# Survey response 10

## Software

| Team Name |
|---|
| Bold Hearts |

| Is your software fully or partially OpenSource. If so, where can it be found: |
|---|
| Our code is partially open source. We open source code when it is entirely stable and tested in competitions. Software can be found: https://gitlab.com/boldhearts/ |

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

| Are you using Inverse Kinematics? If so what solution (analytic, (pseudo)inverse jabcobian, etc...) are you using? |
|---|
| The IKWalk library is a C++ implementation of an open-loop walk engine for small and mid-size humanoid robots. Motor target positions are generated online based on splines in Cartesian space and inverse kinematics. No ZMP is used but several parameters have to be manually tuned. |

| Are you simulating your robot? If so what are you using simulation for? |
|---|
| Yes, we have a model and libraries for the simulation in Webots. We also have some unmaintained integrations in Gazebo. |

| What approach are you using to generate the robot walking motion? |
|---|
| We use the IKWalk library with a ROS2 wrapper. |

| What approach are you using to generate motions for standing up? |
|---|
| We use manual key frame animations. We are exploring at adaptive motions as part of ongoing research. |

| What approach are you using to generate kicking motions? |
|---|
| We use key frame animations, where an appropriate script is selected depending on the scenario. |

| Do you use any other motions than the previously mentioned? If so, what approaches are you using to generate them? |
|---|
| A similar approach is used for all motions, where motions are scripted in Python and frame properties can be dynamically changed based on sensor data. |

| Which datasets are you using in your research? If you are using your own datasets, are they public? |
|---|
| We use a combination of data collected during RC competition (currently not released), and dataset created by the Hamburg Bit-Bots and built by the RC community. |

| What approaches are you using in your robot's visual perception? |
|---|
| We wrap the Darknet framework as a ROS2 node and use the xYOLO network developed by the Electric Sheep team. Additionally, we have improved their image pre-processing and network detection accuracy for a greater number of object detections. |

| 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 currently use image space and decisions are made in the agent frame. |

| How is your robot localizing? |
|---|
| The robot makes decisions in the agent frame, with the ball's location relative to the goal keeper used to determine the side of the field the ball is in. We are actively implementing a custom localisation solution as a ROS2 node. |

| Is your robot planning a path for navigation? Is it avoiding obstacles? How is the plan executed by the robot (e.g. dynamic window approach)? |
|---|
| We detect obstacles (other agents), but do not currently plan routes around them. |

| How is the behavior of your robot's structured (e.g. Behavior Trees)? What additional approaches are you using? |
|---|
| We currently use Behaviour Trees. |

| Do you have some form of active vision (i.e. moving the robots camera based on information known about the world)? |
|---|
| We have a look-around behaviour to search for objects of interest and the robot 'fixates' on objects of interest. |

| Do you apply some form of filtering on the detected objects (e. g. Kalman filter for ball position)? |
|---|
| We have a moving window for detections where a detection ratio threshold must be met to avoid false-positives and false-negatives. |

| Is your team performing team communication? Are you using the standard RoboCup Humanoid League protocol? If not, why (e.g. it is missing something you need)? |
|---|
| We currently use the mitecom protocol but look to migrate to the standard RoboCup Humanoid League protocol. Currently we deploy source code and compile on the robot, and re-compiling protobuf is too slow. |

| Please list contributions your team has made to RoboCup |
|---|
| We ran a Robotics module at the University of Hertfordshire for Bachelors students. Our team members also are part of the RC-HL organisation committees. Since the last two years, we started an annual RC hackathon event for students in both Schools of Engineering and Computer Science for robot design and software solutions. Team members are participating at demos, presentations for students and general public, and providing interview and promoting research related to RoboCup. We have several open-source ROS2 nodes to share with the community. |

| Please list the scientific publications your team has made since the last application to RoboCup (or if not applicable in the last 2 years). |
|---|
| The team is actively involved in editing and contributing to the unique journal project, titled "The human in the loop: perspectives and challenges for robots' behaviours in RoboCup 2050". |

| Please list the approaches, hardware designs, or code your team is using which were developed by other teams. |
|---|
| We are currently using IKWalk Engine from Rhoban team, and our vision is a modified version of xYOLO developed by Electric Sheep. |

| What operating system is running on your robot and which middleware are you using (for example Ubuntu 22.04 and ROS2 Galactic)? |
|---|
| We run Ubuntu 22.10 and ROS 2 humble. |

| Is there anything else you would like to share that did not fit to the previous questions? |
|---|
| N/A |