Software Survey 2025

Team Name

Tsinghua Hephaestus

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

Not at this moment, but it will be OpenSource in the near future.

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 design our robot CAD model using Solidworks and it can be easily converted into Unified Robotics Description Format by the OpenSource ROS sw_urdf_exporter. Once the URDF has been created, the dynamic properties can be loaded in simulation and algorithm.

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

Yes, we use inverse kinematics. We compute the inverse kinematics iteratively using a damped Levenberg-Marquardt method (also known as Damped Least Squares method).

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

Yes, we are using simulation for locomotion, perception, localization, path planning, and decision making.

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

We are using reinforcement-learning for locomotion and push recovery.

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

Reinforcement learning.

What approach are you using to generate kicking motions?

Reinforcement learning.

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

No.

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

We haven't adopted any publicly available datasets yet for perception model training. Our own perception datasets, which contain more than 20k images collected mainly inside our lab, are not public at presen

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

YOLO v8 is adopted in our perception system for object detection. Thanks to the changes in model architecture, 30FPS is now available with TensorRT acceleration on Nvidia AGX Orin, accuracy on small and faraway objects has also improved significantly.

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?

All planning is done in Cartesian space, with respect to a newly defined robot base frame, whose center is set at the midpoint between the two feet, with the x-axis pointing forward, the y-axis pointing to the left, and the z-axis pointing upward. Given the objects' uv coordinates in image space, a ray originating from the camera can be projected using the camera's intrinsic parameters. This ray is then transformed from the camera frame to the robot base frame, using the head-to-base transform (acquired from robot kinematics) and the camera-to-head transform (obtained from extrinsic calibration). Assuming that all objects lie on the ground (where their z-coordinates in the base frame are zero), the length of the ray can be fixed, and the coordinates in the robot base frame can be computed.

How is your robot localizing?

A particle-filter based on lines and crossings on the field, combined with odometer which is based on an IMU and state estimation.

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)?

Yeah, we do have path planning module and it is obstacle avoidance oriented. The global path is planned by traditional sampling method like Bi-RRT with kinematic constraints. The global path is transformed to local by the dynamic window approach as well as Bezier splines.

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

We use an OpenSource behavior tree library (https://github.com/BehaviorTree/BehaviorTree.CPP), which receives information about the position of the ball, goal and obstacles from the visual node, the position and orientation of the robot on the field the information about the game status from the game controller node, the angular state information (yaw, pitch) of the head node and the information about the robot's motion state from the gait node. According to these information, after calculation and processing of the decision node, corresponding decisions are made, and movement commands are issued to the head node and the gait node, so that the robot can make corresponding actions in the game.

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

Yes, the robot's camera will actively rotate in the direction that maximizes the probability of detecting the ball and improves its own localization accuracy.

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

Yes, we are using PySOT for ball tracking to improve ball position estimation when the ball is obscured.

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)?

No. Because humans only rely on sound and vision for information interaction between players in soccer games.

Please list contributions your team has made to RoboCup

The Tsinghua Hephaestus is a RoboCup Humanoid League team running at Dept. of Automation, Tsinghua University, China, since July 2006. The team had taken part in the RoboCup2007 both in KidSize and TeenSize. Our TeenSize team got the 2nd place in RoboCup2008, the 3rd place in 2009 and 2010. From 2011, we started to participate in Adultsize competition. We got the 2nd in Robocup2012 and the 3rd place in 2011, 2013, 2014, 2018 and 2019. We Also got the 2nd place in the AdultSize Technique Challenge in 2017, 2018, 2019 and 2024.

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

[2023]

1. Qingkai Li, Yanbo Pang, Wenhan Cai, Yushi Wang, Qing Li, and Mingguo Zhao, Xiu Li, Mingguo Zhao*, An Overview of Multi-task Control for Redundant Robot Based on Quadratic Programming, CIAC-2023.

2. Xueying Chen, Yong Yang, Wenhan Cai, Songrui Huang, Hopping Motion on Heavy-Legged Bipedal Robot Based on SLIP Model and Whole Body Control, FRSE2023.

3. Songrui Huang, Wenhan Cai and Mingguo Zhao*, Three-Rigid-Body Model based NMPC for Bounding of a Quadruped with two Spinal Joints, CLAWAR2023.

4. Qilun Wang, Qing Li, Mingguo Zhao, Xiaozhu Ju, Fast Terrain-Adaptive Motion of Humanoid Robots Based on Model Reference One-Step-Ahead Predictive Control, IEEE Transactions on Control Systems Technology.

5. Yushi Wang, Yanbo Pang, Qingkai Li, Wenhan Cai, and Mingguo Zhao, Trajectory Tracking Control for Robot Manipulator under Dynamic Environment, ICIRA2023.

6. Qingkai Li, Yanbo Pang, Yushi Wang, Xinyu Han, Qing Li, Mingguo Zhao, CBMC: A Biomimetic Approach for Control of a 7-DoF Robotic Arm, Biomimetics.

[2022]

7. Yan Xie, Jiajun Wang, Hao Dong, Xiaoyu Ren, Liqun Huang and Mingguo Zhao, Dynamic Balancing of Humanoid Robot with Proprioceptive Actuation: Systematic Design of Algorithm, Software,and Hardware, Micromachoines,2022,13,1458.

8. Xiaozhu Ju, Jiajun Wang, Gang Han and Mingguo Zhao, Mixed Control for Whole-Body Compliance of a Humanoid Robot[C]// ICRA2022.

9. Gang Han, Jiajun Wang, Xiaozhu Ju and Mingguo Zhao, Recursive Hierarchical Projection for Whole-Body Control with Task Priority Transition, IROS2022.

10. Jiajun Wang, Gang Han, Xiaozhu Ju and Mingguo Zhao, Whole-Body Control with Motion/Force Transmissibility for Parallel-Legged Robot, IROS2022.

11. Wenhan Cai, Qingkai Li, Songrui Huang, Hongjin Zhu, Yong Yang, Mingguo Zhao, Squat Motion of a Bipedal Robot Using Real-Time Kinematic Prediction and Whole-Body Control, IET C&R.

12. Yong Yang, Jiyuan Shi, Songrui Huang, Yuhong Ge, Wenhan Cai, Qingkai Li, Xueying Chen, Xiu Li and Mingguo Zhao, Balanced Standing on One Foot of Biped Robot Based on Three-Particle Model Predictive Control, Biomimetics.

13. Sun Yiyong, Zhao Haotian, Chen Zhang, Zheng Xudong, Zhao Mingguo, Liang Bin, Fuzzy model-based multi-objective dynamic programming with modified particle swarm optimization approach for the balance control of bicycle robot, IET Control Theory and Applications(1751-8644, IF2.670), 2022, 16(1), p.7-19.

14. Luo Guifu, Du Ruilong, Zhu Shiqiang, Song Sumian, Yuan Haihui, Zhou Hua, Zhao Mingguo, Gu Jason, Design and Dynamic Analysis of a Compliant Leg Configuration towards the Biped Robot's Spring-Like Walking, JOURNAL OF INTELLIGENT & ROBOTIC SYSTEMS(0921-0296, IF3.129), 2022, 104(4).

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

None

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

Ubuntu 22.04

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.