Position Rectification with Depth Camera to Improve Odometry-Based Localization

Lan Anh Trinh
Electronics Engineering Dept.
Posts and Telecommunications Institute of Technology
Ho Chi Minh, Vietnam
lananhtrinh@ptithcm.edu.vn

Nguyen Duc Thang
Biomedical Engineering Dept.
International University, Vietnam National University-Ho Chi Minh, Vietnam
ndthang@hcmiu.edu.vn

Nguyen Vu Duy Hau
Biomedical Engineering Dept.
International University, Vietnam National University-Ho Chi Minh, Vietnam
nvdhau@gmail.com

Tran Cong Hung
Science Technology Dept.
Posts and Telecommunications Institute of Technology
Ho Chi Minh, Vietnam
conghung@ptithcm.edu.vn

***Abstract*— Indoor localization plays an important role for many applications especially robotics where the location of robot is necessary for the tasks of tracking and controlling. Among efforts proposed to address this problem, odometry-based localization is presented as an effective method with simple installation. This approach is based on the movement information of the robot wheels that are obtained by motion sensors to estimate position changes of robots. However, the errors of estimating the poses of robot by odometry are accumulated and increases overtime due to the wheel skidding, sensor errors, or inaccurate information of wheelchair configuration. This paper presents an approach to overcome these drawbacks through the tuning process using a depth camera. A randomized forest is trained to map depth images captured by a depth camera to labeled locations. As a sequence, this information is used to refine the results of odometry-based localization. Experiments are conducted with a real electronic wheelchair to show that the proposed approach helps increase the accuracy of the conventional odometry-based methods. Therefore, the approach is applicable for many applications related to indoor localization.**

Keywords—Indoor localization; Depth camera; Randomized forest; Odometry-based localization

I. INTRODUCTION

Localization plays a critical role in many fields of daily life such as medical, military, aerospace, and robotics. For outdoor environments, the global position system (GPS) based on satellites is the common approach for applications with the vast range of terrain [1]. However, the navigation system with GPS is not suitable for indoor environment where the signals from the satellites are weak to reach an object inside a building. Therefore, many approaches have been proposed to replace GPS to address indoor localization.

The indoor localization can be done by means of radio signals when the wireless network with many available access points (APs) [2]. It has been shown that the energy attenuation received from the antenna of a device is proportional to the square of the distance from the device to the corresponding AP. As the results, by evaluating the signals obtained at the antenna, the distances from it to several APs are estimated, in other words distance of arrivals (DOAs) are available. From at least three available DOAs, the location of a device with respect to APs as the landmarks is found. However, the localization estimated by this method is inaccurate due to the rapid attenuation of the radio signals at far distances. Therefore, another kind of methods is based on ultrasound wave to estimate a distance for DOA localization. Although this method is able to achieve the high accuracy of localization, the ultrasound equipment is extensively expensive. Therefore, it has been applied mainly in medical areas to determine the position of probes or sensors within a narrow 3D space. Similarly, optical sensors and sonar sensors are limited by short distance determination for the indoor localization [3].

Currently, odometry-based localization [4,5] is presented as an effective method for a wide range of applications in robotics. It is based on the movement data from the wheels of robot which are obtained by motion sensors to estimate position changes of robots over time. Although the odometry-based localization outperforms conventional systems based on radio signals regarding accuracy, it poses drawbacks of increasing errors. The differences between the actual and the estimated pose of robot found by odometry tend to be increased over time due to wheel slips, inaccurate information about the distance between two wheels, unequal radii of two wheels, etc. By such, after an amount of time, tuning process is needed to reinitialize the position of the robot that is a complicated task with a manual input from a user.

In order to address the location rectification for odometry-based localization, we have attempted addition solution with visual based localization using a depth camera. A depth camera has been developed to provide a depth image to let us know the distance from the interested points to the camera. Some depth cameras utilize light reflection from the objects to reveal depths, yet this camera is complicated in implementation. Recently, with current development of a depth camera using a structured-dot pattern such as Kinect released by Microsoft, the depth camera can provide a good quality of a depth image with the depth resolution of one centimeter and the observation range within ten meters with reasonable price.

In this work a new algorithm for indoor localization system is proposed with the uses of odometry with two encoders and with a depth camera to refine the estimated position. Those two sensors have affordable price and acceptable accuracy so that they can be accepted for many

978-1-4673-6547-5/15/$31.00 ©2015 IEEE

indoor applications. The whole system is mounted on an electronic wheelchair and tested with real environment.

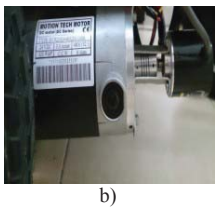
The rest of paper is organized as follows. Section 2 describes our devices and methods. We present experimental results with discussion in Section 3. Finally, we conclude the paper with Section 4.

II. DEVICES AND METHODS

A. Devices

The wheelchair that is utilized in this work is an electric powered wheelchair with two motors as depicted in Fig. 1(a) manufactured by Motion Technology Electric & Machinery Co. LTD, Taiwan [6]. The wheelchair's length, width and height are 1.10*m*, 0.6*m,* and 1*m* respectively. The wheel radius is 0.17*m* and the distance between two wheels is 0.52*m*. Two rotary encoders are installed for the self-localization of the wheelchair. Each rotary encoder OMRON E6B2-CWZ6C [7] is fixed onto a motor of wheelchair as illustrated in Fig. 1(b). How two rotary encoders help localize the wheelchair will be discussed later in the following section.

a)



b)

c)

Fig. 1. The configuration of a) the wheelchair, b) encoders, and c) depth camera.

To provide the depth information, a Kinect camera is applied. The Kinect was first introduced by Microsoft in 2010 as a new controller for Microsoft Xbox 360 video game to capture gamers' motion without requiring markers attached on their bodies. The two valuable sensors of Kinect are the basic RGB camera and the depth camera which contains an infrared (IR) emitter and an IR receiver. The camera is well operated to capture depths from 0.8*m* to 3.5*m*. Besides, it can provide two images with full VGA resolution of 640×480 pixels at a recording rate of 30 frames per second, the former for 24 bit RGB images and the later for 11 bit depth image with 2048 levels of depths [8].

Because of its low cost and capability to produce depth image, it becomes the potential sensor for wide applications. Therefore besides the official Window SDK provided by Microsoft to support application development for Kinect, additional open-source drivers as well as processing tools of third party such as OpenNI or Libfreenect were created to make Kinect application easy to be implemented and available to cross platforms. OpenNI is the most used for Kinect development among these libraries so it is utilized in this work. After a depth image is captured, the coordinates of a point in 3-D is estimated from the pixel indices as follows,

$$X = \frac{(u - u_0)Z}{f} \qquad (1)$$

$$Y = \frac{(v - v_0)Z}{f}$$

where u and v are the row and column index of a pixel, u_0, v_0, and f are the parameters configured by a depth camera.

Finally, in order to gather all information from Kinect, encoders and to generate control commands for navigating the wheelchair, the center controller is designed with a software program that runs on a Mac Mini with an Intel Core i5 2.3 GHz, 2GB RAM, Windows 7 32bit OS.

B. Kinemact Model of Wheelchair for Odometry-based Localization

For odometry-based localization the kinematic model of the wheelchair is established as shown in Fig. 2 below. Two big filled rectangles present the two wheels driven by motor and two small blank rectangles present the two front castor wheels, L is the distance between two driven wheels, R is the wheel radius, O_w is the center of the axis of the driven wheels and the origin of wheelchair coordinates frame $\{O_w, X_w, Y_w\}$. The configuration of the wheelchair is represented on the world space coordinates frame $\{O,X,Y\}$ by a set of three parameters $\{x, y, \theta\}$ also known as the pose where (x, y) is the position of O_w with respect to the world frame $\{O,X,Y\}$. Finally, θ is the angle between the wheelchair orientation vector $O_w X_w$ and the world frame axis OX.

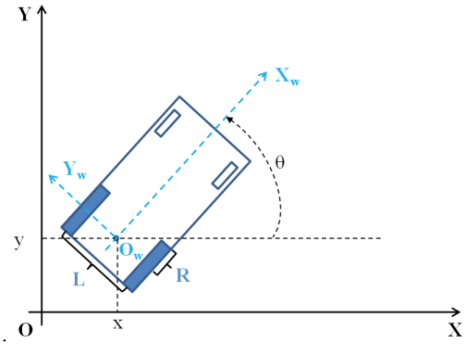


Fig. 2. The kinematic model of the wheelchair in 2D.

In fact, the wheelchair can be considered as a differential wheeled robot because its movement is based on two separately motor driven wheels. The direction of the

wheelchair can be adjusted by varying the relative rate of rotation of its motors. So the kinematic model representing the movement constraints of the wheelchair is the same as the kinematic model of a differential wheeled robot as described in [9,10]. According to this model, the wheelchair motion can be tracked if we are able to determine the distance travelled by each driven wheel, the wheel radius – R and the distance between two driven wheels – L. Let d_l be the distance travelled by the left driven wheel in an interval d_t, d_r be the distance travelled by the right driven wheel in the interval d_t. Correspondingly, the distance travelled – d_Δ and the change in orientation $-\theta_\Delta$ of the wheelchair in the interval d_t are approximated as

$$d_\Delta = \frac{d_r + d_l}{2} \qquad (2)$$
$$\theta_\Delta = \frac{d_r - d_l}{L} .$$

With these approximations, the wheelchair's pose at time t $\{x_t, y_t, \theta_t\}$ can be updated according to

$$x_t = x_{t-d_t} + d_\Delta \cos(\theta_{t-d_t}) \qquad (3)$$
$$y_t = y_{t-d_t} + d_\Delta \sin(\theta_{t-d_t})$$
$$\theta_t = \theta_{t-d_t} + \theta_\Delta ,$$

where $\{x_{t-dt}, y_{t-dt}, \theta_{t-dt}\}$ is the previous wheelchair's pose.

C. Implementation of Odometry-based Localization

The encoder is a device that translates rotation of motor shaft or wheel into electrical signals which contain information such as speed, direction, angular position of the motor shaft. There are two types of encoder: absolute encoder and incremental encoder. The absolute encoder provides angular position of the motor shaft directly. On the contrary, the incremental encoder requires additional processing step to determine the angular position of the motor shaft. Regarding the price and installation, the incremental encoder is chosen.

The working principle of incremental encoder is based on optical measurement. A light beam from a light source to photo detector is interrupted by the optical grid when rotor disc rotates with the motor shaft. This result is shown in several sine wave signals which are converted into a discrete square wave signals using a threshold value defined by the light and dark states. The encoders used in this work have three channel outputs: channel A, channel B, and channel I corresponding to three photo detectors blue, green, and purple. The photo detector of channel A and B are placed in the way that the signal outputs of two channels are 90 degree out of phase. With the special property of channel A and channel B, the direction of motor shaft can be detected. The channel I produces a reference pulse when encoder disc finishes a revolution. Resolution of optical encoder is defined by number of optical grid of channel A or B and measured in counts per revolution (CPR). CPR is an important parameter used to calculate the minimum angular resolution or traveled distance of the driven wheels such as d_r and d_l mentioned in the above section.

Note that the microcontroller is chosen in such a way that it is able to meet the objectives of the controller hardware. There are two encoders so the microcontroller needs two external interrupt pins for counting pulses at the two channels A and two input pins for checking logic level at the two channels B to detect rotation direction. The original controller of wheelchair uses two analog signals so the microcontroller needs two pulse width modulation (PWM) channels which can be converted to analog signals using analog low-pass filter. With specific requirements, the *ATmega8* microcontroller is chosen.

Our program on microcontroller *ATmega8* has three main parts which are getting data from two encoders, creating and controlling two PWM signals, communicating with computer for sending encoders data, and receiving control command to adjust PWM signals. The program is designed for fast processing because several interrupts and timers are utilized at the same time: two external interrupts for encoders, *Timer0* timeout interrupt for sending encoders data to a computer, *Timer1* for creating two PWM signals. All of these interrupt tasks have highly priority while receiving control command task has the lowest one.

The wheelchair has two DC motors with maximum speed of 4700*rpm* or 78.33 rounds per second, ratio 32:1 (for thirty two rounds of the motors, the corresponding wheel finish a round), and the wheel radius is 0.17*m*. With this information and CPR, the maximum pulses per second generated by encoder, the maximum external interrupts and the distance resolution which indicate travelled distance of wheel per pulse are computed. All information is calculated by the following equations.

$$MaxGeneratedPulse = MaxMotorSpeed * CPR \qquad (4)$$
$$MaxExternalInterrupts = MaxGeneratedPulse * 2$$
$$DistanceResolution = \frac{2\pi R}{32 * CPR} .$$

As the distance resolution is 0.33*mm* per pulse, the encoder resolution of 100 CPR is chosen.

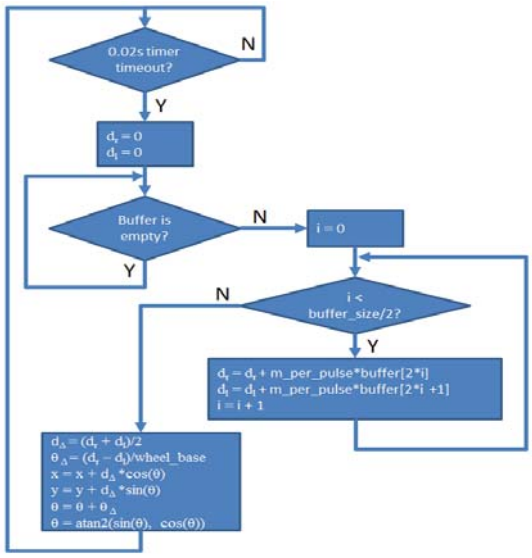


Fig. 3. The flowchart of odometry-based localization.

The motion sensors are the rotary encoders mounted on two driven wheels. The encoder data is received from controller hardware and estimates the position of wheelchair

based on the kinematic model described in Section II.B above. Once the encoder data are received, they are stored in buffer for further processing. The encoder data frame is two bytes which store the pulse count of the right and left wheels. Odometry function is executed periodically for every 0.02*s*. The flowchart of the odometry-based localization algorithm is overviewed in Fig.3 in which the wheel data from motion sensors is used to estimate the change in position of wheels object over time with the corresponding parameters: *m_per_pulse* is 0.33*mm*; *wheel_base* is 0.52*m* (distance between two driven wheels); d_r and d_l are the travelling distance of the right and left wheel in duration 0.02*s* and *x, y,* θ are the position and orientation of wheelchair. Note that *atan2* function is called to ensure that θ is normalized in range $[-\pi, \pi]$.

D. Position Tuning with Depth Camera and Randomized Forest

As aforementioned, the differences between the actual and estimated position of a robot are accumulated that leads to increasing errors of the odometry-based localization. To overcome such drawbacks, extra information from visual system can be used to refine the position of the wheelchair after estimated by odometry. In this work, a depth camera is investigated. A set of depth and RGB images are captured by a depth camera and matched to some labeled locations that are annotated with labels. To create the training data, the wheelchair is moved around the building while the depth camera mounted on the wheelchair continuously records the depth and RGB data. After that, a set of RGB and depth images are labeled with different areas of the building to create training data. To be suitable with the huge image features, a randomized forest is applied to learn the training data to match the images with the labeled areas. Since the coordinates of each area are known in advance, the coordinate x_t of the wheelchair at the time index t is updated by equation,

$$x_t = \alpha x_{t-1} + \beta x_t^{\text{odometry}} + \gamma x_t^{\text{camera}} \quad (5)$$

where x_t^{odometry} is the coordinate computed from odometry data and x_t^{camera} is the coordinate along the x-axis of the area found by randomized forest for the RGB and depth image captured at the time index t by a depth camera. With some areas where x_t^{camera} is available, then we have $\alpha + \beta + \gamma = 1$. Otherwise, $\alpha + \beta = 1$ and $\gamma = 0$. The coordinate y_t is calculated by the same procedure.

Basically, a randomized forest is constructed by several decision trees and each tree is trained randomly from a part of given training data. This leads to the independence of individual trees, and in turn to improve generalization. Therefore the randomized forest model helps achieve high robustness with respect to noisy data.

The decision tree as described in Fig. 4 is made by a set of nodes in a hierarchical fashion for making decisions. Each split node stores a test function to be applied to classify the incoming data and each leaf stores the final answer (predictor). The functioning of a decision tree can be separated into two processes: training and testing. Training a decision tree includes sending all training data [11] into the tree and then optimizing the parameters of the split nodes to optimize an energy function. On the other hand, testing the internal node is applied a test to the input data and sent it to the appropriate child. The process is repeated until a leaf node is reached. In this paper, the randomized forest is used for the task of classification.

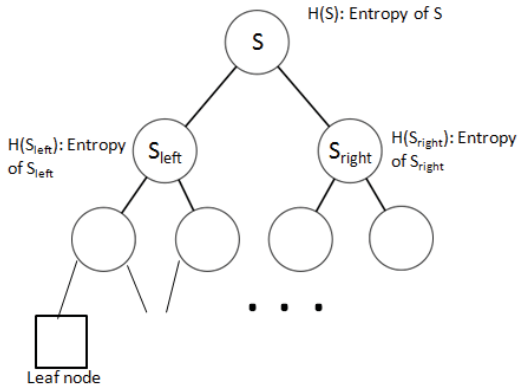


Fig. 4. A decision tree.

Besides, the energy function is defined by the entropy and information gain (IG). The entropy represents the impurity of a node with respect to the distribution of data classified to this node. The higher entropy is, the less certainty of the distribution data of the node is. Alternatively, IG measures the expected reduction in the entropy or uncertainty. Mathematically, those two measures are equivalent as expressed by

$$H(X) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \quad (6)$$

$$IG = H(S) - \sum_{i\in\{\text{left,right}\}} \frac{|S_i|}{|S|} H(S_i)$$

where $H(X)$ is the entropy of node X, $p(x_i)$ is the probability of the label x_i, n the number of classes or labels, $|S|$ is the number of elements classified to a node. Here, node S has two child nodes that are the left node S_{left} and right node S_{right}.

Each split node in the tree is associated with a binary split function ϕ. The data at the split node is sent to its left or right child dependent on the output of ϕ (0 indicates "left" and 1 indicating "right"). Actually, the split function is a weak classifier that is described by some parameters that are chosen to maximize the IG of the data when passed through this node.

After a tree is trained, the useful information for prediction will be stored in leaf nodes. In particular, each leaf node contains the distribution of classes $p(c|v)$ associated to the part of training data that has reached to that leaf. During testing, each test data v is simultaneously pushed through T trained trees until it reaches the leaf of each tree. The final output is ensemble by averaging,

$$p(c|v) = \frac{1}{T}\sum_{i=1}^{T} p(c_i|v). \quad (7)$$

The effects of randomness, the forest size T, maximum allowed tree depth D, choices of the weak learner model, features selection, etc. are discussed further in [11].

III. EXPERIMENTAL SETUP AND RESULTS

The experiments are conducted on the fourth floor of the International University, Ho Chi Minh, Vietnam. The map and the routes of the moving wheelchair for testing are illustrated in Fig. 5. The trajectory of the wheelchair estimated by odometry-based algorithm is shown in Fig. 8. Obviously, there are significant differences between the estimated trajectory and the ground-truth. The errors become larger when the wheelchair is turned left or right at each corner of the building.

To perform the position tuning with the depth camera, a set of 1433 depth and RGB images with annotated labels of areas are used to train the randomized forest. An example of a pair of RGB and depth images is given in Fig. 6. The RGB image is converted into grayscale. For each grayscale or depth image, the histogram is estimated. The number of bins in a histogram is 100, therefore the data of each video frame is reduced to 200 features (100 for the grayscale image and the later for the depth image). There are eight labels corresponding to eight regions as depicted in Fig. 7. Regarding x_t^{camera} and y_t^{camera} expressed in equation (5), the center coordinates x_1, x_2, y_1, and y_2 represent the coordinate for each area. For areas 2, 4, 5, and 6, both x_t^{camera} and y_t^{camera} are available, meanwhile for the rest, only either x_t^{camera} or y_t^{camera} is able to be used.

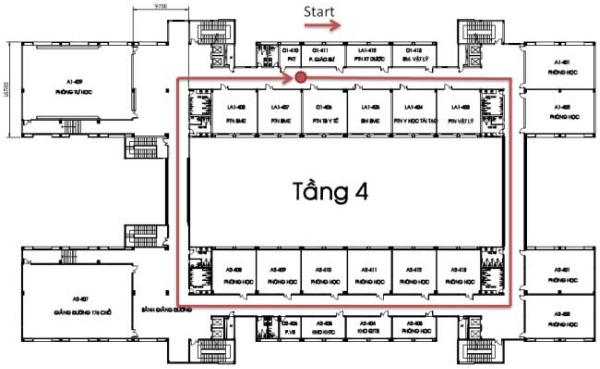


Fig. 5. The map and the routes of the moving wheelchair.

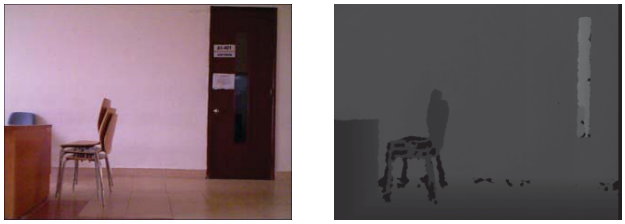


Fig. 6. A sample of RGB and depth images.

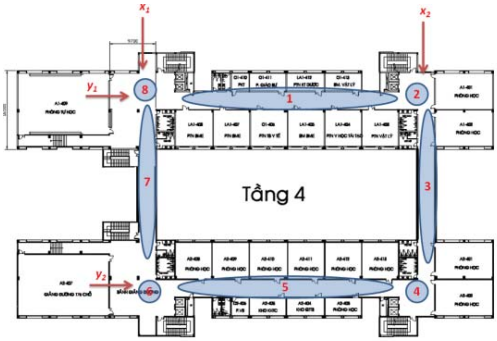


Fig. 7. Different labels of various areas in the building.

To evaluate the accuracy of the proposed approach, the root mean squared error (RMSE) $RMSE = \frac{\sum_{i=1}^{n}\sqrt{(x_i^{est}-x_i^{orig})^2+(y_i^{est}-y_i^{orig})^2}}{n}$ is computed where i is the frame index, (x_i^{est}, y_i^{est}) and (x_i^{orig}, y_i^{orig}) are the cordinates of the estimated and original location of the wheelchair respectively. As shown in Table I, the odometry-based localization combined with rectification with a depth camera significantly reduce the RMSE. Fig. 8 shows the trajectory of the wheelchair tracked by the proposed odometry-based localization combined with rectification (the number of tree is 6). The wheelchair is better tracked with the new approach.

TABLE I. COMPARISONS OF RMSE ESTIMATED BY ODOMETRY-BASED LOCALIZATION BETWEEN WITH AND WITHOUT RECTIFICATION

Number of trees in the randomized forest	RMSE (m) with rectification	RMSE (m) without rectification
1	10.4	26.4
2	10	
3	9.4	
4	9.1	
5	9.1	
6	8.9	

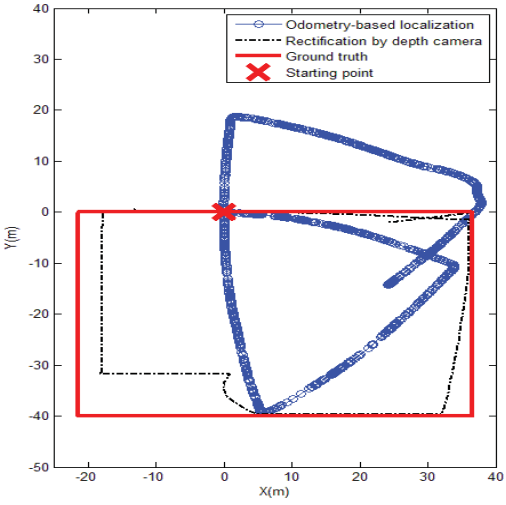


Fig. 8. The trajectories of the wheelchair tracked by the proposed odometry-based localization with rectification.

IV. CONCLUSIONS

This paper presents an approach to improve the odometry-based localization using tuning process with a depth camera. A set of depth and RGB images are used to train the randomized forest to match them with several labeled areas. As the results, this matching is further utilized to reduce the localization errors caused by odometry methods. We have performed experiments with real data to show the effectiveness of the proposed approach. Our future work will focus on collecting more training data to detect more labels of a building to better refine the localization outcomes.

ACKNOWLEDGMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.05-2013.11 and also funded by Posts and Telecommunications Institute of Technology Ho Chi Minh, Vietnam.

REFERENCES

[1] C. Gao, M. Sands, and J. R. Spletzer, "Towards autonomous wheelchair systems in urban environments," in *Field and Service Robotics*, 2010, pp. 13-23.

[2] S. Teller, J. Battat, B. Charrow, D. Curtis, R. Ryan, J. Ledlie*, et al.*, "Organic indoor location discovery," *Computer Science and Artificial Intelligence Laboratory Technical Report,* vol. 75, p. 16, 2008.

[3] T. Mohammad, "Using ultrasonic and infrared sensors for distance measurement," *World Academy of Science, Engineering and Technology,* vol. 51, pp. 293-299, 2009.

[4] M. Oishi, A. Cheng, P. Bibalan, and I. Mitchell, "Building a smart wheelchair on a flexible software platform," in *RESNA International Conference on Technology and Aging*, 2011.

[5] R. Simpson, E. LoPresti, S. Hayashi, I. Nourbakhsh, and D. Miller, "The smart wheelchair component system," *Journal of Rehabilitation Research and Development,* vol. 41, pp. 429-442, 2004.

[6] http://www.motiontech.com.tw/.

[7] http://www.ia.omron.com/product/item/e6b27007h/.

[8] L. Cruz, D. Lucio, and L. Velho, "Kinect and rgbd images: Challenges and applications," in *Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2012 25th SIBGRAPI Conference on*, 2012, pp. 36-49.

[9] F. C. Vieira, A. A. Medeiros, P. J. Alsina, and A. P. Araújo Jr, "Position and Orientation Control of a Two-Wheeled Differentially Driven Nonholonomic Mobile Robot," in *ICINCO (2)*, 2004, pp. 256-262.

[10] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*: MIT press, 2011.

[11] J. Criminisi and E. Konukoglu, "Decision Forests for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning.," Microsoft Research2011.