Inha-United 2026 Team Description Paper

Minho Lee[1], Dongjin Cho[1], Minho Lee[1], Jungtae Kim[1],
Gunwoo Park[1], Jiyun Kim[1], Jihyun Han[1], Sanghyun Lee[1], Wonhyuk Jung[1],
Yonggun Cho[1,2], Inwook Shim[1,3], Junwoo Jang[1,4], Woojin Ahn[1,5]

[1]Inha University, Incheon, Korea,
[2]SPARO Lab.,[3]RCV Lab.,[4]Artemis Lab.,[5]RILS Lab.
Corresponding Email to:
yg.cho@inha.ac.kr, **iwshim@inha.ac.kr**,
junwoo@inha.ac.kr, **wjahn@inha.ac.kr**
`https://inha-united.github.io/Home2026/`

November 30, 2025

Abstract. Team Inha-United is a research-driven group formed by four robotics laboratories at Inha University. We integrate our strengths in SLAM, robot learning, multimodal perception, and large language models into a unified system capable of operating in dynamic domestic environments. Our goal is to build a robust and adaptable service-robot system for the RoboCup@Home Open Platform League. Using the RB-Y1 humanoid robot, our system combines a hybrid 2D–3D mapping framework with semantic spatial reasoning, reliable object manipulation for both known and unknown items, and an LLM-guided Behavior Trees for consistent task execution and human–robot interaction. These capabilities support core RoboCup@Home tasks such as reception, person guidance, and object delivery. Through our participation, we demonstrate the robustness and adaptability of our system in realistic domestic environments and contribute reproducible methodologies to the community.

1 Introduction

Our team, Inha-United, is a multi-lab research group at Inha University with diverse academic backgrounds. We have gained substantial experience across different robotic domains through a wide range of projects using various robot platforms equipped with diverse sensing modalities, as shown in Fig. 1. We have also participated in multiple international competitions listed in Table 1 as opportunities to validate our technical skills and acquire practical insights, and have received the awards indicated. Building on this foundation, we aim to realize reliable and versatile robot systems capable of performing various tasks in real-world environments and interacting naturally with humans. In this paper, we present how we integrate our modular framework into our system and outline the key components developed for the upcoming 2026 RoboCup@Home OPL competition.

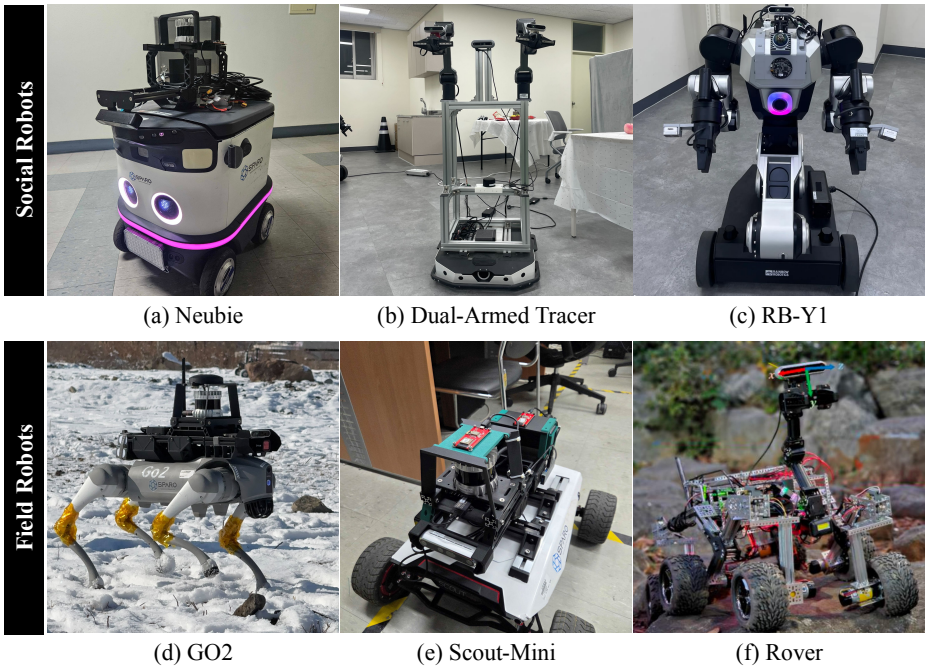


Fig. 1. Mobile robot platforms with diverse sensing modalities in our research.

2 Approach

2.1 System Overview

Hardware Setup: Because our team's goal is to develop a general-purpose home-service framework, we use RB-Y1[1], a commercial platform that can be deployed in various indoor environments. RB-Y1 consists of an upper body with dual arms and a height-adjustable torso, mounted on a two-wheeled differential-drive mobile base. The main control computer is the Jetson AGX Orin Module, and an additional Jetson Thor is installed for perception workloads. For home-service tasks, we mount a Realsense D435f camera on the robot's head, which is used by our visual perception algorithms and vision-language model (VLM). To support manipulation tasks, each arm is also equipped with a Realsense D405 camera for object pose estimation and grasp execution. A Livox MID-360 LiDAR is attached to the body and is mainly used for precise depth sensing, which enables accurate environmental perception. A more detailed description is given in Appendix A.

[1] `https://www.rainbow-robotics.com/rby1`

Table 1. Results of competitions

Competitions	Results
DARPA Robotics Challenge Final'15	1^{st} Place
ICRA'24, Workshop on Construction Robots	Best Research Award
Virtual RobotX Challenge'19	1^{st} Place

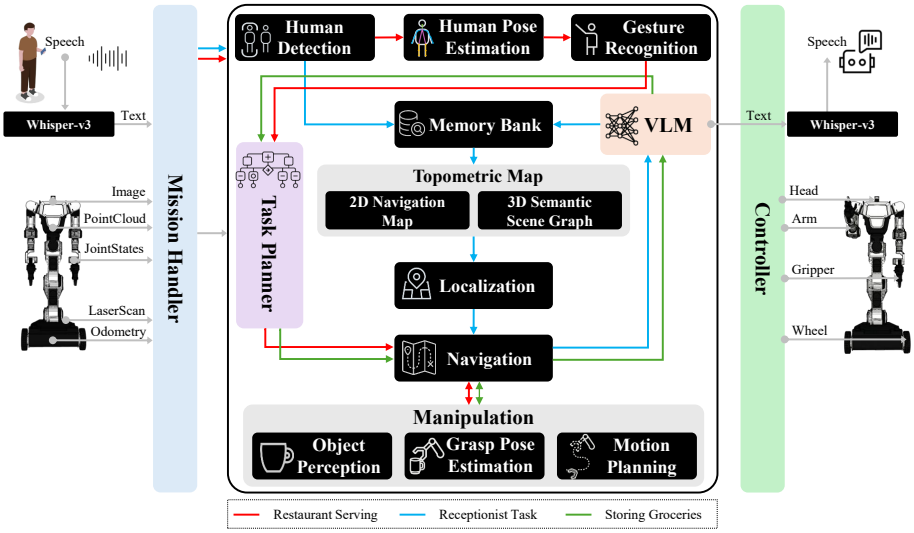
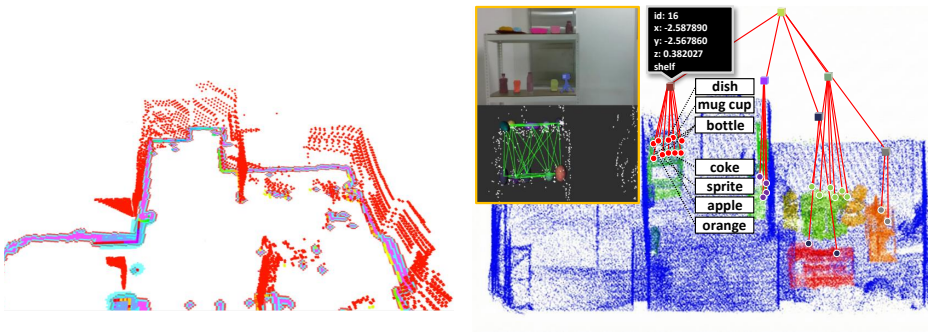


Fig. 2. The architecture of our system pipeline for Robocup@Home tasks.

Software Framework: Fig. 2 summarizes the software architecture used to handle the three competition missions. At run time, the Mission Handler receives multimodal inputs (e.g., speech, vision, and joint state), infers which mission is requested, and translates it into a set of functional requirements. These requirements are dispatched to three core capability stacks: Mapping and Navigation (Section 2.2), Manipulation (Section 2.3), and Human–Robot Interaction (Section 2.4). Each mission is realized by combining reusable modules from these stacks, rather than designing a separate pipeline for every task.

2.2 Mapping and Navigation

Topometric Map Construction: RoboCup@Home environments often include cluttered furniture layouts and dynamic human motion, which can degrade the performance of conventional 2D SLAM. To improve robustness, our system employs a hybrid 2D–3D topometric representation that integrates geometric



(a) 2D Navigation Map (b) 3D Semantic Scene Graph

Fig. 3. Visualization of the 2D navigation map and the 3D semantic map.

occupancy with object-level semantics. As shown in Fig. 3, our mapping module combines a 2D navigation map and a 3D semantic scene graph. The 2D navigation map is derived from the LiDAR point cloud by projecting height-based occupancy onto a 2D grid, providing an efficient costmap for global and local planning. This approach preserves the speed of classical 2D planners while leveraging 3D structure to enhance safety in narrow, cluttered indoor spaces. The 3D semantic scene graph is constructed by applying open-vocabulary object detection and segmentation to RGB-D images. By maintaining semantic consistency across objects and rooms, the robot can perform object-goal navigation and scene-aware decision-making.

Navigation: Our navigation module leverages a 2D navigation approach for the robot equipped with a 3D LiDAR sensor. Using the cost map generated in the previous stage, the system computes the global path with the A* algorithm. To handle cluttered and dynamically changing environments, we employ a DWB local planner that generates and updates the robot's trajectory based on newly received sensor data while following the global path. This allows the robot to avoid dynamic obstacles and maintain safe navigation.

2.3 Manipulation

Object Perception: To enable reliable grasp planning, our system explicitly distinguishes between known and unknown objects, then estimates their 6-DoF poses separately. The robot first collects multi-view images from the head and hand-eye cameras and performs object recognition on each view. By analyzing the uncertainty of the recognition network, the system determines whether an object belongs to a known category or should be treated as an unknown object. In the case of known objects, we leverage FoundationPose [1], which supports

model-based 6-DoF pose estimation by aligning the detected object with its corresponding CAD model. For unknown objects, we reconstruct the object's 3D geometry by aggregating multi-view depth observations over multiple frames and subsequently estimate its pose in the scene.

Grasp and Motion Planning: Given the estimated object pose, the system computes an appropriate grasp pose and generates a collision-free motion plan to physically execute the grasp. For known objects, the grasp pose is determined by retrieving a pre-defined optimal configuration linked to the corresponding 3D CAD model. For unknown objects, we employ the GraspGen framework [2] to infer a feasible grasp pose directly from the reconstructed object geometry. The resulting grasp pose is then provided to the motion planning framework (MoveIt[2]), which integrates the robot's current perception of the 3D scene to generate a collision-free trajectory that guides the end-effector to the target pose.

2.4 Human-Robot Interaction

Human Perception: As a foundation for human-robot interaction, the system incorporates a human perception module that enables the robot to recognize, remember, and interpret human users. Using InsightFace[3], the robot stores user identities in a memory bank to recognize previously encountered individuals and support personalized interaction. In addition, human pose information is extracted using the MediaPipe framework [3] and interpreted as gesture cues, enabling the robot to infer both user actions and implicit intent.

Natural Language Understanding: To support diverse human-robot interaction tasks involving natural language, such as interactive dialogue and instruction-following, we leverage foundation models, Whisper-v3 [4] for Automatic Speech Recognition (ASR) and Gemma3 [5] for Natural Language Understanding (NLU). Rather than maintaining separate fine-tuned models for each downstream task, the system introduces a task manager that applies task-specific prompts and routes the parsed inputs accordingly, enabling a single language model to efficiently support multiple interaction scenarios.

Context-Aware Decision-Making: Our decision-making module interprets user intent and environmental context to generate structured high-level commands. These commands are combined with a Behavior Tree (BT), which provides a reliable, interpretable, and modular high-level control structure that organizes task execution through sequence, fallback, and parallel nodes. This integration leverages the stability of BT-based control and the flexibility of LLM-driven reasoning, enabling the robot to adapt to diverse tasks while maintaining predictable and context-aware behavior.

[2] `https://moveit.ai/`
[3] `https://insightface.ai`

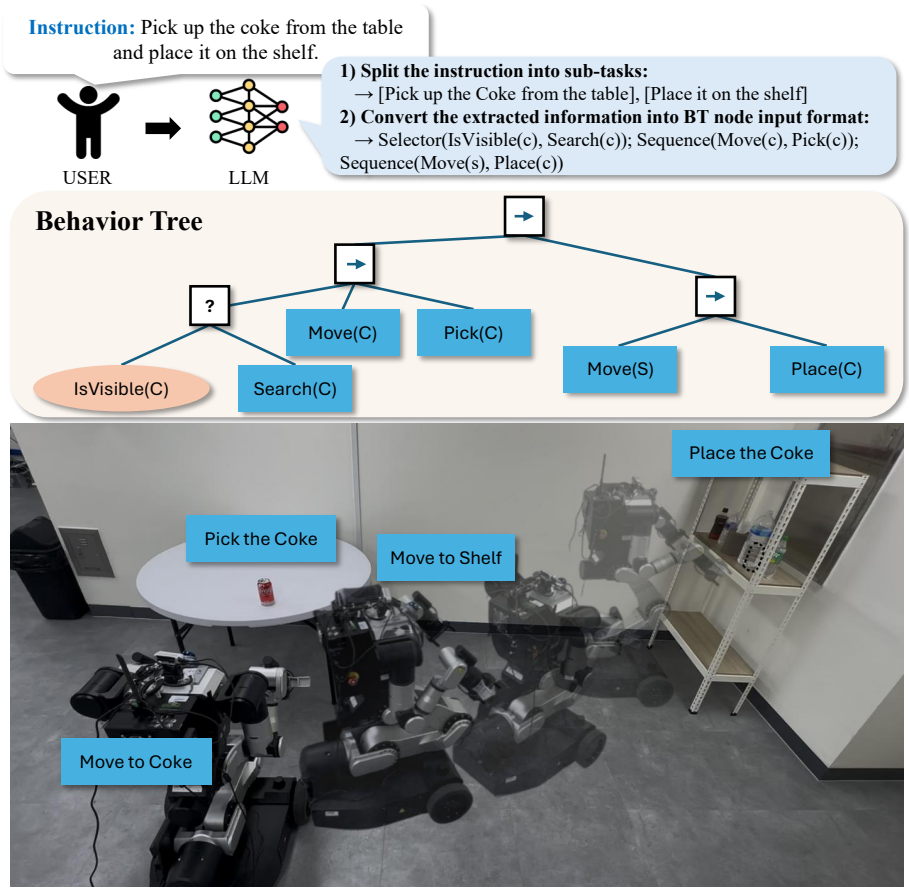
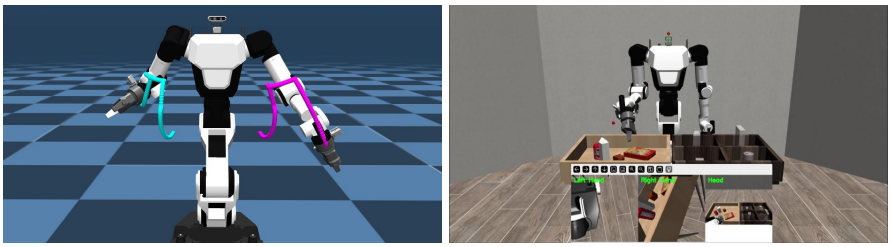


Fig. 4. The process of LLM-based task parsing to Behavior Tree node mapping.

2.5 Simulation and Sim2Real

Simulated Environments: We built simulation environments in MuJoCo[4] and Isaac Sim[5] (see Fig. 5). MuJoCo enables fast controller prototyping, while Isaac Sim provides high-fidelity physics for testing complex interactions. These environments allow us to evaluate navigation and manipulation behaviors under varied lighting, layouts, and occlusions before deploying to the RB-Y1. By iterating in simulation first, we can quickly identify failure cases and refine robot policies prior to real-world testing.

[4] `https://mujoco.org/`
[5] `https://developer.nvidia.com/isaac`



(a) Simulated RB-Y1 robot in the MuJoCo simulator

(b) Simulated RB-Y1 robot in the Isaac simulator

Fig. 5. Simulated environments for the RB-Y1 in MuJoCo and Isaac Sim.

Sim2Real Transfer: To facilitate Sim2Real transfer, we aim to minimize the reality gap at both the simulation design and policy learning stages. The simulation environment is first constructed to closely match the real world, and policies are initialized through training in simulation. Domain randomization is applied to both visual and physical properties, including lighting, textures, layouts, sensor noise, friction, mass, and collision parameters, to expose the policy to a wide range of operating conditions. Given the close alignment between the simulation and the real-world setup, we further apply domain adaptation using real-world demonstration data. This adaptation is performed through two complementary strategies: (i) policy-level fine-tuning using real-world demonstrations, and (ii) learning domain-invariant feature representations that provide robust feedback to the simulation-trained policy. Specifically, the policy is trained to prioritize task-relevant geometric cues over low-level pixel variations, enabling stable generalization across both simulated and real environments.

3 Contribution

Our system makes three main scientific contributions. First, we propose a hybrid mapping approach to efficiently capture both immediate and accumulated environmental changes that arise in real-world scenarios. Second, we introduce an adaptive dual-arm manipulation that leverages LLM-guided reasoning to select

the appropriate arm based on task demands and geometric constraints, overcoming the limitations of typical single-arm systems. Third, we integrate an LLM-based task planner with Behavior Trees, enabling robust, interpretable, and context-aware task execution for general-purpose service robots.

4 Conclusion

In this paper, we presented the overall design of our home-service robot system, including the RB-Y1 platform, our modular software framework, and the core capabilities in mapping, navigation, manipulation, and human–robot interaction. We developed a hybrid 2D–3D topometric mapping pipeline, an adaptive dual-arm manipulation system guided by an LLM-based task planner, and a unified behavior-tree execution structure suitable for the three main missions of the 2026 RoboCup@Home competition. Our system has been validated in a series of indoor scenarios, showing reliable navigation in frequently changing layouts, consistent shelf and tabletop manipulation tasks, and successful language instructions through our LLM–BT pipeline. Nevertheless, several challenges remain, including efficient map update, the latency in LLM-based reasoning, and the need for further refinement of dual-arm reaching performance. We are currently addressing these limitations and integrating the remaining modules required for full-mission execution. We expect to demonstrate robust and flexible performance at RoboCup@Home 2026.

References

1. Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.
2. Adithyavairavan Murali, Balakumar Sundaralingam, Yu-Wei Chao, Wentao Yuan, Jun Yamada, Mark Carlson, Fabio Ramos, Stan Birchfield, Dieter Fox, and Clemens Eppner. Graspgen: A diffusion-based framework for 6-dof grasping with on-generator training. *arXiv preprint arXiv:2507.13097*, 2025.
3. Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019.
4. Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023.
5. Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Inha-United@Home RB-Y1 Robot Hardware Description

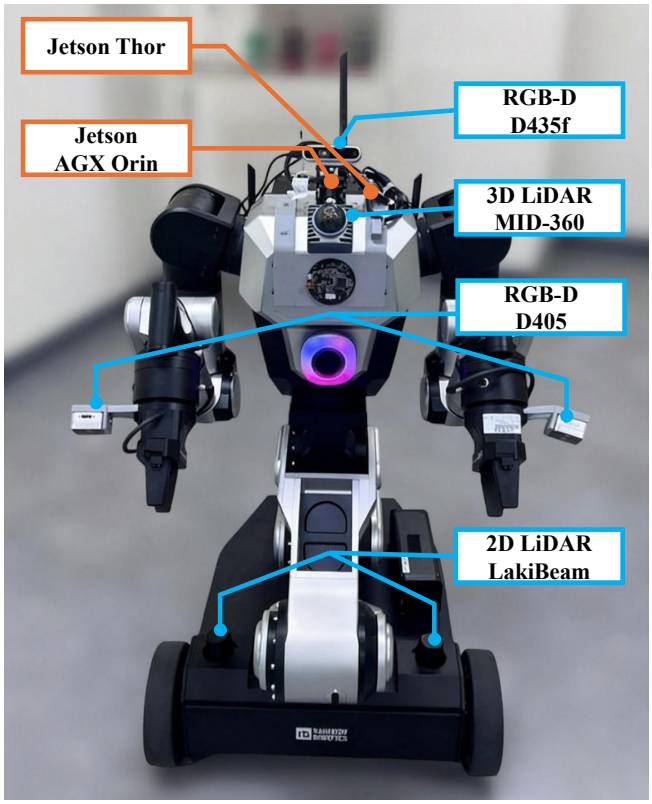


Fig. 6. Hardware configuration of the RB-Y1 platform. Internal computing units are located in a backpack at the rear of the torso. Camera modules are attached using custom-designed mounts, and 2D LiDARs on the mobile base are merged to expand sensing coverage for navigation.

Our robot platform, Rainbow Robotics' RB-Y1, is a general-purpose service robot equipped with two 7-DoF arms and a 3-DoF leg mounted on a high-speed, wheel-based mobile platform (see Fig. 6). RB-Y1 overcomes the limitations of single-arm collaborative robots and fixed industrial manipulators, enabling repetitive and precise operations across diverse industrial and service domains. We integrate robust mobile navigation, dual-arm manipulation, and multi-sensor perception into the RB-Y1 platform, providing a versatile foundation for a wide range of embodied AI tasks.

Specifications of the RB-Y1 robot are as follows:

- **Base:** 2-wheel differential-drive with rear caster (maximum speed of 1.5 m/s)
- **Base Footprint:** 700 mm
- **Robot Height:** 1470 mm (maximum)
- **Robot Weight:** 131 kg
- **Arm:** Dual 7-DOF
- **Arm length:** 750 mm (maximum)
- **Arm payload:** 3 kg
- **Head:** 2-DoF
- **Torso:** 6-DoF
- **Battery:** 50 V, 25 Ah (1,270 Wh) Li-ion battery.
- **Computing Units:**
  - **Main PC:** NVIDIA Jetson AGX Orin 64 GB
  - **Auxiliary Perception PC:** NVIDIA Jetson Thor 128 GB
  - **Internal Control PC:** UP Xtreme i12 with 12th Gen Intel® Core™ processor

Also our robot incorporates the following sensors:

- **LakiBeam 2D LiDAR** used for 2D SLAM and navigation.
- **Livox MID-360 3D LiDAR** used for 3D perception.
- Head-mounted **Intel RealSense D435f** and wrist-mounted **Intel RealSense D405**
- **ReSpeaker Mic Array v3.0** used for voice acquisition.

Robot's Software Description

For our robot we are using the following software:

- **2D Mapping:** SLAM ToolBox
- **Motion Planning:** MoveIt2
- **Grasp Pose Generation:** GraspGen
- **Object Pose Estimation:** FoundationPose
- **Camera Calibration:** MoveIt Hand-Eye Calibration
- **Automatic Speech Recognition:** Whisper-v3
- **Large Language Model:** Google Gemma3
- **Object Recognition:** Grounded-SAM2
- **Human Pose Detection:** MediaPipe
- **Face Recognition:** InsightFace

External devices

Our robot relies on the following remote high-performance computing resource:

- **Remote High-Performance Computing Server:** Two NVIDIA RTX A6000 GPUs, 256 GB RAM, 8TB SSD
- **Desktop:** Intel Core Ultra 9 285K CPU, 64 GB RAM, 1 TB SSD, NVIDIA RTX 5090 GPU