Hibikino-Musashi Team Description Paper

Yuichi Kitazumi[1], Shuichi Ishida[1], Yu Ogawa[1], Kota Yamada[1], Yusuke Sato[1], Mariko Oki[1], Hiroshi Thoriyama[2], Noriyuki Shinpuku[2], Yasunori Takemura[1], Amir A.F. Nassiraei[1], Ivan Godler[2], Kazuo Ishii[1] and Hiroyuki Miyamoto[1]

[1]Kyushu Institute of Technology, Japan
[2]The University of Kitakyushu, Japan
kitazumi-yuichi@edu.brain.kyutech.ac.jp

Abstract. Development of "Musashi Robot" which is an omni-directional mobile robot started three years ago. In this paper, we introduce the robot's system and the software algorithms. At first, we describe the safety concept. In the industrial world, safety of many kinds of robots has been considered. Now, in spite of using the robot by many amateurs, RoboCup engineers don't consider the safety. We propose safety rules for Middle Size League. Secondly, we introduce a new goalie arm system and a new kicking device which can change force and height of the kick. On the software side, we describe a self-localization algorithm using Monte Carlo Localization, and a ball getting algorithm using fuzzy control. Additionally, we present our scientific innovation contents: development of color constancy algorithm by using Self Organization Map.

1 Introduction

"Hibikino-Musashi" is a joint middle-size league RoboCup [1] soccer team. Members of the team are from three different research and educational organizations, all located in the Kitakyushu Science and Research Park, Kitakyushu, Japan. The three organizations are: Kyushu Institute of Technology, The University of Kitakyushu, and Kitakyushu Foundation for the Advancement of Industry Science and Technology.

The paper is organized into two parts. The first part introduces hardware system of our robot, and the second part explains about software system of the robot.

2 Hardware system

The omni-directional mobile "Musashi Robot" is shown in Fig.1. The design concept and the main features are as follows.

(i) Omni-directional motion
(ii) Omni-vision system
(iii) Strong kicking mechanism
(iv) Cable-free system as much as possible
(v) Simple modular architecture for easy assembly and maintenance

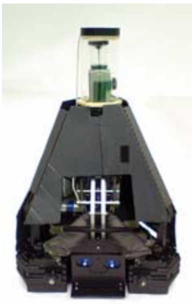

Fig.1 Photo of "Musashi Robot"

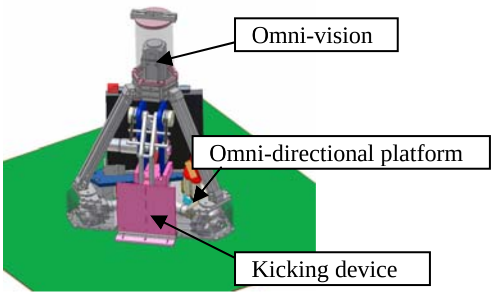

Fig.2 3D CAD design of "Musashi Robot"

"Musashi Robot" base is an omni-directional platform, and it has omni-vision (Fig.2). The dynamic and kinematics characteristics of the design allow for high maneuverability on the playing field. Each robot is equipped with 3 omni-wheels, each of them driven by a 70W DC motor. Gearboxes with reduction ratios of 12:1 are used to reduce the high angular speeds of the motors (7000 rpm) and to amplify the wheel's mechanical torques.

The velocity feedback is done by using 500 pulses digital incremental encoders. The velocity of the wheels is controlled by three Faulhaber motor drivers (MCBL 2805), each equipped with a RS232 communication port. The controllers read the pulse trains from the motor encoders and produce PWM output voltages for the motors based on a PID algorithm. The result is a mobile robot with maximum linear speed of 2.4m/s and acceleration of 2.5m/s^2.

As shown in Table 1, the only sensors using in the "Musashi Robot" are an omni-directional camera and three DC motor encoders. Figure 3 shows a chart diagram of "Musashi Robot" power system including a main Li-Polymer battery (nominal voltage is 25.9V). The necessary voltage for the camera and a micro computer power supply are produced by converting 25.9V to 12V and 5V, respectively. The power consumption of the robot is in average about 40W, and the operation duration of the robot is estimated to be 0.5h.

Figure 4 shows the "Modularity concept". It is necessary to have a clear overview of the robot system. Based on the chart diagram of the robot architecture; the possible modules should be defined by considering the similar hardware structures or similar mechanical connections. Musashi robot's modular architecture is realizing these functions in the concept of easy assembling, maintenance and transportation [2][3].

Table 1 Specifications the robot

Dimensions	Triangle: 500mm Height: 800mm Weight: 17.5kg
Actuators	DC-motor x 3 (Maxon, 24V, 70W) Motor driver x 3 (Faulhaber, MCDC2805) Omni-wheel x 3
Battery	Li-Polymer battery 7cells (25.4V, 2000mAh)
Operation	0.5h
Sensor	Omni-directional camera, DC-motor encoder x 3
Kicking device	DC-motor and torsion spring x 2

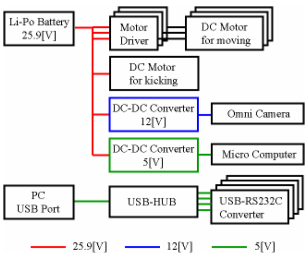

Fig. 3 System architecture of Musashi Robot

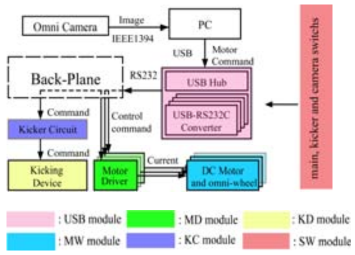

Fig. 4 System architecture of Musashi Robot

2.1 Safety Considerations in the RoboCup MSL

One of RoboCup's final goals is to win against the human World Cup champion team. Therefore, it is important to think about safety strategy. In our team, we try to reduce the risk through a risk-assessment by using international standard [4]. Additionally, we interviewed participants of the RoboCup Soccer MSL 2008 competition, and made a survey of the critical hazard in other team's robots so that we understand the present situation regarding safety. The following eight teams answered: EIGEN, Nubot, CAMBADA, CoPS, MRL, ADRO, MINHO, Strive, and the summary of the responses is given in Table 2. The questions were:

Q1: Are there any parts where safety design is considered in your robot?
Q2: Are there any parts which you feel are dangerous?
Q3: Do you have the experience of being injured by the robot?
Q4: Do you have the experience you feel danger in RoboCup competition?

The comments to each question were: Q1 – our robots have a sponge rubber around the platform, Q2 – the guard for keeping the ball has sharp ends, Q4 – when I was setting the ball on a free kick point as a referee, I was kicked by the robot.

Thus, it is important to survey both the necessity of safety measures and how to apply them in detail. Consequently, attitude to safety will improve. Furthermore, it will also become possible for the robots to play the soccer with humans. Our team is trying to improve the robots in reference to that goal.

As a first step, we observed the comment of Q4. Therefore, we studied the referee situation. Now, referee is a person that has the highest risk of being injured by the robot during the game in the robot soccer field [5]. Therefore, if a referee is provided with safety, the risk of the accident can be reduced dramatically. In RoboCup MSL case, it is impossible to fence the robots and keep the referee far from the robots moving area. In fact, for the referee it is necessary to have a direct interaction with the robots during the game, the same as in human soccer, to change the ball position, judge the game and so on. However, to decrease the risk, game organizers should explain to the referee and to the participants about the specifications of the robots, describe the expected mistaken behavior, and provide the results of risk assessment. After achieved accountability, the designers of the robot can escape the responsibility. However, the present RoboCup MSL rules aren't written about this significant subject. Therefore, when an accident happens, no one can take responsibility. It is important for the participants to remove critical hazards and to implement the safety measures beforehand. For the future of the RoboCup, we should make a global safety standard. As one of the efforts in that direction, a questionnaire would be beneficial.

Table 2 Results of questionnaire

	Yes [%]	No [%]
Q1	87	13
Q2	62	38
Q3	38	62
Q4	62	38

2.2 New Kicking Device

Musashi Robot has been using a spring kicking device for three years. The spring kicking device uses “Cam Charger” mechanism. In general three mechanisms are necessary to use energy of a spring: the first is a mechanism for charging the spring; next is a mechanism to lock the spring, and the last is a mechanism to release the spring [3]. The spring kicking device has these mechanisms realized by special designed of the cam shape. The spring kicking device’s advantages and disadvantages are as follows:

Advantages

- Strong kicking force: The spring kicking device is able of a strong shoot (up to 5.0 m/s) by series of torsion springs.
- Saving of the power: Current only is needed for rotating a DC motor [24.0V] in point of comparing the spring type kicker with solenoid type one.

Disadvantages

- Low response speed: “Cam Charger” mechanism needs time for charging the spring. Therefore, a series of kicks is impossible in a short time period.
- Difficult power management: Force of charging is generated by cam radius, so that kick power management with current is impossible.

In this year, we focused on a second disadvantage, and try to solve it on the new kicking device. This second disadvantage is a direct cause of a problem in which the robot always kicks the ball with constant power, regardless of its position and cannot score a point near the goal area. The Musashi Robot can kick the loop shoot, but the height of loop ball is almost constant so that the ball overshoots a goal’s upper bar.

To solve this problem, we developed a “variable kick unit” in this year. In the variable kick unit a kick plate is separated into two parts: an upper part and bottom part which has a lift plate to scoop the ball up. The bottom part joints to the upper part by passive link, and this passive link can be fixed or freed by micro solenoid coil on the upper part. The kicking device can kick a loop shoot when the passive link is fixed (the solenoid is ON,) because the kicking device’s function is the same previous one. When the passive link is free (the solenoid is OFF,) the lift plate cannot move with the upper part. Due to this, the kicking device can kick a grounder shoot, because the lift plate cannot lift up the ball. The solenoid coil is controlled by a PIC.

2.3 New Goalie Arm

In RoboCup 2008 MSL competition, some teams exhibited strong shooting abilities and high accuracy loop shoots (for example, Tech. United, CAMBADA). It is difficult to respond to such a shoot only by the goalie robot’s own motion. Therefore, we are developing a goalie robot with an arm that can expand and contract. The arm extends and contracts for 100 millimeters in one second, only in one direction (right, left and upwards), to comply with the RoboCup rules. In fact, three goalie robot’s arms are attached to left side, right side and upper side of the robot, respectively. The frame bar is made of PVC pipes, and the frame joint parts are made of ABS resin. The goalie robot’s arm is extended by releasing a compressed spring. The spring is compressed by reeling a wire that is connected to the moving parts (storage of arm) by using a DC motor. At the time of releasing the spring, a momentary operation that is quicker than reverse motion of the motor is needed. Therefore, an electromagnetic clutch is attached to the motor, and the compressed spring is released by releasing the electromagnetic clutch. The actuator is controlled by a PIC, which also controls the limitation of 4 seconds wait time.

3 Software system

In this section, we introduce the software system of "Musashi Robot". The software is composed of three parts: image processing, communications, and behavior. In the image processing part, the lines, the ball, and the goal on the field are recognized by extracting necessary information from the omni camera images obtained from IEEE1394 camera. The type of information that is used is the distance and the angle to the object. Position of the robot is estimated by self localization method based on the Monte Carlo Localization (MCL) algorithm by using the white lines distance information. The communications part transmits the signals from referee box and data between the robots as a multipoint-to-multipoint network. From these two processes, the robots choose behavior and execute their behavior. Furthermore, this year improvements in the robot's behavior were especially made in the approach to the ball that uses fuzzy control.

3.1 Self Localization System

Robots need to recognize own position on the field in order to play. This section describes the implemented self-localization algorithm. The method is based on the field lines detection and calculation of the self-position [6].

We employ a Monte Carlo Localization (MCL) method [7]. For our algorithm, the field model is a Cartesian coordinate system with the origin at the center of the field. The robot's state is represented by a vector $\mathbf{x} = [x, y, \theta]^T$ which consists of a position (x, y) and an orientation θ. We provided the algorithm, which detects only orientation, made the posture θ ingredient known in MCL, and planned the dimension reduction of the state vector. The orientation detection is explained further below. For localizing, we have to construct the posterior density $p(\mathbf{x}_t \mid \mathbf{y}_1 ... \mathbf{y}_t)$ from the state of a robot $\mathbf{x}_t$ and the sensor data $\mathbf{y}_t$ at the current time t. In the particle filter methods, a probability density is represented by a set of N random samples (particles). The method proceeds in two phases.

In the first phase we predict a current state of the robot. That is specified as a conditional density $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1})$ from the previous state $\mathbf{x}_{t-1}$ and a control input $\mathbf{u}_{t-1}$. The predictive density is obtained by the following integral. For our algorithm, we set the control input $\mathbf{u}_{t-1}$ as odometry data and add it to each particle.

$$p(\mathbf{x}_t \mid \mathbf{y}_1 .. \mathbf{y}_{t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{y}_1 ... \mathbf{y}_{t-1}) d\mathbf{x}_{t-1} \quad (1)$$

In the second phase we update the density according to the sensor data $\mathbf{y}_t$. The likelihood of $\mathbf{y}_t$ at state $\mathbf{x}_t$ is represented as $p(\mathbf{y}_t \mid \mathbf{x}_t)$. The posterior density is obtained using Bayes theorem.

$$p(\mathbf{x}_t \mid \mathbf{y}_1 ... \mathbf{y}_t) = \frac{p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{y}_1 ... \mathbf{y}_{t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_1 .. \mathbf{y}_{t-1})} \quad (2)$$

Sensor data $\mathbf{y}_t$ is distance to the field line. The state is compared to $\mathbf{y}_t$ and the likelihood is updated of each particle. After that, weighted particles are normalized and re-sampled. Re-sampling is done according to the weight of each particle: new particles are generated around the particles that have high likelihood.

In the RoboCup soccer field most constituents are straight lines or perpendicular segments of lines. The robot's orientation is detected by searching for inclination of the straight line ingredients in the circumference seen from the robot. This approach is simple, high-speed, and the derivation of the robot orientation becomes efficient.

Our task is to use MCL in the RoboCup environment. It is possible to falsely detect orientation because the form of the field is symmetric against the center of the field. We solved the false detection by using compass sensor for this problem.

3.2 Approach to the Ball by Fuzzy Control

In our team's algorithm, the decision of which robot goes to get a ball is based on the distance of ball to robot. The robot which is the nearest to the ball will go to get the ball. However, this method has a problem that the ball which is behind the robot is difficult to reach. Because of that reason, the orientation of the robot to the ball is also considered in the decision.

In a fuzzy program, we can make rules to cope with various situations [8]. The rules are given by Potential Membership Function (PMF). The x-axis of PMF is orientation of the robot and y-axis of that is a grade which represents priority of the orientation. The direction which robot move is decided by combined PMF that has the highest grade.

In the following we describe the main rule to define PMF. First, we make a basic PMF represented by a linear function that has grade 1 in the direction of the ball, and grade 0 in the direction opposite to the ball. In short, a triangle is drawn with the top in the direction of the ball. With this one rule, the robot moves in the direction of the ball because the grade of the ball direction is the highest. However, if the robot goes to get a ball that is behind the robot without changing the robot's direction, the robot cannot get the ball because the direction of the robot doesn't look to the ball. Additionally, if the robot changes direction to face the ball, it is difficult to turn for shoot motion after catching the ball. To deal with such situations, we need other rules for going around behind the ball. The rule uses a value of angle between the goal and the ball and it changes the direction of the robot motion. If there is large angle, the robot goes around by bigger angle. On the other hand, if there is small angle, the robot goes almost directly to the ball. Finally, there is another rule that is the changing the robot's direction for getting the ball more smoothly. The robot changes its direction depending on the distance between the robot and the ball. If the robot is near to the ball, it needs to turn large angle to catch the ball.

According to these rules, the robot faces in direction of the goal when it gets the ball. Due to this, it is easy to change into shoot motion. We take in consideration such method so that robot moves not only directly to the ball, but also goes into a position which is preferred for after movement like shooting to the goal.

3.3 Development of Color Constancy Algorithm using Self Organizing Map

In this section, we present the color recognition algorithm based on color constancy algorithm by using Self-Organizing Map (SOM) for the robot vision system. SOM is an unsupervised learning algorithm that performs topology-preserving transformation from higher-dimensional vector data spaces to low map spaces. The SOM has become a powerful tool in many areas such as data mining, data analysis, data classification and data visualization [9]-[12]. Our robot vision system is based on YUV and HSV color map spaces. Both color maps have different vector spaces, then, some objects are recognized by using thresholds to use logical addition to YUV and HSV thresholds [13]. To realize a robust robot vision system against light changing environment, in our new robot vision algorithm, recognition of the light color environment and the threshold parameters are both estimated by using SOM. We put the four kinds of basic colors: green, red, blue and white under the omni-directional camera in Musashi Robots (Fig. 5).

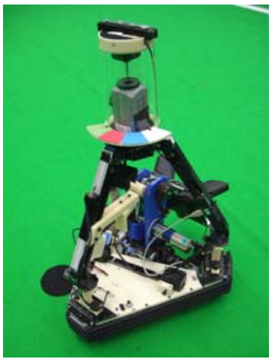

Fig.5 Musashi Robot with color calibration board

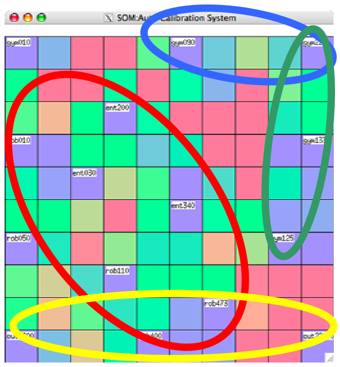

Fig. 6 Feature map of the table 3 learning data

Table 3 Environment of light condition (4 kinds of condition)

Class	Location	Kinds of light	Light (lx)
1	RoboCup room	No light	10
2	RoboCup room	SN	50
3	RoboCup room	FL	110
4	RoboCup room	FL	400
5	RoboCup room	FL and SN	473
6	Entrance	SN	30
7	Entrance	FL and SN	200
8	Entrance	FL and SN	340
9	Outside	SN	7600
10	Outside	SN	22000
11	Gym	SN	10
12	Gym	SN and WM	90
13	Gym	SN and OM	125
14	Gym	SN, OM andWM	133
15	Gym	SN, OM an WM	220

SN: Sunshine, FL: Fluorescent light, WM: White mercury lamp, OM: Orange mercury lamp

The learning data $\mathbf{\theta} = (x, y)$ are made manually. At first, input data $\mathbf{x} = (Y_g, U_g, V_g, Y_r, U_r, V_r, Y_b, U_b, V_b, Y_w, U_w, V_w)$ are written in the learning data file by using the three color calibration points' YUV values average in the image. *Y*, *U* and *V* are values of YUV in 8bit color map. This input data vector is normalized to -1.0 to 1.0 vectors. Then, output data $\mathbf{y}$ are also written in the learning data file, and output data thresholds are adjusted by the conventional "Musashi Robot" vision system method.

The environment of light condition sets under the many kinds of the lights. Table3 shows data classes of light conditions in the 4 kinds of rooms which are RoboCup practice room, gym, entrance, and outside. The entrance and RoboCup practice room is used with fluorescent light condition and sunshine. The gym is used with white mercury lamp and orange mercury lamp and sunshine. For using the SOM algorithm, we expect to interpolate the thresholds values (that means input and output value).

At the result, we can recognize the non-learning data about 86%. Figure 6 shows the feature map of the SOM. This lattice means the one unit (neuron). Color of the lattice means blue is the shortest of the distance between each lattice, red is the most faraway of the distance between each lattice. The number means light condition and class number. Red circle means fluorescent light condition, Blue circle means white mercury lamp condition. Green circle means orange mercury lamp condition. Yellow circle means sunshine. Left upper side is dark light condition, and right down side is bright light condition. The leaning data sets are clustered in each environment. Therefore, this algorithm has the possibility to adjust the thresholds robustly.

3.4 The simulator

We developed a simulator by ODE (Open Dynamics Engine) last year. It is used to study the algorithms to get the ball, to check the team behavior, and other robot and team's features. In this year, we mounted the getting ball algorithm which was developed by this simulator on several robots, and succeeded to develop a good algorithm. More details about this can be found at http://hwm7.gyao.ne.jp/shimpuku/.

Acknowledgements

This work is supported by 21st Century COE program, Ooba Sangyou Co., Ltd., Nippon Telephone Center Corporation Co., Ltd., and Neural Image Co., Ltd.

References

[1] H.Matsubara, M.Asada, H.Kitano, "History of RoboCup and Prospects of RoboCup-2002", RSJ, Vol.20, No.1, pp.2~6, 2002, In Japanese

[2] A.A.F.Nassiraei, "Concept of Intelligent Mechanical Design for Autonomous Mobile Robots", Kyushu Institute of Tech., Ph.D thesis, 2007

[3] A.A.F. Nassiraei, Y.Takemura, et. al. "Concept of Mechatronics Modular Design for an Autonomous Mobile Soccer robot", CIRA 2007, Jacksonville, pp.178~183

[4] Y.Ogawa, Y.Takemura, et. al. "Introduction of the safety system to RoboCup Soccer Middle Size League", ROBOMEC 2008

[5] Y.Takemura, Y.Ogawa, et. al. "A System Design Concept Based on Omni-Directional Mobility, Safety and Modularity for an Autonomous Mobile Soccer Robot", Journal of Bionic Engineering, Supp1, pp.130-137

[6] "Line based robot localization under natural light conditions", A. Merke, S. Welker, and M. Riedmiller, In ECAI 2004 Workshop on Agents in Dynamic and Real-Time Environments, Valencia, Spain, 2004.

[7] F. Dellaert, D. Fox, W. Burgard, S. Thrun, Monte Carlo localization for mobile robots, in: Proc. IEEE International Conference on Robotics and Automation (ICRA-99), Detroit, MI, 1999.

[8] R.Tsuzaki, K.Yoshida, "Motion Control Based on Fuzzy Potential Method for Autonomous Mobile Robot with Omnidirectional Vision", Proc. of the Robotics Society of Japan, Vol.21, No.6, 656-662.

[9] T.Kohonen, "Self-organized formation of topologically correct feature maps", Biol. Cybernetics, vol.43, pp.59-69, 1982.

[10] Saalbach G.Heidemann and H.Ritter, "Parametrized SOMs for Object Recognition and Pose Estimation", Proc. of ICANN 2002, pp.902-907, 2002.

[11] Barreto, G.A., Araujo, A.F.R. Ducker. C. and Ritter. H., "A distribution of a simulator for robots and its application to position estimation by self-organization", Technical Report of IEICE, Vol.100, No.687, NC2000-147, pp.149-156.

[12] K.Ishii, S.Nishida, T.Ura, "A Self-Organizing Map Based navigation System for an Underwater Vehicle", Proc. of ICRA'04, pp.4466-4471, 2004.

[13] K.Azeura, I.Godler, "Color Sampling with Omni-Directional Camera in Robot Soccer", Proc. of RSJ 2006, Okayama, 1B25.pdf, 2006, In Japanese.