is executable.

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

We compare ManiTaskGen-FRL with existing long-horizon manipulation benchmarks in Tab. 1. Notably, despite using only 6 scenes, ManiTaskGen-FRL contains significantly more tasks than other benchmarks. Furthermore, our method is scene-agnostic, meaning it can be applied to any given scene, allowing for the incorporation of additional scene data sources to further



GPT-TaskGen

ManiTaskGen (ours)

Figure 5: **The ''lightmap''s which show the diversity of generated tasks.** Each time an object or location is mentioned in a task, we add a highlight at the corresponding position. By analyzing the brightness intensity and distribution of highlights, it becomes evident that our method generates tasks that cover more objects and locations.

Benchmark	Scene Number	Instruction Number
GenSim (Wang et al., 2023)	-	100
$\lambda$ (Jaafar et al., 2025)	20	521
M3Bench (Zhang et al., 2024)	119	31,050
Language Rearrangement (Szot et al., 2023)	1	1,000
Embodied Agent Interface (Li et al., 2025)	2	438
EmbodiedBench (Yang et al., 2025)	4	1128
ManiTaskGen-FRL (Ours)	6	199,109

Table 1: Comparison between ManiTaskGen-FRL and other existing benchmarks for long-horizon manipulation tasks.

expand the task set in practical applications.

In the following sections, we present experimental results and analyses from two key aspects: In Sec. 5.1, we assess the validity and diversity of the generated tasks; In Sec. 5.2, we present a direct downstream application of our generated benchmark by evaluating the performance of state-of-theart vision-language models on ManiTaskGen-FRL, and analyzing the underlying failures causes.

Task Validity Rate	Level 1	Level 2	Level 3	Level 4
GPT-TaskGen	40.1%	39.4 %	18.7%	44.3%
ManiTaskGen (Ours)	93.8%	98.4 %	92.3%	74.1%

Table 2: Human-Verified Task Validity Rate for ourmethod and baseline method.

#### 5.1 Task Validity and Diversity

For a fair comparison, we implement a GPT-based task generation approach as a baseline, referred to as GPT-TaskGen. Specifically, we provide each scene's scene graph along with images covering all objects to GPT-40 (Achiam et al., 2024), instructing it to generate the tasks. We apply this baseline method to the same 6 *FRL\_apartment*  scenes to generate 1,000 process-based tasks (corresponding to Level 1, Level 2, and Level 3 tasks in ManiTaskGen) and 1,000 outcome-based tasks. We refer to these generated tasks as GPT-TaskGen-FRL. These task sets serve as a direct comparison to evaluate the validity and diversity of the tasks produced by our method. 468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

502

Validity Assessment. We first evaluate the validity of the generated tasks by conducting human verification on samples from ManiTaskGen-FRL and GPT-TaskGen-FRL, with the results reported in Tab. 2. For Level 1, 2, and 3 tasks, although our generation process ensures that the target location has sufficient space to accommodate the moved object, some tasks may still be infeasible due to occlusions, obstacles, or restricted visibility, making the target position difficult to reach or observe. For Level 4 tasks (outcome-based tasks), the validity rate is expectedly lower compared to process-based tasks, as it relies on a VLM-based voting mechanism.

Nevertheless, our results show that most tasks are valid, with significantly higher validity rates compared to the baseline method.

**Diversity Assessment.** After evaluating the validity of the generated task set, we proceed to assess its diversity, which reflects how well the tasks cover various scenarios to enable comprehensive testing of vision-language planning models.

To compare task diversity, we sample 100 human-verified tasks from one same  $FRL_apartment$  scene from ManiTaskGen-FRL and GPT-TaskGen-FRL. Fig. 5 presents two "lightmap"s that visualizes the distribution of involved objects and locations. Specifically, we

459

451

462

463

464

465

466

	Level 1		Level 2		Level 3		Avg	
	IP	SR (%)	IP	SR (%)	IP	SR (%)	IP	SR (%)
Human	91	65	90	61	90	60	90	62
Random	0.4	0	0.4	0	0.4	0	0.4	0
GPT-4o(Achiam et al., 2024)	42	13	16	4	23	2	27	6
GPT-4o-mini(Achiam et al., 2024)	16	2	7	2	12	0	12	1
Gemini-2.0-flash (Team et al., 2024)	42	12	19	4	30	1	30	6
Gemini-1.5-flash(Team et al., 2024)	47	23	18	9	29	3	31	12
Gemini-1.5-pro (Team et al., 2024)	41	16	21	8	32	2	31	9
Claude-3.5-sonnet (Anthropic, 2024)	52	16	23	8	39	3	38	9
Claude-3.5-haiku (Anthropic, 2024)	36	4	16	2	26	0	26	2
Owen2-VL-72B-Ins (Wang et al., 2024)			8		$ \overline{3} -$		6	
Llama-3.2-11B-Vision-Ins (Meta, 2024)	1	0	2	2	2	0	2	1
Llama-3.2-90B-Vision-Ins (Meta, 2024)		6	9	6	20	1	18	4

Table 3: Evaluation results on 10 VLMs, covering both proprietary (upper part) and open-source models (lower part). Here, IP refers to Intermediate Points, and SR refers to success rate.



Figure 6: Success and failure cases in Level 1 evaluation experiments from Claude-3.5-sonnet . From left to right, we show a breakdown of the long-tail failure modes of each of the components.

light up the centroid of an object or location whenever it appears in a task, with light intensity accumulating when the same area is repeatedly referenced. The figure clearly demonstrates that our method generates a wider range of tasks, covering more objects and locations compared to the baseline.

#### 5.2 Benchmarking Existing VLMs

503

504

505

508

509

510

As a direct downstream application of our gener-511 ated tasks and benchmarks, we evaluate existing 512 VLMs on ManiTaskGen-FRL and analyze their per-513 formance. Specifically, we randomly sample 100 514 tasks (from Level 1, 2, and 3, respectively) for each of the six FRL\_apartment scenes. For each test 516 episode, we set a time step limit of 15 steps. We use 517 two metrics for evaluation: (1) Success Rate (SR): 518 An episode is considered successful only if the 519 520 agent executes the call\_end action and the final scene graph matches the expected scene graph; (2) 521 Intermediate Points (IP): For Level 1 and Level 2 522 tasks, a successful episode should include the fol-523 lowing four substeps, each contributing 25 points: 524

Navigate to the correct starting location. Pick up the correct object. Navigate to the correct destination location. Place the object successfully. For Level 3 tasks, we compute the average points from the two sequential one-step tasks. 525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

Tab. 3 presents the benchmarking results for the evaluated VLMs. We also provide human performance and random-choice performance for reference. Our evaluation indicates that most models achieve an average success rate under 10%, which is significantly below human-level performance. And even the best-performing model, Claude-3.5sonnet, attains only 52 intermediate points, highlighting the substantial challenge posed by the generated tasks. To further analyze the failure cases, we present a Sankey Diagram of mistakes in Fig. 6, illustrating the distribution of Intermediate Points (IP). As wrong item picking and wrong placement account for the largest proportion of mistakes, this suggests that VLMs' perception and spatial understanding capabilities may serve as the bottleneck for long-horizon manipulation task planning.

#### 6 Conclusion

In this paper, we introduce ManiTaskGen, an automated method for task generation and benchmark construction for any interactive scene. Mani-TaskGen can generate a comprehensive set of longhorizon manipulation tasks, covering both processbased and outcome-based tasks, thereby providing a diverse set of testing scenarios for visionlanguage planning agents. Our experiments demonstrate the validity and diversity of the generated tasks, while also showcasing its practical usability by benchmarking existing VLMs and revealing their limitations.

# Limitations

560

583

584

585

587

588

592

594

595

596

597

598

599

605

607

610

611

Our method effectively generates long-horizon ma-561 nipulation tasks and supports automated bench-562 marking, but it has some limitations: (1) Lower Validity for Level 4 Tasks: While process-based tasks achieve over 90% validity, Level 4 outcomebased tasks have only around 70%. This is because some templates in the ManiTaskOT-1K dataset are scene-dependent, and determining suitability re-568 quires strong perception and reasoning. Automated success evaluation for these tasks also remains challenging. (2) Less-than-Perfect Validity for Level 571 1-3 Tasks: For Level 1-3 tasks, although our algorithm ensures adequate space for object place-573 ment, it doesn't guarantee visibility or accessibility. We plan to introduce a learning-based method to 575 improve target feasibility assessment. (3) Simpli-576 fied Low-Level Manipulation: Our benchmark 577 abstracts low-level manipulation, focusing on highlevel perception, reasoning, and planning. We aim to extend it to include low-level tasks to evaluate 580 more models like Vision-Langugae-Action models (VLAs).

#### References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, and Shyamal Anadkat et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.
- Amazon. 2024. Amazon mechanical turk. https:// www.mturk.com/.
- Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. 2018. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*.
- Anthropic. 2024. About claude models. https://docs.anthropic.com/en/docs/ about-claude/models. Accessed: 2024-09-03.
- Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. 2020. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pages 2048–2056. PMLR.
- Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839.

Zane Durante, Qiuyuan Huang, Naoki Wake, Ran Gong, Jae Sung Park, Bidipta Sarkar, Rohan Taori, Yusuke Noda, Demetri Terzopoulos, Yejin Choi, et al. 2024. Agent ai: Surveying the horizons of multimodal interaction. *arXiv preprint arXiv:2401.03568*. 612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

- Kuan Fang, Yuke Zhu, Silvio Savarese, and Li Fei-Fei. 2020. Adaptive procedural task generation for hard-exploration problems. *arXiv preprint arXiv:2007.00350.*
- Chuang Gan, Siyuan Zhou, Jeremy Schwartz, Seth Alter, Abhishek Bhandwaldar, Dan Gutfreund, Daniel LK Yamins, James J DiCarlo, Josh McDermott, Antonio Torralba, et al. 2021. The threedworld transport challenge: A visually guided task-and-motion planning benchmark for physically realistic embodied ai. *arXiv preprint arXiv:2103.14025*.
- Jiayuan Gu, Devendra Singh Chaplot, Hao Su, and Jitendra Malik. 2022. Multi-skill mobile manipulation for object rearrangement. *arXiv preprint arXiv:2209.02778*.
- Mengkang Hu, Pu Zhao, Can Xu, Qingfeng Sun, Jianguang Lou, Qingwei Lin, Ping Luo, Saravan Rajmohan, and Dongmei Zhang. 2024. Agentgen: Enhancing planning abilities for large language model based agent via environment and task generation. *arXiv preprint arXiv:2408.00764*.
- Ahmed Jaafar, Shreyas Sundara Raman, Yichen Wei, Sudarshan Harithas, Sofia Juliani, Anneke Wernerfelt, Benedict Quartey, Ifrah Idrees, Jason Xinyu Liu, and Stefanie Tellex. 2025.  $\lambda$ : A benchmark for dataefficiency in long-horizon indoor mobile manipulation robotics. *Preprint*, arXiv:2412.05313.
- Youngwoon Lee, Joseph J Lim, Anima Anandkumar, and Yuke Zhu. 2021. Adversarial skill chaining for long-horizon robot manipulation via terminal state regularization. *arXiv preprint arXiv:2111.07999*.
- Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. 2023. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pages 80–93. PMLR.
- Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Erran Li Li, Ruohan Zhang, et al. 2025. Embodied agent interface: Benchmarking llms for embodied decision making. *Advances in Neural Information Processing Systems*, 37:100428–100534.
- Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. 2023. Code as policies: Language model programs for embodied control. In 2023 IEEE International Conference on Robotics and Automation (ICRA), pages 9493–9500. IEEE.

765

767

768

769

770

771

772

773

Xiao Liu, Tianjie Zhang, Yu Gu, Iat Long Iong, Yifan Xu, Xixuan Song, Shudan Zhang, Hanyu Lai, Xinyi Liu, Hanlin Zhao, et al. 2024. Visualagentbench: Towards large multimodal models as visual foundation agents. *arXiv preprint arXiv:2408.06327*.

667

674

675

677

679

695

702

704

705

710

711

712

713

714

715

716

717

718

719

- Meta. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models. Accessed: 2025-02-15.
- Roberta Raileanu and Tim Rocktäschel. 2020. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*.
- Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020a. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749.
- Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2020b. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*.
- Marta Skreta, Zihan Zhou, Jia Lin Yuan, Kourosh Darvish, Alán Aspuru-Guzik, and Animesh Garg. 2024. Replan: Robotic replanning with perception and language models. *arXiv preprint arXiv:2401.04157*.
- Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. 2023.
  Llm-planner: Few-shot grounded planning for embodied agents with large language models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2998–3009.
- Markku Suomalainen, Yiannis Karayiannidis, and Ville Kyrki. 2022. A survey of robot manipulation in contact. *Robotics and Autonomous Systems*, 156:104224.
- Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. 2021. Habitat 2.0: Training home assistants to rearrange their habitat. In Advances in Neural Information Processing Systems (NeurIPS).
- Andrew Szot, Max Schwarzer, Harsh Agrawal, Bogdan Mazoure, Rin Metcalf, Walter Talbott, Natalie Mackraz, R Devon Hjelm, and Alexander T Toshev. 2023. Large language models as generalizable policies for embodied tasks. In *The Twelfth International Conference on Learning Representations*.

- Andrew Szot, Karmesh Yadav, Alex Clegg, Vincent-Pierre Berges, Aaron Gokaslan, Angel Chang, Manolis Savva, Zsolt Kira, and Dhruv Batra. 2022. Habitat rearrangement challenge 2022. https:// aihabitat.org/challenge/2022\_rearrange.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, and et al. Anja Hauth. 2024. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.
- Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. 2023. Gensim: Generating robotic simulation tasks via large language models. *arXiv preprint arXiv:2310.01361*.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.
- Luca Weihs, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. 2021. Visual room rearrangement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5922–5931.
- Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. 2020. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11097–11107.
- Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. 2025. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *Preprint*, arXiv:2502.09560.
- Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. 2024. Guiding long-horizon task and motion planning with vision language models. *arXiv preprint arXiv:2410.02193*.
- Xiaohan Zhang, Yan Ding, Saeid Amiri, Hao Yang, Andy Kaminski, Chad Esselink, and Shiqi Zhang. 2023. Grounding classical task planners via visionlanguage models. *arXiv preprint arXiv:2304.08587*.
- Zeyu Zhang, Sixu Yan, Muzhi Han, Zaijin Wang, Xinggang Wang, Song-Chun Zhu, and Hangxin Liu. 2024. M3bench: Benchmarking whole-body motion generation for mobile manipulation in 3d scenes. *arXiv preprint arXiv:2410.06678*.

# A Appendix

774

775

776

777

778

779

782

819

# A.1 Details of ManiTaskOT-1K.

The ManiTaskOT-1K dataset contains 1,000 templates for outcome-based tasks. We first collected scene images from real-world and simulated datasets, including Habitat (Szot et al., 2021), Scan-Net (Dai et al., 2017). Next, we used Amazon Mechanical Turk (AMT) (Amazon, 2024) to gather natural language instructions from human annotators. Finally, we manually filtered the instructions for quality and removed specific object names to create task templates.

We provide more template examples as follows:

787	"Design a symmetrical display of [SUB-
788	OBJECTS00] on [OBJECT0]."
789	"Design an aesthetic display of [SUB-
790	OBJECTS00] on [OBJECT0]."
791	"Group all the scattered [SUB-
792	OBJECTS00] on [OBJECT0] by size,
793	and stack the largest ones vertically on
794	the left side of the [OBJECT0]."
795	"Construct a mini obstacle course using
796	small objects on the [OBJECT0]."
797	"Align all [SUB-OBJECTS00] on [OB-
798	JECT0] into a symmetrical arrange-
799	ment."
800	"Arrange the objects on [OBJECT0] into
801	an interactive display that showcases
802	their use and function."
803	"Create a "kaleidoscope" on top of [OB-
804	JECT0] by arranging objects in a color-
805	ful, symmetrical pattern."
806	"Create a "city" on [OBJECT0] by ar-
807	ranging the objects to resemble buildings,
808	roads, and parks."
809	"Transform [OBJECT0] into a "garden"
810	by arranging the surface objects to resem-
811	ble flowers, trees, and paths."
812	"For the collection of small electronic
813	devices on [OBJECT0], use principles of
814	proportion and balance to group them."
815	"Organize the contents of [OBJECT0]
816	into a streamlined and functional layout,
817	and group similar items."
818	We plan to release the dataset to benefit the com-

munity once the paper is public.

#### A.2 Benchmark Test Flow Sample

Below we display an entire history of a successful episode of the benchmark test flow, achieved by Gemini-2.0-flash.

In the text boxes starting on the next page, we provide the text transcripts. For the images uploaded to the VLM during the conversation, we have organized them in Fig. 7. The blue paragraphs in the text indicate parts related to the images, and the numbers preceding the text correspond to the image numbers in Fig. 7.

Note that in practical applications, we may provide photos of an object or location taken from slightly different angles to help the VLM agent obtain a more comprehensive observation when the scene is partially occluded.



Figure 7: History of images provided to the VLM agent.

#### Success Episode by Gemini-2.0-flash

System Message:

Suppose you are an expert home robot. You're now in a room, and are given tasks which mainly involves puttings some thing from one place to another. I want you to act like a robot, interact with the scene and try your best to finish the tasks. In the scene you will like running in an automaton. In each state, all the actions you can do can be described with an action space.

Note: 1: As we've guaranteed that the space you need to put item on must be somehow available at the beginning, if you think there's not enough space for you to put down the item, you can either try moving away obstacles(probably involved by your wrong move), or change your standing position. 2: The system will let you try until reach a maximum time step count. Try your best to finish the task with the least steps of interaction!

Let's get started!

Please process task 5483:

"Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part".

Task Initial Information:

frl\_apartment\_kitchen\_utensil\_09\_73 belong\_to\_platform: kitchen\_counter\_1\_body\_platform\_1. And The destination platform is frl\_apartment\_rack\_01\_76\_platform\_2.

[1] We also provide images of involved objects, the first image I upload to you is about frl\_apartment\_kitchen\_utensil\_09\_73, marked with a grey shadowed cube.

User Message:

Now you're preparing to get in to the scene. Remember the task is: "Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 0 timesteps.

Task Initial Information:

frl\_apartment\_kitchen\_utensil\_09\_73 belong\_to\_platform: kitchen\_counter\_1\_body\_platform\_1. The destination platform is frl\_apartment\_rack\_01\_76\_platform\_2.

Now you have these actions to choose, and you need to select a key as an action from this current action space dict:

'0': 'call\_end', "11': 'go to platform 1: frl\_apartment\_sofa\_10\_platform\_0', '12': 'go to platform 2: frl\_apartment\_sofa\_10\_platform\_1', '13': 'go to platform 3: frl\_apartment\_sofa\_10\_platform\_2', '14': '15': platform frl\_apartment\_chair\_04\_46\_platform\_0', 'go to 4: 'go platform 5: frl\_apartment\_chair\_04\_47\_platform\_0', '16': 'go to platform 'go 6: frl\_apartment\_chair\_05\_7\_platform\_0', '17': to to platform 7: frl\_apartment\_chair\_05\_8\_platform\_0', '18': go to platform 8: frl apartment stool 02 18 platform 0', '19': plat-'go to form 9: frl\_apartment\_stool\_02\_6\_platform\_0', '110': 'go to platform 10: frl apartment rack 01 76 platform 0', '111': to plat-'go form 11: frl\_apartment\_rack\_01\_76\_platform\_1', '112': 'go to platform 12: frl\_apartment\_rack\_01\_76\_platform\_2', '113': 'go to platform 13: kitchen\_counter\_1\_body\_platform\_0', '114': 'go to platform 14: kitchen\_counter\_1\_body\_platform\_1', '115': 'go to platform 15: fridge\_0\_body\_platform\_0', fridge\_0\_body\_platform\_1', '117': '116': 'go to platform 16: 'go to platform 17: fridge\_0\_body\_platform\_2', '118': 'go to platform 18: fridge\_0\_body\_platform\_3', '119': 'go to platform 19: fridge\_0\_body\_platform\_4', 'l20': 'go to platform 20: fridge\_0\_body\_platform\_5', '121': 'go to platform 21: fridge\_0\_body\_platform\_6', '122': 'go to platform 22:

fridge\_0\_body\_platform\_7', '123': 'go to platform 23: frl\_apartment\_bin\_02\_2\_platform\_0', '124': 'go to platform 24: frl\_apartment\_bin\_03\_3\_platform\_0', '125': 'go to platform 25: frl apartment wall cabinet 01 4 platform 0', '126': 'go to platfrl\_apartment\_wall\_cabinet\_01\_4\_platform\_1', form 26: '127': 'go to platform 27: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_2', '128': 'go to platform 28: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_3', '129': 'go to platform 29: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_4', '130': 'go to platform 30: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_5', '131': 'go to platform 31: frl apartment wall cabinet 01 4 platform 6', '132': to plat-'go frl\_apartment\_table\_03\_14\_platform\_0', '133': form 32: 'go to plat-'134': form 33: frl\_apartment\_table\_03\_14\_platform\_1', 'go to platform 34: frl\_apartment\_table\_01\_48\_platform\_0', '135': 'go to platform 35: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_0', '136': 'go to platform 36: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_1', '137': 'go to plat-37: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_2', '138': form 'go to platform 38: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_3', '139': 'go to plat-39: frl apartment wall cabinet 02 21 platform 4', form '140': 'go to platform 40: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_5', '141': 'go to platform 41: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_6', '142': 'go to 42: platplatform frl\_apartment\_chair\_01\_15\_platform\_0', '143': go to form 43: frl\_apartment\_chair\_01\_15\_platform\_1', '144': 'go to platform frl\_apartment\_chair\_01\_16\_platform\_0', '145': 44: platform 'go to 45: frl\_apartment\_chair\_01\_16\_platform\_1', '146': 'go to platform 46: chestOfDrawers\_01\_2\_body\_platform\_0', 'l47': 'go to platform 47: chestOfDrawers\_01\_2\_body\_platform\_1', '148': go to platform 48: chestOfDrawers\_01\_2\_body\_platform\_2', '149': 'go to platform 49: chestOfDrawers\_01\_2\_body\_platform\_3', '150': 'go to platform 50: chestOfDrawers\_01\_2\_body\_platform\_4', '151': platform 51: 'go to chestOfDrawers\_01\_2\_body\_platform\_5', '152': 'go to platform 52: chestOfDrawers\_01\_2\_body\_platform\_6', '153': 'go to platform 53: frl\_apartment\_table\_02\_40\_platform\_0', 'go to platform 54: frl\_apartment\_table\_02\_40\_platform\_1', '155': '154': 'go to frl\_apartment\_table\_02\_40\_platform\_2', '156': 'go to platform 56: platform 55: frl\_apartment\_tvstand\_89\_platform\_0', '157': 'go to platform 57: cabinet\_3\_body\_platform\_0', '158': 'go to platform 58: cabinet\_3\_body\_platform\_1'.

Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character format will directly cause the entire task to fail.

VLM response: 114

User Message:

Now you may see the objects on the platform kitchen\_counter\_1\_body\_platform\_1. Or you may not see them clearly due to bad starting direction. Remember the task is: "Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 1 timesteps.

TaskInitialInformation:frl\_apartment\_kitchen\_utensil\_09\_73belong\_to\_platform:kitchen\_counter\_1\_body\_platform\_1.Thedestinationplatform isfrl\_apartment\_rack\_01\_76\_platform\_2.

[2] We also provide an image from your view, with all objects you may pick up are marked with grey shadowed cube and circled numbers. Now you can call\_end, rotate clockwisely to another freespace, go to another platform or choose to pick up one item here.

	Action space description:'-1': "rotate clockwisely to the nearest freespace that can fit you, the							
robot. In case there's no any other freespace, you'll stay in the same position.", '0': 'call_end',								
'01': 'p	oick up oł	oject 1', 'o2': 'pick up object 2', 'o3': 'pick up object 3',	'o4': 'pick up	p obje	ct 4',			
'o5': 'j	pick up o	bject 5', 'o6': 'pick up object 6', 'o7': 'pick up object 7	", '08': 'pick	c up o	bject			
8', '09	': 'pick u	p object 9', 'o10': 'pick up object 10', 'o11': 'pick up	object 11 <sup>°</sup> , 'c	o12':	'pick			
up obje	ect 12', 'o	013': 'pick up object 13', '014': 'pick up object 14', '01	5': 'pick up	objec	t 15',			
'o16':	'pick up	object 16', 'o17': 'pick up object 17', 'o18': 'pick up o	object 18', 'c	o19':	'pick			
up obje	ect 19', 'o	520': 'pick up object 20', 'o21': 'pick up object 21', 'o2	22': 'pick up	objec	t 22',			
'o23':	'pick up o	object 23', 'o24': 'pick up object 24', 'o25': 'pick up obj	ject 25', 'o26	': <sup>'</sup> pio	ck up			
object	26', '11':	'go to platform 1: frl apartment sofa 10 platform 0',	'12': 'go to j	olatfo	rm 2:			
frl apa	rtment s	ofa 10 platform 1', '13': 'go to platform 3: frl apartmen	nt sofa 10 pl	latfori	n 2',			
'14':	'go	to platform 4: frl_apartment_chair_04_4	6_platform_(	)',	'15':			
'go	to pla	tform 5: frl_apartment_chair_04_47_platform	_0', '16':		'go			
to p	olatform	6: frl_apartment_chair_05_7_platform_0',	'17':	'go	to			
platfor	m 7:	frl_apartment_chair_05_8_platform_0', '18':	'go	to	plat-			
form	8:	frl_apartment_stool_02_18_platform_0', '19':	'go	to	plat-			
form	9:	frl_apartment_stool_02_6_platform_0', '110':	'go	to	plat-			
form	10:	frl_apartment_rack_01_76_platform_0', '111':	'go	to	plat-			
form	11:	frl_apartment_rack_01_76_platform_1', '112':	'go	to	plat-			
form	12:	frl_apartment_rack_01_76_platform_2', '113':	'go	to	plat-			
form	13:	kitchen_counter_1_body_platform_0', '114': 'g	o to plat	form	14:			
kitcher	n_counter	-1_body_platform_1', '115': 'go to platform 15: fridg	e_0_body_pl	atforr	n_0',			
'116':	'go to	platform 16: fridge_0_body_platform_1', '117':	'go to pla	tform	17:			
fridge 0 body platform 2' '118': 'go to platform 18: fridge 0 body platform 3' '119': 'go to								
fridge_	_0_body_	platform_2', 'l18': 'go to platform 18: fridge_0_body_pl	atform_3', '1	19': '	go to			
platfor	_0_body_ m 19: frie	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg	atform_3', 'l ge_0_body_pl	19': ' latfori	go to n_5',			
platfor '121':	_0_body_ m 19: fric 'go to	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122':	atform_3', 'l ge_0_body_pl 'go to pla	19': ' latfori itform	go to m_5', 1 22:			
platfor '121': fridge_	_0_body_ m 19: frid 'go to _0_body_	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_	atform_3', '1 ge_0_body_pl 'go to pla _bin_02_2_pl	19': ' latforn tform atforn	go to n_5', 1 22: n_0',			
fridge_ platfor '121': fridge_ '124':	_0_body_ m 19: frid 'go to _0_body_ 'go ta	platform_2', '118': 'go to platform 18: fridge_0_body_pl dge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_	atform_3', '1 ge_0_body_pl 'go to pla bin_02_2_pl _0', '125':	19': ' latforn tform atforn 'go	go to n_5', 1 22: n_0', 5 to			
platfor '121': fridge_ '124': platfor	_0_body_ m 19: frid 'go to _0_body_ 'go t m 25:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12'	atform_3', 'l ge_0_body_pl 'go to pla bin_02_2_pl _0', 'l25': 26': 'go	19': ' latforn itform atforn 'go to	go to m_5', n_22: m_0', p_to plat-			
platfor 'l21': fridge_ 'l24': platfor form	_0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127	atform_3', '1 'ge_0_body_pl 'go to pla bin_02_2_pl _0', '125': 26': 'go '': 'go	19': ' latforn atforn atforn 'go to to	go to m_5', n_22: m_0', p_ to plat- plat-			
platfor 'l21': fridge_ 'l24': platfor form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128	atform_3', 'l ge_0_body_pl 'go to pla bin_02_2_pl _0', 'l25': 26': 'go ': 'go 3': 'go	19': ' latform atform atform 'go to to to	go to n_5', n_22: n_0', po to plat- plat- plat-			
platfor 'l21': fridge_ 'l24': platfor form form	_0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129	atform_3', 'l 'go to pla bin_02_2_pl _0', 'l25': 26': 'go 7': 'go 3': 'go 9': 'go	19': ' latform atform atform 'go to to to to	go to n_5', a 22: n_0', b to plat- plat- plat- plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_3', '129	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     3': 'go     0': 'go     '': 'go	19': ' latforn itform atforn 'go to to to to to to	go to $n_5$ ', 22: $n_0$ ', p to plat- plat- plat- plat- plat-			
platfor 'l21': fridge_ 'l24': platfor form form form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131	atform_3', 'l     ge_0_body_pl     'go to pla     bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     3': 'go     0': 'go     '': 'go	19': ' latforn atforn atforn 'go to to to to to to	go to n_5', a 22: n_0', b to plat- plat- plat- plat- plat- plat- plat-			
fridge_ platfor 'l21': fridge_ 'l24': platfor form form form form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go	19': ' atforn atforn atforn to to to to to to to to to	go to n_5', 22: n_0', b to plat- plat- plat- plat- plat- plat- plat- plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133':	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     'go	19': ' atforn atforn atforn 'go to to to to to to to	go to n_5', 22: n_0', b to plat- plat- plat- plat- plat- plat- plat- plat- plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_1', '134': frl_apartment_table_03_14_platform_1', '134':	atform_3', 'l 'go to pla bin_02_2_pl _0', 'l25': 26': 'go 7': 'go 3': 'go 1': 'go 1': 'go 1': 'go 2': 'go 2': 'go 2': 'go 2': 'go 2': 'go 2': 'go	19': ' latforn atforn atforn to to to to to to to to to to	go to n_5', 1 22: n_0', 2 to plat- plat- plat- plat- plat- plat- plat- plat- plat- plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_01_48_platform_0', '135':	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go	19': ' latforn atforn atforn 'go to to to to to to to to to to to to	go to n_5', 22: n_0', b to plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	_0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_aj	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_01_48_platform_0', '135': partment_wall_cabinet_02_21_platform_0', '136':	latform_3', 'l 'go to pla 'go to pla bin_02_2_pl _0', 'l25': 26': 'go ?': 'go	19': ' latforn atforn atforn 'go to to to to to to to to to to to to	go to n_5', 22: n_0', b to plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	_0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_a] 36:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_03_14_platform_1', '134': frl_apartment_table_01_48_platform_0', '135': partment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136':	latform_3', 'l 'go to pla bin_02_2_pl _0', 'l25': 26': 'go ': 'go ': 'go ': 'go ': 'go ': 'go ': 'go ': 'go 'go 'go 'go to 'go 7': 'go	19': ' latforn atforn atforn to to to to to to to to to to to to to	go to n_5', 22: n_0', b to plat- pla			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_a] 36: 37:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_03_14_platform_0', '135': partment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '130	latform_3', 'l 'go to pla bin_02_2_pl _0', 'l25': 26': 'go 7': 'go 7': 'go 7': 'go 1': 'go 1': 'go 1': 'go 2': 'go 'go to 'go to 'go 7': 'go 8': 'go	19': ' latforn atforn atforn 'go to to to to to to to to to to to to to	go to n_5', 22: n_0', 2 to plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_aj 36: 37: 38:	platform_2', '118': 'go to platform 18: fridge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_0', '133': frl_apartment_table_03_14_platform_0', '134': frl_apartment_table_01_48_platform_0', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     ''go	19': ' latforn atforn atforn 'go to to to to to to to to to to to to to	go to n_5', 22: n_0', b to plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_a] 36: 37: 38: 39:	platform_2', '118': 'go to platform 18: fridge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform 21: fridge_0_body_platform_6', '122': platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_03_14_platform_0', '135': partment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     3': 'go     '': 'go     ''go	19': ' latforn atforn atforn to to to to to to to to to to to to to	go to n_5', 22: n_0', plat-			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_a] 36: 37: 38: 39: 40:	platform_2', '118': 'go to platform 18: fridge_0_body_pl lge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_03_14_platform_0', '135': partment_table_01_48_platform_0', '136': frl_apartment_wall_cabinet_02_21_platform_1', '134' frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go	19': ' latforn atforn atforn to to to to to to to to to to to to to	go to n_5', 1 22: n_0', 2 to plat- p			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_a] 36: 37: 38: 39: 40: 41:	platform_2', '118': 'go to platform 18: fridge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform 7', '123': 'go to platform 23: frl_apartment_ o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_0', '133': frl_apartment_table_03_14_platform_0', '134': frl_apartment_table_01_48_platform_0', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '139 frl_apartment_wall_cabinet_02_21_platform_5', '131 frl_apartment_wall_cabinet_02_21_platform_5', '131 frl_apartment_wall_cabinet_02_21_platform_6', '132 frl_apartment_wall_cabinet_02_21_platform_6', '134 frl_apartment_wall_cabinet_02_21_platform_6', '135 frl_apartment_wall_cabinet_02_21_platform_6', '135 frl_apartment_wall_cabinet_02_21_platform_6', '135 frl_apartment_wall_cabinet_02_21_platform_6', '135 frl_apartment_wall_cabinet_02_21_platform_6', '135 frl_apartment_wall_cabinet_02_21_platform_6', '135	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     ''go     ''go     ''go     ''go     ''go     ''go     ''go     ''go     'go  'go </td <td>19': ' latforn atforn atforn 'go to to to to to to to to to to to to to</td> <td>go to n_5', 22: n_0', b to plat- pla</td>	19': ' latforn atforn atforn 'go to to to to to to to to to to to to to	go to n_5', 22: n_0', b to plat- pla			
fridge_ platfor '121': fridge_ '124': platfor form form form form form form form f	0_body_ m 19: frid 'go to _0_body_ 'go t m 25: 26: 27: 28: 29: 30: 31: 32: 33: 34: frl_aj 36: 37: 38: 39: 40: 41: m 42:	platform_2', '118': 'go to platform 18: fridge_0_body_platform_4', '120': 'go to platform 20: fridg platform 21: fridge_0_body_platform_6', '122': platform_7', '123': 'go to platform 23: frl_apartment_o platform 24: frl_apartment_bin_03_3_platform_ frl_apartment_wall_cabinet_01_4_platform_0', '12 frl_apartment_wall_cabinet_01_4_platform_1', '127 frl_apartment_wall_cabinet_01_4_platform_2', '128 frl_apartment_wall_cabinet_01_4_platform_3', '129 frl_apartment_wall_cabinet_01_4_platform_4', '130 frl_apartment_wall_cabinet_01_4_platform_5', '131 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_wall_cabinet_01_4_platform_6', '132 frl_apartment_table_03_14_platform_0', '133': frl_apartment_table_01_48_platform_0', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_1', '136': frl_apartment_wall_cabinet_02_21_platform_2', '135 frl_apartment_wall_cabinet_02_21_platform_3', '135 frl_apartment_wall_cabinet_02_21_platform_5', '14 frl_apartment_wall_cabinet_02_21_platform_6', '143'	atform_3', 'l     'go to pla     'go to pla     _bin_02_2_pl     _0', 'l25':     26': 'go     '': 'go     3': 'go     '': 'go </td <td>19': ' latforn atforn atforn to to to to to to to to to to to to to</td> <td>go to n_5', 22: n_0', plat- pl</td>	19': ' latforn atforn atforn to to to to to to to to to to to to to	go to n_5', 22: n_0', plat- pl			

44: frl\_apartment\_chair\_01\_16\_platform\_0', '145': platform 45: 'go to frl\_apartment\_chair\_01\_16\_platform\_1', '146': 'go to platform 46: chestOfDrawers\_01\_2\_body\_platform\_0', 'l47': 'go to platform 47: chestOfDrawers\_01\_2\_body\_platform\_1', chestOfDrawers\_01\_2\_body\_platform\_2', '148': 'go to platform 48: '149': 'go platform 49: chestOfDrawers 01 2 body platform 3', '150': 'go to platto form 50: chestOfDrawers\_01\_2\_body\_platform\_4', '151': 'go to platform 51: chestOfDrawers\_01\_2\_body\_platform\_5', '152': 'go to platform 52: chestOfDrawers\_01\_2\_body\_platform\_6', '153': 'go to platform 53: frl\_apartment\_table\_02\_40\_platform\_0', 'go to platform 54: frl apartment table 02 40 platform 1', '155': '154': 'go to frl\_apartment\_table\_02\_40\_platform\_2', '156': platform 55: 'go to platform 56: frl\_apartment\_tvstand\_89\_platform\_0', '157': 'go to platform 57: cabinet\_3\_body\_platform\_0', '158': 'go to platform 58: cabinet\_3\_body\_platform\_1'You need to select an action from this current action space. Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character format will directly cause the entire task to fail.

# VLM response: 09

### User Message:

Now you are standing at kitchen\_counter\_1\_body\_platform\_1 with object frl\_apartment\_kitchen\_utensil\_09\_73 in hand.Or you may not see the platform clearly due to bad starting direction.Now you have the following choices and corresponding outputs:-1: switch clockwisely to the next free space of the platform 0: call\_end 11-58: go to platform i startplace: place the object in hand.

Remember the task is: "Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 2 timesteps. Task Initial Information:

frl\_apartment\_kitchen\_utensil\_09\_73 belong\_to\_platform: kitchen\_counter\_1\_body\_platform\_1. The destination platform is frl\_apartment\_rack\_01\_76\_platform\_2.

[3] We also provide an image from your view, with all objects you may pick up are marked with grey shadowed cube and circled numbers. The item you've just picked up should disappear. Now you can call\_end, rotate clockwisely to another freespace, go to another platform or start placing the item on your hand here.

The current Action space description is as below. It is in the form of a dictionary, where the key represents an executable action, and the value describes the details of that action.

-									
Action space description:'-1': "rotate clockwisely to the nearest freespace that can fit you,									
the robot. In case there's no any other freespace, you'll stay in the same position.", '0':									
'call_end','l1': 'go to platform 1: frl_apartment_sofa_10_platform_0', 'l2': 'go to platform 2:									
frl_apartment_sofa_10_platform_1', '13': 'go to platform 3: frl_apartment_sofa_10_platform_2',									
'14':	'go	to	platform	4:	frl_apartment_ch	air_04_46	_platform_	0',	'15':
'go	to plat	form	5:	frl_apar	tment_chair_04_47_	_platform_	0', '16	':	'go
to p	olatform	6:	frl_a	partment	_chair_05_7_platfor	m_0',	'17':	'go	to
platfor	m 7:	frl	l_apartme	nt_chair_	_05_8_platform_0',	'18':	'go	to	plat-
form	8:	frl_a	partment_	_stool_02	_18_platform_0',	'19':	'go	to	plat-
form	9:	frl_a	partment_	_stool_02	_6_platform_0',	'110':	'go	to	plat-
form	10:	frl_a	apartment	_rack_01	_76_platform_0',	'111':	'go	to	plat-
form	11:	frl_a	apartment	_rack_01	_76_platform_1',	'112':	'go	to	plat-
form	12:	frl_a	apartment	_rack_01	_76_platform_2',	'113':	'go	to	plat-

form kitchen\_counter\_1\_body\_platform\_0', '114': platform 13: 'go to 14: kitchen\_counter\_1\_body\_platform\_1', '115': 'go to platform 15: fridge\_0\_body\_platform\_0', fridge\_0\_body\_platform\_1', '117': 'go to platform 17: '116': 'go to platform 16: fridge\_0\_body\_platform\_2', '118': 'go to platform 18: fridge\_0\_body\_platform\_3', '119': 'go to platform 19: fridge 0 body platform 4', '120': 'go to platform 20: fridge 0 body platform 5', 'go to platform 21: fridge\_0\_body\_platform\_6', '122': 'go to platform 22: '121': fridge\_0\_body\_platform\_7', 'l23': 'go to platform 23: frl\_apartment\_bin\_02\_2\_platform\_0', '124': 'go to platform 24: frl\_apartment\_bin\_03\_3\_platform\_0', '125': 'go to 25: frl apartment wall cabinet 01 4 platform 0', '126': platform 'go to platfrl\_apartment\_wall\_cabinet\_01\_4\_platform\_1', form 26: '127': 'go to plat-27: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_2', '128': form 'go to platform 28: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_3', '129': 'go to platform 29: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_4', '130': 'go to platform 30: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_5', '131': 'go to plat-31: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_6', '132': 'go form to platform 32: frl\_apartment\_table\_03\_14\_platform\_0', '133': 'go to plat-33: form frl apartment table 03 14 platform 1', '134': plat-'go to form 34: frl\_apartment\_table\_01\_48\_platform\_0', '135': platform 'go to 35: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_0', '136': 'go to platform 36: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_1', '137': plat-'go to form 37: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_2', '138': 'go to plat-38: frl apartment wall cabinet 02 21 platform 3', '139': form 'go to platform 39: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_4', '140': to plat-'go form 40: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_5', '141': 'go to plat-41: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_6', form '142': 'go to platform 42: frl\_apartment\_chair\_01\_15\_platform\_0', '143': plat-'go to frl\_apartment\_chair\_01\_15\_platform\_1', 43: 'go form '144': to platform '145': 44: frl\_apartment\_chair\_01\_16\_platform\_0', platform 45: 'go to frl\_apartment\_chair\_01\_16\_platform\_1', '146': 'go to platform 46: chestOfDrawers\_01\_2\_body\_platform\_0', 'l47': 'go to platform 47: chestOfDrawers\_01\_2\_body\_platform\_1', '148': 'go to platform 48: chestOfDrawers\_01\_2\_body\_platform\_2', '149': 'go chestOfDrawers 01 2 body platform 3', to platform 49: '150': 'go to plat-50: chestOfDrawers\_01\_2\_body\_platform\_4', '151': 'go to platform 51: form chestOfDrawers 01 2 body platform 5', '152': 'go to platform 52: chestOfDrawers\_01\_2\_body\_platform\_6', '153': 'go to platform 53: frl\_apartment\_table\_02\_40\_platform\_0', 'go to platform 54: frl\_apartment\_table\_02\_40\_platform\_1', '155': '154': 'go to platform 55: frl\_apartment\_table\_02\_40\_platform\_2', '156': 'go to platform 56: frl\_apartment\_tvstand\_89\_platform\_0', '157': 'go to platform 57: cabinet\_3\_body\_platform\_0', '158': 'go to platform 58: cabinet\_3\_body\_platform\_1': 'go to platform 58: cabinet\_3\_body\_platform\_1', 'startplace': 'place the object in hand to the current platform'You need to select an action from this current action space. Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character format will directly cause the entire task to fail.

#### VLM response: 112

User Message:

Now you are standing at frl\_apartment\_rack\_01\_76\_platform\_2.with object frl\_apartment\_kitchen\_utensil\_09\_73 in hand. Or you may not see the platform clearly due to bad starting direction.Now you have the following choices and corresponding outputs:-1:

switch clockwisely to the next free space of the platform 0: call\_end 11-58: go to platform i startplace: place the object in hand.

Remember the task is: "Move frl apartment kitchen utensil 09 73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 3 timesteps. Task Initial Information: frl\_apartment\_kitchen\_utensil\_09\_73 belong to platform: kitchen\_counter\_1\_body\_platform\_1. The destination platform is frl\_apartment\_rack\_01\_76 platform 2.

[4] We also provide an image from your view, with all objects you may pick up are marked with grey shadowed cube and circled numbers. Now you can call\_end, rotate to another freespace, go to another platform or start placing the item on your hand here.

The current Action space description is as below. It is in the form of a dictionary, where the key represents an executable action, and the value describes the details of that action.

Action space description:'-1': "rotate clockwisely to the nearest freespace that can fit you, the robot. In case there's no any other freespace, you'll stay in the same position.", '0': 'call end', '11': 'go to platform 1: frl apartment sofa 10 platform 0', '12': 'go to platform 2: frl\_apartment\_sofa\_10\_platform\_1', '13': 'go to platform 3: frl\_apartment\_sofa\_10\_platform\_2', '14': platform 4: frl\_apartment\_chair\_04\_46\_platform\_0', 'go to '15': platform 5: frl apartment chair 04 47 platform 0', 'go '16': 'go to platform frl\_apartment\_chair\_05\_7\_platform\_0', to 6: '17': 'go to frl\_apartment\_chair\_05\_8\_platform\_0', platform 7: '18': 'go to platform 8: frl\_apartment\_stool\_02\_18\_platform\_0', '19': 'go platto platform 9: frl\_apartment\_stool\_02\_6\_platform\_0', '110': 'go to form 10: frl\_apartment\_rack\_01\_76\_platform\_0', '111': 'go to platform 11: frl\_apartment\_rack\_01\_76\_platform\_1', '112': 'go to platfrl\_apartment\_rack\_01\_76\_platform\_2', '113': form 12: plat-'go to platform form 13: kitchen\_counter\_1\_body\_platform\_0', '114': 'go 14: to kitchen\_counter\_1\_body\_platform\_1', 'l15': 'go to platform 15: fridge\_0\_body\_platform\_0', '116': 'go to platform 16: fridge\_0\_body\_platform\_1', '117': 'go to platform 17: fridge\_0\_body\_platform\_2', '118': 'go to platform 18: fridge\_0\_body\_platform\_3', '119': 'go to platform 19: fridge\_0\_body\_platform\_4', '120': 'go to platform 20: fridge\_0\_body\_platform\_5', '121': 'go to platform 21: fridge\_0\_body\_platform\_6', '122': 'go to platform 22: fridge\_0\_body\_platform\_7', '123': 'go to platform 23: frl\_apartment\_bin\_02\_2\_platform\_0', '124': 'go to platform 24: frl\_apartment\_bin\_03\_3\_platform\_0', '125': 'go to platform 25: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_0', '126': 'go to platform 26: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_1', '127': 'go to platform 27: frl apartment wall cabinet 01 4 platform 2', '128': 'go to platform 28: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_3', '129': to plat-'go 29: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_4', '130': form 'go to platplatform 30: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_5', '131': 'go to form 31: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_6', '132': 'go to plat-'133': 'go form 32: frl\_apartment\_table\_03\_14\_platform\_0', to platform 33: frl apartment table 03 14 platform 1', '134': 'go to platform 34: frl\_apartment\_table\_01\_48\_platform\_0', '135': platform 'go to frl\_apartment\_wall\_cabinet\_02\_21\_platform\_0', '136': 'go 35: to platfrl apartment wall cabinet 02 21 platform 1', form 36: '137': 'go to platform 37: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_2', '138': 'go to platform 38: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_3', '139': 'go to plat-39: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_4', form '140': 'go to plat-

form 40: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_5', '141': plat-'go to form 41: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_6', '142': 'go to frl\_apartment\_chair\_01\_15\_platform\_0', platform 42: '143': go to platform 43: frl\_apartment\_chair\_01\_15\_platform\_1', '144': to platform 'go 44: frl apartment chair 01 16 platform 0', '145': 'go 45: to platform frl\_apartment\_chair\_01\_16\_platform\_1', '146': 'go to platform 46: chestOfDrawers\_01\_2\_body\_platform\_0', 'l47': 'go to platform 47: chestOfDrawers\_01\_2\_body\_platform\_1', '148': 'go to platform 48: chestOfDrawers\_01\_2\_body\_platform\_2', '149': go chestOfDrawers 01 2 body platform 3', to platform 49: '150': 'go to platchestOfDrawers\_01\_2\_body\_platform\_4', '151': 50: platform 51: form 'go to chestOfDrawers\_01\_2\_body\_platform\_5', '152': 'go to platform 52: chestOfDrawers\_01\_2\_body\_platform\_6', '153': 'go to platform 53: frl\_apartment\_table\_02\_40\_platform\_0', '154': 'go to platform 54: frl\_apartment\_table\_02\_40\_platform\_1', '155': 'go to platform 55: frl\_apartment\_table\_02\_40\_platform\_2', '156': 'go to platform 56: frl\_apartment\_tvstand\_89\_platform\_0', '157': 'go to platform 57: cabinet\_3\_body\_platform\_0', '158': 'go to platform 58: cabinet\_3\_body\_platform\_1', 'startplace': 'place the object in hand to the current platform'You need to select an action from this current action space. Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character format will directly cause the entire task to fail.

#### VLM response: startplace

# User Message:

Now you are standing at an empty platform. You have the following choices and corresponding outputs: 0: call\_end 1-9: place the object in hand on the ith 9-grid of the platform. There are images with each place's number marked using colorful rectangles and circled numbers, and each of these images also have a copy without marks.

999: place the object in hand around the platform, system will help you find an arbitrary fitable place Remember the task is: "Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 4 timesteps. Task Initial Information:

frl\_apartment\_kitchen\_utensil\_09\_73 belong\_to\_platform: kitchen\_counter\_1\_body\_platform\_1. The destination platform is frl\_apartment\_rack\_01\_76\_platform\_2.

[5] We also provide an image from your view, with the 9-grid of the platform marked with colorful rectangles and circled numbers. Now you can call\_end, place the object in one part of the grid, or just put it down regardless of the direction. You'll be told if the item is too big to fit in the place you choose.

The current Action space description is as below. It is in the form of a dictionary, where the key represents an executable action, and the value describes the details of that action.

Action space description: '0': 'call\_end', '1': 'place the object in the corresponding grid 1', '2': 'place the object in the corresponding grid 2', '3': 'place the object in the corresponding grid 3', '4': 'place the object in the corresponding grid 4', '5': 'place the object in the corresponding grid 5', '6': 'place the object in the corresponding grid 6', '7': 'place the object in the corresponding grid 7', '8': 'place the object in the corresponding grid 8', '9': 'place the object in the corresponding grid 9', '999': 'place the object around the platform'You need to select an action from this current action space. Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character

format will directly cause the entire task to fail.

VLM response: 5

User Message: Now you may see the objects on the platform frl\_apartment\_rack\_01\_76\_platform\_2.Or you may not see them clearly due to bad starting direction.Remember the task is: "Move frl\_apartment\_kitchen\_utensil\_09\_73 to frl\_apartment\_rack\_01\_76\_platform\_2's front part". Please note that you have a time limit. There are a total timelimit of 15 timesteps, and you have already used 5 timesteps.

TaskInitialInformation:frl\_apartment\_kitchen\_utensil\_09\_73belong\_to\_platform:kitchen\_counter\_1\_body\_platform\_1.Thedestinationplatformisfrl\_apartment\_rack\_01\_76\_platform\_2.Frl\_apartment\_rack\_01\_76\_platformfrl\_apartment\_rack\_01\_76\_platformfrl\_apartment\_rack\_01\_76\_platform

[6] We also provide an image from your view, with all objects you may pick up are marked with grey shadowed cube and circled numbers. Now you can call\_end, rotate clockwisely to another freespace, go to another platform or choose to pick up one item here.

The current Action space description is as below. It is in the form of a dictionary, where the key represents an executable action, and the value describes the details of that action.

Action space description:'-1': "rotate clockwisely to the nearest freespace that can fit you, the robot. In case there's no any other freespace, you'll stay in the same position.", '0': 'call\_end', 'o1': 'pick up object 1', 'l1': 'go to platform 1: frl\_apartment\_sofa\_10\_platform\_0', 'l2': 'go to platform 2: frl\_apartment\_sofa\_10\_platform\_1', '13': 'go to platform 3: frl\_apartment\_sofa\_10\_platform\_2', 'l4': 'go to platform 4: frl\_apartment\_chair\_04\_46\_platform\_0', '15': 'go platform 5: frl\_apartment\_chair\_04\_47\_platform\_0', '16': 'go to platform frl\_apartment\_chair\_05\_7\_platform\_0', to 6: '17': 'go to platform 7: frl\_apartment\_chair\_05\_8\_platform\_0', '18': plat-'go to '19': 8: frl\_apartment\_stool\_02\_18\_platform\_0', form 'go to platform 9: frl\_apartment\_stool\_02\_6\_platform\_0', '110': plat-'go to frl\_apartment\_rack\_01\_76\_platform\_0', form 10: '111': 'go to platform 11: frl apartment rack 01 76 platform 1', '112': 'go platto form 12: frl\_apartment\_rack\_01\_76\_platform\_2', '113': 'go to plat-'114': form 13: kitchen counter 1 body platform 0', 'go to platform 14: kitchen\_counter\_1\_body\_platform\_1', '115': 'go to platform 15: fridge\_0\_body\_platform\_0', go to platform 16: fridge 0 body platform 1', '117': '116': 'go to platform 17: fridge\_0\_body\_platform\_2', '118': 'go to platform 18: fridge\_0\_body\_platform\_3', '119': 'go to platform 19: fridge\_0\_body\_platform\_4', 'l20': 'go to platform 20: fridge\_0\_body\_platform\_5', '121': 'go to platform 21: fridge\_0\_body\_platform\_6', '122': 'go to platform 22: fridge 0 body platform 7', '123': 'go to platform 23: frl apartment bin 02 2 platform 0', '124': 'go to platform 24: frl\_apartment\_bin\_03\_3\_platform\_0', '125': 'go to 25: platform frl\_apartment\_wall\_cabinet\_01\_4\_platform\_0', '126': 'go to platform 26: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_1', '127': 'go to platform 27: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_2', '128': 'go to platform 28: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_3', '129': to 'go plat-29: frl apartment wall cabinet 01 4 platform 4', '130': form 'go to plat-30: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_5', form '131': 'go to plat-'132': form 31: frl\_apartment\_wall\_cabinet\_01\_4\_platform\_6', 'go platto form 32: frl\_apartment\_table\_03\_14\_platform\_0', '133': 'go to platform 33: frl\_apartment\_table\_03\_14\_platform\_1', '134': 'go to platform 34: frl\_apartment\_table\_01\_48\_platform\_0', '135': to platform 'go 35: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_0', '136': 'go to plat-

form 36: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_1', '137': 'go to platform 37: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_2', '138': 'go to platform 38: frl apartment wall cabinet 02 21 platform 3', '139': plat-'go to 39: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_4', form '140': 'go to platform 40: frl apartment wall cabinet 02 21 platform 5', 'l41': plat-'go to form 41: frl\_apartment\_wall\_cabinet\_02\_21\_platform\_6', '142': 'go to 42: frl\_apartment\_chair\_01\_15\_platform\_0', platform '143': 'go platto form 43: frl\_apartment\_chair\_01\_15\_platform\_1', '144': 'go to platform 44: frl apartment chair 01 16 platform 0', platform '145': 'go to 45: frl\_apartment\_chair\_01\_16\_platform\_1', '146': 'go to platform 46: chestOfDrawers\_01\_2\_body\_platform\_0', 'l47': 'go to platform 47: chestOfDrawers\_01\_2\_body\_platform\_1', 'go to platform 48: '148': chestOfDrawers\_01\_2\_body\_platform\_2', '149': 'go to platform 49: chestOfDrawers\_01\_2\_body\_platform\_3', '150': 'go to platchestOfDrawers\_01\_2\_body\_platform\_4', form 50: '151': 'go to platform 51: chestOfDrawers\_01\_2\_body\_platform\_5', '152': 'go to platform 52: chestOfDrawers\_01\_2\_body\_platform\_6', '153': 'go to platform 53: frl\_apartment\_table\_02\_40\_platform\_0', 'go to platform 54: frl apartment table 02 40 platform 1', '155': '154': 'go to platform 55: frl\_apartment\_table\_02\_40\_platform\_2', '156': 'go to platform 56: frl\_apartment\_tvstand\_89\_platform\_0', '157': 'go to platform 57: cabinet\_3\_body\_platform\_0', '158': 'go to platform 58: cabinet\_3\_body\_platform\_1'You need to select an action from this current action space. Note: Please output only the corresponding action key (e.g., 110) without any additional output. Only output one current action. Outputting extra characters or using an incorrect character format will directly cause the entire task to fail.

VLM response: 0

(call\_ended)

Checker Information: Target object placed in correct direction. Task\_status:Success