Sensors and Materials, Vol. 31, No. 3 (2019) 1021–1036
MYU Tokyo

S & M 1832

Obstacle Avoidance of Mobile Robot Based on HyperOmni Vision

Shih-An Li,* Li-Hsiang Chou, Tsung-Han Chang,
Chao-Hsu Yang, and Yu-Cheng Chang

Electrical and Computer Engineering, Tamkang University,
151, Ying-Chuan Rd., Tamsui, New Taipei City, Taiwan (R.O.C.)

(Received November 23, 2018; accepted February 7, 2019)

Keywords: dynamic window approach, artificial potential field, HyperOmni Vision, mobile robot, obstacle avoidance

In this paper, we present a method that combines an improved dynamic window approach (IDWA) and an artificial potential field to avoid obstacles for autonomous wheeled robots with HyperOmni Vision. The experimental environment follows the rule of RoboSot avoidance challenge in the Federation of International Robot-soccer Association (FIRA) competition. There are three parts of this study, namely, (1) image processing, (2) an IDWA, and (3) an improved artificial potential field. In the avoidance challenge in FIRA, the robot must avoid eight obstacles that are placed randomly in the playing field. The results indicate that the three-wheeled robot can avoid obstacles autonomously by using the method proposed in this paper. Our team won the FIRA avoidance challenge championship this year by the method proposed in this paper.

1. Introduction

Designing a robot soccer system involves a combination of many disciplines that require various academic theories and techniques. There are two essential parts in completing the avoidance challenge, for example, obstacle detection and path planning. It is important to consider the robot's path plan to avoid obstacles. Some researchers used an artificial potential field(1,2) to seek targets and obstacles. The robot regarded a target as an attractive force and an obstacle as a repulsive force. It then combined all the forces to establish the direction of motion. This method has been widely used to avoid obstacles and move smoothly. However, owing to local optimization problems, it easily causes a local minima problem before arriving at its destination.

The method we proposed in this paper is based on the dynamic window approach (DWA), which establishes the dynamic window and the predicted trajectories through an OmniHyper camera. After that, the DWA was combined with the artificial potential field to renew the predicted trajectories and finish the obstacle avoidance.

This study is carried out to increase the stability of the robot and ensure that the robot avoids collision with obstacles so that it can move smoothly. The experimental environment is an

*Corresponding author: e-mail: lishyhan@gmail.com
https://doi.org/10.18494/SAM.2019.2226

ISSN 0914-4935
https://myukk.org/

avoidance challenge field in the RoboSot competition. In this study, the avoidance challenge field is used as the experimental environment, and a three-wheeled robot is used as the testing platform. This test platform is equipped with sensors such as an omnidirectional camera, an inertial measurement unit (IMU), and a motor encoder. The omnidirectional camera will capture the environmental image of the robot. The IMU detects the angle of rotation of the robot and the acceleration of the movement. The motor encoder is used to detect the moving distance and speed of the robot. The robot can calculate the better obstacle avoidance path through these parameters.

2. Background Discussion

In this section, our robot and experimental environment are described. This section is divided into the following two parts: (1) image processing system and (2) path planning. These parts are briefly described below.

2.1 Image processing system

In this study, an omnidirectional vision system is used to capture an image. This omnidirectional vision system includes a hyperbolic mirror and a high-speed camera, as shown in Fig. 1. The high-speed camera, Prosilica GC655, is made by Allied Vision. Moreover, the features are shown in Table 1. Prosilica GC655 is a high-performance vision camera with a GigE Vision compliant gigabit ethernet interface. This camera performs up to 90 FPS (frame per second) in the full resolution mode. The vision system is located at the center of the sixth-generation robots designed in our laboratory, as shown in Fig. 2. The computer on the robot can obtain a 360° circular image using the camera, as shown in Fig. 3.

After obtaining the image, the soccer and goal are identified by using color models, such as RGB, HSV, and CMYK color models. The RGB color model exhibits a nonlinear change in color saturation. Moreover, all colors are composed of the values of the three components in RGB, thus resulting in a high correlation between the three components. When the

Fig. 1. (Color online) Omnidirectional vision system.

Table 1

Model no.	Prosilica GC 655	Interface	IEEE 802.3
Resolution	659 (H) × 493 (V)	Sensor	Sony ICX414
Sensor type	CCD Progressive	Sensor size	Type 1/2
Max. frame rate at full resolution			90 FPS

contrast, brightness, or color saturation in the image is changed, the effects among these three components must be considered. The RGB color model is easily affected by shadows or light, thus reducing the rate of color recognition. In contrast, the HSV color model(3) is more resistant to the aforementioned effects. In this study, the HSV color model shown in Fig. 4 is used. The HSV color model consists of the following three components: hue (H), saturation (S), and value (V).

Image noise is a common issue in image processing. Therefore, we binarized the image to filter the noise. In this study, the HSV color model is used to separate obstacles. However, the shadows of obstacles have negative effects on binarization, misleading image noise between different color models. Therefore, breadth-first search(4) is used to filter image noise.

The breadth-first search is an uninformed search, so we need to use queues to record the order of visits. We assume that the gray area is an obstacle. In the breadth-first search, we initially set queue-containing vertex 1 and search for its surrounding adjacent vertices. We then dequeue vertex 1 and enqueue vertices 2, 3, and 4. The queue now contains vertices 2, 3, and 4, as shown in Fig. 5(a) Assume that we first visit vertex 2 and search for its surrounding adjacent vertices and then dequeue vertex 2 and enqueue vertices 5 and 6, as shown in Fig. 5(b) Moreover, the breadth-first search terminates until the queue is empty.

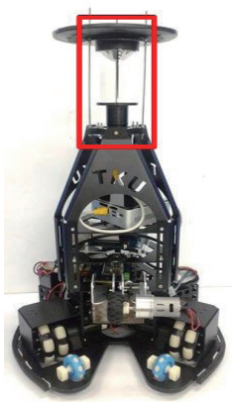


Fig. 2. (Color online) Sixth-generation robot.

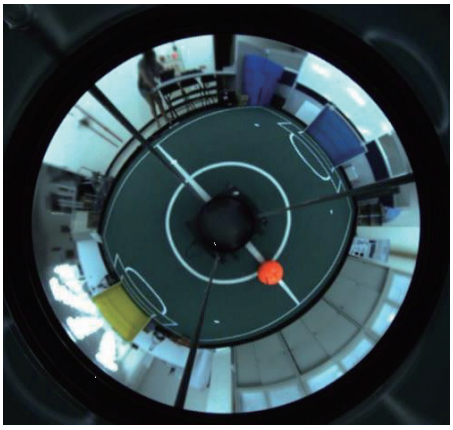


Fig. 3. (Color online) Image of an omnidirectional vision.

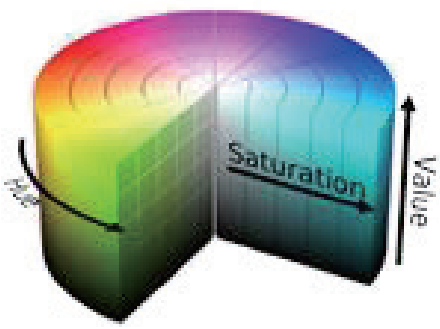
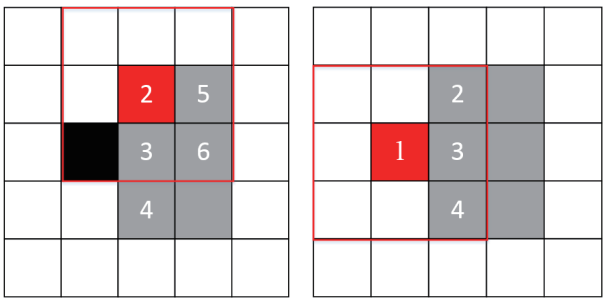


Fig. 4. (Color online) HSV color model.



(a) (b)

Fig. 5. (Color online) (a) Breadth-first search. (b) Breadth-first search.

In this study, we defined a specific size of the black area in this image. Objects under the size of the definition will be considered as image noise; otherwise, objects will be considered as obstacles (Fig. 6).

2.2 Path planning

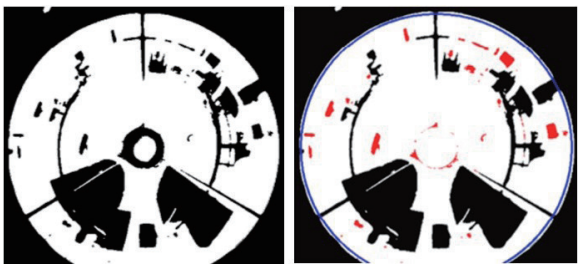
2.2.1 Dynamic window approach

Path planning is the ability of an autonomous robot to move to a destination. The DWA was proposed by Fox *et al.* in 1997.[5] It is a collision avoidance algorithm that is efficient and robust.[6,7] The DWA is an algorithm that takes a variety of velocity samples from the speed space. These velocity samples are used to simulate the trajectories of the robot to the target within a certain time. Pick the best trajectory from these trajectories. There are two parts in the DWA algorithm, namely, (1) search space and (2) optimization, which will be described below.

In the search space, there are three parts, namely, (1) circular trajectories, (2) admissible velocities, and (3) dynamic window. In the circular trajectory section, there are linear moving speed and angular velocity (v, ω) in the velocity space. In the next n time interval, the velocity sample is selected from the velocity space as the moving path of the robot to the target point. In the admissible velocity section, when obstacles are placed around the robot, the translation and rotation speeds of the robot are limited. Assume the sample speed is (v, ω) and dist(v, ω) is the length of the curve trajectory generated by the movement that will collide with the nearest obstacle based on the sample speed. The length can be obtained from the radius "r" of the curved path and the angle "γ" between the collision point and the robot, so the arc length can be obtained from "r", as shown in Fig. 7. Let $\dot{v}_b$ and $\dot{\omega}_b$ be accelerations for breakage. According to the above conditions, the suitable speed sample can be defined as

$$V_a = \{(v,\omega) \mid v \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{v}b} \wedge \omega \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{\omega}b}\}. \tag{1}$$

In the dynamic window section, according to the acceleration limit of the motor, the entire speed space can be reduced to the speed space that can be reached in the next moment. The reduced speed space is the dynamic window. Suppose t is the acceleration $\dot{v}$ and angular



(a) (b)

Fig. 6. (Color online) (a) Binarized image and (b) breadth-first search.

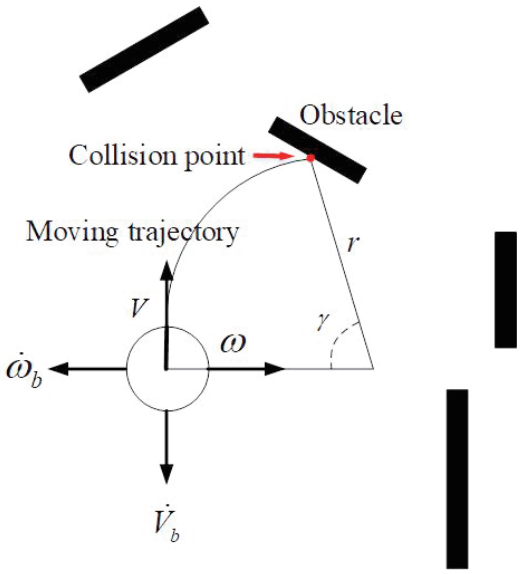


Fig. 7. (Color online) Calculation of "*dist*(*v*, *ω*)".

acceleration $\dot{\omega}$ that the motor can provide at each moment, and (v_a, ω) is the actual speed. The velocity space V_d of the dynamic window can be defined as

$$V_d = \{(v,\omega) \mid v \in [V_a - \dot{v} \cdot t, V_a + \dot{v} \cdot t] \wedge \omega \in [\omega a - \dot{\omega} \cdot t, \omega a + \dot{\omega} \cdot t]\}. \tag{2}$$

The dynamic window is a speed space centered on the actual speed. In the next moment, all the curved tracks are expected not to exceed this speed space and not collide with the obstacles in the space. According to the above three parts, V_r denotes the velocity sample contained in the final search space and V_s denotes the speed space that the robot can provide, as shown in Fig. 8, and V_r can be defined as

$$V_r = V_s \cap V_a \cap V_d. \tag{3}$$

After determining the resulting search space V_r, a velocity is selected from V_r. The algorithm scores the curve trajectory generated by the velocity sample through an evaluation function and selects the curve trajectory with the highest score as the path of the next robot movement. The evaluation function is

$$G(v,\omega) = \sigma(\alpha_g \cdot heading(v,\omega) + \beta_g \cdot dist(v,\omega) + \gamma_g \cdot velocity(v,\omega)) . \tag{4}$$

The target heading (v, ω) measures the alignment of the robot with the target direction. It is given by $180 - \theta$, where θ is the angle of the target point relative to the robot's heading direction, as shown in Fig. 9.

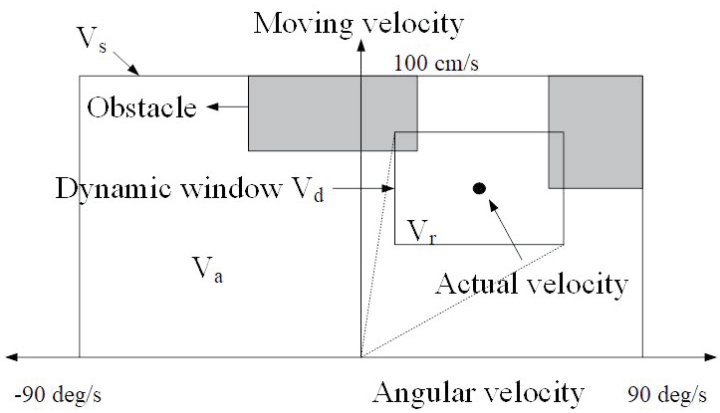


Fig. 8. Dynamic window.

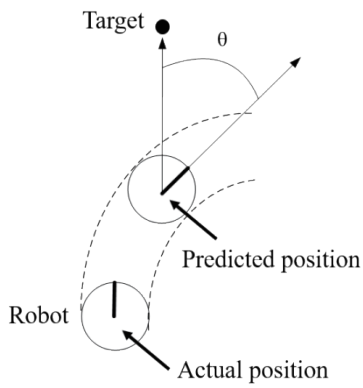


Fig. 9. Illustration of angle θ.

2.2.2 Artificial potential field

The artificial potential field algorithm is a robot path planning algorithm proposed by Khatib in 1986 at Stanford University.[8] It regards the target and obstacle as objects that are gravitational and repulsive to the robot, and the robot moves along the combined force of gravity and repulsive force, as shown in Fig. 10.

The artificial potential field has the advantages of simple mathematical description, low computation speed, high response speed, and smooth trajectory planning. In recent years, the artificial potential field has been widely used as a method in the robot path planning,[9,10] This method has become one of the most mature and effective traditional calculation methods.

There are problems associated with the characteristic of the artificial potential field. According to Eqs. (5.1)–(5.3), we learned that F_{att} decreases when the robot is close to the target and F_{rep} increases rapidly when the robot is close to the obstacle.[11] Therefore, the artificial potential field is a non-optimal path solution and has a local minima problem,[12] as shown in Figs. 11(a) and 11(b).

The equations used in the artificial potential field method are as follows:

$$F_{att} = \alpha \times v, \tag{5.1}$$

$$F_{rep} = \begin{cases} \beta \times \left(d_s - d_{ob}\right)^2 \times v, & d_{ob} \le d_s \\ 0, & d_{ob} > d_s \end{cases} \tag{5.2}$$

$$F = F_{att} + F_{rep}, \tag{5.3}$$

where F_{att} is the attractive force, F_{rep} is the repulsive force, F is the composition of forces, α is the parameter of the attractive force, β is the parameter of the repulsive force, v is the vector of the robot velocity, d_{ob} is the distance between the robot and the closest obstacle, and d_s is the safe distance between the robot and the obstacle, as shown in Fig. 12.

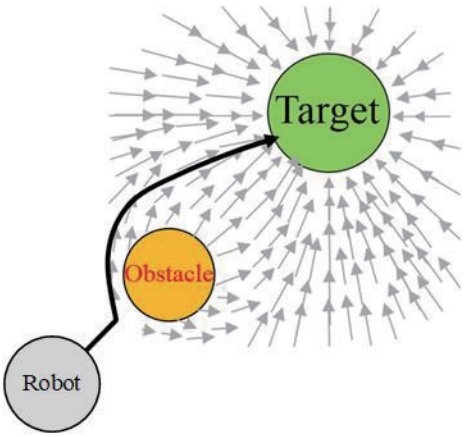


Fig. 10. (Color online) Path planning through Artificial Potential Field.

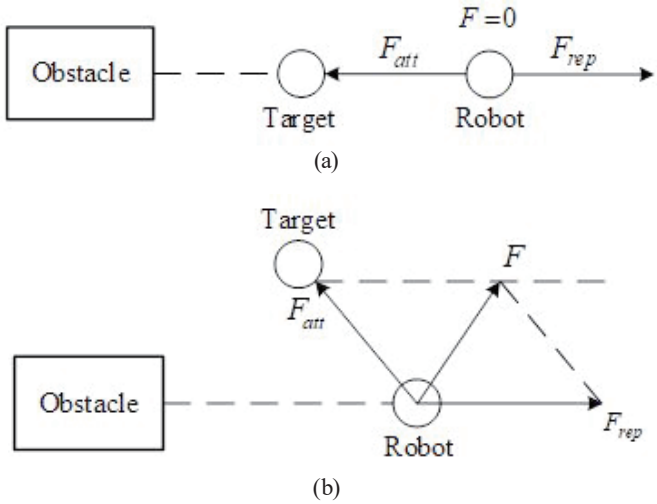


Fig. 11. (a) Non-optimal path. (b) Local minima problem.

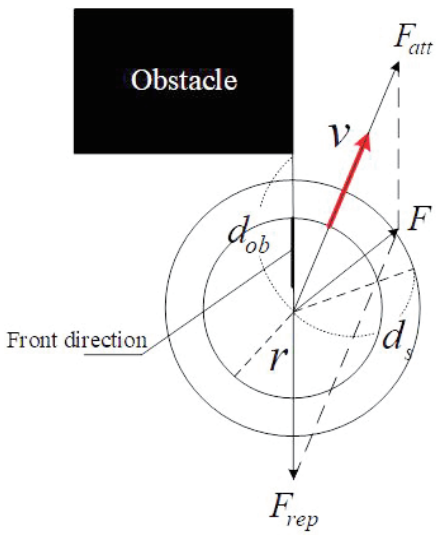


Fig. 12. (Color online) Artificial potential field.

In the omnidirectional mobile platform, the equations for the concept of the artificial potential field are as follows:

$$F_{att} = \alpha \times v_{new}, \tag{6}$$

$$F_{rep} = \left(\left(d_{ob} - d_s\right)\cos\theta_R + \left(d_{ob} - d_s\right)\sin\theta_R\right) \times \beta \times v, \tag{7}$$

$$F = F_{att} + F_{rep}, \tag{8}$$

where v_{new} is the new velocity vector of the robot calculated by DWA, and v is the vector of the robot velocity, which is set by the user interface. The schematic diagram of the artificial potential field method of the omnidirectional mobile platform is shown in Fig. 13.

3. Research Methods

The flow chart of the wheeled robot obstacle avoidance method proposed in this paper is shown in Fig. 14. The industrial personal computer placed on the robot first takes the image captured by the camera for image processing. Then, the improved DWA method proposed in this chapter is used to determine the most suitable path of motion.(13) Subsequently, the path is recorrected by the artificial potential field algorithm to avoid nearby obstacles. Finally, a motion vector is produced to make the robot move.

Take the movement of differential robots as an example. The velocity space in the DWA contains v and ω. The robot determines whether the trajectory will hit the obstacle, and the algorithm searches for a velocity space over time and calculates the new robot's trajectory, as shown in Fig. 15.

Compared with the differential mobile platform, the omnidirectional wheeled platform moves in a 360 degree linear motion.(14) Therefore, the study only considersand v when moving, and then θ and v are respectively converted into an X vector and a Y vector, as shown in Fig. 16. We set the x-axis for the front direction. We calculated the angle using the front direction. Moreover, we considered that counterclockwise is positive and clockwise is negative. The equations are as follows:

$$x = |v| \cos\theta, \tag{9.1}$$

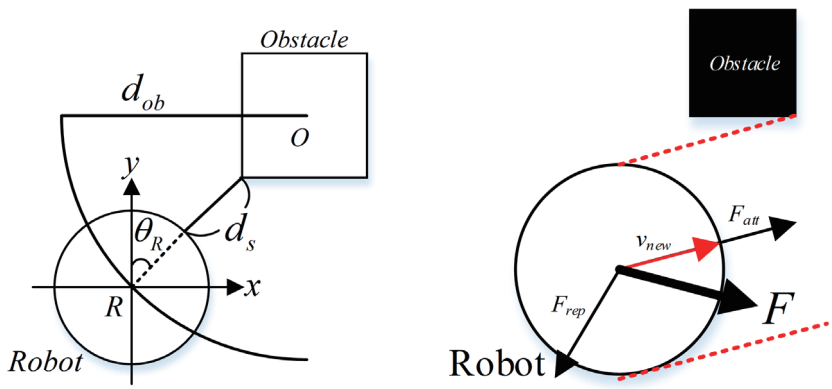


Fig. 13. (Color online) APF of the omnidirectional mobile platform.

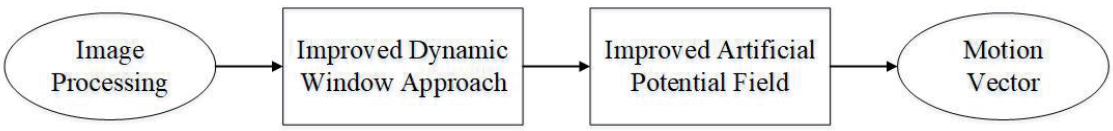


Fig. 14. Flow chart of obstacle avoidance.

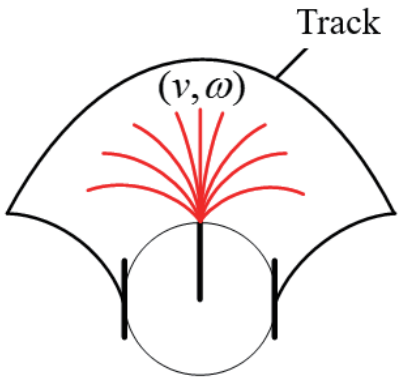


Fig. 15. (Color online) Trajectory of the differential robot.

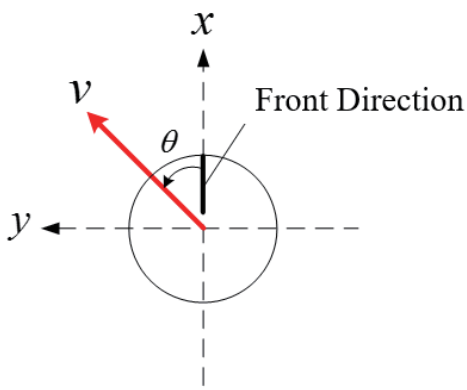


Fig. 16. (Color online) Vector conversion.

$$y = |v| \cos\theta. \tag{9.2}$$

Since the experimental environment in this study follows the rule of the Federation of International Robot-soccer Association (FIRA)[15] obstacle avoidance challenge, the scan line of image recognition was used to determine obstacles. The robot determines the free area according to the position of the obstacle, as shown in Fig. 17.

The robot prioritizes the active areas, and the area closer to the target point takes precedence. Therefore, the robot first determines that area 2 is the priority path, then it is area 1, and the final is area 3. After completing the above steps, the robot will calculate the width of area 2 and determine if the width can accommodate it to ensure that it does not hit obstacles. Take two obstacles as the identification of three scan lines. The distance between the robot and the obstacle is calculated using the distance information obtained from the known scan line. The robot uses R1 as the baseline to operate with all the scan lines L1, L2, and L3 of another obstacle. Then, it can calculate the distances D_1, D_2, and D_3 as θ_1, θ_2, and θ_3, respectively, as shown in Fig. 18. The calculation of R_2 and R_3 is the same as that of R_1. There will be $i*j$ distance information in total, and finally, the shortest distance is found to be smaller than the fuselage. The equations are as follows:

$$V = \{D_k \,|D_k : IR^2\}, \quad i, j, k = 1, 2, 3, \ldots, n, \tag{10}$$

$$W = \{\min D_k \,|D_k = \sqrt{R_i^2 + L_j^2 - 2R_i L_j \cos\theta_k}\}, \quad \forall D_k \in V. \tag{11}$$

The moving speed vector v_{new} is first obtained by Eqs. (10) and (11). Then Eq. (12) can obtain the minimum angle between the robot and the obstacle. From this angle, the obstacle distance D_R can be calculated as shown in Fig. 19.

$$\theta_{min} = \theta_{V_{new}} - \theta_R \tag{12}$$

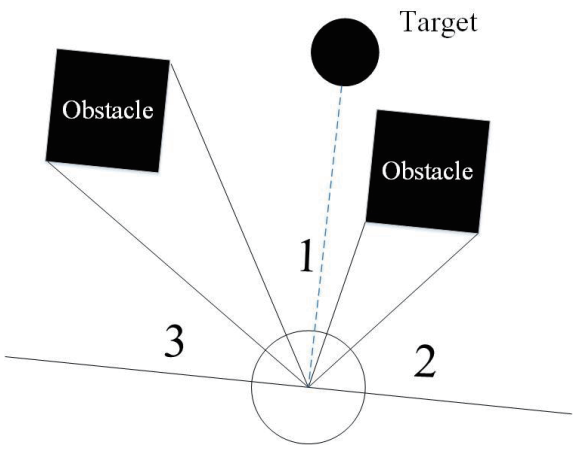


Fig. 17. Active area of the robot.

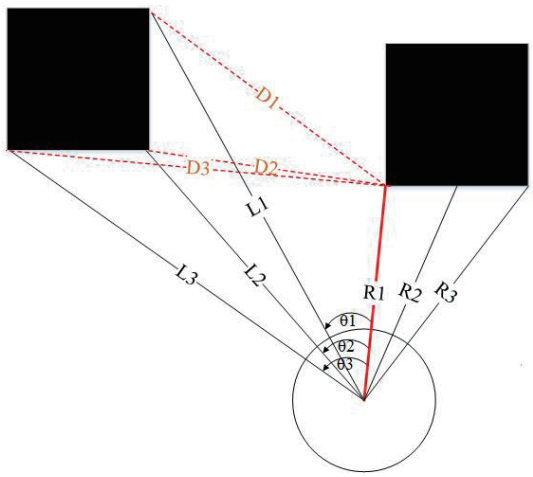


Fig. 18. (Color online) Calculation of the distance between two obstacles.

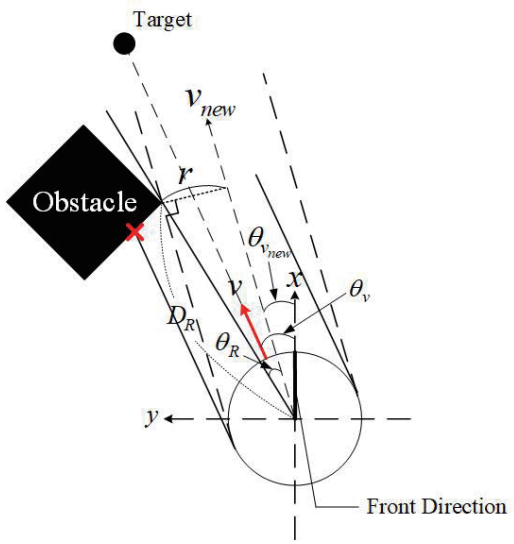


Fig. 19. (Color online) DWA to avoid the trajectory of obstacles.

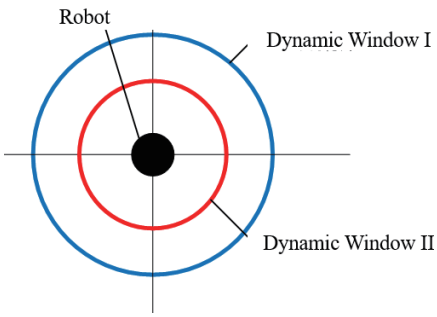


Fig. 20. (Color online) Improved dynamic window approach.

In addition to changing the original differential mobile platform into an omnidirectional mobile platform, two layers of dynamic windows called dynamic windows I and II are used, which is considered an improved DWA method, as shown in Fig. 20.

When the robot moves, it uses V to predict the robot's trajectory. In addition, it plans the trajectory of obstacle avoidance when there are obstacles in the track. The equations are as follows.

$$\theta_V = \cos^{-1}\left(\frac{\vec{X}\cdot\vec{T}}{|X||T|}\right) \tag{13}$$

$$\begin{cases} \theta_{V_{new}} = \overline{X} - \tan^{-1}\left(\left|\dfrac{r}{D_R}\right|\right), & r > D_R \cdot \tan(\theta_R) \\ \theta_{V_{new}} = \theta_V, & r \le D_R \cdot \tan(\theta_R) \end{cases} \tag{14}$$

Here, $\vec{T}$ is a target vector and θ_{min} is the angle of the obstacle closest to the target.

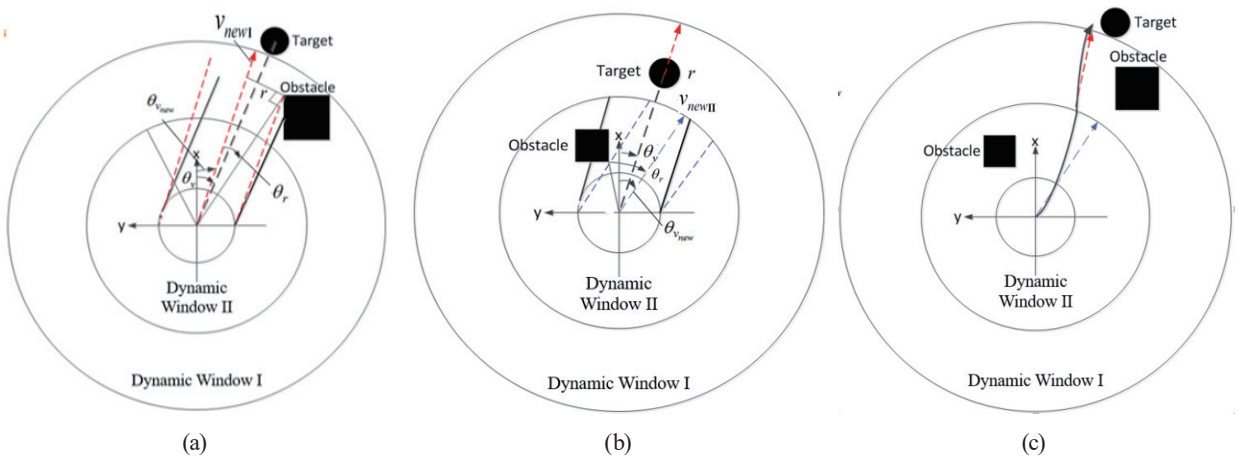
The detection range of dynamic window I is expanded so that the robot can detect relatively far obstacles to select a better path and plan the trajectory, as shown in Fig. 21(a). Then, the robot sets the target in Dynamic Window II and detects relatively close obstacles to replan the trajectory, as shown in Fig. 21(b). In addition, the artificial potential field is used to ensure that the robot will not touch the obstacles. In this manner, the robot filters out the better path, as shown in Fig. 21(c).

4. Research Results

The method proposed in this paper was used to combine and improve the DWA and artificial potential field methods, and the authors named it space-perception approach (SPA). In this study, the experimental results are divided into two parts, namely, simulation and implementation.

In the simulation part, the advantages and disadvantages of the three algorithms, namely, DWA, artificial potential field, and SPA, are compared. The simulation results in the MATLAB program are shown in Fig. 22–25 below.

It can be seen from the results that, when the DWA encounters obstacles around it, it reduces its moving speed and leads to a narrower measurement range. In the artificial potential field, when there are more than two obstacles around the robot, the movement trajectory causes oscillation, making the robot hit the obstacles. In the SPA method, the robot's movement path is the smoothest and fastest, as shown in Fig. 24.



(a) (b) (c)

Fig. 21. (a) Path planning of dynamic window I. (b) Path planning of dynamic window II. (c) Path planning of improved DWA.

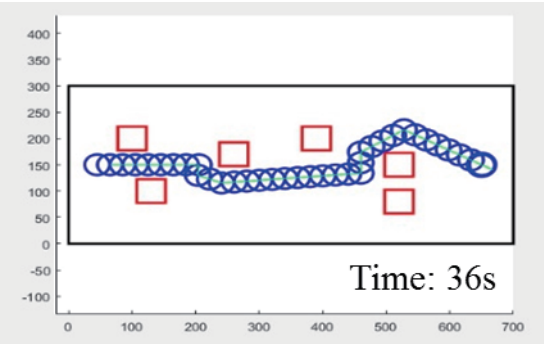


Fig. 22. (Color online) Dynamic window approach.

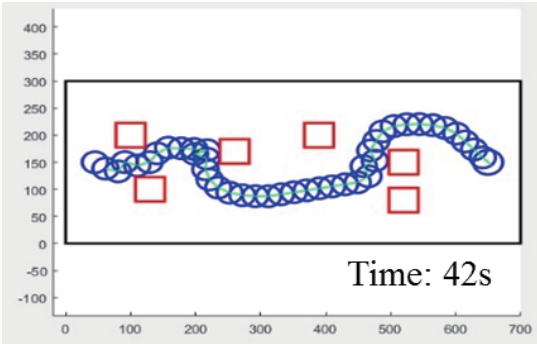


Fig. 23. (Color online) Artificial potential field.

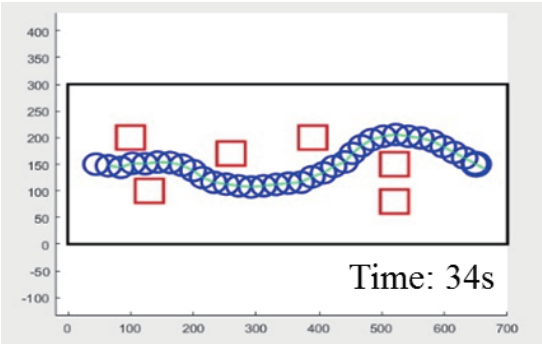


Fig. 24. (Color online) Space-perception approach.

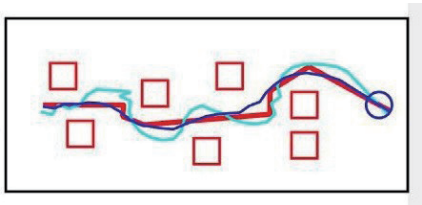


Fig. 25. (Color online) Three different methods of robot trajectory.

In the implementation part, the proposed method is compared with the other three methods and verified in the FIRA avoidance challenge, as shown in Fig. 25. The avoidance challenge rule sets the yellow goal as the destination. A robot must move from a blue goal to a yellow goal without touching any obstacles. The referee randomly places eight obstacles on the path. The size of each obstacle is similar to that of a black box with a length of 45 mm, a width of 45 mm, and a height of 70 mm. The distance between the obstacles is at least 60 cm. The robot is 45 cm in length and width, so it has an error range of only 15 cm when crossing an obstacle. The speed of the wheeled robot is set to 25 cm/s. The distance between the starting point and the destination is 4.2 m. We tested each method 20 times and took the average data, as shown in Table 2. The method proposed in this paper has the highest success rate and is also the fastest.

A piece of pink paper was stuck on top of the robot and the movement trajectory of the robot is recorded according to the fisheye set on the ceiling. The trajectory is drawn after image processing, as shown in Fig. 26.

The first method uses the SPA as the obstacle avoidance algorithm. This method not only makes the trajectory smooth but also plans a better path. When the robot moves to the midfield, owing to the dynamic window I, it senses an obstacle farther to the right and passes through a wider path on the left side. In this manner, path planning is more efficient and more stable, as shown in Fig. 27.

Table 2
Comparison of four different methods.

	Fig. 27	Fig. 28	Fig. 29	Fig. 30
Time (s)	19	21	24	Fail
Distance (m)	4.75	5.25	6.00	Fail
Stability	Excellent	Great	Bad	Worst
Efficiency	Excellent	Great	Good	Worst
Success rate	95%	90%	40%	0%

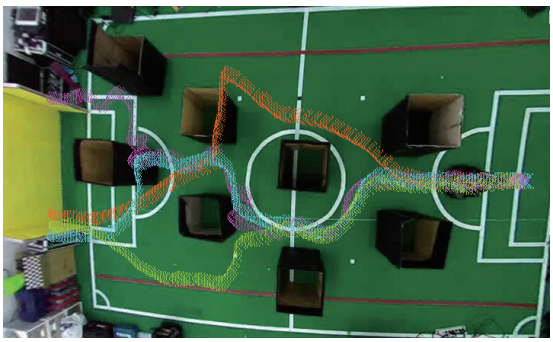


Fig. 26. (Color online) Comparison of four different methods.

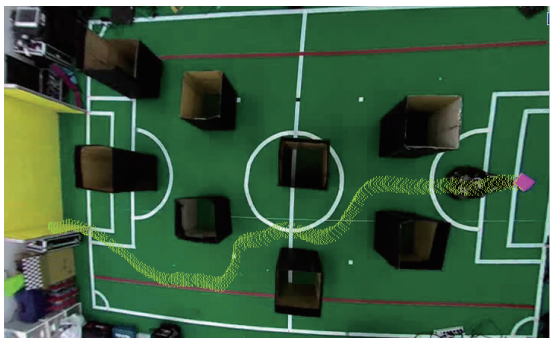


Fig. 27. (Color online) Space-perception approach.

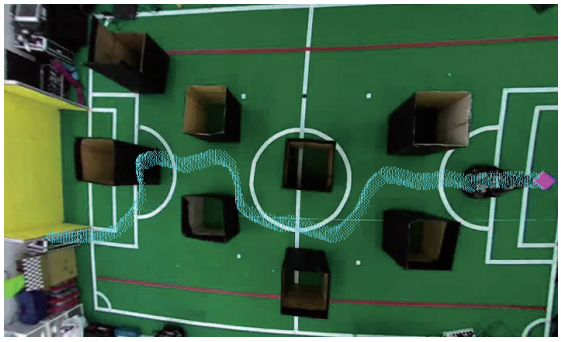


Fig. 28. (Color online) Original DWA with artificial potential field.

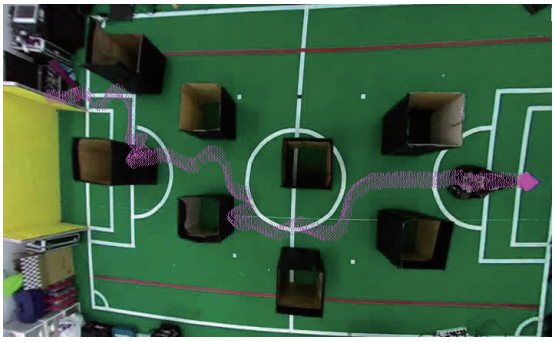


Fig. 29. (Color online) Artificial potential field without DWA.

The second method uses the original dynamic window approach with the artificial potential field as obstacle avoidance protection. Its trajectory is also smooth, but the path selection is much worse than the SPA. When the robot moves to half of the field, since the detection range of the DWA window is small, it selects a poor path on the right side. Although the stability is excellent, the path planning efficiency is worse, as shown in Fig. 28.

The third method only uses the artificial potential field. The path selection is uninformed and very inefficient. This is because the presence of repulsive forces makes the trajectory produce many shocks. In this approach, the stability is not sufficient, probably producing local minima problems. Moreover, the robot touches the obstacles easily when the distance between the obstacles is decreased, as shown in Fig. 29.

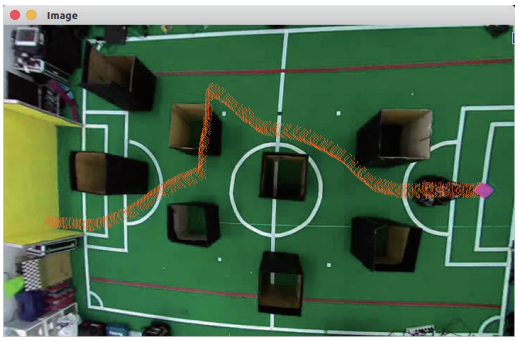


Fig. 30. (Color online) DWA algorithm only.

The fourth method uses only the DWA without the artificial potential field. The result is the worst; the robot can only plan the path. The robot touches the obstacles to produce numerous collisions without the artificial potential field when it goes through the half field. Thus, the test shows us the importance of the artificial potential field as the obstacle avoidance protection, as shown in Fig. 30.

5. Conclusions

In this study, an effective obstacle avoidance method is proposed to achieve an omnidirectional mobile platform with an omnidirectional vision system. Moreover, winning the championship in the 2017 FIRA avoidance challenge proves that the method is robust. In future developments, as long as obstacles can be identified through an imaging system, the method can be applied to various experimental environments. The method that we proposed in this paper improves these two algorithms and combines them. It uses the breadth-first search to filter image noise, which makes the image recognition more stable and accurate. Moreover, the success rate applied in the competition approaches 100%. Therefore, this not only avoids a collision to reduce the loss of the robot equipment but also has a higher efficiency to accomplish the mission.

Acknowledgments

This research was supported in part by the Ministry of Science and Technology (MOST) of the Republic of China under contract MOST 106-2221-E-032-007.

References

1 M. C. Lee and G. P. Min: Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics **2** (2003) 735.
2 M. G. Park, H. J. Jae, and C. L. Min: IEEE Int. Symp. Industrial Electronics **3** (2001) 1530.
3 R. Gonzalez and R. E. Woods: Digital Image Processing (Pearson Prentice Hall, NJ, 2011) p. 470.
4 T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein: Introduction to Algorithms (MIT Press Cambridge, Massachusetts, 2001) p. 43.
5 D. Fox, W. Burgard, and S. Thrun: IEEE Rob. Autom. **4** (1997) 23.

6 M. Seder and I. Petrovi´c: IEEE Int. Conf. Robotics and Automation (IEEE, 2007) 1986.
7 C. C. Chou and F. L. Lian: Proc. 48th IEEE Conf. Decision and Control 2009 held jointly with the 2009 28th Chinese Control Conf. (CDC/CCC) (IEEE, 2009) 5971.
8 O. Khatib: IEEE Int. Conf. Robotics and Automation **2** (1985) 500.
9 Q. D. Zhu, Y. J. Yan, and Z. Y. Xing: Int. Conf. Intelligent Systems Design and Applications **2** (2006) 622.
10 L. Zhou and W. Li: IEEE Int. Symp. Computational Intelligence and Design **2** (2014) 429.
11 C. L. Liu, J. Tang, and J. Y. Yang: Int. Conf. Natural Computation **2** (2011) 1011.
12 L. Tang, S. Y. Dian, G. X. Gu, K. L. Zhou, S. H. Wang, and X. H. Feng: Int. Conf. Computer Science and Information Technology **9** (2010) 633.
13 S. Piyapat, N. Nattee, and S. Attawith: Int. Conf. Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTIC, 2013) 1.
14 G. Y. Li, Y. Y. Wu, and W. Wei: World Congress on Intelligent Control and Automation (WCICA, 2008) 5759.
15 FIRA: http://www.fira.net

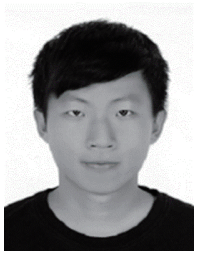
About the Authors

Shih-An Li received his B.S. degree in Electrical Engineering from National Taipei University of Technology, Taipei, Taiwan, in 1997. He received his M.S. and Ph.D. degrees in Electrical Engineering from Tamkang University, Taiwan, in 2004 and 2008, respectively. He joined the Electrical Engineering Department of Tamkang University in 2010 and is now an assistant professor. His research interests include a fuzzy system, intelligent control, SOPC design, FPGA design, and genetic algorithms.

Li-Hsiang Chou received his B.S. and M.S. degrees in Electrical Engineering from Tamkang University, Taipei, in 2014 and 2016, respectively. He is currently a Ph.D. student in the Department of Electrical and Computer Engineering, Tamkang University. His primary research interests include FPGA design, intelligent control, and robot system.

Tsung-Han Chang received his B.S. and M.S. degrees in Electrical Engineering from Tamkang University, Taipei, in 2014 and 2016, respectively. He is currently a Ph.D. student in the Department of Electrical and Computer Engineering, Tamkang University. His primary research interests include image processing, system integration, and robot system.

Chao-Hsu Yang received his B.S degree in Electrical Engineering from Tamkang University, Taipei, in 2018. He is currently an M.S. student in the Department of Electrical Engineering, National Taiwan University of Science and Technology. His primary research interests include robot system and deep learning.



Yu-Cheng Chang received his B.S degree in Electrical Engineering from Tamkang University, Taipei, in 2018. He is currently an M.S. student in the Department of Mechanical and Electromechanical Engineering, National Sun Yat-sen University. His primary research interests include system integration and self-driving cars.