An effective depth map navigation for mobile robot in indoor environments

Article *in* JP Journal of Heat and Mass Transfer · July 2018

DOI: 10.17654/HMSI218221

5 authors, including:

Dang khanh hoa
Hanoi Open University

T.V. Duc
Hanoi University of Science and Technology

Hoang Vu Minh
Hanoi University of Science and Technology

Tien Dzung Nguyen
Hanoi University of Science and Technology

JP Journal of Heat and Mass Transfer

http://www.pphmj.com
http://dx.doi.org/10.17654/
Volume …, Number …, 2018, Pages … ISSN: 0973-5763

AN EFFECTIVEDEPTH MAP NAVIGATION FOR MOBILE ROBOT IN INDOOR ENVIRONMENTS

Dang Khanh Hao, Than Viet Duc, Vu Minh Hoang, Vu Song Tung and Nguyen Tien Dzung*

School of Electronics and Telecommunications (SET)
Hanoi University or Science and Technology (HUST)
No. 01, Dai Co Viet Street, Hai Ba Trung District
Hanoi City, Vietnam

Abstract

This article presents a solution of navigation in an indoor environment for a mobile robot that only use a depth stream. In this work, a real-time navigation method is integrated in the self-built robotic system fitted with a Kinect sensor. First, the method applies a viable mobility principle that is a road vehicle only be able to travel on the ground. So, this method starts with finding the ground from the depth video which is provided steady by Kinect. Based on shapes of this reliable ground plane, a robust moving direction selection algorithm is proposed. In the meantime, the robot system always follows the minimum path distance strategy to the given target. At the same time, with a predetermined map of operating environment, robot navigation problem was resolved with the same successful rate of recent methods such as Fast Sampling Plane Filtering (FSPF) method. The robotic system is navigated more robust and refined than the neural network

Received: January 16, 2018; Accepted: January 31, 2018
Keywords and phrases: depth map, ground plane, path direction, navigation.
A preliminary version of this paper was presented at the ICIDB-2018, Hanoi, Vietnam.
*Corresponding author
Communicated by Gyanendra Prasad Joshi

method with a 10-step adjustable angle. Experiments have shown that the stability and accuracy of the proposed navigation algorithm is up to 98%. The processing time of the navigation proposed method is minimized to meet the robot's real-time applications.

I. Introduction

In recent years, the problem for robot navigation in indoor environments is paid a lot special attention. Many researchers have tried to use a robust sensor that can collect rich information such as a camera. Many RGB-D image sensors have been marketed as Kinect by Microsoft or Bumblebee by FLIR Systems, Inc, ... etc. These products are so powerful that they improve the way people interact with the world around them, enhance public safety and well-being, increase energy efficiency, and enable healthy and entertained communities [1].

Three approaches have been presented that perform Simultaneous Localization and Mapping (SLAM) with a RGB-D camera system [2, 3, 4, 12]. These applied methods are optimized for only small workspaces such as small rooms but not for large environments. The best results of works [2, 3, 4] still have two disadvantages include modest precision and not satisfying the minimum processing time for continuous movement of the robot. The application of both the RANSAC algorithm and 3D mapping makes the processing speed decrease dramatically.

According to article [5], it has presented some encouraging results, but the authors were aware well that mapping implementations from RGB-D sensor are not real time. It is obvious that there is a need for a more plenty of hardware resource with a modern GPU. Then it still needs to improve the performance of the system implementation. The experienced results of [6] have satisfied the real-time condition but the camera system is forced to operate at a resolution lower than the VGA standard. Also, the system could only update the map from sensor at less than 6 times per second.

More recently, the authors of manuscript [7] have done quite a lot of work consists of making 3D reconstruction and recognition of visible terrain in front of the mobile robot for successive obstacle avoidance that bases on the pyramidal data structure and dynamic programming technique. Despite of these amazing results, the installation system has several shortcomings such as low camera's resolution with 128×128. This parameter is not quality enough to recognize all major obstacles on the ground plane. In addition, the blinded area within approximately 2m from the robot is too large in comparison with the size and velocity of the robot.

This robust trend is possible with the support of powerful hardware at an ever-lower cost. It's easy to build a high-performance system with a compact physical size. Operating system and software are also supported by more and more hardware and firmware resources with plenty of functional libraries. So many researchers are trying to put artificial intelligence into almost vehicle with the goal of helping these robots automate many intelligent behaviors. The proposed system in manuscript [8] uses a stability sensor named Kinect to collect the stream of depth video data that will be fed into an artificial neural network (ANN). This network recognizes different kinds of path in the environment comprising path ahead, left path, right path and intersections. However, the experienced result is not comprehensive because the navigation problem is only suitable for restrictive indoor environment. The innovation of [9] is only achieved to combine the RGB-D mapping and neural network training for achieving an Indoor Positioning System. The main contribution of [10] method is the combination between partial depth estimation and the particle filtering. Partial depth estimation only approximates the depth of the feature points without computing the depth of all the pixels in the image. Particle filtering ensures that approximation noisy are filtered out during the estimation of location of the robot. So, the experienced results are so amazing, but computation time should be reduced more unless size of the particle set is not flexible.

In this manuscript, the proposed method retains the advantages of the approaches [8, 9, 10] 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 describes in detail plane mathematics, ground plane based depth map, introducing operation model of the indoor robot and some useful calculation of size of pixel. In the third section, this work presents the architecture of the robot software system to perform navigation as well as applied algorithms. The fourth section is the obtained results and some important discussions to improve outcomes in different environments. The fifth section shows several conclusions and the needed future work.

II. Basic of Planes Mathematics and System Architecture

(a) Mathematical definition gradient of the plane

Mathematically, if a plane exists, it is a set of adjacent consecutive points and satisfies a represented plane equation defined as follows:

$$Ax + By + Cz + D = 0. \tag{1}$$

Obviously, (1) must satisfy the condition $A^2 + B^2 + C^2 > 0$. From (1), z is drawn on a side of the (1) and z becomes a function of two variables x, y is written as follows:

$$z = -\frac{A}{C}x - \frac{B}{C}y - \frac{D}{C}. \tag{2}$$

Taking partial derivatives of the function (2) with respect to x variable and y variable respectively:

$$\frac{\partial z}{\partial x} = -\frac{A}{C} \tag{3}$$

$$\frac{\partial z}{\partial y} = -\frac{B}{C}. \tag{4}$$

Thus, the gradient vector is defined:

$$\nabla z = \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}\right) = \left(-\frac{A}{C}, -\frac{B}{C}\right). \tag{4}$$

From (5), the depth gradient of a predetermined plane is constant along with both x axis and y axis directions. As such, the adjacent points have the same depth gradient values they belong to the same plane. This is a reliable characteristic for the object in the image is considered to be planar.

(b) Ground plane based on depth map

Capture devices with focal length f are placed at O with height h from the ground and its direction is parallel to the ground which is flat absolutely (Figure 1) [16]. Let O' be the perpendicular projection of O on the ground plane and O'' be the projection of O on the image. Let M_1 be a considered ground points and p denotes a distance from O'' to the projection of M_1 on the image. Assuming that OO'' is parallel to $O'M_1$, we have the distance z from O to M_1 :

$$z = h\sqrt{1 + \frac{f^2}{p^2}}. \tag{6}$$

Taking the differential both sides of formula (6), it leads to (7) as below:

$$dz = -hf \frac{1}{p^2\sqrt{p^2 + f^2}} d\rho. \tag{7}$$

From (7) and the actual figures of the camera we can approximate the differential Δz follow Δp.

Suppose there is one more point M_2 that is also located on the ground plane and M_1M_2 is perpendicular to the direction of the camera's view. Then the images of M_1M_2 is a horizontal segment. The length of that segment:

$$\Delta z = \sqrt{\Delta M^2 + z^2} - z. \tag{8}$$

From (8), the differential of horizontal direction Δz will be smaller than z. Now, assuming the depth images obtained from the camera horizontally x axis (direction from left to right), the vertical axis is y (direction from top to bottom); depth value is quantized and only get a finite value. Considering the obtained image, it is a digital image so from (8) we find that the depth difference of the ground along x axis will be 0 and the other only be kept within near zero.

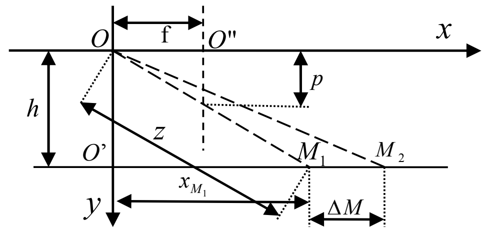


Figure 1. Principles of calculation the differential depth.

From (7) and going over a few simple changes, we can see that the differential depth along y axis is different from zero for the pixels which have a depth value greater than a certain value T. And it maybe gets zero if the point's depth is less than T.

So, candidates for the voting of the ground plane must meet the following conditions satisfy the following constraints of the ground plane:

- The number of pixels of the region must be greater than a predetermined threshold;

- $\frac{\partial z}{\partial x} = 0$ 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.

(c) Operation model of the indoor robot

Robots include components as described in Figure 2. Kinect sensor is mounted at the top of the system with height h_2. The Kinect's optical axis is turned down so that it can see the nearest point on the ground $G_{\min}$. Knowing the open angle of the Kinect is 53^0, it is easy to get the farthest ground point $G_{\max}$ at approximately 295cm from the camera position.

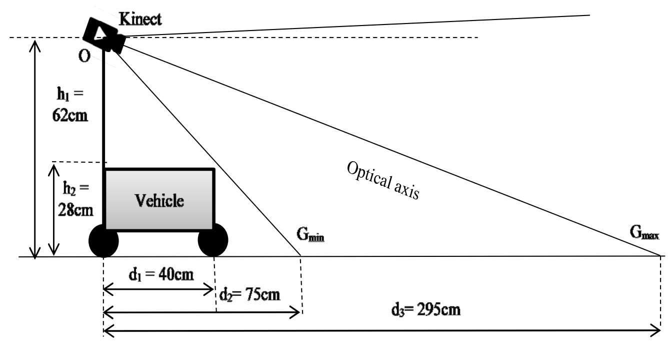


Figure 2. The model of robot using stereo vision based navigation system.

(d) Determining the 2D size of the depth point

Microsoft's Kinect provides distance from any point of view to the location of the camera. Then, with the constructed robot architecture in II.c, it could transform that point into a coordinate in the space that deals with distance, elevation, and viewing angles. Placed the camera at position *O*, the *Oz* axis coincides with the camera's optical axis, the *Oy* axis is directed from bottom to top, the *Ox* axis is directed from right to left, as illustrated in Figure 3.

At the minimum viewing distance of Kinect $OI = 0.8\,m,$ the horizontal field of this sensor is therefore approximately $AB = 87\,cm,$ and the vertical field is $BC = 63\,cm$ approximately, resulting in a resolution of just over $1.359 \times 1.312\,mm$ per pixel along with *x* axis and *y* axis, respectively.

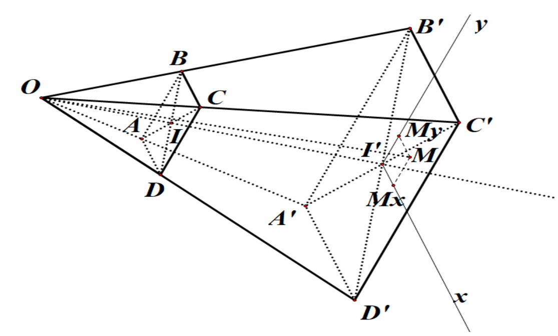


Figure 3. Mathematical model for calculating the size of the depth point.

To calculate the point size M along x axis with the specified depth, use the congruence of two $OI' M_x'$ and OIM_x triangles. After some transformations, the result of the size of the point on the x-axis M_x is depicted in formula (9).

$$M_x = \frac{I_x p_x}{1000}(mm). \tag{9}$$

Similarly, by using the congruence of two $OI' M_y'$ and OIM_y triangles, the result of the size of the point M on the y axis M_y is depicted in (10).

$$M_y = \frac{I_y p_y}{1000}(mm). \tag{10}$$

Finally, it could get the dimension of M in the 3D space $Oxyz$ includes (M_x, M_y, M_z).

III. The Depth Map Based Navigation System Implementation

The system includes three successive stages as shown in Figure 4. The first functional block is responsible forenhancing 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 neighbors 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.

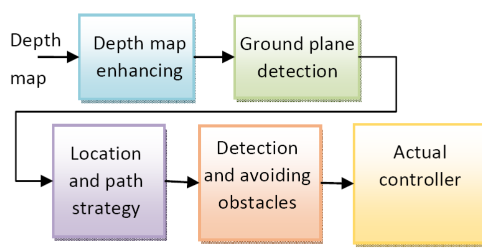
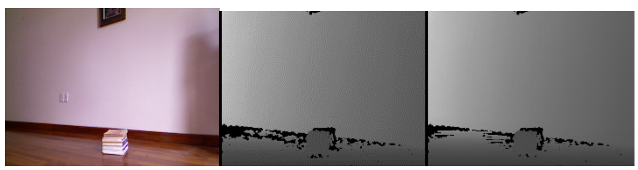


Figure 4. Block diagram of depth based navigation system.

(a) Depth map enhancing

To decrease this type of the noise, it is easy to see that if a review of the scope of a window *W* is small enough, they should always receive the correct values and the variability of depth value is not too strong. Thus, in the reviewed window, if it contains some points with unreasonable depth value then we can put them on the average value of the actual value of the points as shown in Figure 5. The result is the number of black points is greatly reduced. If the ratio between wrong value points and size of window *W* is more than 50%, the repair work is not effective because of lack of average value information. However, if this work reference experience from observing the actual depth data, this kind of error mostly occurs in regions which are far from the location of the camera. So, it has a less important role than the near zone from camera's position.



(a) (b) (c)

Figure 5. Illustration of captured data from Kinect and improving result. (a): Color Image; (b): Depth map; (c): Enhanced depth map.

(b) Ground plane detection

The block diagram of the proposed method is presented in Figure 6. The task of this stage is to create a map of depth difference, also called a gradient map from the depth map input performed by calculation of gradients in y and x directions, respectively. The resulting gradient depth map is further smoothed by consideration of depth difference of a point within a small enough window with size $w = 2 \times 2$, because of possible presence of noise in the input depth map.

Then, the second stage groups the adjacent pixels that have a similar gradient into a range. The candidate ground plane is then formed by the ranges that meet the ground hypotheses.

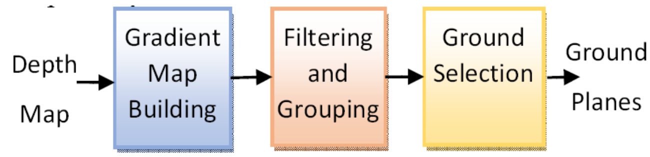


Figure 6. A block diagram of the GDM Ground Plane Calculation.

In order to extract more exact and smooth ground plane, the third correction stage starts dividing the initial difference depth map into square blocks of size B and then estimates the ratio R between the ground pixels inside each block and the block size. This is an important parameter used to classify the blocks into ground or non-ground ones and then generate the final map which includes ground and non-ground regions. If R is greater than a given threshold θ, then the block is considered as ground, and vice versa. In order to evaluate the value of θ, the smallest rectangular bounding the detected ground regions is determined, and the ratio between the number of all ground pixels *Pground_of_ranges* over the square of the rectangular *Prec* as depicted in equation (11).

$$\theta = \frac{\sum p_{ground\_of\_ranges}}{P_{rec}}. \tag{11}$$

Obviously, the non-ground areas which belong to obstacles appearing with a large size enough would be detected.

(c) 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.

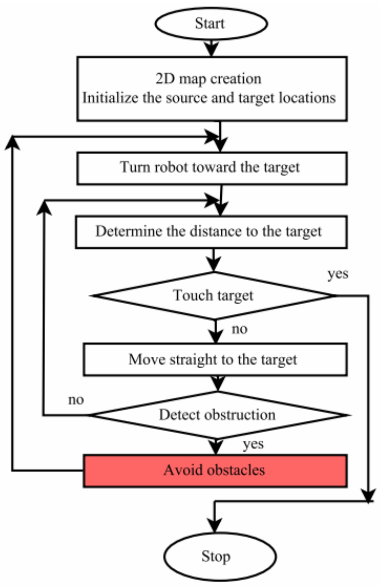


Figure 7. Strategy of depth based navigation for indoor mobile robot.

The navigation strategy consists of two tactical components that are target moving tactic and obstacle avoiding tactic (Figure 7). The principle of target moving tactic always strives to 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.

(d) Detection and avoiding obstacles

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

1. *The obstacles in front*: 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.

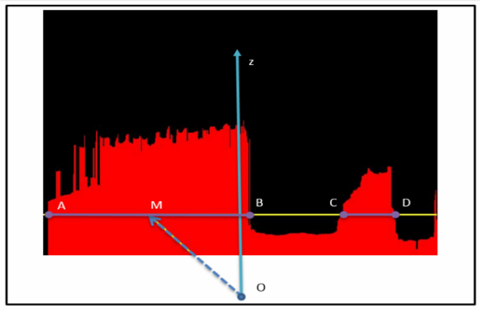


Figure 8. Mathematical model of the directional selection.

2. *The obstacles in the left*: 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. 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 Figure 8.

3. *The obstacles in the right*: The scenario is repeated like that where the obstacles appearing in the left in the opposite manner, however.

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.

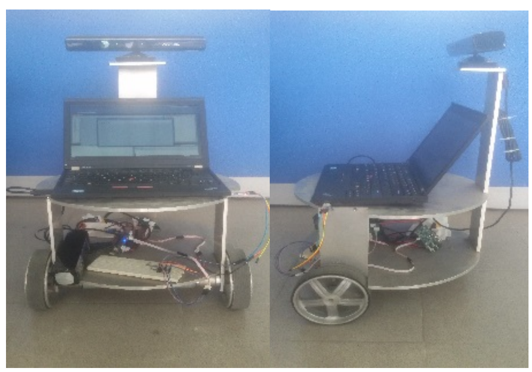
IV. Experiences Results and Discussions

The program is written in C# programming language in Visual Studio 2015 environment, and implemented on a laptop with the configuration as follow: Core i5-2520 processor, maximum clocked 2.5Ghz, Windows 7

Ultimate 64-bit SP1 as shown in Figure 9. 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.

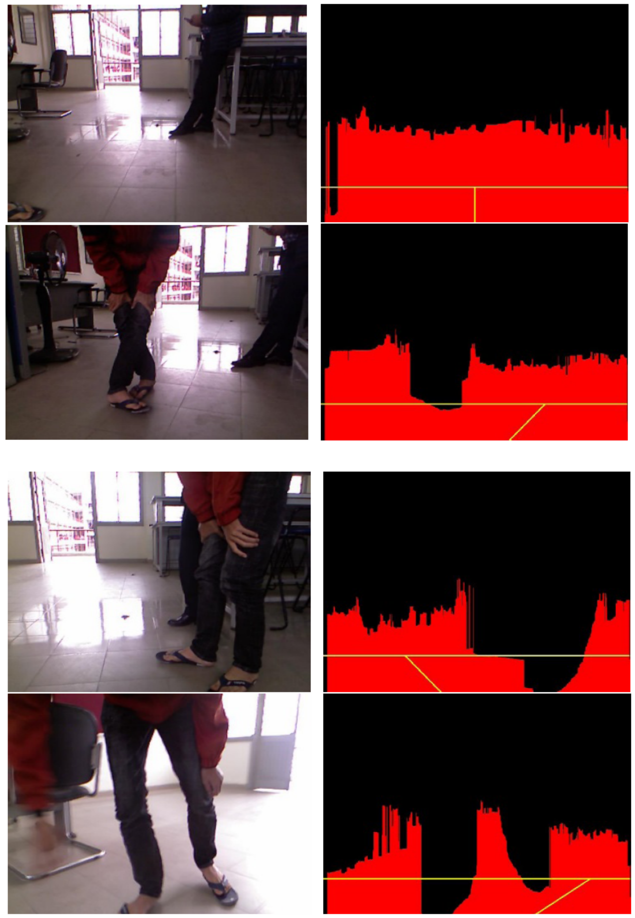
In this section, the authors describe the experimental results using the proposed method. The experiment is conducted in a real environment where the scenes is set up with different obstacles for testing the stability of the proposed method as depicted in Figure 10. It also shows the movement log of the robot controlled by the proposed algorithm installed on the platform in Figure 9. As one can see, the robot can avoid successfully the obstruction appearing in its moving path from the original to the destination.

The test was conducted nearly 30 times with a random static obstacle arrangement. In addition, there are the appearance of moving objects such as the phenomenon of a person entering the robot's view. When the robot moves on the floor of test room, both depth based navigation video and color video are saved. They are useful to evaluate the quality of the test.



(a) (b)

Figure 9. Robotic system. (a) Front view. (b) Side view.



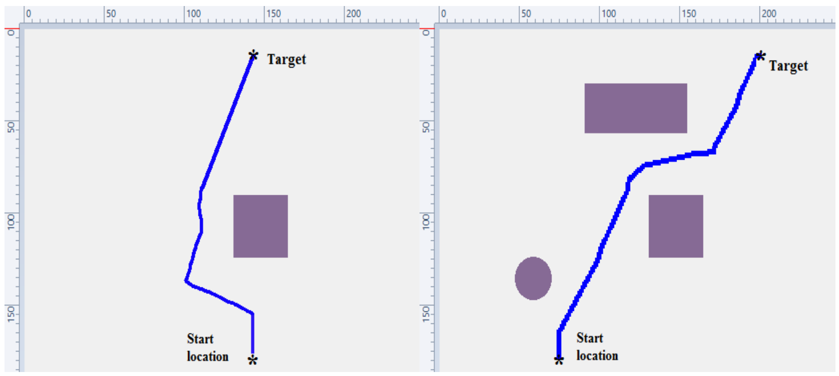
(a) (b)

Figure 10. Tested frames captured by the Microsoft's kinect sensor. (a) The original frames; (b) The extracted ground planes maps including the determined moving direction of the robot: First row: moving straight; second: moving in the right; third row: moving in the left; last row with multiple obstacles: moving to the right.

Table I. Success Navigation rate.

Method (Authors)	No of tests	Success Navigation Rate (%)
(Correa, et al)[8]	**-**	**92**
FSPF [11]	**-**	**98**
Our method	**27**	**98**

In addition, the success rate of success was compared with other methods (Table I). The result has verified the efficacy of the algorithm for robot indoor autonomous navigation. Figure 11 illustrates the results of tracking the robot location with two kinds of scenario consisting of one obstruction and multiple obstructions. The robot operates in an area of $10\text{m} \times 8\text{m}$. The distance between the starting point and the target is about 9m. The robot reached its destination with a success rate of 98%, however the timings of each test were not the same, absolutely.



(a) (b)

Figure 11. Trace of robot location. (a): Navigation with a simple script. (b) Navigation with a complicated script.

V. Conclusions

In this study, we focus on developing the indoor navigation system for mobile robot applications using a depth camera. It has addressed the problem of robot navigation in an unknown and dynamic environment. Kinect sensor is well-suited for robust localization and reliable obstacle avoidance in complex indoor environments. All the experimental results suggest that the localization and navigation are very efficient for indoor robot with the proposed method.

This manuscript shows various tested results which demonstrate the robust proposed method. By comparing of three common parameters among interesting methods, the applied algorithm illustrates a high performance certainly. In future work, this project should be improved by using real-time depth video. 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] FLIR Systems, Inc, https://www.ptgrey.com/corporate/about-us.

[2] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers and Burgard, An evaluation of the RGB-D SLAM system, In Robotics and Automation (ICRA), 2012 IEEE International Conference on, 1691-1696.

[3] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox and N. Roy, Visual odometry and mapping for autonomous flight using an RGB-D camera, In Int. Symp. of Robotics Research (ISRR), 2011.

[4] R. A. Newcombe, S. J. Lovegrove and A. J. Davison, DTAM: Dense tracking and mapping in real-time, In IEEE Int. Conf. on Computer Vision (ICCV), 2011.

[5] Peter Henry et al., RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments, Experimental robotics, Springer Berlin Heidelberg, (2014), 477-491.

[6] Daniel Maier, Armin Hornung and Maren Bennewitz, Real-time navigation in 3D environments based on depth camera data, 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), IEEE (2012), 692-697.

[7] 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.

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

[9] Chun C. Lai and Kuo L. Su, Development of an intelligent mobile robot localization system using Kinect RGB-D mapping and neural network, Computers & Electrical Engineering 2016.

[10] Selvaraj Prabu and Guoqiang Hu, Stereo Vision based Localization of a Robot using Partial Depth Estimation and Particle Filter, IFAC Proceedings 47(3)

(2014), 7272-7277.

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

[12] Khalid N. Al-Mutib, Ebrahim A. Mattar, Mansour M. Alsulaiman and H. Ramdane, Stereo Vision SLAM Based Indoor Autonomous Mobile Robot Navigation, International Conference on Robotics and Biomimetics, December 5-10, (2014), Bali, Indonesia, pp. 1584-1589.