Software Survey 2026

Team name

HERoEHS

Which division(s) are you applying for? If your used software differs between divisions, please fill out the survey once per division.

Large Size (height < 190 cm, weight < 80 kg)

Is your software fully or partially OpenSource? If so, where can it be found?

Our source code is completely closed source. It is managed privately on GitHub. We have developed most of the code ourselves, while partially utilizing open-source libraries. The open-source libraries we used include BehaviorTree.CPP, microstrain_mips, pinocchio, and OSQP, all of which are documented in the submitted Robot Specification.

Are you using any software developed by other teams? If so, list every component that you are reusing and the team that originally developed it.

We do not use software developed by other teams.

Are you using any datasets in your research? If you are using your own datasets, are they public?

We are using approximately 90,000 data samples, which we collected ourselves, for Object Detection and Segmentation models. This dataset is not publicly available.

Please list the scientific publications your team has made since the last application to RoboCup (or if not applicable in the last 2 years).

Park, S., Noh, D., & Han, J. (2025, May). A Study on Power Consumption of a Humanoid Welcome Robot Using Battery Usage Data. In 2025 22nd International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 1-6). IEEE.

Yong, J., Kim, J. Y., Noh, D., & Han, J. (2025, May). A Study on the Interaction Between Humanoid Robots Equipped with Korean Dataset-Based Gesture Generation AI Models and Korean Users.

In 2025 22nd International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 1-6). IEEE.

Lee, S. H., Noh, D., Kim, J. Y., Yong, J., Kim, J. Y., & Han, J. (2025, May). Enhanced Localization for Bipedal Humanoid Robots in Dynamic and Noisy Environments. In 2025 10th International Conference on Control and Robotics Engineering (ICCRE) (pp. 56-60). IEEE.

Eum, Y., Park, C., Kang, G., Chun, Y., & Han, J. (2024). A Study on the Appearance and Behavioral Patterns of Robots for Fostering Attachment in Users.

Are there any other contributions you would like to share with the RoboCup community?

We have consistently participated in the Main Competition and Technical Challenge of the RoboCup Humanoid AdultSize League for the past seven years, achieving increasingly better results. Our best achievements are 2nd place in the Main Competition and 1st place in the Technical Challenge. Our team leader, Younseal Eum, has been a member of the Organizing Committee since 2023 and contributed to hosting RCAP2023 in Pyeongchang, South Korea. Furthermore, we successfully secured Incheon, South Korea, as the host city for RoboCup2026. Additionally, our team member, Seongjoong Kim, is serving as a member of the Technical Committee.

Which approach are you using to generate the robot walking motion?

We previously employed a position-control-based preview controller for locomotion. Currently, we are developing and benchmarking whole-body model predictive control (MPC) approaches.

Which approach are you using to generate other motions of the robot (e.g. kicking, standing up)?

Previously, kicking motions were generated via inverse kinematics with position commands, enabling adaptive motion generation based on the ball position, while other motions were implemented using predefined sequence-based position commands. Currently, we are extending these approaches toward more generalized and flexible motion generation frameworks for various robot actions.

Do you have a kinematic or dynamic model of your robot? If so, how did you create it (e.g. measure physical robot, export from CAD model)?

We create the kinematic and dynamic model of our robot by measuring from the CAD model. The mass of the components that make up the robot is obtained through the measurement of the actual parts, which is used for dynamic analysis.

What approaches are you using in your robot's visual perception?

We perform vision recognition based on image and point cloud data from the ZED2i. Additionally, we utilize three deep learning models Object Detection, Segmentation, and Classification to recognize objects and perform localization

Are you planning with objects in Cartesian or image space? If you are using Cartesian space, how do you transform between the image space and cartesian space?

We utilize the PointCloud (3D) data obtained from the ZED 2i camera and use a TransformMatrix to calculate the distance between specific objects and the robot. Additionally, based on the distance between the objects and the robot, we transform the coordinates into a global coordinate system for further use.

Do you have some form of active vision (i.e. moving the robots camera based on information known about the world)?

During the game, the robot tracks the ball and searches for it by looking left, right, forward, and downward when the ball is not visible. Additionally, if the recognized field lines and key features are insufficient, the camera is adjusted to better capture these features.

What approach are you using to localize your robot?

We detect field landmarks (e.g., l-points, t-points, and goalposts) for field perception, and generate odometry using a state estimator that fuses IMU and force sensor data. Global localization is performed by combining visual SLAM and a particle filter to estimate the robot's pose on the field.

Is your team performing team communication? Which communication protocol are you using?

We use the standard protocol of the RoboCup Humanoid League and perform team communication to implement various strategic plays.

What approach are you using for navigation? Are you avoiding obstacles?

Our robot generates a path using Bezier curves to navigate to the target location. During the path generation process, if obstacles are detected, the robot creates waypoints to create a new path for avoiding these obstacles. Through this modified path, the robot can effectively avoid obstacles and reach its destination.

How is the behavior of your robots structured? (e.g. Behaviour Trees)

The robot uses the behavior tree method with the "BehaviorTree.CPP" library. It also uses the ROS2 messaging system to command the robot's joints and receive sensor data.

Are you simulating your robot? If so, which simulator are you using and for what purpose do you use simulations?

We use Unity3D, MuJoCo, and Isaac Sim in our simulation pipeline. Unity3D is used to evaluate overall gameplay strategies and multi-agent scenarios, while MuJoCo is used to verify and validate detailed robot motions and dynamics. Isaac Sim is used for large-scale simulation, data generation, and high-fidelity physics validation.

What operating system is running on your robot and which middleware are you using (for example Ubuntu 22.04 and ROS2 Galactic)?

The control PC runs Ubuntu 24.04 with ROS 2 Jazzy, while the inference PC runs Ubuntu 22.04 with ROS 2 Humble.

Is there anything else you would like to share that did not fit any previous question?

None.