Team Water: The Champion of the RoboCup Middle Size League Competition 2013

Song Chen, Zhe Zhu, Ye Tian, Charles Lynch, Di Zhu, Ye Lv, Xinxin Xu, Wanjie Zhang, Liang Zhao, Xiaoming Liu, Miao Wang, Peng Zhao, Chenyu Wang, Wei Liu, Feichan Yang, Binbin Li, Jieming Zhou, Zongyi Zhang, and Xueyan Wang

Beijing Information Science & Technology University,
No. 12, East Qinghexiaoying Rd., 100192, China
water.bistu@gmail.com
http://jdgcxy.bistu.edu.cn/robocup/blog

Abstract. This paper addresses the problems encountered and the advancements made to the robots of Team Water, the champion of the RoboCup Middle Size League Competition in 2013. This paper describes the robot from both the hardware point of view and the software. The hardware description includes details of the design of an omnidirectional wheel and an improved ball handling device. The software part focuses on the improvements on the masking function and the localization algorithm.

1 Introduction

The middle size league is an important part of RoboCup (Robot World Cup) which mainly focuses on advancing MAS (Multi-Agent System)[1] and DAI (Distributed Artificial Intelligence). During the competition, each team dispatches five autonomous robots on an 18 meters long and 12 meters wide soccer field. The winner is the team who scores more goals. There are many technical challenges. First, the robot collects information of the field in real time and follows that up with decisions, while the robot keeps moving simultaneously. Second, due to the special design of the omnidirectional camera, the image got from the camera is distorted. The program must correct the image efficiently and provide the robot with precise location information. Third, because of the fierce antagonism during the match among robots, the robot demands a high tensile structure. At last, on our course towards the goal of beating a human soccer team in 2050, the intelligent cooperation among robots is a big challenge to the researchers.

Team Water of BISTU devoted itself to the RoboCup middle size league since 2003, when the team designed its first generation robot with two differential wheels and a coach program. In 2005, Team Water turned its focus to improve performance of the robot's hardware and software. Consulting the paper on the control of omni-wheels[2], the second generation robot was designed with a three wheeled omnidirectional chassis and a 360 degree mirror. The image processing and motion strategy was also redesigned and implemented on the new robot, and the coach

S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 37–48, 2014.

program improved. In 2010 and 2013, Team Water won the championship of the RoboCup middle size league. In the past year, the team improved the performance of the omnidirectional wheel, ball handling mechanism, mask function and localization module.

Remainder of this paper is organized as follows. Section 2 details on the improvements to the hardware, including the omnidirectional wheel and ball handling mechanism. Section 3 addresses the improvements to software, including the mask function and the localization functions. Section 4 describes the results of our improvement at the competition. Section 5 concludes the paper.

2 Hardware System

Robots of Team Water are shown in the Fig. 1. The camera captures the images and sends them to the onboard computer. The onboard computer makes decisions after receiving information that come from the sensors and the electrical circuits of the control system. Then the onboard computer sends the decisions to the servo.

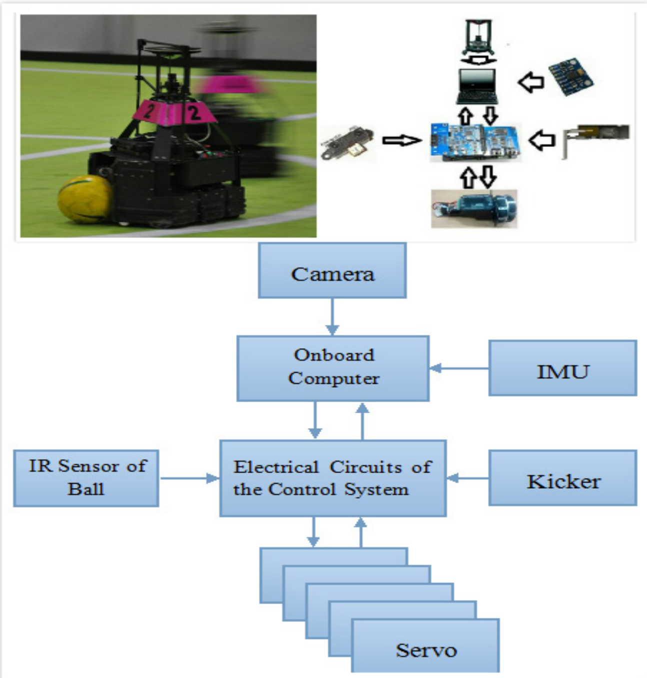

Fig. 1. Hardware system

Hardware defects have been fixed after the RoboCup in 2012 where we missed the championship. Most of the improvement has been made in the Omni-directional wheels and ball handling mechanism.

2.1 Omnidirectional Wheel

Omni-directional Wheels have small rollers to allow the wheels to move freely in any direction. They can rotate along the primary diameter, just as any other wheels. The smaller rollers along the outer of the diameter allow rotation along the orthogonal direction. With these two rotations combined, the robot can move along any direction no matter in which direction it's heading.

The Omni-directional Wheels used in our robots before 2013 have many defects. To start with, the wheel hubs are made of plastic, so they can break easily when crashed by other robots, as shown in Fig. 2(Left). Moreover, the thin rubber wrapped around the rollers is easy to wear and fall off after one or two competitive games, as shown in Fig. 2(Right). Additionally, it makes disassembling of the wheel hub almost impossible because all the parts of the wheel hub are glued together, which makes the replacement of small rollers impossible, resulting in a great waste.

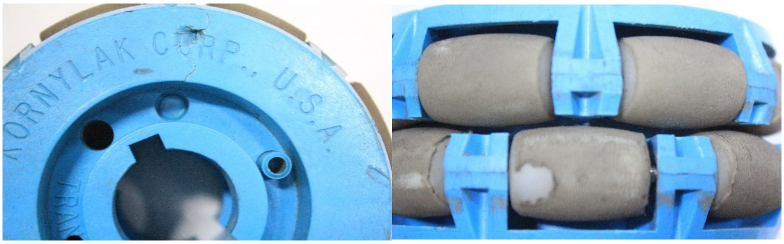

Fig. 2. Left: A break caused by hits. Right: Worn rollers

To solve those problems, we design a new kind of Omni-directional Wheel, as shown in Fig. 3(Left). We use aluminum alloy, which is stronger than plastic, to make the wheel hubs, the shafts and the bearings of the small rollers so that the wheels can withstand more powerful hits. The diameter of the roller bearing has been increased in order to support more weight. Moreover, new rollers that have thicker and tougher rubber wrapped around them are used, and with this additional rugged tread they are able to provide larger friction force to actuate the robots. A new structure with rollers fixed on the wheel independently of each other has been proposed, as shown in Fig. 3(Middle) and Fig. 3(Right), to improve the time and economy efficiency when the rollers need to be replaced.

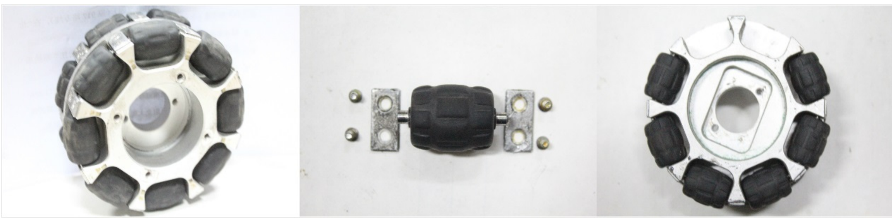

Fig. 3. Left: the new Omni-directional Wheel. Middle: Roller. Right: Wheel hub

2.2 Ball Handling Mechanism

The ball handling mechanism is comprised of a servo motor, two spring shock absorbers and components which are used to fix and link. The two ball handling mechanisms are fixed on each side of the kicker. The robot can dribble the ball better with the servo motors rotating. While colliding with other robots, the ball handling mechanism can control the ball and protect the possession of the ball from other robots [3]. Moreover, with the help of the ball handling mechanism, the robot can perform some movements like making a sudden stop or turning with speed.

In 2011, to pursue a better effect of dribbling, we managed to use active ball handling mechanism through fixing servo motors on each side of the kicker. Then, the robot could dribble the ball with more motions. It also decreased the failure rate of dribbling and increased the success rate of attack. Based on the result of tests in the 2011 and 2012 RoboCup competitions, the solution performed as well as we expected (Table 1). Some defects, however, were exposed as well. Due to the fierce fighting among robots during a match, the huge impact damaged the ball handling mechanism. Because a precise relative position between the ball handling mechanism and the ball were required, any deformations of the damaged ball handling mechanism would influence dribbling performance. Even more, sometimes the robot might not draw the ball into the ball handling mechanism or could not dribble the ball stably, finally result in losing the possession of the ball and failing to execute the robot's motion strategy.

Table 1. The average times of losing ball per game

Before 2011	In 2011	In 2012
30	20	14.5

In order to resolve these defects, we considered two schemes. One was to use stronger materials and increase the components' thickness. Another was to add new components to absorb the force of the impact. Considering the available mounting space and the weight of the robot, we decided to use scheme two.

We fixed spring shock absorbers which can absorb the force of the impact from fighting to protect the ball handling mechanism. The suspending cushion structure and the axial cushion structure convert the fixed ball handling mechanism into a floating structure. Therefore, with the help of two absorbers, the ball handling mechanism can effectively absorb the impact coming from both the axial direction and radial direction. Finally, this design protects the ball handling mechanism from damage.(Fig.4)

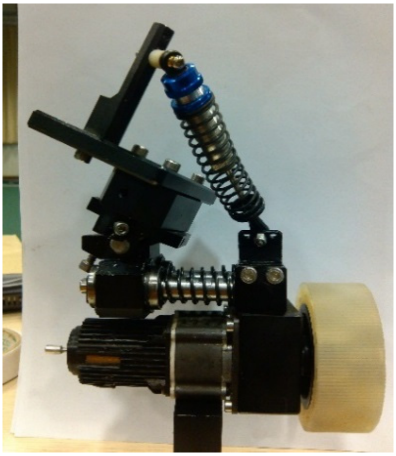

Fig. 4. Spring shock absorber

3 Software System

The software system of Water's robot is comprised of the robot module and the coach module (Fig.5). The software system starts with the robot image processing module which receives images of the robot's 360 degree of view from the camera, and then processes the images to calculate the coordinates of the ball and the robot's position and send to the coach module. The coach module is going to make decisions based on the coordinates received from the image processing module. Then, the coach module sends these decisions, such as the destination coordinates of the robots' motion, the

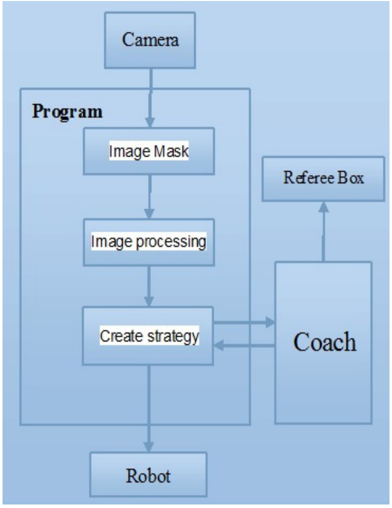

Fig. 5. The flow chart of robot's match system

robots' role in the game and so on, to each robot. Based on the commands from the coach module and the output of the image processing module, the robot itself will plan motion its strategy through its motion decision module.

This year, in order to improve the match system's efficiency and precision in localization, some improvements were made to the image process module and localization function. Following are the details.

3.1 Mask

Last year, the image processing module using OpenCV[4] segment the input image received from the camera by color directly (Fig.6).

According to the new rules, audiences have been allowed to stand around the field at their will. The colorful clothes audience members wear means much more interference for the robots. Therefore, detection approach that is based on simple color division and shape recognition is not reliable. More interference means that there is more information that the image processing module has to process, which is a heavy burden for the robot.

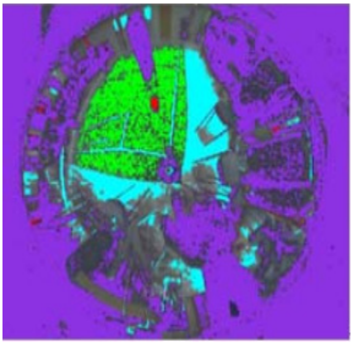

Fig. 6. The image divided by color

In order to improve the reliability and recognition efficiency of the image processing module, a mask function is introduced before the image processing module. Thus, the image received from the camera is going to be preprocessed by the mask function to reduce computation complexity.

Based on the location information received from the last image, the mask function generates a set of mask data named MD1 which excludes pixels that are outside the match field (as determined from the previous image). Meanwhile, the mask function also generates another set of mask data named MD2 which excludes pixels occluded by the robot. Then, the mask function combines MD1 with MD2 and gets the final set of mask data named MDF. Next, the mask function compares the picture from the camera (Fig.7) with MDF (Fig.8). This step excludes the image's pixels of robot occlusion and objects outside the match field. This result is sent to the image processing module as input.

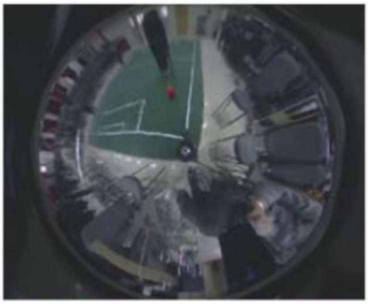

Fig. 7. The 360 degree picture camera takes

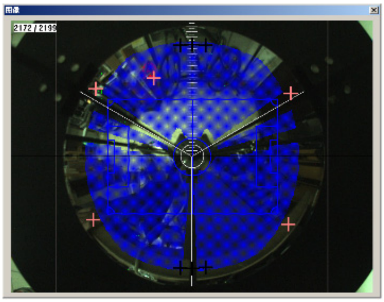

Fig. 8. The picture the mask function processes

3.2 Localization

Before 2012, based on the paper of Brainstormers Tribots[5], the localization function for determining the robot's location in the field first binarizes the image, in which the pixels including in the color threshold of green are defined as the field, the pixels included in the color threshold of white are defined as lines, and the other pixels are bypassed. The image devolved by the first step is named BD1. Then, the localization function scans the BD1 by a set of radials which set its origin at the center of BD1 with a 2.5 degrees difference in their direction. At the same time, the localization function also scans the BD1 by lines of 20 pixels' displacement from left to right and from top to bottom (Fig.9). When a scan line detects white pixels located between green pixels, the function is going to work out the weighted center of the white pixels. After scanning, all centers are saved in an array named WA1. Then, the function is going to restore the WA1 to a new array which can compare with the match field by distortion correction. This new array, without distortions, is named LTA. Next, we compare the LTA with the recent period's localization data, to exclude pixels outside the match field, and finally get an array named FLA. The FLA is continuously compared with the field template (Fig.10) by the point to point ICP method[6]. Finally, it solves for the robot's location, named FRL, and orientation, named FRA, relative to the field. Due to the fierce antagonism between robots during the match, the robot's orientation would suddenly change or the camera's image becomes blurry and the robot is going to locate itself in the wrong location. It becomes clear that the localization function which is only based on images is not reliable.

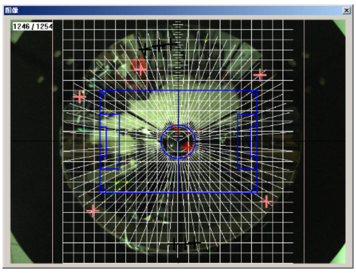

Fig. 9. Grid method to scan the white line of image

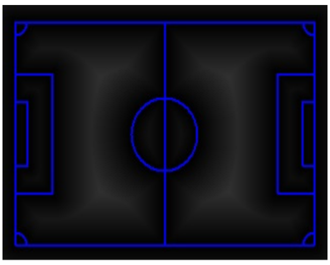

Fig. 10. Field template

In order to improve localization precision, efficiency and reliability, the team manages to ask help from AI area. With high integration and low cost IMUs, however, being popularized by their use in cellphones and other electronic products in recent years, such as MPU 6050[7], we decide to use the IMUs to assist the robot localize itself, which is cheaper and easier to realize. More sensors are added to help the robot to determine its location. The localization function fuses the angle got from the IMUs (Fig. 11) with the FRA, then determines the robot's orientation.

When fusing, the localization function first gets the initial angle named A1' from the IMUs. Then, the localization function gets an angle named A1, from IMUs, by subtracting AVGDA from A1'. Subsequently, the A1 and the FLA together are compared with the field template by the ICP method and the Steepest Descent method[8]. This process works out a final angle name A2. DA results from subtracting A1 from A2. AVGDA is the average of the sum of the DAs from all the previous iterations.

To further improve the precision of robot's location, the localization function also collects data from the odometer of the servo motors. The data from the odometer is called A3. Using the weighted average method, A2 and A3 are calculated. The output angle A3 is sent to the coach module. At this stage, the estimation precision of the robot's orientation can reach less than 3 degrees.

Fig. 11. 6050 IMUs

4 Experimental Result

4.1 Omnidirectional Wheel

Before 2012, a robot at least needed two to three sets of wheels. The cost was more than a thousand dollars. Robot failure caused by damage of the wheels happened frequently. This year, after we designed new wheels, all the wheels stayed intact. The new wheels also improved the time of the robot on duty. As the robot needs fewer wheels than before, the cost of the robot on wheels drops to four hundred dollars. Its reliability, robustness and durability is much better than the wheels we bought before. Moreover, these wheels can adapt to most environments, which enables us to use it in the real world and ordinary environment.

During the competition, we did not replace any wheels. Moreover, the new wheels provided larger grip force than before.

4.2 Ball Handling Mechanism

Through the test in 2013 RoboCup World Cup, the failure rates of dribbling fell, the average times of losing the ball per game are shown in Table 2. In fierce competition, the ball handling mechanism still can hold the ball well.

Table 2. Dribble Failure Times Per Game

Dribble condition	Previous Year	Current Year
Free dribble	4.3	1.7
Dribble in fierce competition	10.2	5.9

4.3 Mask

As Fig.12(Left) shows, we put two balls in the field. Ball1 is the target we wish to detect. Ball2 is an interference. Without the mask function, the image processing module binarizes both of the two balls in the image and fixes the center on Ball2 which does not belong to the match field. The solution is to use the mask function (Fig.12.Right) The mask function will pre-process the image before the image processing module takes over. The function excludes Ball2 which is outside the range of the mask. Then, the picture is sent to the image processing module which only needs to process Ball1. As the result, the center correctly is identified as Ball1.

During debugging, we always assigned our members to stand around the field to simulate interference to the robot. Therefore, we were able to test the effectiveness of the mask function (Table 3). The rates of fault recognition dropped by forty percent and the computation time cost of the image processing module decreased (Table 4).

Table 3. Wrong Recognitions Per Ten Tests

	Without Mask	With Mask
Ten balls around the Field	5	1
Ten people wearing green	7	2

Table 4. Image Processing Unit Time Per Iteration

	Without Mask	With Mask
Time	3ms	2ms

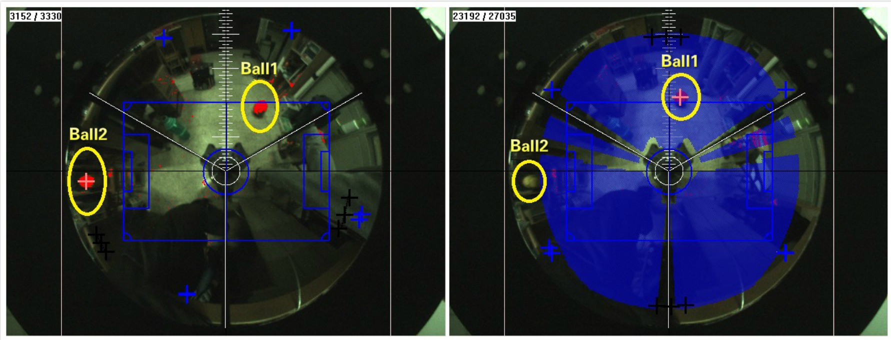

Fig. 12. Left: Not using template, red cross is on the Ball2. Right: Using template, red cross is on the Ball1.

In these pictures, the left ball is Ball2 and right is Ball1.The Red Cross is centered on the ball.

4.4 Localization

During a match, robots crash frequently while trying to get the ball. When a crash happens, the robots may be suddenly turned into a new direction. Therefore, the image received from the camera will suddenly change. Using the FLA which is just derived from the lines cannot be matched to the previous FLA due to the sudden change in physical location. It will result in the FLA being unable to match to the field template (Fig.13 Left). With the help of the IMUs (Fig.13 Right), the A1 corrects the FRA, so that the FLA can match with the field template and ensure the robot is able to determine its correct location, as shown in Fig. 13(Right).

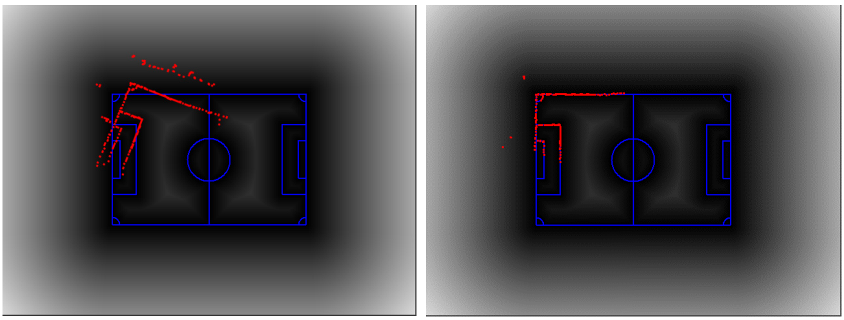

Fig. 13. Left: Not using IMUs, field data can't match field template. Right: Using IMUs, field data match field template after using IMUs data correction (red dots are field array data in the picture).

4.5 Competition

Via the test in this year's competition, all the changes we made this year resulted in a good effect. We did not change any omnidirectional wheels from preparing phrase to the end of competition. Moreover, the rubber wrapping around the rollers was almost always intact. Second, the failure of ball handling devices reduces. We just repaired a few times and replaced none. On software part, after introducing mask function, the robot could exclude the white interferences around the field. Furthermore, the mask function also decreased the time spent on image processing module from 5ms to 3ms. The localization also performed well. While collision with other robots, the IMUs could fix the wrong location information got from the suddenly changing image. It improved the precision of the robot's localization. Additionally, the data got from the IMUs reduced the computation time of the ICP method from 3ms to 2ms. In conclusion, all improvements made this year influenced the performance of our robots in a positive way. These improvements reduced the time we spent on repair and improved the time of the robot on duty. Finally, based on these improvements and the good performance of whole robots, we won the championship of 2013 RoboCup MSL.

5 Conclusion

Through these years' improvements, the collision among RoboCup MSL robots is more intense than before. Therefore, the robot requires more reliable mechanical design than before. Moreover, in order to hold and dribble the ball stably, the robot's ball handling device is improved, so that the robot can handle ball even violently impacted. The robot's software system is also updated. Thus, the robot can validly avoid the collision and possess the ball well.

How to command robots to cooperate efficiently and develop abundant tactics remains a very interesting research topic. Additionally, to test the efficiency between cooperation among five robots with different specific is also very challenging.

All of the above is what Team Water is going to focus on in the next year.

References

1. Zhang, W., Zhang, B., Zhu, M.: Intelligent Soccer Robot System. Tsinghua University Press, China (2010)
2. Ribeiro, F., Moutinho, I., Silva, P., Fraga, C., Pereira, N.: Controlling Omni-directional Wheels of a MSL RoboCup Autonomous Mobile Robot
3. de Best, J., van de Molengraft, R.: An Active Ball Handling Mechanism for RoboCup. In: 2008 10th Intl. Conf. on Control, Automation, Robotics and Vision, Hanoi, Vietnam, pp. 17–20 (December 2008)
4. Bradski, G.R., Kaehler, A.: Learning OpenCV: Computer Vision with the OpenCV Library. Reilly Media, Inc., USA (2008)

5. Lauer, M., Lange, S., Riedmiller, M.: Calculating the Perfect Match: An Efficient and Accurate Approach for Robot Self-Localisation. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) RoboCup 2005. LNCS (LNAI), vol. 4020, pp. 142–153. Springer, Heidelberg (2006)
6. Low, K.-L.: Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration
7. MPU-6000 and MPU-6050 Product Specification Revision 3.2. InvenSense Inc. (November 16, 2011)
8. Haykin, S.O.: Neural Networks and Learning Machines, 3rd edn. China Machine Press (2011)