Owaribito-CU

Tomoichi Takahashi, Yasunori Nagasaka, Hironobu Fujiyoshi, and Takashi Fujii

Chubu University, 1200 Matsumoto, Kasugai, AICHI, 487-8501 Japan
ttaka@isc.chubu.ac.jp

1 Introduction

The Owaribito robot soccer team has competed in RoboCup at Stockholm (1999) and Seattle (2001) as a joint team of Aichi Prefectural University and Chubu University. This year, two universities participate in RoboCup 2002 as individual teams.

This paper describes features of Owaribito-CU robot soccer team. Our robot is the same as Owaribito, that has no special devices for kicking and dribbling a ball. The features are

- carrying a ball by cooperative movement of robots,
- target location by calibrated camera model,
- viewer of robot motions.

.

2 Team Development

A number of students contribute to Owaribito-CU 2002 that was based on Owaribito 2001. They are Kazuhiro Ooura, Hiroyuki Orio, Hidetsugu Kondou, Kuniyasu Shibata, Hiroaki Sawano, Masanori Kizaki, Daisuke Tasaki, Yoshihiro Kitou, Syouichi Shimizu, Takayuki Nishi, Takuya Ina, Tomokazu Terai, Motoki Ido, Takehisa Shinma, Isao Harada.

3 Robot motion in formation

Robots' movements in a form are demonstrated in Cornell Univ.'s team [2]. They move and pass a ball with keeping specified formations such as circle formation.

The purpose of our robot control is to carry a ball. Fig. 1 shows the image of the cooperation play of three robots. Robots carry a ball with holding the ball in the center of them. The robots move in the formation, as a result of it, it blocks the ball from opponents' attack. The play looks like maul play in rugby not soccer [1].

We compare two methods to calculate positions of robots at next steps (Fig. 2). The middle solid line in the figure shows a planned trajectory of the midpoint of robots. The dots on the trajectory indicate the point at t_i.

method 1: Positions and directions of robots are calculated independently from the trajectory.

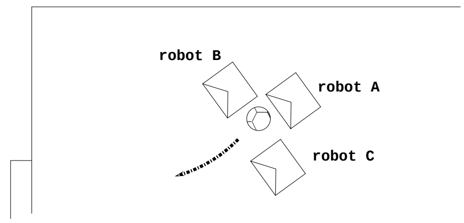

Fig. 1. image of maul play by three robots

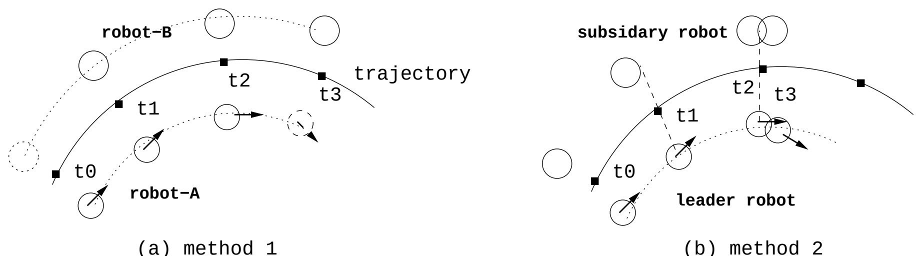

Fig. 2. methods of position calculation

method 2: Position and direction of a robot (the robot is called a leader) are calculated at first. The other robot's position and direction are calculated from the leader robot.

The two methods are equivalent in a simulated world, however, they give different results in a real world where a robot may not move as commands, or the images may not be grabbed clearly. The right figure in Fig. 2 shows a case that the leader robot is stacked at t_2. Then the other robot must stay there to hold a ball between them and not go as in the left figure. Preliminary experiments say method 2 is better than method 1.

4 Estimating target location by calibrated camera

Calibration of the internal and external parameters of the camera is necessary in order to determine how each pixel in the image is related to a target (a robot or a ball) location in the field.

Given a calibrated camera and an image pixel corresponding to the center of a target, the location of the target is calculated by ray intersection as shown in Fig. 3.

step 1: A viewing ray (x_v, y_v, z_v) is computed by internal parameter and a position of the target in the distorted image plane (X_d, Y_d).

$$\begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} = R^T \cdot \begin{bmatrix} SX_d(1 + kr^2) \\ Y_d(1 + kr^2) \\ f \end{bmatrix} \tag{1}$$

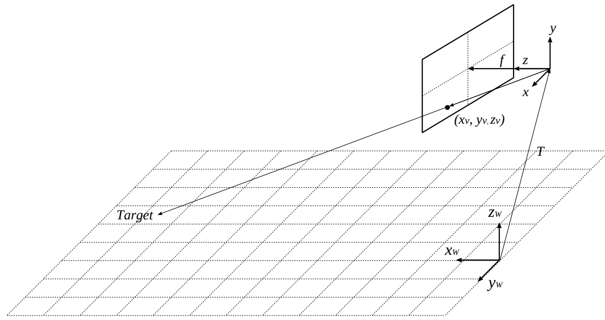

Fig. 3. Ray intersection

where R is the orientation of the camera; f is the focal length; S is the aspect ratio; k is the radial distortion coefficient and r is the distance to the image center. All these parameters are estimated in a preceding camera calibration step.

step 2: A line in the 3D world is constructed as follows,

$$\begin{bmatrix} x_W \\ y_W \\ z_W \end{bmatrix} = \alpha \begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} + T \tag{2}$$

where T is the position of the camera center in the world coordinate system and it is given by the camera calibration step as well.

step 3: We assume the ground plane is horizontal and $z_W = 0$. α is estimated as following.

$$\alpha = -\frac{T_z}{z_v} \tag{3}$$

Then, the world coordinate location of a ball $(x_W, y_W, 0)$, is the position at which a ray passes through the filed plane.

$$x_W = -\frac{T_z}{z_v} x_v + T_x \quad , \quad y_W = -\frac{T_z}{z_v} y_v + T_y \tag{4}$$

In the case of estimating location of a robot, since the height of a robot is known in advance, z_W can be set as the height of the robot.

We used 21 points in the field to calibrate a camera. In the experiments of estimating robot locations, the average prediction error was 13[mm]. It is accurate enough for controlling the robot motion as described in Section 3.

5 Viewer of planned motions and recorded motions

In real world, robots are not controlled as commands. It is necessary to monitor the difference between planned movements and the real movements. Logviewer records commands from a host computer, and the robots locations recognized from vision. Fig. 4 shows the log record.

The first S_expression indicates record kind - command or recognized robot's location - and time. The following S_expressions are the position of a ball, the positions and directions of teammate robots, and the positions of opponent robots in case of recognized robot's location. Fig. 5 shows the display of viewer.

```
(v(0)(134.869416,73.473804)
 ((0,326.000000,-24.886186,-7.956355)
  (1,0.000000,83.006598,89.055034)
  (2,333.000000,50.792165,67.023144)
  (3,326.000000,-24.886186,-7.956355)
  (4,326.000000,-24.886186,-
7.956355))
 ((0,215.686529,68.706128)
  (1,217.188591,83.667973)
  (2,132.572440,151.866150)
  (3,254.740137,73.925376)
  (4,0.000000,0.000000)))
(v(1)(134.869416,73.473804)
 ((0,326.000000,-24.886186,-7.956355)
  (1,0.000000,83.006598,89.055034)
        :
(c(2)((0,0,0)(1,0,0)(2,0,0)(3,0,0)(4,0,0)
(v(2)(134.869416,73.473804)
 ((0,326.000000,-24.886186,-7.956355)
        :
```

Fig. 4. Log file

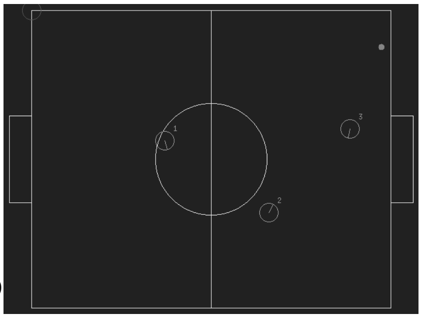

Fig. 5. Display of Logviewr

6 Conclusion

Maul play can be used as two ways to attack to the opponent goal and defend by holding the ball. We employ calibrated camera model to calculate targets' location and develop tools for estimating their movement.

References

1. http://www.rugby365.com/stories/laws/law_book/LAW_000812_19228.shtml
2. R. D'Andrea,*et al.*; The Cornell Roboucp Team, in *RoboCup 2000: Robot Soccer World Cup IV*, Lecture Notes in Computer Science, Springer, 2001, pp. 41-51