Software Survey 2025

Team Name

Blenders FC

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

Our software is fully OpenSource and can be found in https://github.com/Blenders-FC

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, we have a kinematic model that was developed with official measurements of the physical robot and represented as links where the final effector is the foot in order for it to be always parallel to the floor.

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

Yes, we are using Jacobian according to Kajita in the Introduction to Humanoid Robotics book.

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

We are simulating our robots in Gazebo, mainly with the intention to test localization software in ideal conditions and a full field, since we don’t have one. Other secondary uses are for general testing for proofs of concept of robot behavior and algorithms.
We are also simulating the kinematic calculations in MATLAB, to visualize and generate the walking algorithm along with key poses.

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

The robot's walking motion is generated using Kajita’s mathematical model, which models the robot as an inverted pendulum under constraint. This ensures balance by keeping the robot’s center of mass within the support polygon, allowing for stable walking motion.

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

Inverse kinematics is used to define a starting and a final position, these are defined in a function that is called when the IMU surpasses a threshold and in between points are interpolated to create a full standing up motion.

What approach are you using to generate kicking motions?

To generate kick motions we are using the kinematic model by processing key poses in the movement. This allows us to maintain stability and control how far we can kick. Because of an internal controller in the robots' servos, poses can be interpolated without having to provide an exact position for each sample time.

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

In the motion area, we also worked on the robot's walk, where the programming included in the robot is used as a base, but we modified certain parameters to make the walk more stable, according to the type of terrain that we had (artificial grass). We also focused on the way the robot gets up whether it falls forward or backward, and to implement this we based ourselves on Kajita's algorithm in the book "Introduction to humanoid robotics".

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

We are using the dataset called RhobansDetection-V2 for our vision, it can be found in https://universe.roboflow.com/starkit-detection/rhobansdetectionv2

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

We trained a YOLOv4-Tiny computer vision model, That enables the robot to detect soccer balls and goal posts.

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 plan in Cartesian space. The transformation from image space to Cartesian space is achieved using camera calibration parameters and trigonometric functions. First, the

ball’s pixel coordinates are obtained from the image. Then, using the camera’s intrinsic properties and the known height and tilt angles of the camera, we estimate the real-world (X, Y, Z) position of the ball relative to the robot.

How is your robot localizing?

Our robot is currently undergoing localization testing using different algorithms. We are evaluating the performance of ORB-SLAM and VINS-Mono to determine the most suitable real-time option. Additionally, we are developing a custom localization algorithm that uses goal triangulation and field line positions to estimate the robot’s position within the soccer field.

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)?

Currently, the robot does not implement a full path planning algorithm that takes obstacles into account, but future iterations may incorporate a path planning algorithm to navigate while avoiding obstacles, such as potential fields.

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

The robot's behavior is structured using a Finite State Machine (FSM), where it transitions between different predefined states based on sensor inputs and internal conditions. These states include searching for the ball, tracking, walking, kicking, and recovery from falls. We are currently exploring and testing Behavior Trees reliability for future improvements.

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

Yes, our robots use a computer vision model to detect objects within its camera frame. When specific objects are identified (such as balls or goal posts), the robot dynamically adjusts its camera to gather more information or maintain focus on the detected objects. This will enable the robot to perform actions such as preparing for kicking the ball.

Do you apply some form of filtering on the detected objects (e. g.

Kalman filter for ball position)?

At the moment, once the object is detected we apply a Kalman filter inside of the robot functioning as the goalkeeper, with the intention of predicting the general trajectory of a near ball and being able to stop it with the robot's body.

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 have implemented and tested Protobuf communication from the NuBots team, Protobuf message protocol for the Robocup Humanoid league, and we are currently deciding which information from our robots will be transmitted by this protocol.

Please list contributions your team has made to RoboCup

We haven't made contributions to RoboCup at the moment.

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

We haven't published anything yet, but our first scientific publication is in progress

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

Currently, we are utilizing a dataset created by Rhoban's team to train our computer vision model, it is called RhobansDetection-V2 and can be found in https://universe.roboflow.com/starkit-detection/rhobansdetectionv2

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 as its operating system and using ROS Noetic as the middleware.

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

N/A

If you have additional materials you would like to show, please

link to them here.

N/A