Water Team Description 2024

Beijing Information Science & Technology University,
NO. 12 East qinghexiaoying Rd., Haidian District, Beijing, 100192, China
www.teamwater.cn,water@teamwater.cn

Abstract. This paper delves into Team Water's robotic systems, spotlighting key enhancements in robot vision, software architecture, path-planning, self-positioning, and collaborative strategies. It outlines the team's triumphs in RoboCup and prestigious platforms, emphasizing the integration of YOLOv8 for football recognition, master selection algorithm refinement in the Middle Size League, and inter-system communication for performance optimization. Additionally, the paper addresses version compatibility challenges in vision detection program development, proposing an innovative solution to swiftly adapt to new demands and environments while maintaining compatibility with existing operating systems.

Keywords: Robocup Middle Size, Water, goallkeeper, robot, YOLOv8

1 Introduction

The Beijing Information Science & Technology University proudly presents Team Water, its distinguished Middle Size Robot Team, established in the year 2003. Renowned for their participation in the prestigious China Open spanning from 2006 to 2022, and their commendable appearances in the World Championships from 2010 to 2019, the team exemplifies excellence in robotic innovation. Team Water's expertise is exemplified through their profound research in cutting-edge domains including Robot Vision, advanced Software Architecture Frameworks, intricate Path Planning mechanisms, precision-oriented Positioning techniques, sophisticated Communication forms, and robust Control Models, placing them at the forefront of technological advancement in the field of robotics.

World Championships: 1st place: 2010,2011,2013,2015,2017
2nd place:2014,2019
3rd place: 2013,2018

RoboCup China Open: 1st place: 2010,2014,2017,2018,2019， 2021， 2023
2nd place: 2008,2009,2011,2013,2016
Top 1：2022

Asia Pacific RobotCup: 1st place: 2019， 2021， 2023
3st plase:2022

Capability video： https://youtu.be/HqMzqoP1ENk

2 Hardware Structure

The original robot was made manually in 2004.The first two wheels robot was made manually by the team in 2006.Then in 2007 we made a large-scale transformation to it, and advanced the visual parts and motor driver parts of the robot. So the effect of the image and the speed of robot were upgraded.(**Fig. 1 and 2**)In 2008, our first Omni-directional three wheels robot was finished which was also made manually.(**Fig.3**)This robot was more flexible. In 2009, our robot was advanced and new robot was born.

Dimensions: 50*50*80(cm)
Weight: 39kg

Fig.1

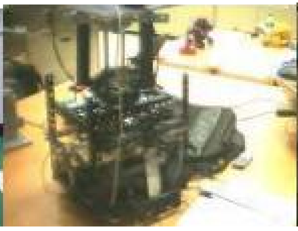


Fig.2

Fig.3

2.1 Motion

When we made the robot, we put an Omni-directional wheel, a 160w DC electric machinery and a pro-Motion BDMC3606SH Servo Driver together to make a Machinery Servo Driver. And placed them on the evenly Equilateral triangle robot machine chassis. The angle of the adjacent otor axis is 120 degrees. So the connection between the Machinery servo Driver is simpler and easier to replace, remove or fix.(**Fig.4 and 5**)

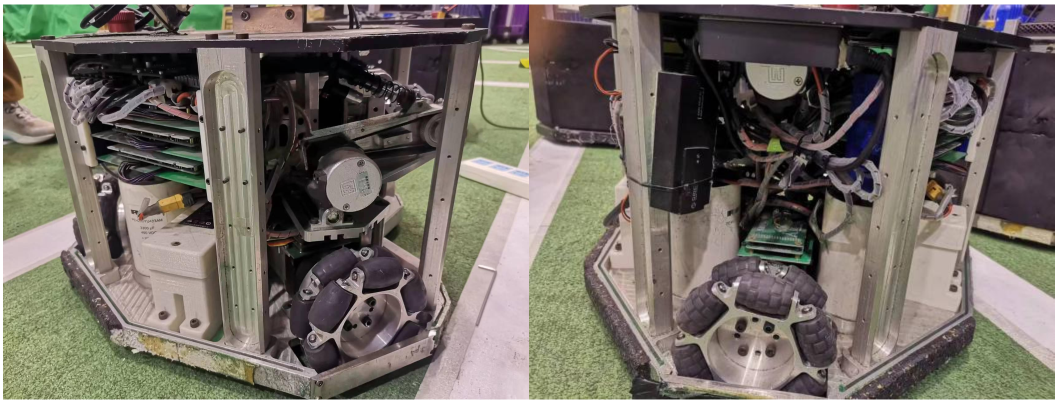


Fig.4

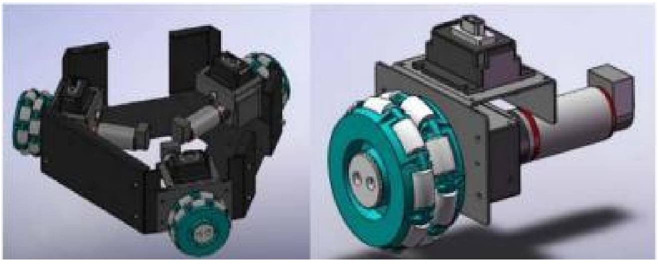


Fig.5

2.2 Control circuit

Our control circuit could make unified controlling for all functions of the robot. The inboard computer could communicate with it through USB. Using this interface, we could control the 5 axes servo motor of our robot and collected the current of every motor, encoder position and the cultivate volume from dribbling sensor.

2.5 Camera

The minimum requirement for cameras is 640*480 resolutions and 45 fps. Therefore we select the point Gray's fumy camera, which can reach 60 furthest color dynamic range was higher than 90db, so we had a better suitability for light.

2.6 Omni

The visual range of omni line and the ball we made was much more than 9m. It made it possible that if there was no shelter materials in the ring, our robot would overlook the court nearly about more than 90%.This character greatly helped our robot to judge the trace and predict the position of ball.

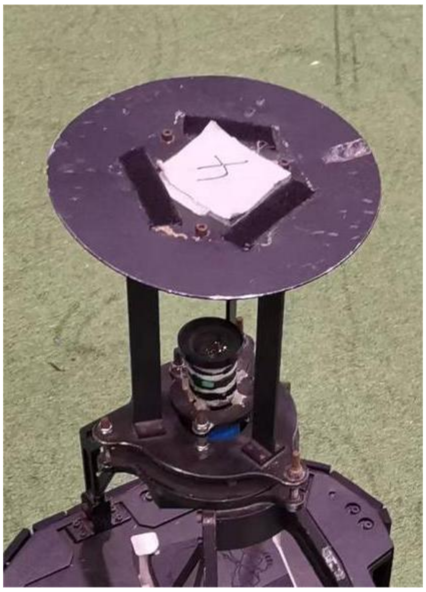


Fig.6

3 Software System

3.1 Path-Planning

Our path-planning not only could provide best walking route, but also made the robot generate his own action initiatively. The robot would produce movement direction, speed and corner speed based on his own direction in polar coordinates and other model which formed his own movement. This feature had several advantages. It combined the tactic part and controlling part together cleverly and generated unified way to solve the movement action. Such as in defense, the coach sent an easy command to assign the robot to one place and he would generate his own movement pattern. It provided an easy, effective and flexible way for motion.

Due to this feature, we also could set up any of tactics flexibility. Our robot would do technical adjustments for different team, strategy and formation. We often did some special adjustment in the second halfjust for the problem in the first half.

3.2 Self positioning

Our robot self-locates by the white lines of the field relative to the position and angel of the robot. We will make correction to the omni-image first and recovery the shape of field line.(**Fig. 7**)The picture is Omni-image before correction and after correction. Through the white line in the image and template of field line, we get the location relative to the field line. From **Fig.8**, the color of template field line is blue. The white field line was expressed by pink matrix. The software will match the pink matrix with the blue lines until they get together. We calculate absolute coordinate in the field with the turning of matrix angel and distance.

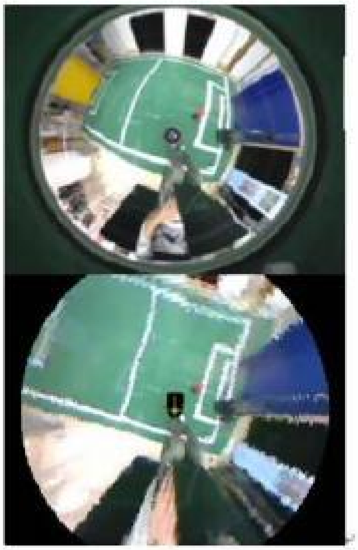


Fig.7

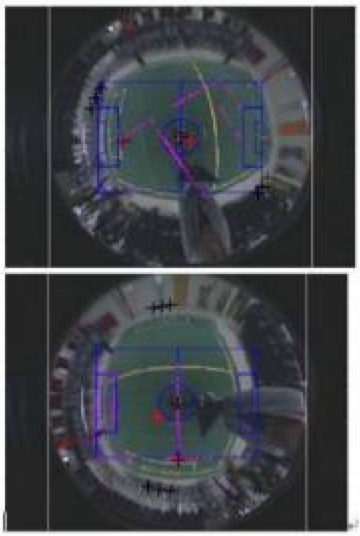


Fig.8

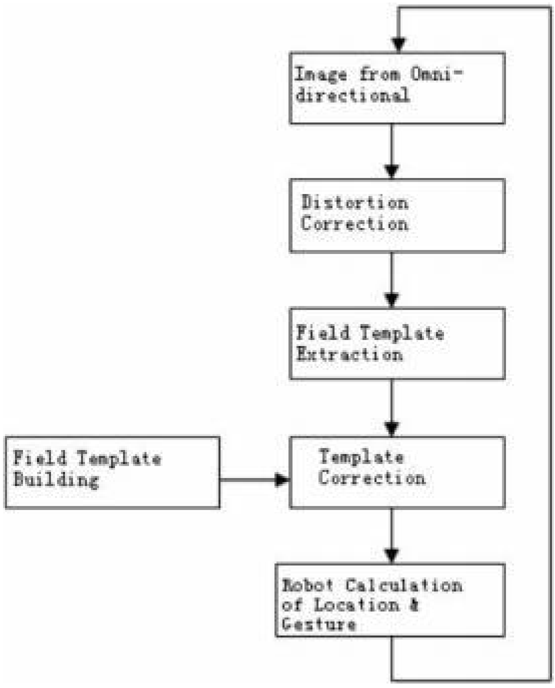


Fig.9

5 Pioneering Ingenuity

Over the past year, our team has made substantial strides in the field of robotics technology, achieving encouraging progress. As researchers and learners, we have consistently maintained a strong desire for knowledge and a pursuit of innovation. This year, we concentrated on three pivotal projects, which not only enhanced our practical experience but also profoundly impacted our understanding and application capabilities in technology.

Firstly, we endeavored to implement the cutting-edge YOLOv8 technology in our football recognition system. This effort not only advanced our skills in object detection but also ignited our passion for real-time image processing techniques. Secondly, we meticulously refined the master selection algorithm used in the RoboCup Middle Size League (MSL) football robot competition. This process not only honed our algorithm design skills but also deepened our comprehension of team collaboration and strategic planning. Finally, by realizing inter-system communication technology on our Water football robots, we not only broadened our understanding of robotic communication networks but also laid a solid foundation for future multi-system integration projects.

5.1 Football Robot Following System Based on YOLOv8

In alignment with the evolution of the RoboCup Middle Size League (MSL) competition towards human-robot confrontation modes, we proposed a diverse football recognition solution based on the YOLOv8 model. This system integrates the principles of binocular ranging for spatial localization, enabling the robot to follow the movement of the football. The overall software system was initially tested on the Jetson Xavier NX embedded platform. Experimental results indicate that the visual recognition operates at approximately 56 frames per second, fulfilling the real-time requirements of the competition. The recognition accuracy in a complex environmental validation set reached 0.767mAP@0.5, meeting the precision demands of the competition.

The vision system approach of the RoboCup MSL has transitioned from simple color-based recognition methods to advanced recognition algorithms based on deep learning, offering multiple possibilities for enhancing robot performance in complex environments. Building on this, our team developed a football recognition system based on the YOLOv8 deep learning network. This system, integrated with depth information provided by depth sensors, localizes the football and guides the robot to track it. The system uniquely identifies and locates footballs of various colors and patterns, addressing the gap in robot recognition when the football is airborne.

The vision recognition and localization node detects the football's position in real-time and communicates with the robot's motion control node to execute the movement control of the football robot. The control outcomes are illustrated in **Fig.10**. The experimental procedure involved kicking the football along the midline of the field, with the robot's tracking performance observed from a third-person perspective. Image (1) depicts the initial state, and in Image (2), it is evident that upon receiving the football's position, the robot calculates the direction, adjusts its posture, and aligns itself towards the football. Images (3) to (8) display the continuous movement of the football, with the robot promptly adjusting its direction and ultimately reaching the position to kick the football, as shown in Image (9). **Fig.11** presents the first-person perspective images corresponding to **Fig.10** during the experiment, illustrating the robot's consistent detection of the football (Image 9 captures the robot's perspective when it has seized the ball, with other sensory methods determining that it's in a ball possession state).

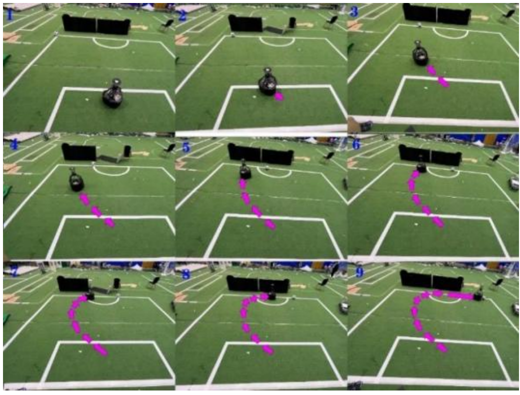


Fig.10 Robot Follows Soccer Ball Movement Trajectory

Fig.11 Robot Vision Recognition

To validate that this vision solution can complement the robot's recognition of airborne footballs, the following experiment was designed: the football was placed in a stationary position on the ground, and the recognition system was activated. The ball was then slowly elevated from the ground in the vertical direction, as depicted in **Fig.12**. The direction of the ball's ascent was consistent with the negative direction of the y-axis in the camera's coordinate system. The football positions obtained by the recognition and localization algorithm are presented in **Fig.13**.

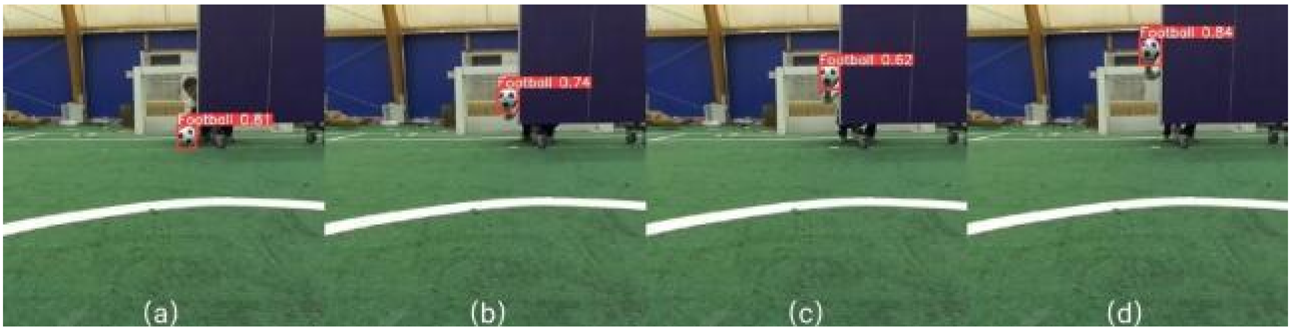


Fig.12 Football Lifting Process

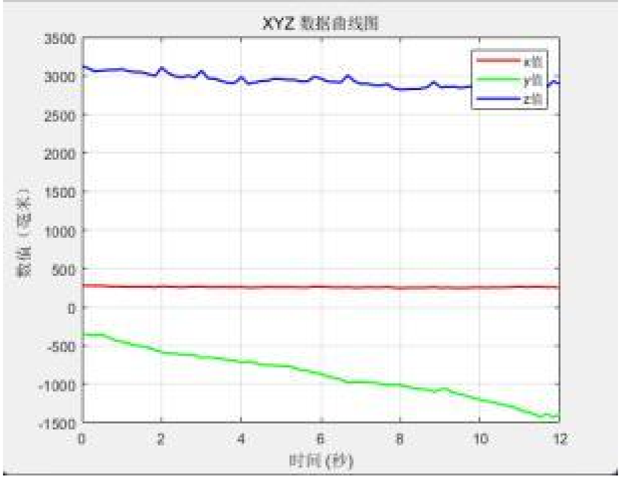


Fig.13 Positioning Results of the Ball Lifting Process

As illustrated in the figure, the football is gradually elevated from (a) to (d), during which the corresponding y-values continuously decrease, while the x and z-values remain relatively constant. This is in accordance with the motion trajectory of the football, thus validating that the system is capable of accomplishing the task of recognizing airborne footballs.

5.2 Technical Report on the Implementation of Inter-system Communication in Water Football Robots

The primary objective of this study is to address the issue of version compatibility in the development of vision detection programs for football robots. Control programs for football robots typically require operation in specific system environments, such as running a VC6.0-based football robot control program on an older Windows XP system. However, with ongoing iterations of the robot, rapid development of new functionalities is necessary to meet the requirements of different competition or working environments. Often, this rapid development involves the use of off-the-shelf integrated algorithm programs, such as the OpenCV Python package for vision processing. However, many powerful algorithm integration programs are not compatible with relatively outdated operating systems, making integration into existing robot programs challenging.

When faced with the aforementioned issues, one solution is to comprehensively replace the robot program to adapt to newer versions of the operating system and to be compatible with new integrated programs. However, this approach is not always feasible. Especially when the robot control program is highly complex and contains a significant amount of algorithm code dependent on the current operating system environment, replacing new algorithm code typically requires considerable time to resolve compatibility issues with the new system. Furthermore, although this method is more effective and standardized in the long term, it is not suitable for meeting the demands of rapid development of new functionalities and is impractical in certain scenarios. Programs adapted for specific types of operating systems are not suitable for addressing integration issues with incompatible programs through updating the operating system.

Therefore, this study aims to propose a solution to version compatibility issues to validate and implement new functionalities in the short term. Specifically, this research focuses on the vision processing functionalities within football robot programs, exploring communication technology-based solutions to address version compatibility issues. This will enhance the flexibility of the robot programs, enabling rapid adaptation to new requirements and environments, while reducing compatibility issues associated with operating system updates.

Experiments primarily involved establishing P2P message queue communication through ActiveMQ on XP and Win11 systems, implementing the transmission of vision data on the Water robot program, and completing the processing of image data on a Win10 system.

Subsequently, the processed vision data information is returned to the XP system through the message queue method.

The implementation process is shown in the figure：

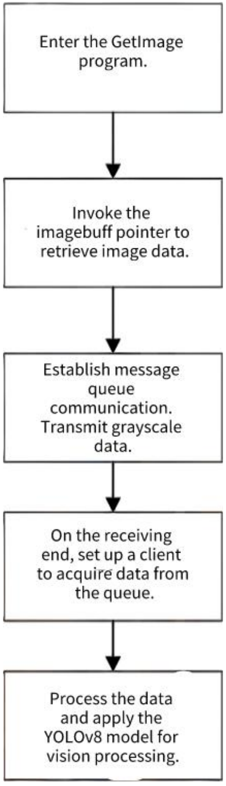


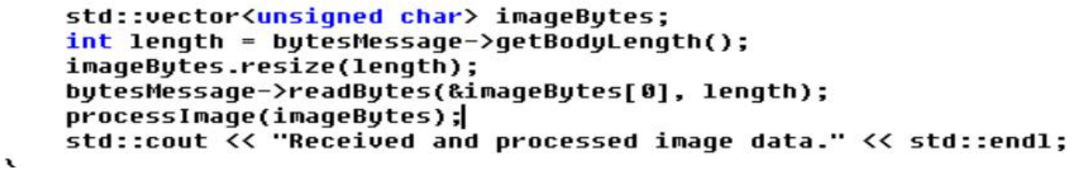
Fig.14 Program implementation basic flow

In the Windows XP system, specifically within the subordinate machine of the football robot, a message queue is established, and the byte-type array of vision data pointed to by the buffer pointer is transmitted. The critical code for this part is illustrated in the figure.

```
std::vector<char> buffer((std::istreambuf_iterator<char>(file)), std::istreambuf_iterator<char>());

auto_ptr<BytesMessage> message(session->createBytesMessage());

message->setBodyBytes(reinterpret_cast<unsigned char*>(buffer.data()), buffer.size());

producer->send(message.get());
```

Fig.15 The server sends visual data

After retrieving and transmitting the vision data from the main program of the Water robot in the subordinate machine system, the vision data is acquired and processed through deep learning on the Win10 operating system. The key code for the data reception portion is depicted in the accompanying figure.



```
std::vector<unsigned char> imageBytes;
int length = bytesMessage->getBodyLength();
imageBytes.resize(length);
bytesMessage->readBytes(&imageBytes[0], length);
processImage(imageBytes);
std::cout << "Received and processed image data." << std::endl;
```

Fig.16 The client receives a visual data array

After addressing issues related to continuous dynamics and data overwriting, the data transmission is completed, along with the vision processing based on the YOLOv8 model.

6 Summary

The study encapsulates the comprehensive development of Team Water's robotic systems, detailing the hardware structures, control circuits, and camera configurations that form the backbone of the robots. It delves into the software system, elaborating on the sophisticated path-planning, self-positioning, and vision processing functionalities that empower the robots with advanced tactical and strategic capabilities. The core of the research lies in the pioneering ingenuity applied to the football recognition system, leveraging YOLOv8 technology and robust inter-system communication to navigate complex environments and achieve real-time responsiveness. The paper addresses the crucial aspect of version compatibility in software development, proposing a practical solution that ensures the swift implementation of new features without compromising the existing system structure. Through this exploration, Team Water continues to push the boundaries of robotic technology, aiming for excellence in future RoboCup and other robotic competitions.

7 References

Dou Zhenzhong.The Application Design and Examples of PIC.Beijing.Beijing University of eronautics and Astronautics Press

http//www.Robocup.org.

Li Shi, Xu Xuming, Ye Zhen, Zhang Zengqi.The International Robot Football match and Related Technology.Robot.000, (9)：420-426.

Predicting away Robot Control Latency Sven Behnke2, Anna Egorova1, Alexander Gloye1, Ra′ul Rojas1, and Mark Simon1，2004

RoboCup Middle Size Robot League Rules and Regulations for 2013