WrightOcean Team Report 2015

Fei Liu, Gaohuan Lv, Hanjun Xiao,
Haiming Ding, Yongjin Qu, Weizhen guo, and Ge Sun

Ludong University, Yantai, Shandong, China
{liufei,lvgh}@ldu.edu.cn,
{572733610,371518282,787498532,296236668}@qq.com,dakesun@126.com
http://www.wrightocean.org

Abstract. The report introduces the information of WrightOcean team and presents four techniques that were used in RoboCup 2015. These techniques consist of a vision algorithm optimizing white goal detection, some behavior approaches, the communication method using standard message protocol for the drop-in player competition and the ball recognition method in realistic ball challenges.

Keywords: RoboCup, Vision, Behavior, Communication

1 Introduction

WrightOcean, a team which belongs to Lab of Robotics, School of Information and Electrical Engineering, Ludong University in China, was established in April, 2012. We have two V5 and three V3.3 H25 NAO robots now. It is a young and passionate team, and all of the members are undergraduates keen on robotics. The team consists of the following numbers at present.

- Instructors: Fei Liu, Gaohuan Lv
- Students: Ge Sun, Haiming Ding, Yongjin Qu, Yanbin Wang, Chunxue Li, Changchun Pu, Shian Wang, Wang Xu

The WrightOcean team participated in the Standard Platform League of RoboCup China Open in 2013 for the first time, and we were the runner-up to the TJArk (a team from Tongji University) in a team competition. In RoboCup China Open 2014, we took part in three technical challenges and shared the third place with Dalian University of Technology. In 2015, we took part in RoboCup S-PL held in Hefei, China for the first time. And we won the third place in technical challenges and one of best drop-in only teams in drop-in player competition[1].

This team report is organized as follows: Sect. 1 is about our team and team members' information. Sect. 2 is about white goal detection. In Sect. 3, we describe two behaviors for drop-in player competition and corner kicks challenge. In Sect. 4, the communication method using standard message protocol is presented. Sect. 5 states the ball recognition method in realistic ball challenge.

2 White Goal Detection

In rule 2015[2], the color of goalpost is white, different from ones in former years. When robots scan up and down, goal detection algorithm will mark the robots in the horizon with two white vertical lines and then those vertical lines will confuse the recognition of the goal. For this problem, the goalpost recognition algorithm needs to be corresponding changed.

We develop a scanning algorithm with the robots' information. In the case when the game starts, it can use the cross bar who is the longest white line to judge the position of goalpost. But in the normal cases, the robots may ward the cross bar in the horizon image. The algorithm marks the start point and the end point of all robots and then excludes the robots from goalposts. It can improve the accuracy of white goal detection.

2.1 Fundamental Principles

Firstly, the crossbar is found from one image when the robot starts. Secondly, the algorithm would look for the lowest point scanning down from the far left and right of the crossbar. Finally, it finds the information of the goalposts. This algorithm can be applied to a simple field scene.

2.2 Rejecting the Interference Induced by Robots

Because the white points in a robot can be viewed as several white lines, and the lines may appear as the same as the goalposts, the robot must discriminate the two different lines. To solve the problem, we adopt the following algorithm. First, all the white vertical lines are found, and labeled by their start and terminate points. Second, every line is judged whether it is in the robotbox by using RobotPerceptor module. If a line is in the robotbox, then it will be removed from the detected line set. Thus the remainder lines will be the expected poles. when more than tow goal post are recognized, the far left and right in the sight are affirmed as goal posts based on guesswork.

3 Behavior

We have developed some independent behaviors for the drop-in player competition and three technical challenges[3]. The behaviors are designed as follows.

3.1 Striker Tactics

At the beginning of competition, the striker finds a ball automatically. If there is a ball in its vision field, then the robot will turn to finding-ball status (turning around and scanning up to down). If it can not find a ball after it has turned 360 degrees, then it will turn to patrolling status (walking randomly for 6 seconds). When a ball is found, the robot will adopt its position and make its body face the ball directly. Then it will approach the ball and aim at the goal. By adjusting the position between its foots and the ball, the robot turns its status to shoot.

Fig. 1. The striker's decision in drop-in player competition

3.2 Tactics in Corner Kicks Challenge

At the beginning of the game, the robot at position R1 (called R1 robot) has to firstly move and the robot at position R2 (called R2 robot) keeps the original state. If R1 robot cant touch the ball, R2 robot is in the same state. If R1 robot touches the ball at the first time, R2 robot starts to move and R1 robot kicks the ball to R2 robot until R2 robot touches the ball.

If R1 robot and R2 robot see the ball at the same time, the robot which is closer to the ball moves and kicks ball first. If they are equal to distance from the ball and no obstructions, R2 robot has the priority to kick. If the robot can't see the goal, it should kick the ball to its teammate.

4 Communication Using Standard Message Protocol

Communications in the B-Human framework consist of three parts, the inter-process communication, debug communication and team communication. All the parts are based on message queue. The message queue encapsulates the implementations of network communications, and the user can add and process different messages based on the framework without considering the fundamental communication implements. In the mechanism of the message queue, the child class of message queue, i.e., InMessage, is used to read messages from the message queue, and the other child class called OutMessage is responsible for putting the messages into the message queue.

Communication between multiple robots is called team communication. It is a mechanism exchanging messages between different robots in the same team. All the messages are broadcasted to all the team members. A message queue embedded in a UDP packet is employed by the team communication. The first message in this queue is idRobot, which contains the ID of the robot who transmitted the message. All the messages are sent in the $TEAM\_OUTPUT(< id >$

$, < format >, < sequence >$) of the TeamDataSender block, and are transformed to standard message format by a function named haddleMessage() in SPLStandardMessage module. On the contrary, when a robot receives a message, it will transform the message from a standard format to inner format and put the message to data flow. The message will be processed in TeamReceive module.

5 Ball Recognition in Realistic Ball Challenge

To determine the number of balls in a target image, noises must be suppressed. In the process, the green component of the picture is filtered firstly, and then the resulted picture is transformed to gray level format. Edge detection is performed based on the gray level image and results in a binary image containing edge information. After performing Hough transform on the binary image, centers of detected circles are obtained and thus their positions are determined.

6 Conclusions

The report presents the techniques that we used in RoboCup 2015. In 2015, we have designed one algorithm optimizing white goal detection, some behavior approaches, the communication method using standard message protocol for the drop-in player competition and the ball recognition method in realistic ball challenges.

Acknowledgement

This project has been supported by the Shandong Science and Technology Development Planning under grant 2012YD03110 and the Research Foundation of Ludong University under grant LY2012021, as well as the lab of robotics in Ludong University. We sincerely thank open source movement of B-Human, Nao Devils, UT Austin Villa, Northern Bites, Kouretes, UNSW Australia and all other SPL teams. Because of your contribution, we can go ahead and contribute our strength for the development of SPL.

References

1. Standard Platform League Results 2015. `http://www.tzi.de/spl/bin/view/Website/Results2015`
2. RoboCup Standard Platform League (NAO) Rule Book, `http://www.informatik.uni-bremen.de/spl/pub/Website/Downloads/Rules2015.pdf`
3. RoboCup Standard Platform League (NAO) Technical Challenges, `http://www.informatik.uni-bremen.de/spl/pub/Website/Downloads/Challenges2015.pdf`