INDOOR NAVIGATION AND WORKPIECE OUTLINE RECOGNITION FOR AUTONOMOUS CONSTRUCTION MACHINERY

Makoto Oshio, Kenniti Hamamoto, Yuji Hiramatsu, Yoshinori Matsunaga
Technical Research Institute
Kajima Corporation
2-19-1 Tobitakyu, Chofu-Shi, Tokyo 182-0036

Abstract: We study basic functions for autonomous construction machines, such as indoor navigation, workpiece outline recognition and machine learning method. The indoor navigation is based on laser scanners and 2D map matching of the surrounding walls. The positioning results are accurate in spite of obstacles. We simplify object recognition problem without any visual markers by focusing on the workpieces with straight edges and a single chromatic color. The silhouettes of the workpieces are separated by a saturation filter and symbolized as the groups of line segments. Finally, new machine learning method is proposed. We focus on an improvement of the Q-Learning reward system. In our proposed method, machines can obtain the rules and surrounding maps with less parameter and action rules known in advance.

Keywords: Autonomous, Indoor Navigation, Vision System, Recognition, Machine Learning

1. INTORODUCTION

Unmanned machinery operation would drastically improve construction work environment. But it is very difficult to apply automation methods widely used in many factories to the construction field, because of unpredictable changes in the field.

To adapt these changes, one possible approach is a remote control. A remote shovel system used in pneumatic caisson construction site works fine and eliminates dangerous work in high-pressure environment. But applied cases are generally limited such as disaster restoration work due to its poor efficiency in complex environment.

There are two kinds of approaches to improve the work efficiency. One is aiding a machine operator by improving the interface using technology such as bilateral control and virtual reality [1]. The other approach is autonomous machines. Currently, it seems nearly impossible to develop completely autonomous machines for industrial use, because of their reliability and the cost. Therefore, semi-autonomous machines, partly autonomous machines, are realistic [2]. In either case, basic autonomous functions are required. We study some of them, such as indoor navigation, workpiece recognition and machine learning method. This paper is the overview of our researches.

2. INDOOR NAVIGATION

Unlike outdoor navigation system, GPS receivers cannot be used in indoor environment. However there are some indoor positioning system available, they requires some devices attached to environment. Considering changing environment and maintenance work, it is not preferable to attach any devices to somewhere other than machines in construction fields. Therefore, we study laser scanner based navigation system. We choose the navigation method with known map matching rather than the method with self-mapping, because there are many things in construction site, which cannot be modeled in advance and may be moved from one place to another. We choose the surrounding walls as stable navigation reference.

2.1 Test field and machine

Our test field is set in a large building. The walls are the same shapes vertically, however, there are some uneven parts horizontally. Figure 1 shows the ground plan of the test field. Figure 2 shows the test machine, which is a remote-controlled backhoe modified for computer control.

Two SICK laser scanners are set to the test machine. They are attached back to back to acquire 360-degree horizontal distance data with 0.5 degree resolution.

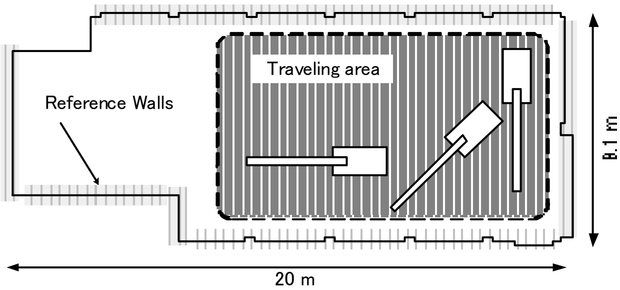


Fig. 1 Indoor Test Field

Fig. 2 Test Machine and Laser Scanner Setting

The machine is also equipped with other sensors, such as stroke sensors and encoders.

2.2 Data process

The Data process consists of the following three parts, finding wall groups from laser scanner data, walls identification and final data matching.

a) Finding wall groups

After converting acquired laser scanner data to coordinate data, line segments are picked up based on the linearity of data coordination and contained data number. Then, the line segments are grouped into walls by the direction and position of the approximate line of each segment. A position allowance is set to prevent the line segments of the same wall from wrong separations. Figure 3 illustrates this process.

Once wall groups are obtained, wall parameters, approximate direction and position of each group, are calculated based on weighted average of the contained segments.

b) Walls identification

The system identifies the wall groups with reference walls by relative angles and distances. If the all reference walls are identified, the approximate machine position and angle is uniquely estimated. However, there are dead angles on the laser scanner due to the machine arm and obstacles. Because of these dead angles, multiple approximate positions and angles case could exist. In that case, the system evaluates every approximate positions and angles in final data matching stage.

With the wall layout in the test field, the system needs at least one vertical wall pair and the wall parallel to the one of them at the first estimation.

c) Final data matching

Since there are no specific reference points on the test field plane, the system evaluate the distances between projected data points and walls in final matching stage. A wall map is generated from the building's CAD data. Each wall section is represented by a wall ID number and line parameters including start and stop points on the test field plane. The system evaluates only the data in the identified wall groups to eliminate obstacle's data.

An initial position and angle is calculated by the weighted average of the identified wall data and projected points are calculated based on the initial position and angle. Then, the distances between projected data points and the walls are calculated to estimate the position error. Threshold of distance is set to eliminate effect of outliers. This matching process is iterated until the position error is lower than a preset value. Since the initial errors of rotation are very small, the rotation angles are not updated during the matching process.

In multiple initial position and angle cases, the system chooses the best result, which contains the largest number of the matched data.

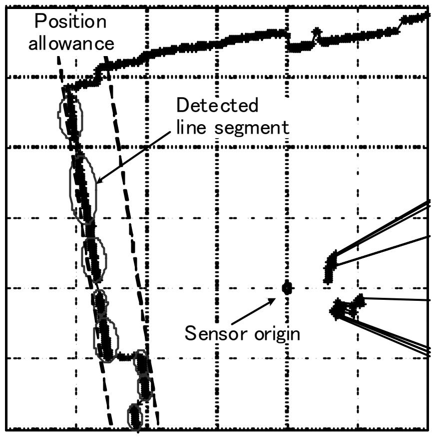


Fig. 3 Wall group finding process

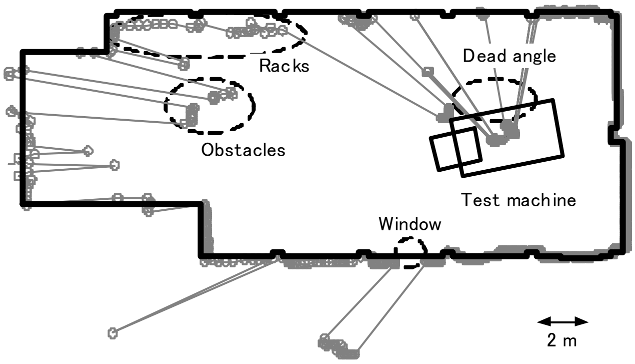


Fig. 4 The result of estimated position and posture

d) Map matching result

Figure 4 shows an example of the matching results. The system estimates the machine position and angle very well in spite of some obstacles and dead angle due to the machine arm.

2.3 Path planning

We use Q-learning method for path planning. Q-learning is the one of the Reinforcement Learning (for short RL) methods (for the details of RL, see book [3]).

The test field is divided into 50 by 23 cells and the cell size is 0.3 sq m. The postures (or angles) of the machine are divided into 36 states with 10-degree ranges. Available actions are four (forward, backward and two rotations). The map information including obstacles is given by the machine navigation system. Figure 5 shows the example of the numerical simulation results. The following are the parameters for this result, learning rate a = 0.3, discount rate g = 1, rewards are r = 1 and –10 for valid and invalid movements respectively. Using the e-greedy, the machine learns optimal path with obstacles avoidance. e is given by the following equation.

$$\begin{cases} e = 0.1 & (\text{if } 0 < ne < te/4) \\ e = 0.1/(1+(ne-te/4)) & (\text{if } ne \geqq te/4) \end{cases} \quad (1)$$

Where ne and te are the current and total episode number respectively. The eligibility trace parameter is 0.8. The reward, r, is set to be 10 and this leads the machine to choose same path in the following episodes.

Due to the large state numbers, the calculation time tend to be long in some obstacles settings. We find that setting some sub goals reduces the calculation time.

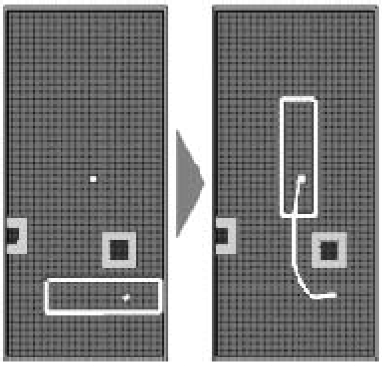


Fig. 5 Numerical simulation result of path planning

2.4 Indoor traveling test

Indoor traveling tests, path tracking and obstacle avoidance, are conducted in the test field. There is little dynamics about the machine traveling. Therefore, path tracking traveling test is done well. However, the machine sometimes takes undesirable rotation during the obstacle avoidance tests, because the machine is programmed to choose the shortest distance path after avoiding obstacles in spite of the restriction of movement direction. It would be easily improved, if smooth trajectory path is generated dynamically.

3. WORKPIECE OUTLINE RECOGNITION

3.1 Objective and test conditions

It is necessary to recognize workpieces by a semi-autonomous machine itself for an easy and efficient operation. However, visual object recognition is still challenging research area. One does not easily find general solution. Using visual markers is possible and reliable way. They have been widely used in motion capture systems. However, it may not be preferable that system totally relies on the visual markers in construction works, because additional on-site work may be needed due to the visual markers.

To simplify object recognition without any markers, we focus on object outlines and set the following test condition. The condition is that workpieces are composed of straight edges and the surface is a single chromatic color or painted in that. This seems reasonable considering assembly works in construction sites. We also use a gray baseboard to separate the outlines from images by a saturation filter.

3.2 System and test images

Our vision system is composed of an IEEE1394 VGA color camera (SONY DFW-V500) and a computer. The camera is calibrated to remove the lens distortion before tests. Four unpainted wood workpieces are placed on a gray

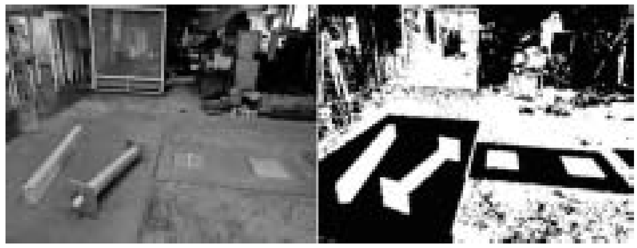


Fig.6 (left) Test image, (right) Filtered image

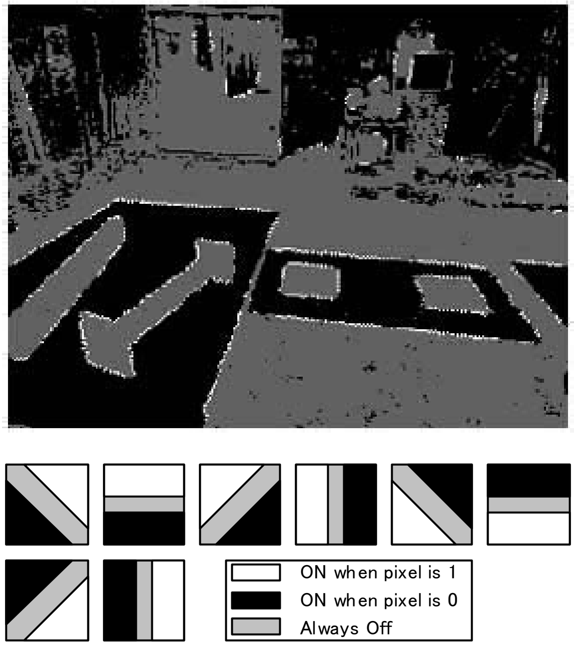


Fig.7 (upper) Detected Point, (lower) 2-D filter patterns

baseboard. Figure 6 shows an example of the test images after compensating distortion and the binary image with the saturation filter.

3.3 Edge detection by 2-D digital filter

The test images are converted to the binary images by the saturation filter. 2-D digital filters are applied for edge detection of line segments. To reduce calculation time, sampling points are selected every four pixels both x and y directions, when 2-D digital filters applied. Figure 7 shows the detected edge points and the 2-D digital filter patterns respectively.

3.4 Line segment and corner detection

Since the detected edge points may not be the exact edge points due to the sampling process, slightly position correction, moving to the nearest edge point, is applied to the necessary points. After that, the direction of each detected edge point is calculated based on an edge direction of the surrounding 15 by 15 pixels area. An iterated least squares fit is used to remove outliers. Figure 8 illustrates this process.

The detected edge points are grouped based on their positions and directions. Then, every edge point along with

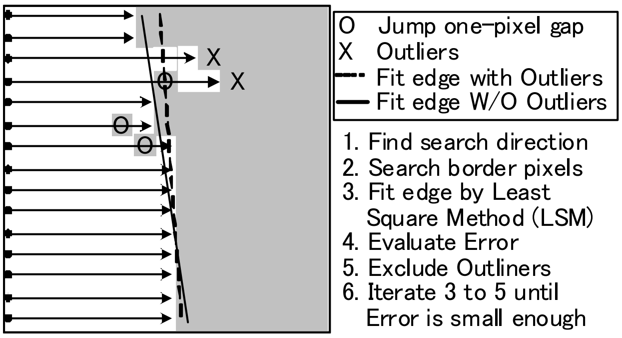


Fig.8 Edge direction finding process

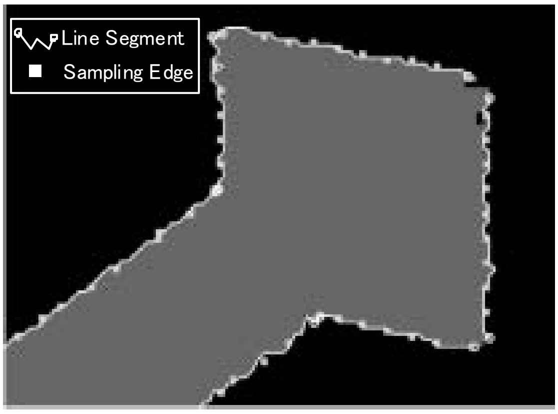
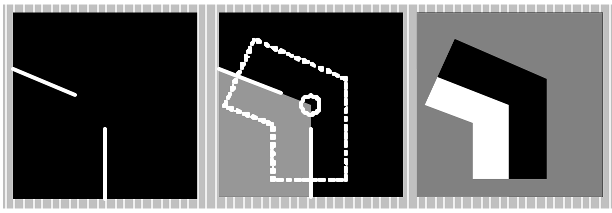


Fig.9 Detected line segments



(Left) The two line segments, which possibly connect at the corner.
(Center) Predict the corner point and determine the filtered area.
(Right) Generated 2-D filter

Fig.10 2-D filter for Corner detection

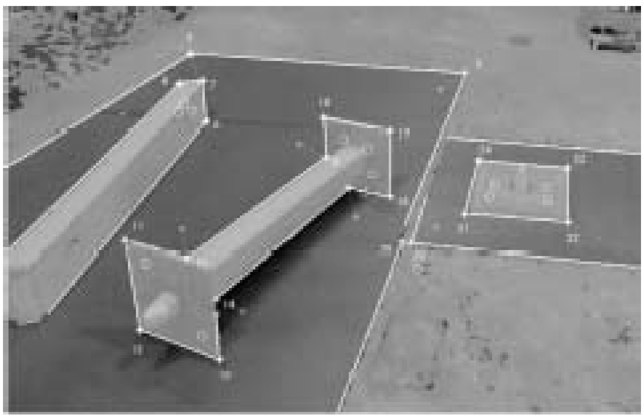


Fig.11 Final outline detection result

the groups is searched and included in the corresponding groups. Line segments are determined by applying a least square fit to each group. Error threshold level is set to remove arcs with large radius. The parameters of the fitted line segments are compared to check wrong division of the same line segment. Figure 9 shows an example of the line segment detection results.

The corners are detected using 2-D filters. The 2-D filters are generated from neighboring line segment pairs. Since the line segments and the corners are detected, the workpiece outlines are recognized not by the areas of the binary image but by the groups of the line segments. Figure 10 illustrates 2-D filter generation. Figure 11 shows an example of the final detection results.

3.5 Idea for workpiece Identification

As mentioned above, the silhouettes of the workpieces in the test image are symbolized as the groups of the line segments. One possible way to identify the workpieces is a geometric method, which searches the fitted pose based on the reference model. However, the edges may not be always captured correctly due to threshold setting of filters, shadow and reflection. The search calculation could result in failure.

To identify the workpieces robustly, one could use the qualitative characteristics of the group of the line segments rather than quantitative ones. The qualitative characteristics are relative lengths (e.g. short, long) and angles (e.g. wide, narrow, interior, exterior) and their connective pattern. These characteristics are independent of image size and unchangeable by slight noises. Therefore, searching these characteristics could be the solution for robust workpiece recognition.

4. NEW LEARNING METHOD OF ACTION RULE TABLE FOR PATH PLANNING

In this section, we state a new learning method with an appropriate action rule table for autonomous and semi-autonomous construction machines under constrained workspace, e.g. obstacles existence. As mentioned before, construction field is non-steady state environment. There is always something different from a construction plan or a survey map in construction field. To solve the problem, a machine has to recognize the context based on current own state and sensor information and selects appropriation action from action candidates.

In our proposed learning algorithm, machines can obtain the rules and surrounding maps by the Simultaneous Localization And Mapping (SLAM) method [4] with less parameters and action rules known in advance. We focus on an improvement of the Q-Learning reward system. The Q-learning is one of the reinforcement learning method [3]. Our proposed algorithm is useful for not only vehicle movement but also a machine manipulator.

In the following subsections, Firstly, concept system of (semi-) autonomous machines is argued. Secondly, the detail of proposed leaning algorithm is mentioned. Finally, the effectiveness of the algorithm is demonstrated by a numerical simulation.

4.1 Concept system of (semi-) autonomous machines

Figure 12 is the scheme of a cognition model for autonomous or semi-autonomous machines. This system consists of the following three functional parts.

a) Mapping and Localization

Even GPS system is available, a machine cannot sense surrounding area without other sensors, such as a laser scanner and a vision sensor. With such sensors, a machine can recognize the shapes of unknown objects around it. A machine can localize own position and posture and update map information simultaneously with sensor data and a known map. This function is known as SLAM [4].

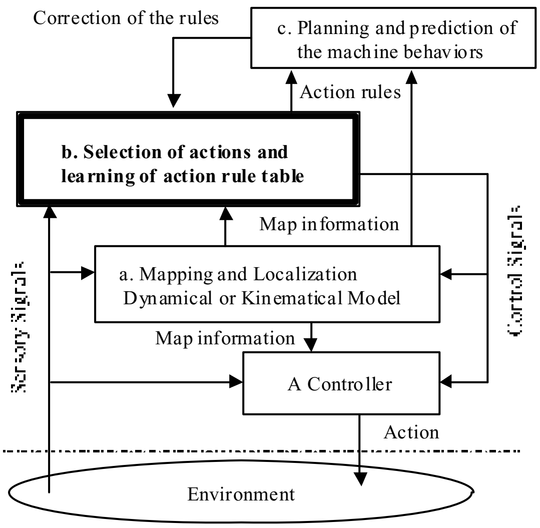


Fig. 12 Cognition model scheme for autonomous vehicles

b) Selection of actions and learning of action rule table

In this functional part, firstly, the next “appropriate” action is selected from the obtained rule tables. Then, the selected action is evaluated from the action result. Lastly, the rule tables are updated based on that evaluation. These steps are iterated. The details of this algorithm are mentioned in the next subsection.

c) Planning and prediction of the machine behavior

Based on the map information generated from the functional part a) and the action rule tables given by the part b, an optimal action sequence is predicted to accomplish the given task, e.g. path planning.

The part b is the most fundamental functional parts among them, because the part b influences other parts. It is part b or learning action rule table to enable the part c to make flexible and robust decisions. Although the part a and c are well-studied problems, the solution to the part b is not established except some proposed concepts [5, 6].

4. 2 A learning algorithm of action rule tables

In this subsection, the details of our proposed algorithm are stated. Our proposed method is aimed to improve the reward system of Q-learning. The following two learning systems are considered in our method.

1. Global learning system for task accomplishment
2. Local learning system for action rule derivation

Reinforcement learning is a kind of machine learning, which is characterized by employing “rewards”. In Q-learning, action rule tables, often called as action value functions, are updated by the following equations for machine states and actions.

$$\delta_t = r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \tag{2}$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\delta_t, \alpha \in (0,1] \tag{3}$$

Where s_t and a_t are state and the action of the machine at time t respectively. $Q(s_t, a_t)$ is the action value function with respect to s_t and a_t. γ is the discount rate, which is set by no more than 1. $r(t+1)$ is the reward at time t. δ_t in equation (2) is called a temporal difference error. a is a learning rate which converge to 0 as time passes. In equation (3), the left arrow means to update the component of action value function $Q(s_t, a_t)$.

The reward system is given and fixed in conventional methods. This means that local rules at some situation and the rewards are designed heuristically by trail and error. For example, when a machine reaches in front of a wall, A designer could set rewards 100 for backward movement and 10 for left or right turn.

In our methods, such a pre-design of the reward system is almost omitted. What a designer has to do is only choose a reward, when the given task is accomplished. The local learning system gives the rewards about the states and the actions during the task. The local learning system interacts with the global learning system with respect to the rewards. The interaction of these two systems is represented by the following equations.

$$r_g = r_{eg} + \lambda Q_l(s_l, a) \tag{4}$$

$$r_l = Q_g(s_g, a) - \min_a Q_g(s_g, a) \tag{5}$$

Where, Q_l and Q_g are the action value functions for local and global learning system, respectively. r_g and r_l are the reward for Q_l and Q_g. r_{eg} is the reward when the given task is accomplished. λ is a proper constant value. Subscripts l and g in the values show the relationship between the local learning system and the global one.

The following is the algorithm for path planning with the local and global learning systems.

Step 1: Initialize Q_l and Q_g
Step 2: Create a new map and $Q_{l,\mathrm{fix}} \leftarrow Q_l$
Step 3: learn Q_g; r_g is calculated by $Q_{l,\mathrm{fix}}$
learn Q_l; r_l is calculated by Q_g
Step 4: if norm($Q_l - Q_{l\_\mathrm{fix}}$) > ε, where ε is a constant return to Step 2

In step 3, to assure learning stability of Q_g, r_g is calculated from $Q_{l,\mathrm{fix}}$. The convergence of Q_l is expected from stable Q_g. Finally, the action value functions of the local learning system are obtained as the action rule tables.

4. 3 Numerical simulation

In this subsection, a numerical simulation is shown for the evaluation of our proposed method. Problem settings are as follows

1. Map world is an n x n cell world
2. Given task is to go to the goal
3. Each cell is defined as either obstacle or movable space
4. Actions are four movements; up, down, right, left
5. State of local learning system is the condition of neighboring four cells (up, down, right, left / obstacle or movable).
6. State of global learning system is the position on the map.

In the learning process, 100 examples of 5 by 5 cell maps are used with r_{pg} = 10 and λ = 0.1 in equation (4). Note that the learning result of the action rule table is represented by the action value function Q_l. Figure 13 shows the result of Q_l. In this figure, the given cell world is 10 by 10, which is four times larger than the learning maps. The optimal actions are shown as the arrows on the cell and decided based on only the Q_l obtained from the learning process. Although those actions are not always appropriate globally (see the actions circled), the resultant planning is sufficient for the local path search.

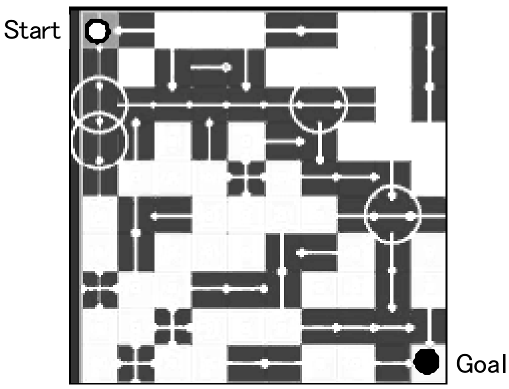


Fig. 13 The results based on the action rules which are learned by the proposed method

5. DISCUSSION

We study some basic functions for autonomous and semi-autonomous construction machines. Our navigation method is robust in spite of many obstacles, when enough reference walls are detected. However, it is possible to happen that no wall can be detected due to the obstacles near the laser scanners. Therefore, an additional obstacle avoidance algorithm needs to be added to the machine control system.

We show that the images of workpiece silhouettes are symbolized by the group of line segments with an achromatic or gray board. A pattern search of line segment connections may identify each workpiece. However, further study is needed for the occlusion and overlap cases.

We propose a new learning method of action rule table for path planning by improving Q-learning reward system. Our proposed method enables machines to obtain action rules and surrounding maps with the less parameters and action rules known in advance.

REFERENCES

[1] Hitoshi Hasunuma, Katsumi Nakashima, et al: “The Development of a Remote Control System for a Humanoid Robot -The Application of a Humanoid Robot for driving of Industrial Vehicles-“, the Journal of the Robotics Society of Japan, Vol. 22, pp. 46-54 (2004)

[2] A. Stentz, C. Dima, C. Wellington, H. Herman, and D. Stager, "A System for Semi-Autonomous Tractor Operations," Autonomous Robots, Vol. 13, No. 1, pp. 87-103 (2002)

[3] R. S. Sutton and A. G. Barto: “Reinforcement Learning: An Introduction”, MIT Press (1998)

[4] M. Montemerlo: “FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem With Unknown Data Association”; Ph. D thesis of CMU (2003)

[5] M. Fujita: “Intelligence Dynamics: A new approach toward intelligence”, In proc. Of Epigenetic Robotics (2005)

[6] M. Haruno, D. M. Wolpert, M. Kawato: “Mosaic model for sensorimotor learning and control”; Neural Computation, Vol. 13, pp. 2201-2220 (2001)