

Tidyboy-OPL: RoboCup@Home Open Platform League Team Description Paper

Taewoong Kang¹, Joonyoung Kim¹, Jaebong Yi¹, Shady Nasrat¹,
Min-Seong Jo¹, Min-Seong Jo¹, Jae-Moon Park², Tae-Young Kim²,
Byeong-Ju Kim³, Seung-Joon Yi¹, Byoung-Tak Zhang^{2,3}

¹Department of Electrical Engineering, Pusan National University, Busan, Korea

²Department of Computer Science, Seoul National University, Seoul, Korea

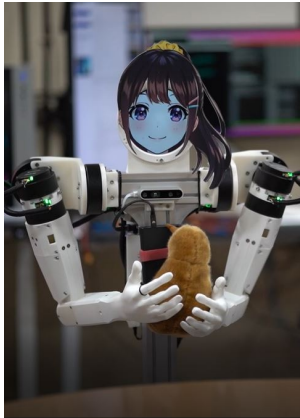
³Department of Mechanical Engineering, Seoul National University, Seoul, Korea

Corresponding email to: seungjoon.yi@pusan.ac.kr

Abstract. Team Tidyboy-OPL is a joint RoboCup@Home Open Platform League (OPL) team that consists of members from Pusan National University and Seoul National University. We first started as a Domestic Standard Platform League (DSPL) team using the Toyota HSR platform, participating in the RoboCup@Home DSPL and World Robot Summit (WRS) competitions over years. Since 2023, we have been also participating in the RoboCup@OPL to further leverage the modular nature of our service robot software framework. In this paper, we present the updated Lucio platform we have developed for the RoboCup@Home OPL, and how our modular service robot software framework can be seamlessly used on the new hardware platform, with help of modern deep learning-based machine learning algorithms.

1 Introduction

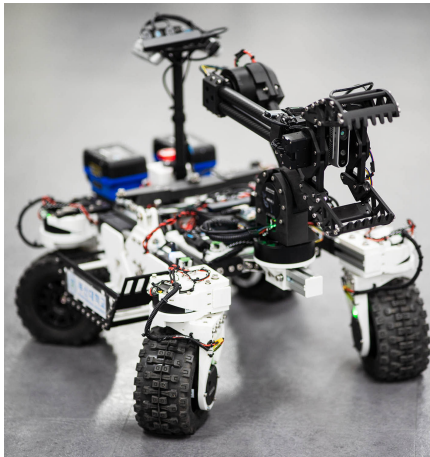
Team Tidyboy-OPL consists of members from Pusan National University and Seoul National University, jointly participating in the RoboCup@Home leagues. Originally formed as the RoboCup@Home DSPL team Tidyboy, we've actively competed in several international service robotic competitions using the Toyota HSR platform with notable success, including victories in the RoboCup@Home Virtual 2021 DSPL and the World Robot Summit 2020 Partner Robot Challenge [Real Space] remote participation league. Since 2023, we have strategically branched into OPL and DSPL teams, to further leverage our extensive experience with designing and building a wide range of custom robotic hardware. This strategy led us to win first place in both OPL and DSPL leagues in the RoboCup@Home 2023, which demonstrates our team's expertise in both machine learning and robotics research. In this paper, we describe how we are updating our hardware platform, Lucio, for the upcoming RoboCup@Home 2024, also we present how we are incorporating recent advancements in machine learning algorithms to further improve its software.



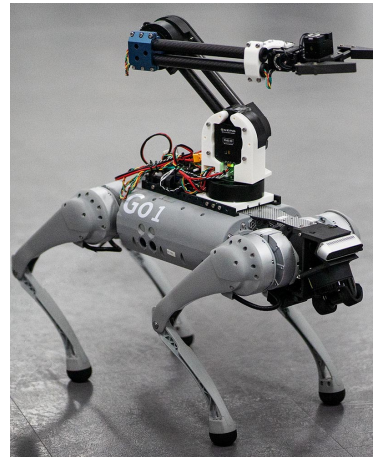
(a) JF2



(b) BM-1



(c) MULE



(d) Armed GO1

Fig. 1: Various custom mobile service robots we have recently developed.

2 Lucio Platform

Since our first participation in RoboCup@Home in 2018, we have developed various types of custom mobile service robots that can autonomously perform RoboCup@Home-type tasks, as shown in Fig.1, with diverse locomotion methods according to own design goals [1, 6]. The Lucio platform, on the other hand, is specially designed to suit the indoor RoboCup@Home environment. In this section, we will present detailed hardware features of the Lucio platform.

2.1 Mobile Base

One of the advantages of the HSR platform over other OPL robots is its smaller footprint. In many circumstances, teams struggle to let their robot simply move around the environment, which may include smaller-than-typical door openings.



Fig. 2: Size Comparison of the Lucio robot and the Toyota HSR

Also, HSR has a circular base with an omnidirectional drivetrain, which makes motion planning in cramped spaces much easier than other robots with either a square footprint or a differential drivetrain.

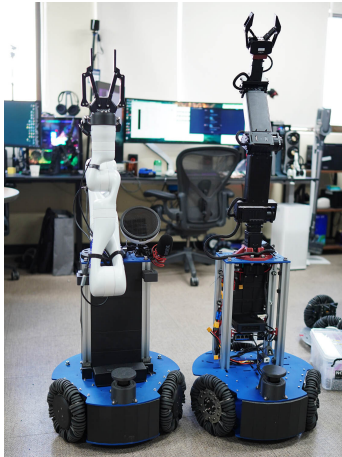
For those reasons, we have chosen a circular base with three large Omni wheels for the Lucio platform. We have decided to use the base size of 501mm, which is 16% larger than the HSR platform as shown in Fig.2, which can keep the robot stable even when the arm is fully extended forward. During the RoboCup@2023 competition, we found the base to be fast, reliable, and trouble-free.

2.2 Arm and Gripper

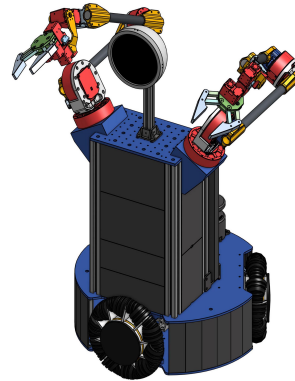
For the RoboCup@Home 2023, we have used the Kinova Gen 3 arm and Robotiq 2F gripper for the manipulator setup. The Kinova arm has a large number of advantages for lightweight home service robot applications, which include a long reach, low weight, high efficiency, and the lack of a separate controller box. Thanks to the long reach of the arm, Lucio does not require a torso lift joint to reach the objects from the ground to the 1.5 meter high. The Kinova arm does have a few disadvantages, which include a high price tag and limited joint range that makes it hard to fold the arm neatly. Currently, we are testing the feasibility of other commercial manipulators, as well as custom-built torque-controlled compliant manipulators on the Lucio base for the RoboCup@Home tasks, as shown in Fig.3.

2.3 Computing Units

For the RoboCup@Home 2023, we have used a NUC small computer inside the robot body that manages the autonomy, localization, base path planning, and



(a) Single arm configuration



(b) Dual arm configuration

Fig. 3: Possible arm configurations for Lucio Platform



Fig. 4: Remote Controller based on the Steam Deck Portable

arm path planning, and an external GPU-equipped backpack laptop that handles all the GPU-required perception tasks. In addition, we started to use the Steam Deck portable as an integrated robot controller since the RoboCup@Home 2023. We use the device to monitor the status of mapping, localization, and object recognition in real-time, as well as to control the robot using the integrated joystick, triggers and buttons, and touch screen as in Fig.4. For the RoboCup@Home 2024, we plan to integrate NVIDIA Jetson embedded computing devices to replace the backpack laptop, which is very power-hungry and requires the hassle of separate charging.

3 Software Framework

We use a modular robot software framework that supports a variety of different robotic hardware including humanoid robots [8], quadrupeds [3], stationary



Fig. 5: Simulated Lucio robot the in Isaac Sim simulator

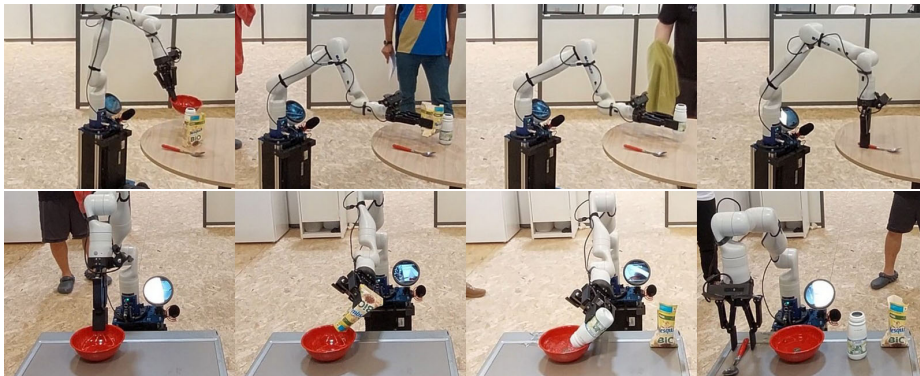


Fig. 6: Lucio performing the Serve Breakfast task at the RoboCup@Home 2023.

robotic manipulators [2], and wheeled service robots [6]. Thanks to the modularity of the framework, most of the code can be shared between the HSR and Lucio platforms except the arm motion controller. Our software framework supports the Gazebo and Webots open-source simulators, and we are currently working to integrate the Isaac Sim simulator as well, as shown in Fig.5.

3.1 Arm motion planning

Our arm motion planning module uses platform-specific motion primitives to reach objects at different heights and approach angles. As Lucio has a longer, high degree of freedom manipulator compared to the HSR platform, it requires fewer motion primitives to perform all the manipulation tasks required for the RoboCup@Home competitions. Fig.6 shows the Lucio robot performing the manipulation-focused Serve Breakfast task. The motion planning module can be straightforwardly used for different manipulators with similar kinematics. In



Fig. 7: Bimanual manipulation based on the imitation learning

in addition to the motion primitive-based arm planning module, we are working on an imitation learning-based arm motion controller as shown in Fig.7, which learns task-specific arm motions from human demonstration provided by real-time teleoperation controller.

3.2 Natural Language Processing

Natural language processing (NLP) takes a very important role for RoboCup@Home Tasks, and correctly understanding long, compound sentences is crucial for General Purpose Service Robot (GPSR) tasks. In our previous approaches, we relied on classic string parsing methods, which often fell short of fully grasping the underlying meaning of sentences, especially when dealing with rephrased or grammatically restructured sentences. As a solution, we have adopted a new approach using Large Language Models (LLM), which is proven to be more robust as it comprehends the contextual nuances within sentences.

We are working on a preliminary LLM-based NLP module, which takes command sentences and generates a series of robot commands. We have built the LLM training data generator based on the RoboCup GPSR and EGPSSR dataset generator. We adjusted the generator to produce training pairs that include both user input and the corresponding LLM response. We created a dataset containing 60,000 user instructions paired with robot responses, wherein the robot responses consist of the ordered actions the robot is capable of performing. We incorporate the Parameter Efficient Fine-Tuning (PEFT) [4] method to ensure efficient training within a one-hour timeframe on a single Nvidia RTX 3090 GPU.

LLM Response	Note
<pre>User: Go to the kitchen and ask James what the time is? Lucio response: Respond('Okay, I will ask him') Move_to('kitchen') Search('James') Respond('Hey James, What is the time?') </s></pre>	<p>Navigating to a specified location and interacting with an individual to inquire data.</p>
<pre>User: Ay lucio Get m tuna can from the ketchen table Lucio response: Respond('Got it i will bring it to you!') Move_to('kitchen table') Search('tuna can') Pickup('tuna can') Give_user() </s></pre>	<p>Handling messy and misspelled inputs.</p>
<pre>User: Retrieve the bowl from the dining room, hand it to charlie he is sitting at the corridor. Lucio response: Respond('Got it i will bring it to him!') Move_to('dining room') Search('bowl') Pickup('bowl') Move_to('corridor') Search('charlie') Give('charlie') </s></pre>	<p>Executing sequential actions, including retrieving an item from one location and delivering it to a specific individual at another.</p>
<pre>User: Search for shady at the entrance and guide him to the bedroom, and bring me knife from the kitchen Lucio response: Respond('Okay, am working on it') Move_to('entrance') Search('shady') Respond('Hey shady, am here to guide you to bedroom , please follow me') Move_to('bedroom') Search('knife') Pickup('knife') Give_user() </s></pre>	<p>Managing identical action structures for rephrased input from the above example.</p>

Table 1: Examples of LLM-based natural language processing.

Table 1 shows some results from the current NLP module, where the module can correctly understand the input sentence and generate the correct sequence of actions for the robot, even if the input sentence is rephrased or has misspelled words.

4 Conclusion

Team Tidyboy-OPL, a collaborative effort between Pusan National University and Seoul National University, has been focusing on combined machine learning advancements with robotic research for home service robot applications. Building on past successes in events like the RoboCup@Home Virtual 2021 DSPL and the World Robot Summit 2020 Partner Robot Challenge, the team Tidyboy has successfully expanded to OPL branch, where the team secured two victories

in the RoboCup@Home 2023 leagues. In preparation for this year’s competition, we are working on incremental improvements to our successful hardware and software modules, incorporating state-of-the-art LMM-based natural language processing, imitation learning-based arm motion control, and on-device artificial intelligence. Furthermore, we have embraced open-source principles by sharing most of their previous codes and datasets, encompassing various competitions such as RoboCup robotic soccer, Turtlebot Autorace autonomous driving, RoboCup@Home DSPL 2019, and WRS 2020. We plan to continue this contribution to the RoboCup@Home league by releasing their codes and data after the competition.

References

1. Kang, T., Kim, J., Song, D., Kim, T., Yi, S.J.: Design and control of a service robot with modular cargo spaces. In: 2021 18th International Conference on Ubiquitous Robots (UR). pp. 595–600 (2021)
2. Kang, T., Yi, J.B., Song, D., Yi, S.J.: High-speed autonomous robotic assembly using in-hand manipulation and re-grasping. *Applied Sciences* 11(1) (2021)
3. Kim, J., Kang, T., Song, D., Yi, S.J.: Design and control of a open-source, low cost, 3d printed dynamic quadruped robot. *Applied Sciences* 11(9) (2021)
4. Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., Raffel, C.: Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning (2022)
5. McGill, S.G., Yi, S.J., Lee, D.D.: Low dimensional human preference tracking for motion optimization. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 2867–2872 (May 2016)
6. Yi, J.B., Kang, T., Song, D., Yi, S.J.: Unified software platform for intelligent home service robots. *Applied Sciences* 10(17) (2020), <https://www.mdpi.com/2076-3417/10/17/5874>
7. Yi, S.J., McGill, S., Hong, D., Lee, D.: Hierarchical motion control for a team of humanoid soccer robots. *International Journal of Advanced Robotic Systems* 13(1), 32 (2016)
8. Yi, S.J., McGill, S., Jeong, H., Huh, J., Missura, M., Yi, H., Ahn, M.S., Cho, S., Liu, K., Hong, D., Lee, D.D.: RoboCup 2015 Humanoid AdultSize League Winner. In: Almeida, L., Ji, J., Steinbauer, G., Luke, S. (eds.) *RoboCup 2015: Robot World Cup XIX*. pp. 132–143 (2015)

Robot Lucio mk2 Hardware Description

Specifications of the Lucio mk2 robot are as follows:

- Base: Custom base with 3 206mm omniwheels
- Base footprint: 501 mm diameter
- Height: 1050 mm (minimum)
- Weight: 30 kg
- Arm: Kinova 3rd gen 7DOF arm
- Arm length: 902 mm
- Arm payload: 4kg
- Gripper: Robotiq 2F-140
- Head: Custom head with circular LCD display
- Battery: Cobalt 24V Li-ion battery
- Computing Unit: Intel NUC and NVIDIA Jetson

Also our robot incorporates the following sensors:

- Asus Xtion Pro Live RGBD camera
- Orbbec Astra Mini RGBD camera
- Hokuyo URG-10LX LIDAR
- Rode microphone

Robot's Software Description

For our robot we are using the following software:

- OS: Ubuntu 20.04
- Middleware: ROS Noetic and in-house codebase
- Localization and Mapping: `hector_slam`, `amcl` and ICP based in-house algorithm
- Arm control: In-house arm motion planner [5]
- Navigation: In-house hierarchical motion planner [7]
- Object recognition: YOLOv8
- Pose estimation: YOLOv8 Pose Estimation
- Pose estimation: YOLOv8 Pose Estimation

External Devices

Our robot relies on the following external hardware:

- Backpack Laptop: Intel i7 CPU, 32GB RAM, NVIDIA 2080 GPU

Cloud Services

Our robot connects the following cloud services:

- Speech recognition: Google Cloud API
- Image recognition: Kairos API

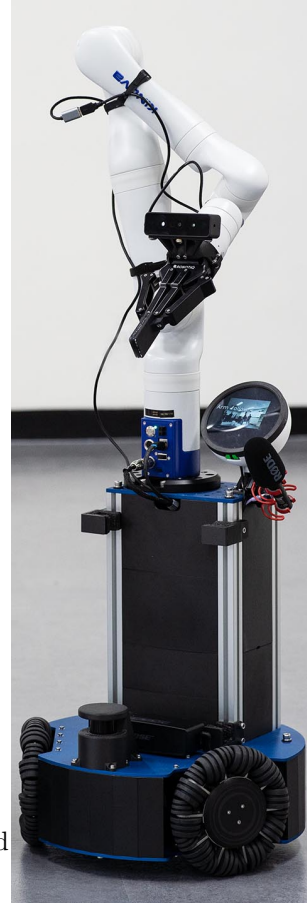


Fig. 8: Lucio mk2 Robot