Software Survey 2025

Team Name

UTRA

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

Fully OpenSource https://github.com/utra-robosoccer/soccerbot

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)?

Yes. The robot was weighed at every joint and link and input into the CAD model

Are you using Inverse Kinematics? If so what solution (analytic, (pseudo)inverse jabcobian, etc...) are you using?

We are using Task-space Inverse Kinematics with constraints from Placo library.

Are you simulating your robot? If so what are you using simulation for?

We use simulation to enhance our workflow to minimise the time development takes. Trying to get the simulation as close to the real life version. Thorough sensor modelling, localisation, computer vision and walking engine testing to at least give us a starting point for tuning and testing in real life. We use Pybullet, Mujoco and webots

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

Placo walk engine that calculates the foot positions based on a LIPM based on the robots urdf and ZMP to minimize disturbances.

What approach are you using to generate motions for standing up?

Static key-frame animations

What approach are you using to generate kicking motions?

Static key-frame animations

Do you use any other motions than the previously mentioned? If so, what approaches are you using to generate them?

Tried using a walking engine developed through RL but have not been able to successfully transfer into the physical domain

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

Bitbot's torso and ball dataset

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

Neural network (yolov8) for object detection and field segmentation. ICP to calcuate odometry from moving field lines.

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?

Cartesian space. For transformation from image to Cartesian space, we use inverse projection mapping to project the pixel to the ground with knoledge of the robots height and cameras rotation from the tf tree.

How is your robot localizing?

Using a UKF that combines measurements from robot movement odometry and visual odometry calculated using field-lines and an ICP algorithm

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)?

Using A* for global planner, using dynamic window approach for local planner, but currently no obstacle avoidance.

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

The behaviour uses py_trees https://github.com/splintered-reality/py_trees

We seperate head and body into seperate trees.

Do you have some form of active vision (i.e. moving the robots

camera based on information known about the world)?

The vision is scanning the enviroment for the ball and following the ball when it sees it.

Do you apply some form of filtering on the detected objects (e. g. Kalman filter for ball position)?

no

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)?

Yes, and Somewhat, it is derived from the standard protocol but a lot of information is removed because it is not necessary

Please list contributions your team has made to RoboCup

N/A

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

N/A

Please list the approaches, hardware designs, or code your team is using which were developed by other teams.

Placo walk engine: https://github.com/rhoban/placo

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

Ubuntu 20.04, ROS1 Noetic

Is there anything else you would like to share that did not fit to the previous questions?

If you have additional materials you would like to show, please link to them here.