

PCMS-HRG Extended Abstract

RoboCup 2026 Humanoid League

AdultSize

Lao Kun Wa, Lam Kin Un, Lei Mou Hong,
Hoi Wai Kio, Kwan Chi Cheng, Lam Si Ioi, Lei Pui Weng,
Lei Ut Ieng, Mui Lek Hou, Wu Iat Cheng, Yao Yi Chen

PUI CHING MIDDLE SCHOOL, MACAU
No. 7, Avenida de Horta e Costa, Macau, China
<https://puiching-fablab.github.io/humanoid/>

Abstract. PCMS-HRG is a first-time RoboCup Humanoid League Adult-Size team. We build on the Booster T1 platform with a ROS-2 stack for perception, locomotion, skill execution, and team coordination, targeting baseline competition readiness within six months. We also use RoboCup to explore learning-based whole-body control, visual imitation learning, and multi-agent coordination.

1 Introduction

PCMS-HRG is a newly formed student team entering RoboCup Humanoid League to build a strong foundation in humanoid robotics. Our development focuses on hardware integration, Python/C++, ROS 2, and AI-driven soccer behaviors. We follow an iterative, hardware-first workflow: implement \rightarrow test on real robots \rightarrow diagnose \rightarrow refine. This document summarizes our platform, current architecture, and a development plan toward baseline competition functionality.

2 Hardware and Software Platform

2.1 Hardware: Booster T1

We use the Booster T1 humanoid robot to accelerate development by leveraging a mature commercial platform. Table 1 summarizes the key specifications.

Table 1. Booster T1 Hardware Specifications

Component	Specification
Height / Weight	1.18 m / 45 kg
Degrees of Freedom	23 (legs, arms, waist, head pan)
Onboard Compute	NVIDIA Jetson AGX Orin
Camera	Stereo global-shutter cameras

2.2 Software: ROS 2 Integrated Stack

We build on ROS 2 Humble with modular publish-subscribe nodes spanning perception, behaviors, and coordination.

- **Perception:** YOLO-based detection for soccer primitives (ball, goalposts, field lines, robots), optimized with TensorRT. Stereo depth supports 3D localization.
- **Locomotion:** Booster SDK provides a stable walking interface backed by a proprietary learned gait, wrapped with ROS 2 safety checks.
- **Skills:** Hierarchical behavior trees (`py_trees`) for modular sequencing (search → approach → align → kick).
- **Coordination:** Team Coordinator node maintains shared world state and broadcasts role commands (attacker/support/defender).

2.3 FollowMe: Closed-Loop Ball Chasing

We implement *FollowMe*, a lightweight closed-loop module for ball chasing and approach-to-kick. Figure 1 shows a three-layer architecture: a **Perception Layer** for target detection and 3D estimation, a **Control Layer** for generating velocity commands, and an **Execution Layer** for executing walk/kick behaviors.

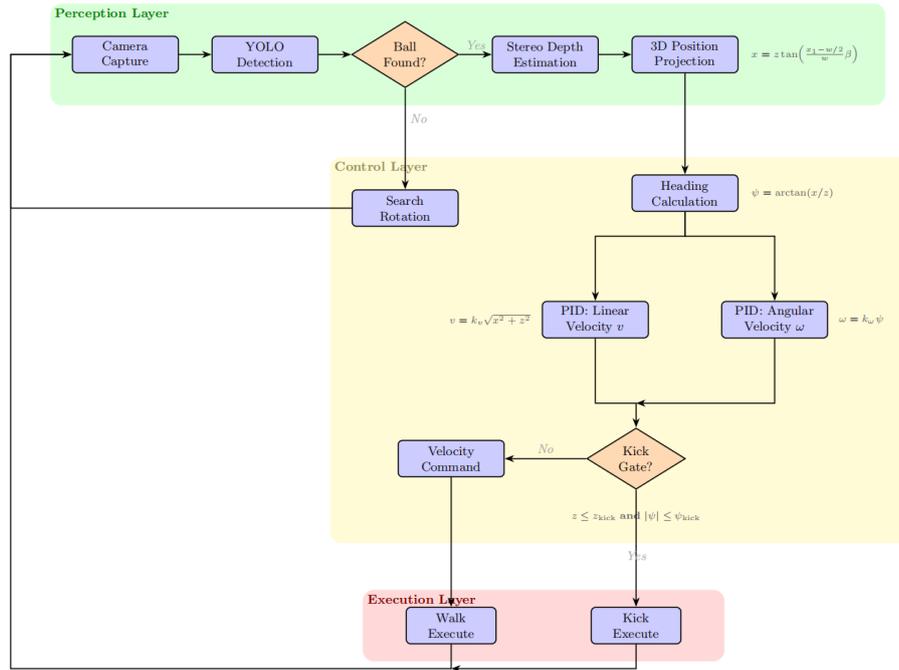


Fig. 1. FollowMe system architecture. The perception layer (green) performs YOLO detection and 3D position estimation. The control layer (yellow) computes velocity commands and triggers a search when the target is lost. The execution layer (red) executes walk or kick.

Target Localization. Given the ball’s image coordinates (x_1, y_1) and depth z from stereo vision, with vertical/horizontal FOV (α, β) and image size (h, w) , we compute the real-world position:

$$x = z \cdot \tan\left(\frac{x_1 - w/2}{w} \cdot \beta\right), \quad y = z \cdot \tan\left(\frac{h/2 - y_1}{h} \cdot \alpha\right) \quad (1)$$

The lateral offset x and distance z are used for planar pursuit control.

Pursuit Control. We compute the heading angle and apply PID controllers for velocity commands:

$$\psi = \arctan(x/z), \quad v = k_v \sqrt{x^2 + z^2}, \quad \omega = k_\omega \psi \quad (2)$$

Before triggering a kick, we require: (1) distance gate: $z \leq z_{\text{kick}}$, and (2) alignment gate: $|\psi| \leq \psi_{\text{kick}}$. If the ball is lost, the robot enters a search routine (rotating in place) until re-acquisition.

3 Development Plan

- **Phase 1 (Months 1–2):** Robot assembly, hardware diagnostics, and sensor/actuator validation for all units. ROS 2 workspace setup with Booster SDK drivers and end-to-end message flow. Perception model deployment with latency/accuracy validation; camera calibration for distance estimation. Walking stability tests on multiple surfaces; establish safe teleop and logging workflows.
- **Phase 2 (Month 3–4):** Improve gait stability for turns, starts, and stops; tune SDK parameters for field surfaces. Data collection under diverse lighting conditions; fine-tune detection model; add temporal filters to reduce noise. Integration of striker/defender behavior trees; achieve autonomous “find ball → approach → align → kick” sequences. Deploy *FollowMe* module with kick-distance and alignment gates.
- **Phase 3 (Month 5–6):** Implement and debug team coordination with shared world state and role assignment. Progressive scrimmages: 1v0 → 2v1 → 3v3 drills to uncover integration issues. Mock competitions aligned with official procedures including referee box integration, with post-game log analysis to identify bottlenecks.

4 Research Interests

1. **Learning Agile Whole-Body Control:** Extend baseline gait toward more agile, robust skills including push recovery, efficient kicks, and smooth transitions. We plan to use deep reinforcement learning with safe training protocols and Sim2Real transfer via Gazebo/Isaac-based simulation pipelines.

2. **Visual Imitation Learning:** Extract motion primitives from human soccer videos through pose estimation techniques. Learn higher-level tactical priors such as positioning and action selection to inform robot skills and game strategies.
3. **Decentralized Multi-Agent Coordination:** Move beyond centralized control toward communication-limited cooperation. Explore adversarial multi-agent reinforcement learning for dynamic role allocation and uncertainty-aware shared world modeling.

5 Conclusion

PCMS-HRG enters RoboCup Humanoid League AdultSize with a Booster T1 fleet and ROS 2-centered architecture. Our *FollowMe* module demonstrates integrated perception-control for ball chasing with kick gates. Our six-month plan targets competition-ready functionality, while using RoboCup as a testbed for embodied AI research.

References

1. Booster Robotics. (2024). *Booster T1 Technical Specifications and SDK Documentation*.
2. RoboCup Humanoid League Technical Committee. (2025). *RoboCup Humanoid League Rules and Regulations*.
3. Macenski, S., et al. (2022). Robot Operating System 2: Design, architecture, and uses in the wild. *Science Robotics*.
4. Open Source Robotics Foundation. (2023). *ROS 2 Humble Hawksbill Documentation*. <https://docs.ros.org/en/humble/>