Modified Vector Field Histogram with a Neural Network Learning Model for Mobile Robot Path Planning and Obstacle Avoidance

Dr. Bahaa I.Kazem *1, Ali H. Hamad *2, Mustafa M. Mozael*3
**1 Mechatronics engineering, Baghdad university, Baghdad, Iraq*
**2 Information and communication engineering, Baghdad university ,Baghdad, Iraq*
**3 Mechatronics engineering, Baghdad university, Baghdad, Iraq*
Drbahaa@gmail.com, alihamed732@yahoo.com, alnoady1982@yahoo.com
doi:10.4156/ijact.vol2. issue5.18

Abstract

In this work, a Modified Vector Field Histogram (MVFH) has been developed to improve path planning and obstacle avoidance for a wheeled driven mobile robot. It permits the detection of unknown obstacle to avoid collisions by simultaneously a steering the mobile robot toward the target; a regular grid map representation for a work space environment is caurried out. A Neural Network (NN) model is used to learn many critical situations of environment during robot navigation among obstacles using MVFH. Also, digital filter has been utilized for improving the robustness of obstacle avoidance trajectory of mobile robot. The proposed MVFH-NN has been implemented and tested by using MobotSim program simulation and MATLAB . The developed algorithm showed good navigation properties and can be used in complex real world (maze-like environment), it also shows good ability to overcome limitations of the traditional VFH algorithm (like wide candidate valley, narrow hallway and target distance limitation).

Keywords: *Mobile Robot, Neural Network, Vector Field Histogram*

1. Introduction

Mobile robots have been successfully employed in industrial settings to improve productivity and to perform dangerous or monotonous tasks. More recently, attention has turned to the use of robots to aid humans outside the industrial environment, in places such as home or office. For example, as the population in the developed world ages, robots can interact with humans in a safe and friendly manner performing necessary home-care/daily living tasks and this would allow more seniors to maintain their independence. Such devices could alleviate some of the non-medical workload from health-care professionals, and reduce growing healthcare costs.

Mapping is the process of generating models of a mobile robot's environment based on sensory information with aim to determine the location of various entities, such as landmarks or obstacles. Most successful navigation algorithms require the availability of dynamic and adaptable maps. An accurate model of the environment surrounding a robot enables it to complete complex tasks quickly, reliably and successfully. Without such a model, a robot neither can plan a path to a place not currently sensed by its sensors, nor may effectively search for an object or place [1].

2.Histogram and Certainty Grid World Model

A new method for real-time building with mobile robot in motion was developed at the University of Michigan by Borenstein and Koren[2]. This method entitled Histogramic In-Motion Mapping. Algorithm uses a two-dimensional Cartesian histogram grid for obstacle representation.
A pioneering method for probability of obstacles in a grid-type world model has been developed at Carnegie-Mellon University (**CMU**) by Moravec [3]. This world model which is called a certainty grid, is especially suited to the accommodation of inaccurate sensor data such as range measurements from ultrasonic sensors.

In the certainty grid, the robot's work area is represented by a two-dimensional array of square elements, denoted as cells. Each cell contains a certainty value (**CV**) that indicates the measure of confidence for an obstacle that exists within the cell area. With the **CMU** method, **CVs** are updated by a heuristic probability function [4].

The active grid **S*** is mapped onto a **1D** structure known as a polar histogram **H**, where, the contents of each active cell in the histogram grid are now treated as an obstacle vector, whose direction is determined by the direction α. from the cell to the Vehicle Center Point (**VCP**). The next step of **VFH** maps the **2D** Cartesian histogram grid map C onto a **1D** structure. To preserve and isolate the information about the local obstacle information rather than using the entire grid map **C**, the **2D** grid used in this step is restricted to a window of **S** called the active window denoted by **S***. with constant dimensions, and centered on the Vehicle Central Point (**VCP**) and Consequently, it moves with the robot. **S*** represents a local map of the environment around the robot. Figure (1) illustrates the cell occupancy of **S***, the active window around the robot, and the angular sectors considered for the evaluation of the **1D** polar histogram.

The active grid **S*** is mapped onto a **1D** structure known as a polar histogram H, where the contents of each active cell in the histogram grid now treated as an obstacle vector whose direction is determined by the direction ***α*** from the cell to the Vehicle Center Point (**VCP**)

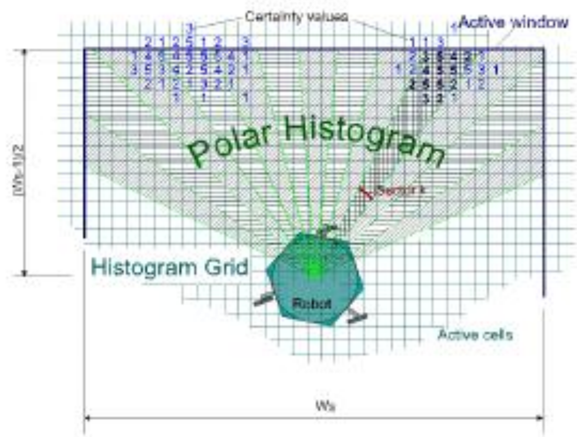


Figure 1. Mapping of active cells onto the polar histogram.[2]

$$\alpha_{ij} = \tan^{-1}\frac{y_j - y_o}{x_i - x_o} \tag{1}$$

and the magnitude is given by

$$m_{ij} = C_{ij}^2 * (d_{\max} - d_{ij}) \tag{2}$$

Where:
d_{max}= Distance between the four farthest cell of **S*** and **VCP**.
C_{ij} = Certainty value of active cell (i,j).
d_{ij} = Distance between active cell (i,j) and the **VCP**.
m_{ij} =Magnitude of the obstacle vector at cell (i,j).
x_o, y_o = Present coordinates of the **VCP**.
x_i, y_j = Coordinates of active cell (i,j).
α_{ij}= Direction from active cell (i,j) to the **VCP**.
H has an arbitrary angular resolution ξ such **n=360/** ξ is an integer (e.g., **ξ=5** and **n=72**). Each sector **k** corresponds to a discrete angle **ρ** quantized to multiples of ξ, such ρ=k×ξ**,** where k = 0, 1, 2... n-1. Correspondence between **S*** and sector **k** is established through.

$$k = \mathrm{int}(\frac{\alpha_{ij}}{\xi}) \tag{3}$$

For each sector **k**, the polar obstacle density is calculated by

$$h_k = \sum_{i,j} m_{ij} \tag{4}$$

In order to alleviate this problem, we use the following function to smooth the polar obstacle density:

$$h_k' = \frac{\sum_{l=-p}^{p} h_{k-l}}{2p+1} \tag{5}$$

where h_k' is the smoothed polar obstacle density **POD**. The parameter $\boldsymbol{p}$ determines how much the polar histogram is smoothed, and we found that $\boldsymbol{p}=5$ produces the best results. The POD of each sector represents the level of difficulty of moving in the corresponding direction. Figure (2) shows the polar histogram H for the momentary situation

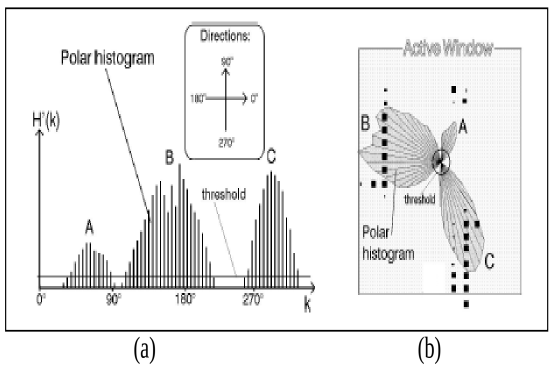


Figure 2. polar histogram density [2]

3. VFH with NN based algorithm

The **VFH-NN** based algorithm uses the sensor data from the environment and the classical find space problem in the strategy was transformed to the procedure ‘learning your environment’. In any position in workspace the robot has information about its distances to all objects in its window (active window) as shown in figure (2). The algorithm uses this information in neural network that learns these situations and in any position gives the free segment of space for safe path as output. The neural network uses as inputs the data measured by the ultrasonic sensors after transformed into polar histogram. The output is a free segment of the robot workspace. For obstacle avoidance purposes, a recurrent type of neural network was used with the gradient back-propagation technique for training the network [5].

4. Improvement of VFH Algorithm

The basic **VFH** algorithm is an obstacle avoidance method and is not fully target-oriented. In other words, it doesn’t necessarily guarantee the robot to reach the target in all cases when it is used in a navigator. In this work, an improvement over the **VFH** algorithm has been made based on neural network. Many drawback of VFH has been resolved using the new technique as illustrated below.

4.1. Wide Candidate Valleys

The robot always moves straight toward the target unless a **POD** peak appears in its heading direction. Without losing generality, we assume that a peak occurs in the histogram when the robot is at point ***c*** as shown in figure (3-a). The obstacle creates a **POD** peak within area ***scq*** and a wide candidate valley outside that area (the right border is ***cs*** and the left border is ***cq***).

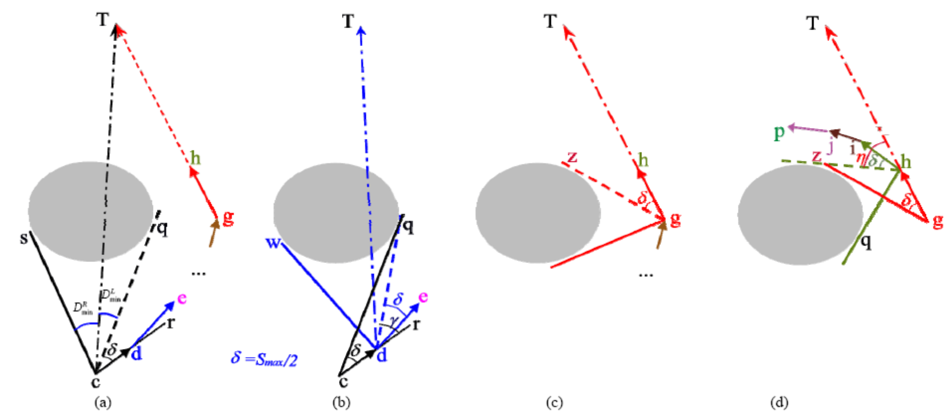


Figure 3. polar histogram density [5]

Applying the basic **VFH** algorithm to the above case, we find that ***cq*** (the left border of the valley) is the "near border" [30][6] of the valley at point ***c***. The robot's movement vector is then offset **δ** from ***cq***. The robot follows the path ***c→ d→e→…→g→h.*** At point ***h***, the near border of the valley is ***hz***. Therefore, the robot's movement vector is then offset **δ** from ***hz*** (figure 3-d). The robot then misses target ***T*** and moves to ***i*** followed by ***j*** and ***p***. At point ***p***, the right border becomes the near border. This switches the robot's movement direction from left to right and the robot moves backwards.
Certainly, if target ***T*** overlaps any one of points ***g, h, i, j,*** and ***p*** (i.e., it happens to be on the robot's obstacle avoidance path), the robot could reach the target. During the obstacle avoidance maneuvering, the problem is resolved by forcing the robot to move straight toward its target according to the equation:

$$k_h = \min(\mathrm{Abs}(\theta_{ck} - \theta_t)) \qquad (6)$$

Where the θ_{ck} is an angle of the candidate sector.

4.2. Target Distance limitation.

In case that a target is located very close to an obstacle, the target is unreachable by the basic **VFH** algorithm. The concept of "NEURAL NETWORK" is proposed to solve this problem. Assume that target **T** is very close to the obstacle, the **PODs** of sector k_t (corresponding to target **T**) and the neighboring sectors are nonzero and they grow as the robot moves closer to **T**. This eventually produces a peak in the histogram and prevents the robot from reaching **T**. To resolve this problem, a cluster of sectors $[k_t - n\ ,\ k_t + n]$ $(n = S_{max}/2)$ is constructed surrounding sector k_t. The histogram indexes of the cells that are inside this cluster are ignored, and the robot will learn on this situation. This treatment then creates a wide valley in $[kt - n\ ,\ kt + n]$, and creates a free target direction ahead of the robot, and thus guides the robot to the target.

4.3. Narrow hallway limitation

According to the basic **VFH** algorithm, When the robot is moving in a narrow hallway that generates two narrow candidate valleys (left and right ones), it moves in a trajectory ***a→b→…→g*** (in the center of the hallway) since the right valley is the closet one at each point and is thus selected as the winning valley. At point ***g*** as shown in figure(4), the right valley is still the closest one to the target vector ***gT*** and is therefore the winning valley. This points the robot's next heading direction at the center of the valley. In this way, the robot moves to point ***h*** and misses the target ***T*** At point ***h***; the left valley is the winning valley since it is the closest to the target vector ***hT*** at this time. The robot then moves back to ***g*** again. It moves back and forth between ***h*** and ***g*** and gets trapped there. This means that the robot may miss a target unless the target is at the center of the hallway. This problem is solved by applying the same scheme for the target distance limitation problem.

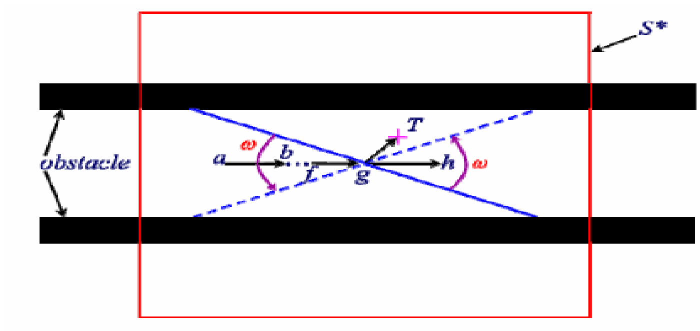


Figure 4. Narrow hallway[5]

5. Low Pass Filter for Improving Obstacle Avoidance Trajectory.

Because of the influence of noisy sonar data, the obstacle avoidance trajectory of **NN-VFH** based algorithm has some dithering as shown in figure(5-a) . For smooth operation of the **NN VFH** method, the following condition between the grid resolution Δs and the sampling period ***T*** must be satisfied:

$$\Delta s > TV_{max} \tag{7}$$

In our case Δs = 0.1 m and TV_{max} = 0.1*0.78 = 0.078 m, therefore, the above condition is satisfied. Since the distance dependent polar obstacle density(POD) is quantized to the grid resolution (10x10 cm), rather drastic changes in the resultant histogram may occur as the robot moves from one cell to another (even with condition (7) is satisfied). This results in an overly vivacious steering control, as the robot tries to adjust its direction to the rapidly changing direction of the heading direction.

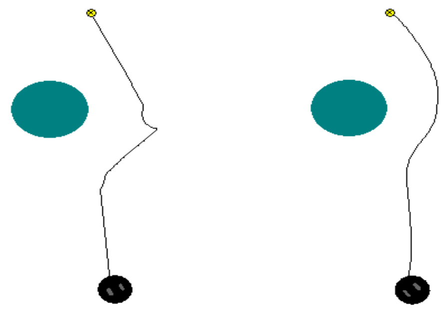
To avoid this problem, a digital low-pass filter, approximated in algorithm τ = 0.4 sec, has been added at the steering-rate command. The resulting steering rate command is given by:

$$\Omega_i = \frac{T\Omega'_{i-1} + (\tau - T)\Omega_{i-1}}{\tau} \tag{8}$$

Where:

Ω_i = Steering-rate command to the robot (after low-pass filtering)
Ω_{i-1}= Previous steering-rate command
Ω_i' = Steering-rate command (before low-pass filtering)
T = Sampling time (here: ***T*** = 0.1 sec)
τ = Time constant of the low pass filter.

The filter smoothes the robot's motion when moving alongside obstacles. As shown in figure(5), the low pass filter is used in this work to improve the obstacle avoidance trajectory.

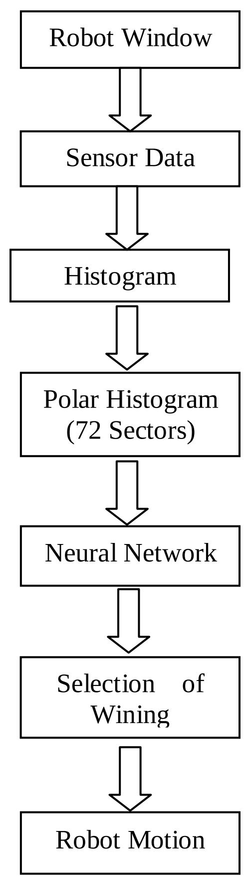


(a) (b)

Figure 5. a- without digital , b- with digital filter

6. Simulation and Results

The NN VFH based algorithm can be shown as flowchart : at first the mobile robot opens a window 33*33cells where the robot is in the center of the window; the ultrasonic sensor will transceiver its wave (24 sensor).The histogram loop include dividing the window into 72 sectors each sector has specific value (Polar Obstacle Density) that indicates the measure of confidence that an obstacle exists within the sector. In the second stage, the Polar Obstacle Density of each sector would be an input to Neural Network, this means that the input layer has 72 neurons. The output will be a 72 index. If the index is one, this means that the sector is free and if it is zero, this means that this sector is not free. For example, figure (6) shows the output pattern for an environment.



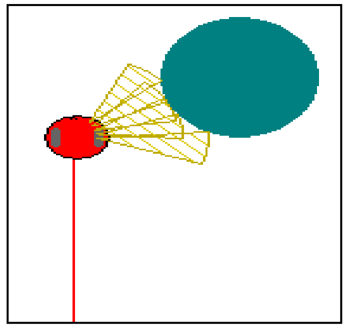


Figure 6. an environment sample

[676, 756, 789, 732, 680, 540, 441, 328, 232, 131, 115, 78, 70, 18, 11, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 95, 168, 257, 404, 556,]

The angle of the sectors is from 5° to 355° steps 5°. The free zone starts from 70° to 300° and the other angles indicate that an obstacle exists within the area. The Neural Network learning is off-line based on several I/O patterns. After learning is completed, the weights are fixed and the final values are then used during recall sessions.

After these steps the mobile robot will select the consecutive sectors that are near the target according to equation (6). Figure (7) shows some obstacle layout used for learning. Figures (8,9) show some simulation results.

7. Conclusions

For the presented work, the main conclusions can be summarized as:

1. The modified Vector Field Histogram algorithm with Neural Network based learning situation algorithm shows good ability to overcome limitations of the traditional **VFH** algorithm.(like wide candidate valley, narrow hallway and target distance limitation).
2. Using Digital filter improves the robustness obstacle avoidance trajectory and smoothes the robots path, but it requires longer time.
3. Complexity of environment affects the time required for the mobile robot to reach the target.
4. The developed algorithm showed good navigation properties and can be used in complex real world (maze-like environment).

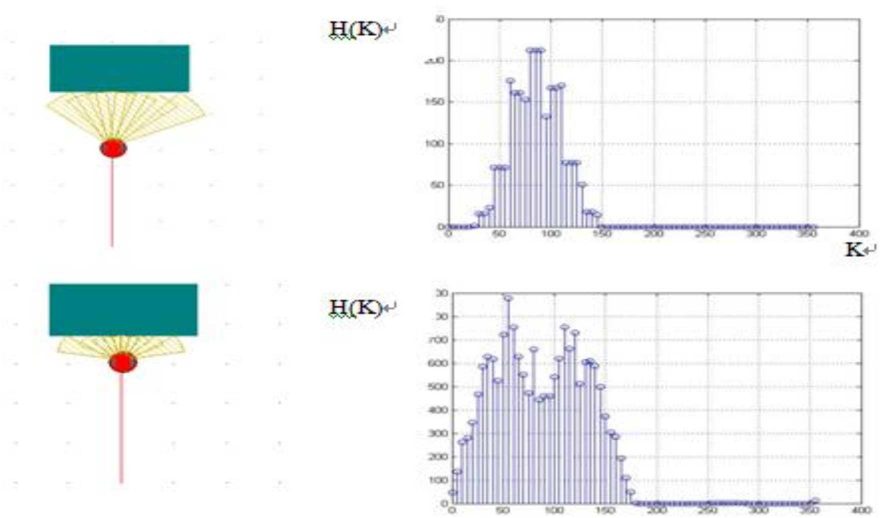


Figure7. the histogram density at two different obstacles distances. K

8. Acknowlgement

Thanks to Rabab Hameed AL-Darraji, Ph.D who proof read the article.

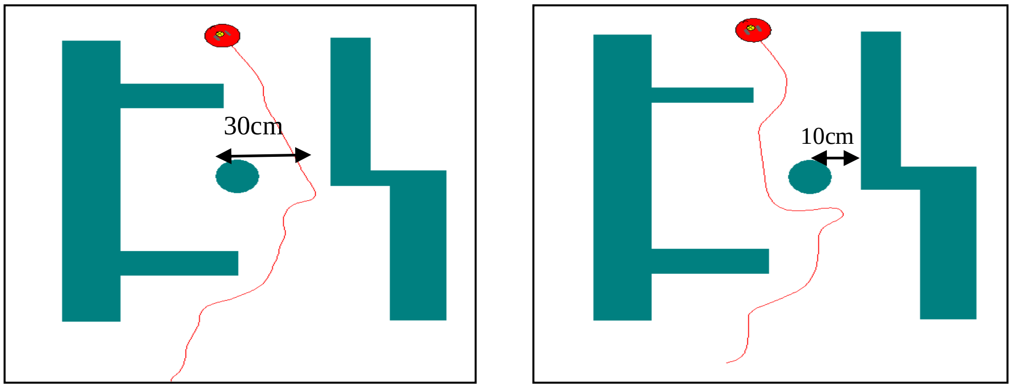


Figure 8. Simulation results

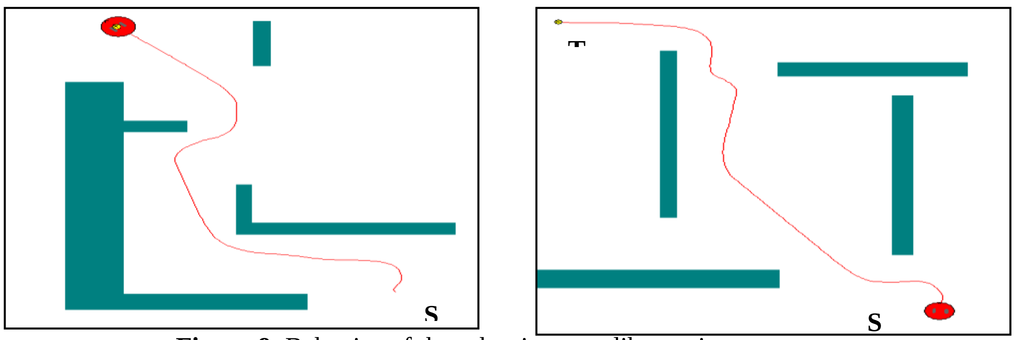


Figure 9. Behavior of the robot in maze-like environment**.**

9. References

1. Maki K. Habib, “Real Time Mapping and Dynamic Navigation for Mobile Robots” ,International Journal of Advanced Robotic Systems, Vol. 4, No. 3 , pp. 323-338, 2007.
2. Borenstein, J. and Koren, Y., “Real-time Obstacle Avoidance for Fast Mobile Robots ”, IEEE Transactions on Systems, Man, and Cybernetics , Vol. 19, No. 5, pp. 1179-1187, Sept 1989.
3. Moravec, H. P. and Elfes, A., “ High Resolution Maps from Wide Angle Sonar”. IEEE Conference on Robotics and Automation, Washington D.C , pp. 116-121.1985
4. U.Raschke & J.Borenstein, “ A Comparison of Grid-type Map-building Techniques by Index of Performance” , IEEE, pp.1828-1832, 1990.
5. Danica Janglová , “Neural Networks in Mobile Robot Motion” , Inernational Journal of Advanced Robotic Systems,Vol. 1, No. 1, pp. 15-22, 2004.
6. Kulic and Croft, “Safe Motion Planning for Human-Robot Interaction: Design and Experiments” , mobile robot towards new application,ISPN,3-86611-314-5, pp 784,ASR/PIV,Germany, December 2006.
7. Modi S., ”Comparison of three obstacle avoidance methods for an autonomous guided vehicle”, M.Sc. Thesis Dep.of Mechanical, Industrial and Nuclear Engineering, University of Cincinnati, 2002.
8. H. Serajy, A. Howard, “Behavior-based robot navigation on challenging terrain: a fuzzy logic approach”, IEEE Trans. on Robotics and Automation, vol.18, no.3, June 2002.
9. P. Vadakkepat, O. Miin, X. Peng, T. Lee, “Fuzzy behavior based control of mobile robots”, IEEE Trans. on Fuzzy Systems, vol.12, no.4, August 2004.