Computational Intelligence for Illuminance Measurement of A Mobile Robot

Naoyuki Kubota[*1*2] , Masayuki Kanemaki[*3] , Kazuhiko Taniguchi[*4] and Atsushi Ueda[*4]

[*1] Dept. of Mechanical Engineering, Tokyo Metropolitan Univ.
1-1 Minami Osawa, Hatioji 192-0397, Japan
email: kubota@comp.metro-u.ac.jp

[*2] "Interaction and Intelligence," PRESTO, Japan Science and Technology Corporation (JST)

[*3] Dept. of Human and Artificial Intelligent Systems, University of Fukui
3-9-1 Bunkyo, Fukui 910-8507, Japan
email: kanemaki@iicx.ia.his.fukui-u.ac.jp

[*4] Kinden Kyoto Institute, Kinden Corporation
3-1-1 Sorakudai, Kizu-cho, Soraku-gun 619-0223, Japan
email: {taniguchi, ueda}@kd-lab.com

Abstract---**This paper proposes the method for illuminance measurement by using a vision-based mobile robot. The robot should perform (1) to build a physical environmental map, (2) to select the measurement points, (3) to measure illuminance according to the environmental map, and (4) to build an illuminance map. To perform the sequence of tasks, the robot must know the self-location. Therefore, this paper proposes a method for behavior coordination and self-location estimation. Furthermore, we conducted several experiments of behavior coordination and illuminance measurement.**

I. Introduction

Robotic systems have been applied to various fields such as manufacturing systems, building industry, aerospace development, and human society. The aims of robots are to assist a human and to perform a task instead of a human. M.Brady defines robotics as the intelligent connection of perception to action [1]. To build an intelligent robot, various methodologies have been developed by simulating human behaviors and by analyzing human brains [2-8]. Actually, computational intelligence techniques based on neural, fuzzy, and evolutionary computing have been proposed to deal with real world problems. As one stream of evolutionary computing, genetic algorithms (GAs) have been effectively used for optimization problems in robotics [12-20]. GAs can produce a feasible solution, not necessarily an optimal one, with less computational cost. The main role of GAs in robotics is the optimization in modeling or problem-solving. In fact, the optimization based on GAs can be divided into three approaches of a direct use, machine learning, and genetic programming. The direct use is often seen in applications to the numerical optimization and the combinatorial optimization for tuning control parameters and for obtaining knowledge and strategies. The machine learning is mainly used for optimizing a set of inference rules in autonomous robots. Finally, genetic programming is applied for obtaining computer programs that realize complicated behaviors or tasks. In this way, GAs are used in various problems of robotics. On the other hand, fuzzy computing is often used for representing human expert knowledge by using membership functions, while neural computing is often used for learning input-output pairs as a function approximator [4]. For example, fuzzy controllers and neural controllers are used for behavior control of robots [24]. We have proposed a method of multi-objective behavior coordination used in unknown and dynamic environments [25]. Furthermore, the proposed method can be applied to practical engineering problems.

This paper proposes the method for illuminance measurement by using a vision-based mobile robot. The robot should perform (1) to build a physical environmental map, (2) to select the measurement points, (3) to measure illuminance according to the environmental map, and (4) to build an illuminance map. To fulfill the sequence of tasks, the robot must know the self-location. Therefore, this paper proposes a method for behavior coordination and self-location estimation. In order to simplify the problem, we use landmark towers. In this paper, we apply a genetic algorithm for detecting landmark towers from an image, and fuzzy control for the robotic behaviors. The effectiveness of the proposed method is demonstrated through several experiments of target tracing and collision avoiding behaviors.

II. Behavior Control of A Mobile Robot

A. A vision-based mobile robots

Figure 1 shows a mobile robot, ActivMedia Robotics Pioneer 2, used for illuminance measurement. This robot is provided with 8 ultrasonic sensors and 2 encoders. Since the sensors and actuators are connected with the Hitachi H8 CPU board in the robot, sensory inputs are sent to the host computer through H8. Furthermore, the robot is provided with an omnidirectional sensor. Images from the omnidirectional sensory are taken directly into the host computer. Therefore, the host computer makes motor outputs according to the image and other sensory inputs.

B. Multi-objective Behavior Coordination

A behavior of the robot can be represented using fuzzy rules based on simplified fuzzy inference [2]. The logical structure written by fuzzy rules is easy for humans to understand and to design. In this paper, we apply the proposed method for a navigation task of a mobile robot. A mobile robot moves from start to target points, and avoids dangerous states. Here we use two behaviors of target tracing and collision avoiding. In general, a fuzzy if-then rule is described as follows,

If x_1 is $A_{i,1}$ and ... and x_m is $A_{i,m}$

Then y_1 is $w_{i,1}$ and ... and y_n is $w_{i,n}$

where $A_{i,j}$ and $w_{i,k}$ are a symmetric triangular membership function for the jth input and a singleton for the kth output of the ith rule; m and n are the numbers of inputs and outputs, respectively. Fuzzy inference is generally described by,

$$\mu_{Ai,j}(x_j) = \begin{cases} 1 - \dfrac{|x_j - a_{i,j}|}{b_{i,j}} & |x_j - a_{i,j}| \le b_{i,j} \\ 0 & otherwise \end{cases} \tag{1}$$

$$\mu_i = \prod_{j=1}^{m} \mu_{Ai,j}(x_j) \tag{2}$$

$$y_k = \frac{\sum_{i=1}^{R} \mu_i w_{i,k}}{\sum_{i=1}^{R} \mu_i} \tag{3}$$

where $a_{i,j}$ and $b_{i,j}$ are the central value and the width of the membership function $A_{i,j}$; R is the number of rules. Outputs of the mobile robot are steering angle and its velocity. In our previous research [24], evolutionary optimization methods are used for obtaining fuzzy rules in various environment conditions. Next, we extend this fuzzy controller to multiple behaviors.

We have proposed a multi-objective behavior coordination mechanism(Fig.2). In general, a mobile robot has a set of behaviors for achieving various objectives. A behavior weight is assigned to each behavior. By extending eq.(3), the output is calculated by

$$y_k = \frac{\sum_{h=1}^{B} \left(wgt_h(t) \sum_{i=1}^{R} \mu_{h,i} \cdot w_{h,i,k} \right)}{\sum_{h=1}^{B} \left(wgt_h(t) \sum_{i=1}^{R} \mu_{h,i} \right)} \tag{4}$$

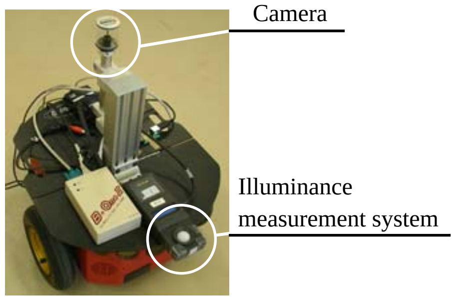


Fig.1 A mobile robot for illuminance measurement

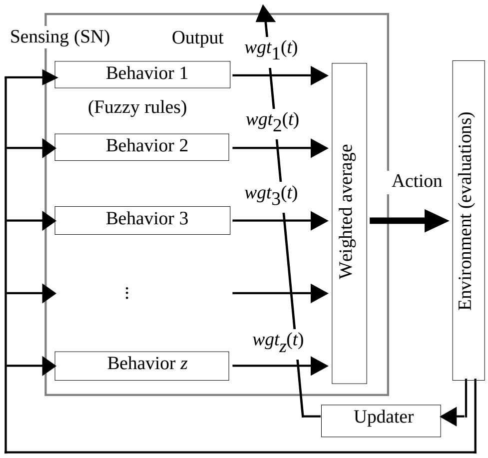


Fig.2 Multi-objective behavior coordination

where B and $wgt_h(t)$ are the number of behaviors; a behavior weight of the hth behavior over the discrete time step t, respectively. By updating the behavior weights, the robot can take a multi-objective behavior according to the time series of perceptual information. This method can be considered as a mixture of experts [22] where the a behavior and behavior coordination mechanism are considered as a local expert and a gating network, respectively.

III. Self-location Estimation Based on Vision

A. A Steady-state Genetic Algorithm

A steady-state genetic algorithm (SSGA) is applied for detecting landmark towers (Fig.3). The SSGA simulates the continuous model of the generation, which eliminates and generates a few individuals in a generation (iteration). A candidate solution (individual) is composed of numerical parameters of the position and size of a landmark ($g_{i,1}$ $g_{i,2}$ $g_{i,3}$) where i indicates the individual number. In SSGA, only a few existing solutions are replaced by new candidate solutions generated by genetic operators in each generation [22]. In this paper, the worst candidate solution is eliminated and replaced with the candidate solution generated by the crossover and mutation. We use elitist crossover and adaptive

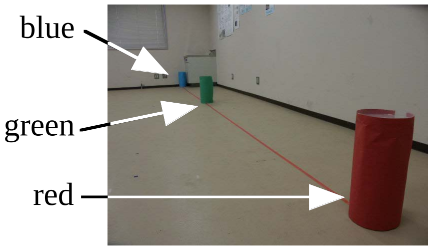


Fig.3 Landmark Towers

mutation. Elitist crossover randomly selects one individual and generates an individual by incorporating genetic information from the selected individual and the best individual. Next, the following adaptive mutation is performed to the generated individual,

$$g_{i,j} \leftarrow g_{i,j} + \left(\alpha_j \cdot \frac{fit_{\max} - fit_i}{fit_{\max} - fit_{\min}} + \beta_j \right) \cdot N(0,1) \tag{5}$$

where fit_i is the fitness value of the ith individual, $fit_{\max}$ and $fit_{\min}$ are the maximum and minimum of fitness values in the population; α_j and β_j are the coefficient and offset, respectively. In the adaptive mutation, the variance of the normal random number is relatively changed according to the fitness values of the population. Fitness value is calculated by the following equation,

$$fit_i = C_{LT} - p \cdot C_{Other} \tag{6}$$

where p is a coefficient for penalty, C_{LT} and C_{other} indicate the number of pixels of the color corresponding to a landmark tower and other colors, respectively. Therefore, this problem results in the maximization problem.

B. Landmark Detection and Self-location Estimation

The robot with the omnidirectional sensor can estimate the self-location by using the angles among landmark towers. In this study, we use three landmark towers of red, green, and blue (Fig.3). Let P, A, B, and C be the position of the robot, the centers of red, green, and blue landmark towers, respectively. The coordinate axes are defined as Fig.4. First, we make two circles satisfying that A, B, and P are on the circle O_1, and B, C, and P are on the circle O_2. Next, let D be the crossing point of the line passing through B and O_1, and let E be the crossing point of the line passing through B and O_2. Because the robot can obtain the directions of A, B, and C according to the result of image processing, the robot can know the angles $\angle BPA = \angle BDA = \theta_1$ and $\angle BPC = \angle BEC = \theta_2$. Let (x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D), (x_E, y_E), and (x_P, y_P) be the points of A, B, C, D, E, and P. To simplify this problem, we restrict the moving range of the robot into $x_A < x_P < x_C$, and $y_A < y_P$. Accordingly we can obtain the following equations,

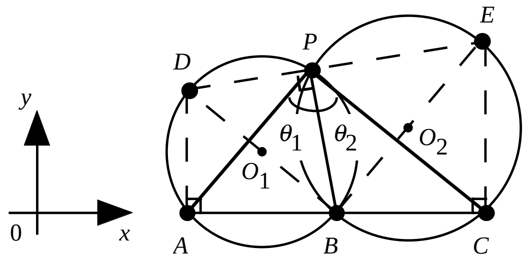


Fig.4 Geometrical relationship of the location of the landmark towers (A, B, and C) and a mobile robot (P)

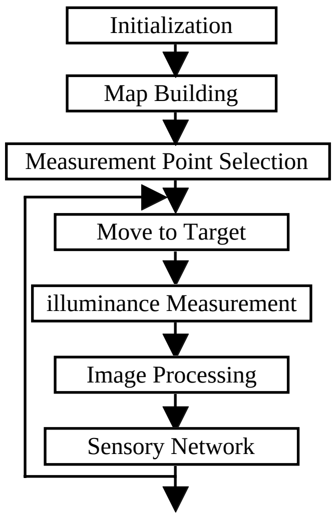


Fig.5 Total procedure of illuminance measurement

$$y_D = y_B + \frac{x_B - x_D}{\tan\theta_1} \tag{7}$$

$$y_E = y_B + \frac{x_E - x_B}{\tan\theta_2} \tag{8}$$

Furthermore, we can obtain the coordinate of the robot by using the similar relationship of the quadrilaterals $ABPD$ and $PECB$,

$$x_P = \frac{x_C \tan\theta_1}{\tan\theta_1 + \tan\theta_2} \tag{9}$$

$$y_P = \frac{(x_C - x_B)\tan\theta_1}{\tan\theta_2(\tan\theta_1 + \tan\theta_2)} \tag{10}$$

In this way, the robot can estimate the position relative to the landmark towers. However, the robot should use several images owing to noise of the images. Therefore, after the robot performs the estimation of the self-location several times, and the robot decides the self-location.

The total procedure for illuminance measurement is shown in Fig.5. After the initialization, the robot performs a

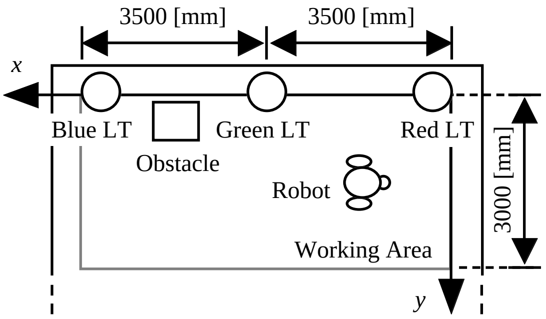


Fig.6 Experimental environment including landmark towers (LT) and obstacle

map building of the physical environment. Next, the robot selects the intermediate points for illuminance measurement. Next, the robot moves to each intermediate target point and measures illuminance at the point. And then, the robot updates the self-location according to the estimated position based on the avobe image processing. Here the sensory network[27] is used for updating the self-location.

The self-locations estimated by dead-reckoning and image processing are (x_R ,y_R) and (x_P ,y_P), respectively. The self-location is updated by

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \frac{1}{W_P + W_R}\left(W_P \begin{pmatrix} x_P \\ y_P \end{pmatrix} + W_R \begin{pmatrix} y_R \\ y_R \end{pmatrix} \right) \tag{11}$$

where W_P and W_R are the weight parameters. These parameters are also updated by

$$W_R = 1 - \frac{\sum_{t=T_i}^{T_{i+1}} \left| v_{r,t} - v_{l,t} \right|}{2 \cdot (T_i - T_{i+1}) \cdot V_{\max}} \tag{12}$$

where $V_{\max}$ is the maximal value of the motor output from T_i to T_{i+1}; $v_{r,t}$ and $v_{l,t}$ are the outputs of right and left motors at the discrete time t. Furthermore, W_P is updated by

$$W_P = \begin{cases} \dfrac{fit_{k,best}}{fit_{upper}} & if \ \ fit_{k,best} < fit_{upper} \\ 1.0 & \end{cases} \tag{13}$$

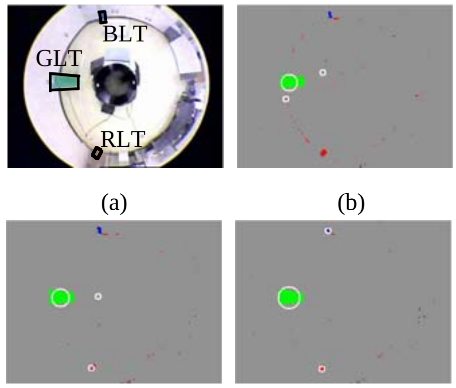
$$k = \arg\min_{c} fit_{c,best}$$

where fit_k,best is the fitness value of the best individual. In this way, the self-location is updated by the state of sensory inputs.

IV. Experiments

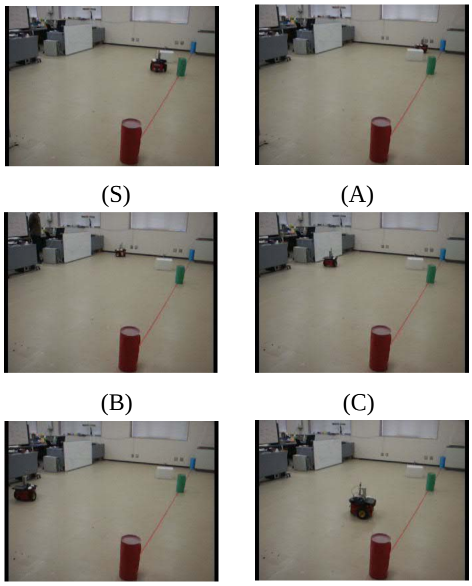
This section shows experimental results of the vision-based mobile robot for illuminance measurement. The size of the environment is 7000 [mm] × 3000 [mm]. We use three landmark towers of blue, green and red colors (Fig.6). The height and radius of each landmark tower are 400 [mm] and 200 [mm], respectively. The robot can move in the area with the edge connecting landmark towers. In the SSGA, the number of individuals is 200, and the number of evaluations is 2000.

Figure 7 shows an extraction result of landmark towers by SSGA in a preliminary experiment. In this figure, RLT, GLT, and BLT are corresponding to red, green and blue landmark towers, respectively. The color condition of an original image is not so good, because this sensor uses an omnidirectional mirror (Fig.7 (a)). Furthermore, the taken image is reversed. Figure 7 (b) shows a searching result of landmark towers, but SSGA could not detect the red and blue landmark towers. SSGA continues to search the landmark towers by using the next image (Fig.7 (c)). At this image, the red landmark tower was detected. Finally, SSGA detected all landmark towers (Fig.7 (d)). The final position on the image



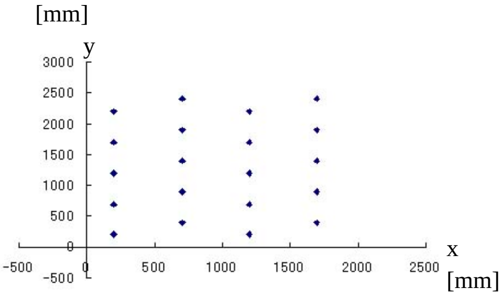
(a) (b) (c) (d)

Fig.7 Extraction results of landmark towers by SSGA

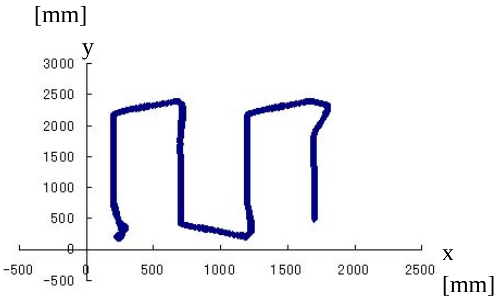


(S) (A) (B) (C) (D) (E)

Fig.8 Snapshots of navigation

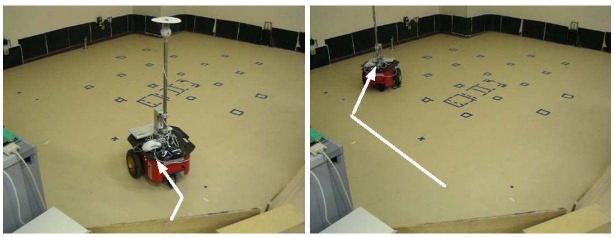


(a)



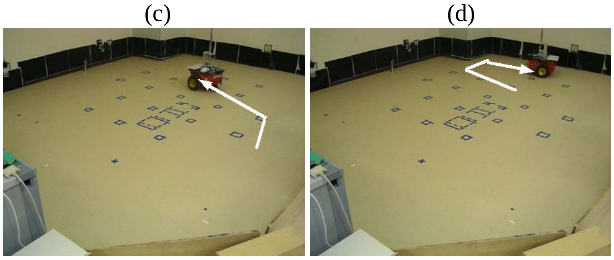
(b)

Fig.9 (a)The target points determined by path-planning
(b)The orbit acquired from encoder information and self-location estimation



(a) (b)

(c) (d)



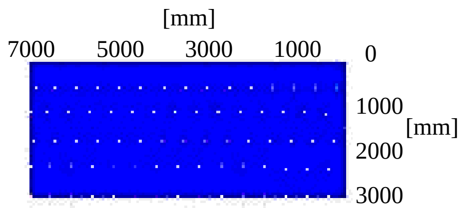
(e) (f)

Fig.10 Snapshot of the Robot in Illuminance Measurement

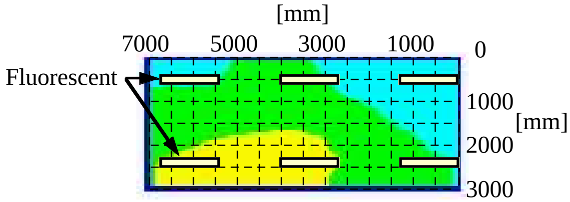
of RLT, GLT, and BLT is (145,212), (98,173), and (157,23)

Table 1 Estimation result of self-location

Point		A	B	C	D	E
Dead-reckoning	x	6019	5998	3480	982	1004
	y	515	2520	2498	2503	472
	θ	0.67	1.63	-3.03	3.14	-1.43
Vision	x	6103	6183	3712	1153	1033
	y	524	2564	2699	2475	735
	θ	0.67	1.68	-3.09	-3.02	-1.35
Estimation	x	6032	5998	3580	1026	1018
	y	516	2520	2585	2496	599
	θ	0.67	1.63	-3.06	-3.11	-1.39
True location	x	6034	5990	3610	1110	930
	y	590	2535	2760	2732	818
	θ	0.66	1.69	-3.12	-3.12	-1.41



(a) An experimental result of illuminance measurement at the observation points



(b) An interpolated results of illuminance distribution in a room
Fig.11 Experimental result of illuminance measurement

and the obtained relative angles; θ_1=79°, θ_2=83°. The estimated self-location of the robot is x_p=3478[mm], y_p=2494[mm] according to the detected landmark towers. The error between the estimated values and the measured values is Δx_p=-22[mm], Δy_p=-6[mm].

Figure 8 shows snapshots of the robot in the navigation, and table 1 shows the estimated results. In Fig.8, (S), and (A) ~ (E) indicate the starting points and intermediate points, that are (3500, 500), (6000, 500), (6000, 2500), (3500, 2500), (1000, 2500), and (1000, 500) [mm], respectively.

Next, we show the experimental result of the illuminance measurement. In order to build an illuminance map, the environmental size is measured by using wall following and collision avoiding behaviors. Next, the intermediate target points for illuminance measurement are determined using the built map (Fig.9(a)). The intermediate points are selected at intervals of 500 [mm] on the map. Figure 9(b) shows the trajectory of the mobile robot in the environment. Furthermore, Fig.10 shows snapshots of the robot inillunimance measurement. Finally, Fig. 11 shows the illuminance map in the experimental result. This map is the half of the environment. We obtained the illuminance map of the room.

V. Summary

This paper proposed the method for illuminance measurement by a vision-based mobile robot. The proposed method includes the behavior coordination method based on fuzzy controllers and image processing for self-location estimation. The experimental results show that the proposed method can perform the illuminance measurement.

As future works, we intend to integrate various image processing methods for behavior control of the mobile robot.

References

[1] M.Brady and R.Paul: Robotics Research, The First International Symposium, The MIT Press, Massachusetts (1984)

[2] S.J.Russell and P.Norvig, Artificial Intelligence, Prentice-Hall, Inc., 1995.

[3] J. M. Zurada, R. J. Marks II, C. J. Robinson (eds.), Computational Intelligence - Imitating Life, IEEE Press, 1994.

[4] J.-S.R.Jang, C.-T.Sun, E.Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall, Inc., 1997.

[5] R.A.Brooks, Cambrian Intelligence, The MIT Press, 1999.

[6] R.C.Arkin, Behavior-Based Robotics, The MIT Press, 1998.

[7] R.Pfeifer and C.Scheier: Understanding Intelligence, The MIT Press (1999)

[8] V.Braitenberg: Vehicles; Cambridge, MA: MIT Press (1984)

[9] U. Neisser, Cognition and reality. W.H.Freeman and Company (1976).

[10] James Gibson, The Ecological Approach to Visual Perception, Houghton Miffilin Company, 1979.

[11] Michael Eysenck, Perception and Attention, Psychology (edited by Michael Eysenck), Prentice Hall, pp.139-166, 1998.

[12] S.Nolfi and D.Floreano: Evolutionary Robotics, The MIT Press (2001)

[13] Y.Davidor: A genetic Algorithm Applied to Robot Trajectory Generation; Handbook of Genetic Algorithms, Van Nostrand Reinhold, pp.144-165 (1991)

[14] J.Xiao, Z.Michalewicz, L.Zhang and K.Trojanowski: Adaptive Evolutionary Planner/Navigator for Mobile Robots; IEEE Trans. on Evolutionary Computation, Vol.1, No.1, pp.18-28 (1998)

[15] T.Fukuda, N.Kubota, and T.Arakawa: GA Algorithms in Intelligent Robots; Fuzzy Evolutionary Computation, Kluwer Academic Publishers, pp.81-105 (1997)

[16] D.B.Fogel: Evolutionary Computation, IEEE Press (1995)

[17] J.Holland: Adaptation in Natural and Artificial Systems, Ann Arbor:University of Michigan Press . (1975)

[18] D.E.Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Welsey (1989)

[19] M.Mitchell: An Introduction to Genetic Algorithms, The MIT Press (1996)

[20] G.Syswerda, A Study of Reproduction in Generational and Steady-State Genetic Algorithms, Foundations of Genetic Algorithms, Morgan Kaufmann, pp.94-101, 1991.

[21] N.Kubota, T.Morioka, F.Kojima, and T.Fukuda, Learning of Mobile Robots Using Perception-Based Genetic Algorithm, Measurement, No.29, pp.237-248, 2001.

[22] N.Kubota, H.Masuta, F.Kojima, T.Fukuda, perceptual System and Action System of A Mobile Robot with Structured Intelligence, The 2002 IEEE World Congress on Computational Intelligence, CD-ROM, 2002.

[23] N.Kubota and T.Fukuda: Sensory Network for Mobile Robotic System with Structured Intelligence, Journal of Robotic and Mechatronics, Vol.10, No.4 (1998).

[24] T.Fukuda and N.Kubota: An Intelligent Robotic System Based on A Fuzzy Approach, Proceedings of The IEEE, Vol.87, No.9, pp.1448-1470 (1999).

[25] N.Kubota, H.Masuta, and K.Taniguchi: Behavioral Dimension of Multi-Objective Behavior Coordination for A Mobile Robot, Proc. of 2001 IEEE International Symp. on CIRA, pp.374-379 (2001).

[26] Y.Nojima, N.Kubota, F.Kojima, and T.Fukufa, Control of Behavior Dimension for Mobile Robots, The Forth Asian Fuzzy Systems Symposium (AFFS2000), pp.652-657, (2000).

[27] Naoyuki Kubota, Hisatoshi Neya and Kazuhiko Taniguchi, Sensory Network and Evolutionary Programming for a Mobile Robot, Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning(SEAL'02), CD-ROM,pp.119-123, (2002.11).