An Improved Navigation Method for Robot in Indoor Dynamic Environment Based on Ground Extraction

Dang Khanh Hoa, Than Viet Duc, Do Trong Anh, Vu Song Tung, Le Dzung, Nguyen Tien Dzung[1*]

Hanoi University of Science and Technology, No. 1, Dai Co Viet, Hai Ba Trung, Hanoi, Viet Nam

Received: September 13, 2018; Accepted: November 26, 2018

Abstract

This paper presents a ground-based navigation method for an indoor robot to reach a predetermined target. By mining depth maps effectively, a mobile robot could find right path and self-locate with a proposed algorithm named Always Move Straight to the Destination (AMDS). The proposed navigation system extracts the ground plane from the depth map provided by an RGB-D camera. Then the navigation system has established an optimal obstacle avoiding strategy with a success rate of 98.7% which is better than some recent comparison methods based on Artificial Neural Network (ANN) classifiers or method of combination of two algorithm of Dynamic Window Approach (DWA) and Anytime Repairing A (ARA*). The robot's navigational capability is more flexible than the comparison methods because the angle of direction adjustment is 1 degree. The proposed ground-based navigation method could be integrated into low cost robots.*

Keywords: Depth map, Ground plane, Navigation, Path direction

1. Introduction

In recent years, mobile robots are being used in the community widely to support vulnerable individuals such as children or the elderly or the sick persons. These subjects require careful care with the supported system requirements to be aware of the context in which they are operating. The data acquisition automation can only meet the full awareness of the environment surrounding the subject. It allows one to recognize the current situation through full collection of parameters. In addition, the system is also capable of predicting the potential risks that may occur depending on the probability of harm to the cared person. Mobile robots have created a great advantage in dynamic environments such as offices, restaurants, hotels, hospitals and airports. In the process of moving to a predetermined destination, this type of robot needs support to solve two problems, including positioning and avoiding obstacles. Positioning is the answer to where the robot is. Avoiding obstacles to get the robot to the destination. To reach the destination, it needs to overcome obstacles along the way.

There several ways for obstacle detection such as common approaches used lasers, ultrasonic and visual sensors. Both of works [1, 2] have implemented a real-time avoidance system for mobile robots with abilities of detection surroundings, obstacle-avoiding and moving toward the target area. But the robot only can reach the target area rather than target point. And the robot needs a powerful sensor that can collect data likely a camera.

In the last few years, visual navigation has been more concerned by using RGB-D camera such as Kinect sensor [3-12]. The works [3, 5, 7, 10, 12] present the development of a perception system for indoor environments to allow autonomous navigation for surveillance mobile robots. The system is composed by an avoiding obstacles subsystem using the Kinect and artificial neural network (ANN) subsystem to recognize five different configurations of the path such as ahead path, left path, right path and intersections. This work contributes for the development of intelligent control and navigation systems for autonomous mobile robots, but the system is trained with a few simple situations.

In recent years, most of researchers assert that the RGB-D camera is an extremely powerful image sensor because of the plenty of information that the computer could exploit from it. So, many researchers are interested in developing integrated RGB-D camera in robot. This sensor makes the system smarter with human-based behavior based on artificial intelligence.

In [4], an algorithm is proposed to avoid obstacles based on stereoscopic vision but it is quite simple and has not yet integrated on a model system. Paper [6, 8] was presented a matching algorithm and

[*] Corresponding author: Tel.: (+84) 988.355.343
Email: dzung.nguyentien@hust.edu.vn

iterative closest point (ICP) algorithm using only data of RGB-D sensor but it needs to improve the accuracy of experienced results. In [7, 8], both depth data and 3D point clouds are used as inputs of filters and cluster processing to avoid obstacles. The quality of 3D point cloud is quite unsatisfactory unless a part of future work is a noise reduction.

In this paper, the authors focus on the problem of robot movement using a unique RGB-D camera named Kinect. The proposed method retains the advantages of the approaches [6-8] by introducing a simple navigation method with no need to integrate artificial intelligent on mobile robot. The rest of the paper is organized as follows: Section 2 presents the problem of designing and installing a indoor robot. Section 3 describes in detail the navigation system implementation to perform applied algorithms. The fourth section is the obtained results and some important discussions to improve outcomes in real different environments. The fifth section shows several conclusions and the needed future work.

2. Design and assembly of mobile robot hardware

2.1 Basic design criteria

A robot typically operates in covered areas, so its design should meet some following basic criteria. The space inside the robot is large enough to fit containing two motors, batteries, controllers, laptops, and Kinect. The structure of the robot car must be stable. The vehicle's weight includes the itself weight, battery, two motors, a laptop and some control cards. An image sensor is located in a good view position. In particular, it looks as far away from the body as possible. The sensor is mounted so that it can see the nearest ground point at distance about $80cm$.

2.2 Modeling of robots

The parameter settings for a robot vehicle is described in Fig. 1. Kinect sensor is mounted at the top of the system of height h_2. The Kinect's optical axis is turned down so that it can see the nearest point on the ground G_{min}. Getting the open angle of 53^0 for the Kinect, the farthest ground point G_{max} is approximately $295cm$ from the vehicle position.

Based on the above requirements, the structure of the robot consists of two layers to provide enough space for the integrated device (Fig. 2). The actuator of the robot consists of two front wheels located controlled by servo motors and a passive wheel. Two direction wheels are in front to help the robot not slip as much as it starts to move. The Kinect camera is mounted on a $62cm$ high mount that is mounted in the center of the rear of the vehicle.

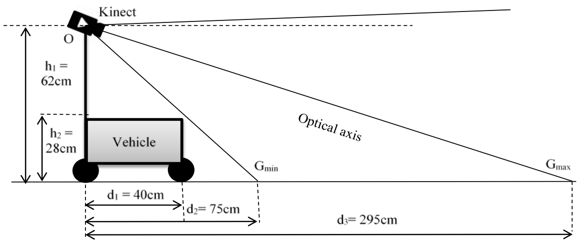
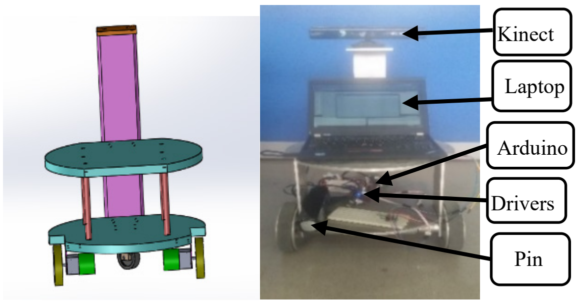


Fig. 1. The model of indoor mobile robot using Kinect



(a) (b)

Fig. 2. The CAD design and platform of mobile robotic system

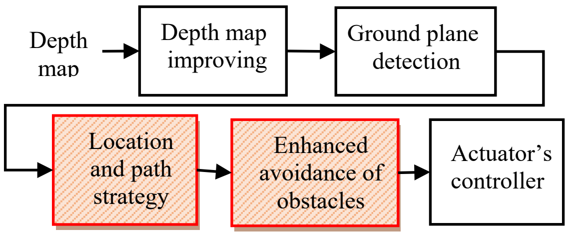


Fig. 3. Block diagram of the navigation system base on proposed algorithm

2.3 The hardware installation

Because the design is relatively reasonable, the installation of robots is done smoothly. The Kinect is mounted on the top of the vehicle to capture the frontal scene with a view angle down of 77^0 as depicted in Fig. 1. Each robot's engine is attached directly to the steering wheels which individually controlled by analyzing the video signals captured by the Kinect. The notebook herein is used to install the developed control software and interconnect with an Arduino kit, which used for control two motor drivers.

3. The Navigation System Implementation

The system includes three successive stages as shown in Fig. 3. The first functional block is responsible for enhancing the quality of depth image that it receives from the Kinect. The main objective of this group is to minimize noise in each image depth. The second function block is the task of creating neighbor point groups. These groups are not overlapping each other and the sum of all the groups are smaller than the size of original depth image. Each group will become a candidate for selection by the last function block based on a set of binding conditions.

3.1 Depth map improving

Depth map captured by RGB-D cameras such as a Kinect is often noisy at object's edges where miss truth values. The wrong depth value pixels had to been detected and removed from the depth map. A simple method estimates missing depth values from sufficient similar patches with available depth values [8]. But a depth map also contains numerous holes where no depth measurements are available. Especially, when a large region misses depth values completely, there is lack of repetitive patches for this surface structure and then this method may fail. Therefore, an efficient approach is applied to improve the quality of Kinect's depth image [9] by using a region growing method and a joint bilateral filter to fill the black holes.

3.2 Ground plane extraction algorithm

The block diagram of the ground extraction based on gradient depth map method is presented in Fig. 4.

3.2.1 Gradient depth map calculation

The task of the first block (Fig. 4) is to create a depth gradient map from the depth map provided by the stereo camera system according to formula (1).

$$dz = -hf \frac{1}{p^2\sqrt{p^2 + f^2}} d\rho \qquad (1)$$

After applying the above calculation to all points in the depth map, the results are two gradient depth maps including *x* axis depth difference map and *y* axis gradient depth maps.

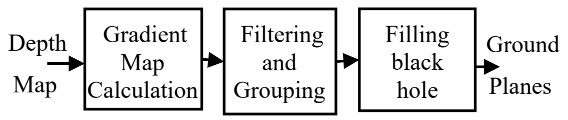


Fig. 4. Block diagram of the ground extraction method based on gradient depth map

The gradient depth maps usually contain a lot of noise due to the fact that the depth map itself is not perfect. The value of the gradient depth of the adjacent points is deviated from the true value even in fact they are in a same flat piece. So, these maps need to be calibrated. An effective and simpler way to solve this problem is to look at the depth difference of a point in an area that is centered. In each region, the vicinity of each point will be determined by sizes of neighborhood zone *w* that takes one of three values 3×3, 5×5 or 7×7.

The depth difference map is divided into two areas, the nearest and the distant, depending on the depth range compared to the median depth of 120. In the near area, the variability rate is low. In the far area, the variability rate of the difference depth rises fast and reach a much higher value than the near area. In the near area, the program uses the window $w = 3\times3$ to reduce the number of calculations. In the farther area, the program selects the size of the neighborhood zone 7×7. Neighborhood zone size 5×5 can be applied experimentally for either close or remote areas in the case of multiple neighborhood sizes. As a result, the depth difference maps have been calibrated more smoothly than the unmodified depth difference maps.

3.2.2 Filtering and Grouping

The grouping and filtering are performed simultaneously. This means that after collecting a group, the algorithm immediately checks that it satisfies ground conditions to keep or discard them. The ground conditions consist of two elements. First, the number of pixels of the region must be greater than a predetermined threshold. The second is $\frac{\partial z}{\partial x} = 0$ and $\frac{\partial z}{\partial y} \neq 0$; or if $\frac{\partial z}{\partial x} = 0$ and $\frac{\partial z}{\partial y} = 0$ then the region must be located completely in the quarter area from the bottom of the input image for higher accuracy in ground plane detection.

The set of zones after the filtering step is raw ground sources. They are the basis for building a complete integrated ground plane.

3.2.3 Filling black holes

After clustering and filtering, we obtain a set of ranges. This set of areas does not completely cover the ground, but as a piece of fabric with holes in different sizes. This is a matter of concern because it greatly influences the direction of movement. To overcome this disadvantage, the output data of grouping and filtering stages need to be processed further to achieve the target of smooth and seamless ground plane. Here are steps processing.

<u>*Step 1*</u>: Divide the depth map into blocks. The result will be a two-dimensional array of blocks with the size of each block *B*. Calculates the ratio of pixels in Ranges to the total number of pixels in block *Rate* .

<u>*Step 2*</u>: Based on *Rate* of each block, it is classified into three kinds of the block by two threshold φ_1 và φ_2 with $1 > \varphi_1 > \varphi_2 > 0$. If $Rate > \varphi_1$, this block is considered as one of full ground pixel. If $\varphi_1 > Rate > \varphi_2$, the block is in a bad area and there

may be obstructions here. If $Rate < \varphi_2$, it must fully contain of the pixels of the obstacles.

Step 3: Combine the results of step 2 to show and store in a convenient format.

3.3 Location and path strategy implementation

The positioning of robots is based on two factors. First, a 2D map of the operating environment is loaded into the robotic system. This map can be created from building drawings or quick sketches by subjective measurements. Placing the robot into the environment provides the program its starting position. These values are loaded into the program including the initial x-coordinate and y-coordinate. The robot's movement will change the initial coordinates and these coordinate values are called the current coordinates. This change is recorded continuously by software installed in the system. By comparing angular velocities of the two motors which drive wheels and mapping from the travel distance to the point coordinates on the map, it can trace the path of the robot. It also allows people to see robot's path through the graphing function of the program.

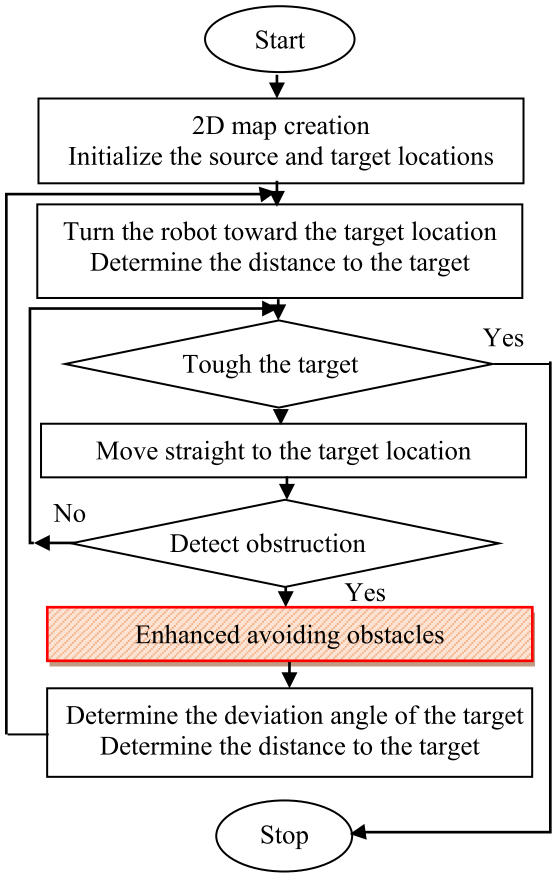


Fig. 5. Strategy of depth map based navigation for indoor mobile robot

Experimental robots are used the tactic named Always Move Straight to the Destination (AMDS). First, the system extracts the ground plane from the depth map provided by the RGB-D camera to find the robot direction. If there is not any obstacle, it always tries to move straight to the destination. If the robot meets obstacles, it will switch to the improved avoidance mode to overcome them (Fig. 5). The principle of moving always strives towards the goal in order to minimize the distance from the current position of the robot to the predetermined location of the target. That is, it always tries to move straight to the destination. However, this strategy might be halted if the robot is blocked by obstacles. From this moment, the tactic of avoiding obstacle has a higher priority than moving to the target.

3.4 Enhanced avoidance of obstacles

When obstacles appear, the ground where the robot is observing becomes narrower. There are two possibilities to occur as follows:

1. Robot is forced to redirect: It occurs when there are obstacles located closely in front of the car (Fig. 6a). Obstacles occupy a part or all the way. The straight ground is becoming increasingly narrowed, and it may lead to unsaved moves for the robot. The algorithm estimates acreages of the left and right ground sides, respectively, and then drives the robot toward a larger zone which corresponds to safer areas. The following is a recommended avoidance tactic.

The orientation angle MOz is the one coordinated by the optical axis of the camera Oz with the line going through the robot's position O and the midpoint M of the horizontal segment AB that connects two edges of the largest ground. It is a quarter of the vertical dimension of depth map from the bottom of the depth map to the horizontal segment as shown in Fig. 6a.

As soon as the robot determines the location of the obstacle, it operates in the avoidance mode by controlling the actuator follow one in two strategies. The first is that the robot can move in the direction of the vector $\overrightarrow{OM}$ (Fig. 6a). However, if the obstruction is too close to the front of the robot, the robot's body will easily collide with it. The second way is that the robot moves in two directions $\overrightarrow{ON}$ and $\overrightarrow{NM}$ sequentially. The directional vector $\overrightarrow{ON}$ allows the robot to keep enough its distance from the obstacle's front nearest corner. This way of avoiding obstacles makes the robot's body safer than the first strategy. The way to determine the turn angle θ and distance ON based on the uniform of the two triangles AMN

and AQO is as follows (Fig. 6a). It leads to (2) and (3) as below.

$$\theta = \arctan\left(\frac{MQ}{OQ - MN}\right) \quad (2)$$

$$\overline{ON} = \sqrt{MQ^2 + (OQ - MN)^2} \quad (3)$$

where $MQ = x_M - x_Q$, $x_M = x_A - x_B$, $OQ - MN = 75cm$ (Fig. 1). MN is the distance from the robot to the obstacle check line which contains segments AB, CD shown in Fig. 6a.

2. Robot continues to move forward: This happens when obstacles appear on the left or right side of the vehicle. Depending on the size of the obstacles, the obscurity of the ground is either wide or narrow. If the obstacle is small enough, the robot continues to move straight. If the invasion of the obstacle increases, the robot tends to move to the right sideway (Fig. 6b).

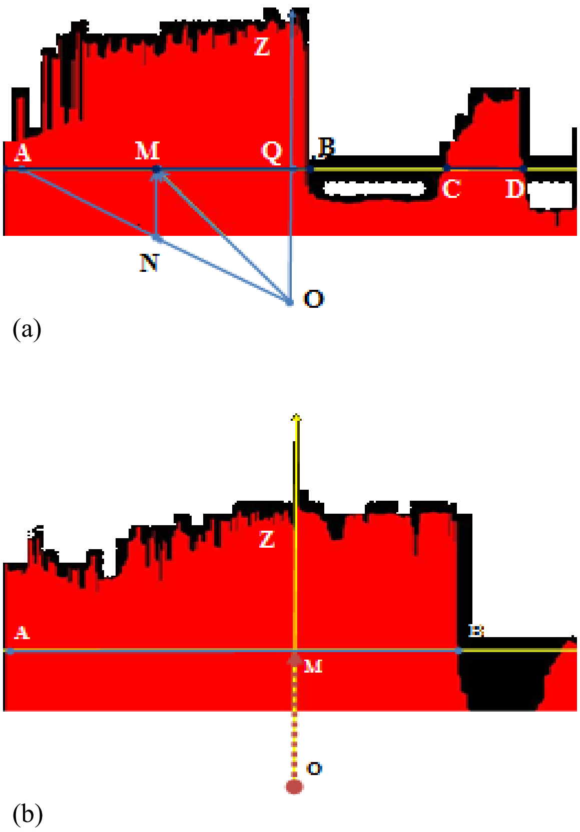


Fig. 6. The improved robot's redirection strategy for front obstacles case

Once the robot has determined the moving direction, obstacle avoiding is finished by moving the robot a segment of the road and the front side of the robot is controlled towards the destination. Then the robot continues to implement the guiding strategy.

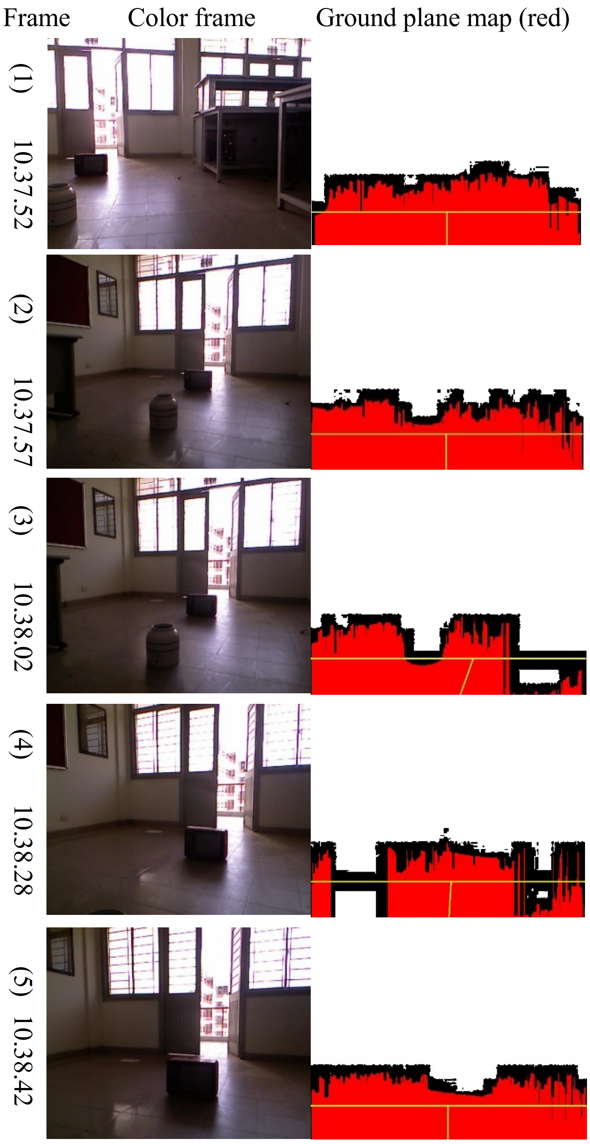
4. Experiences Results And Discussions

Over the duration of the experiment, only the Kinect sensor is used for localization and obstacle detection and avoidance, where this platform is mounted on a wheel-controlled vehicle. In this section, the authors describe the experimental results using the proposed method. The experiment is conducted in a real environment where the scenes is set up with different obstacles for testing the stability of the proposed method as depicted in Fig. 7. The robot can avoid successfully the dynamical obstruction appearing in its moving path from the original to the destination with full kind of navigation decisions such as moving straight or moving to the right or moving to the left. In addition, there are some mobile obstacles such as the phenomenon of a person entering the robot's view. During the robot moves on the floor of test room, both depth-based navigation video and color video are captured. They are useful to evaluate the quality of the test. As shown in Fig. 7, first and second lines illustrates the robot began to observe and result of turning the vehicle towards the destination and moving straight to the destination. The third line to the fourth line, the process performed avoiding the first obstacle. The 5th line shows the second process of navigating by finding out the direction and moving straight to the destination. Lines 6, 7 and 8 show tactics to avoid the second obstacle. The period from line 9 to line 10 is intended for the process of moving the robot. The robot approaches the target exactly at the time shown in line 11.

Figure 8 illustrates the results of tracking the robot location with two kinds of scenario consisting of one obstruction and multiple obstructions. The position of the start and end points are represented by 2D coordinates. The robot is initialized at 12h direction. After the robot receives the coordinates of the start and end points, it determines the rotation angle and moves toward the target as track shown the first blue segment. When the system detects the first obstruction, it determines the rotation angle to avoid obstacles and moves away from obstructions to ensure that they are safe enough (the first red segment). As such, the robot is fully compliant with the proposed move strategy. Robots reach their destination in both cases with one obstruction and many obstacles with nearly stepless adjustment direction angle θ which is much higher definition. So, our method is more flexible and smoother than three others recent works as comparisons shown in Table 1.

Table 1. Comparisons of navigation performance for indoor robot

Method (Authors)	Quantum Step (⁰)	Quantum Error (⁰)	Range of Direction (⁰)	No of Movement /Directions
Correa [3] (2012)	22.5	11.25	180	8/3
Zainuddin [7] (2014)	90	45	180	4/3
XIN Jing [12] (2016)	22.5	11.25	360	16/7
AMDS (Our method)	1	0.5	180	2/>180



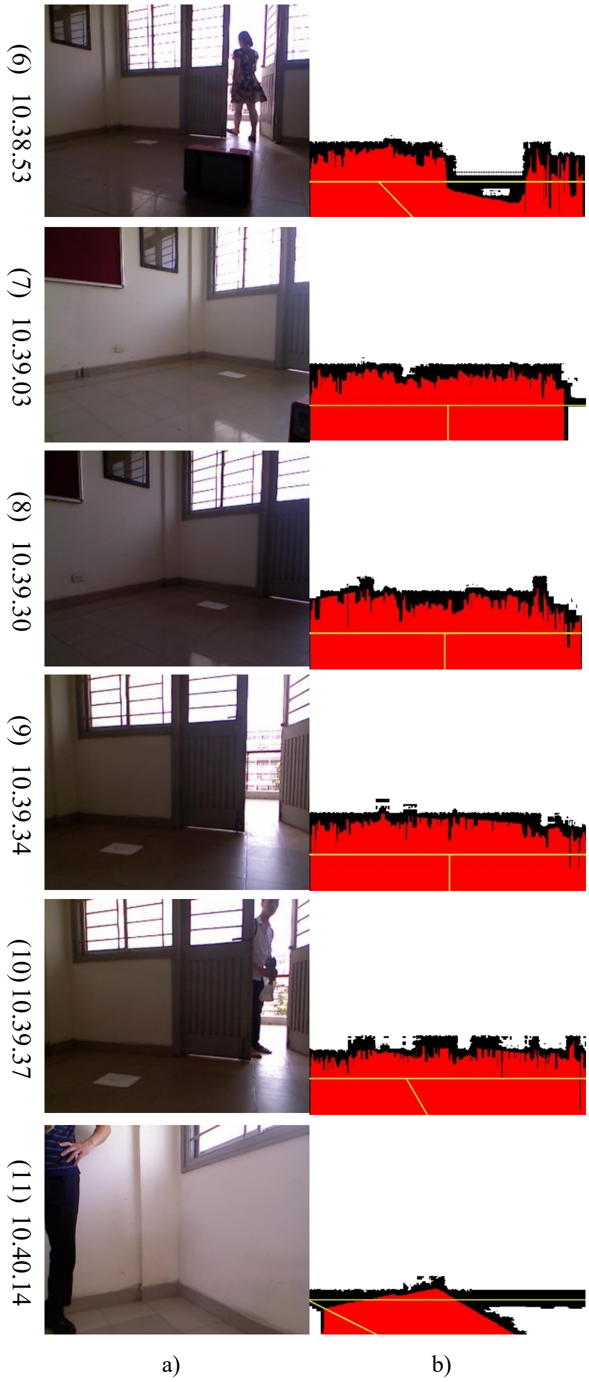
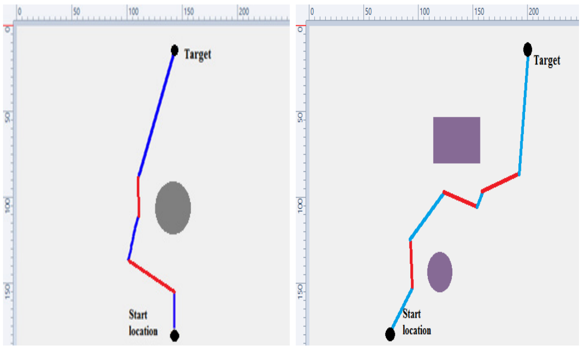


Fig. 7. Strategy navigation for indoor robot using a Kinect sensor. a) The RGB frames; b) The ground plane maps with moving direction of the robot

The success rate of success was compared with other methods (Table 1). The result has verified the efficacy of the algorithm for robot indoor autonomous navigation. The robot reached its destination with a success rate of 98.7%, however the timings of each test were not the same, absolutely.



(a) (b)

Fig. 8. Trace of robot location during navigation with (a) simple script and (b) complicated script

Table 2. Success navigation rate

Method (Authors)	No of tests	Success Navigation Rate (%)
(Correa, et al) [3]	-	92
FSPF [5]	-	98
Our method	75	98.7

5. Conclusions

In this paper, a new method to improve the navigation performance of indoor robot is proposed. An indoor robot system equipped with a Kinect sensor has been executed well with the proposed navigation method in a low light site. This manuscript shows a various of tested results which demonstrate the robust proposed method. The navigation project based on mining real-time depth video has been improved with a higher performance certainly by comparing of successful rate parameters and capacity of direction angles among interesting methods. In the future, it necessary to use a high-resolution depth camera for analysis of geometry of the object observation. It also should mine data further by combining depth data stream with color data stream to deliver more refined and humorous results.

References

[1] Zeping Fang, Jianmin Duan, Obstacle detection for intelligent vehicle based on LabVIEW and laser measurement system, Information and Automation (ICIA), 2013 IEEE International Conference (2013) 202-206.

[2] Ter Feng Wu, Pu Sheng Tsai, Nien Tsu Hu, Jen Yang Chen, Use of Ultrasonic Sensors to Enable Wheeled Mobile Robots to Avoid Obstacles, Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference (2014) 958-961.

[3] Correa, Diogo Santos Ortiz, et al, Mobile robots navigation in indoor environments using kinect sensor, Critical Embedded Systems (CBSEC), Second Brazilian Conference on IEEE (2012) 36-41.

[4] Z. Yong-guo, C. Wei and L. Guang-liang, The Navigation of Mobile Robot Based on Stereo Vision, Fifth International Conference on Intelligent Computation Technology and Automation, Zhangjiajie, Hunan (2012) 670-673.

[5] Biswas, Joydeep, and Manuela Veloso, Depth camera based indoor mobile robot localization and navigation, Robotics and Automation (ICRA), IEEE International Conference on. IEEE (2012) 1697-1702.

[6] László Somlyai, Mobile robot localization using RGB-D camera, ICCC 2013, IEEE 9th International Conference on Computational Cybernetics, Hungary (2013) 131-136.

[7] N.A. Zainuddin, Y.M. Mustafah, Y.A.M. Shawgi, N.K.A.M. Rashid, Autonomous Navigation for Mobile Robot Using Kinect Sensor, 5th International Conference on Computer & Communication Engineering (2014) 28-31.

[8] Yiqun Fu, Guolai Jiang, Wei Feng, Yimin Zhou and Yongsheng Ou, On Real-Time Obstacle Avoidance Using 3-D Point Clouds, Proceedings of the 2014 IEEE, International Conference on Robotics and Biomimetics, Indonesia (2014) 631-636.

[9] Khalid N. Al-Mutib, Ebrahim A. Mattar, Mansour M. Alsulaiman, and H. Ramdane, Stereo Vision SLAM Based Indoor Autonomous Mobile Robot Navigation, Proceedings of the 2014 IEEE, International Conference on Robotics and Biomimetics, Indonesia (2014) 1584- 1589.

[10] Chun C. Lai, Kuo L. Su, Development of an intelligent mobile robot localization system using Kinect RGB-D mapping and neural network, Computers & Electrical Engineering, 67 (2018) 620-628.

[11] Gaoqiang Yang, Fucai Chen, Wen Chen, Mu Fang, Yun-Hui Liu, and Luyang Li, "A New Algorithm for Obstacle Segmentation in Dynamic Environments Using a RGB-D Sensor", Proceedings of The 2016 IEEE, International Conference on Real-time Computing and Robotics, Cambodia (2016) 374-378.

[12] XIN Jing, JIAO Xiao-liang, YANG Yin, LIU Ding, "Visual Navigation for Mobile Robot with Kinect Camera in Dynamic Environment," Proceedings of the 35th Chinese Control Conference, China (2016) 4757-4764.