

DETC2011-47735

SLAM USING 3D RECONSTRUCTION VIA A VISUAL RGB & RGB-D SENSORY INPUT

Helge A. Wurdemann *
PhD Candidate
King's College London
London, United Kingdom

Lei Cui
Postdoctoral Fellow
King's College London
London, United Kingdom

Evangelos Georgiou *
PhD Candidate
King's College London
London, United Kingdom

Jian S. Dai
Professor of Mechanisms and Robotics
King's College London
London, United Kingdom

*joint first authors with equal contribution to this paper

ABSTRACT

This paper investigates simultaneous localization and mapping (SLAM) problem by exploiting the Microsoft Kinect™ sensor array and an autonomous mobile robot capable of self-localization. The combination of them covers the major features of SLAM including mapping, sensing, locating, and modeling. The Kinect™ sensor array provides a dual camera output of RGB, using a CMOS camera, and RGB-D, using a depth camera. The sensors will be mounted on the KCLBOT, an autonomous nonholonomic two wheel maneuverable mobile robot. The mobile robot platform has the ability to self-localize and perform navigation maneuvers to traverse to set target points using intelligent processes. The target point for this operation is a fixed coordinate position, which will be the goal for the mobile robot to reach, taking into consideration the obstacles in the environment which will be represented in a 3D spatial model.

Extracting the images from the sensor after a calibration routine, a 3D reconstruction of the traversable environment is produced for the mobile robot to navigate. Using the constructed 3D model the autonomous mobile robot follows a polynomial-based nonholonomic trajectory with obstacle avoidance.

The experimental results demonstrate the cost effectiveness of this off the shelf sensor array. The results show the effectiveness to produce a 3D reconstruction of an environment and the feasibility of using the Microsoft Kinect™ sensor for

mapping, sensing, locating, and modeling, that enables the implementation of SLAM on this type of platform.

1. INTRODUCTION

Over the past 30 years, researchers have been developing multiple solutions of visual-based mobile robots that are able to navigate within an unknown indoor and outdoor environment. Only during the last decade, this wide area has been focused on function driven navigation such as improving the living standard and increasing the independency of blind people. In this regard, concepts of obstacle avoidance and location as well as path planning using vision have been proposed by [1]. Another area of application is security: In [2], the 'WITH' mobile robot [3] is presented for detection of threat by evaluating unexpected objects and faces. The largest field apart from military use is urban search and rescue robots [4] [5]. Mobile robot systems aimed at this sector should be robust and available at rather low costs. Furthermore, targets are often not only identifiable via vision but via noise.

Mobile robot navigation has been of major interest since the 1980s. The development during this period is summarized the development in [6]. This survey concentrates on indoor and outdoor navigation. These are divided into three groups: Map-based systems depend on pre-defined geometric models or topological maps of the environment, whereas mapless navigation are systems that recognize objects found in the space or track those objects by generating motions based on visual

observations. Map-building-based navigation is an intermediate way where sensors construct their own geometric or topological models of the environment for navigation.

A simple, but robust and efficient algorithm for a mobile robot path planning is discussed in [7]. Here, a path is taught and replayed in indoor and outdoor environments. The system navigates by comparing feature coordinates qualitatively. Obstacle avoidance and global localization is part of the author's future work. Indoor navigation using 2-dimensional vision systems can be another way to explore the environment. Lie et al. [8] presents a two-stages-technique: During the offline part the surrounding is constructed with the Rao-Black-wellized particle filter. A location recognition algorithm then allocates features to the pre-built map in order to move autonomously within the area. A similar approach is used in [9]: Images are taken by a monocular camera, segmented, filtered by an edge algorithm, and modeled as a topological graph, where a certain position of the mobile robot is equivalent to a node. Hwang and Shih [10] use two charge-couple-device (CCD) cameras controlled by two stepping motors each to navigate a car-like wheeled robot. The cameras are mounted overhead and the robot is tagged with two landmarks. During indoor experiments images are locating the mobile robot using the landmarks and obstacles.

Stereo cameras are popular techniques for mobile robot navigation, some vision cameras are also expanded to an omnidirectional system [11] [12] [13]. In [14] a stereo visual system is mounted on an autonomous air vehicle for navigation after having tested this technique on a ground robot. The main contribution of this paper is to self-localize and estimate the change in position over time. A further step [15] describes an online stereo camera algorithm for reconstruction of urban environments. The sequence is as follows: Using a point cloud a 3D model is reconstructed, the environment divided into traversable ground regions, and a local safety map is built. This plot supplies information about safe and unsafe areas that is essential for the robotic system to navigate autonomously. In [16] a method is presented for obstacle avoidance and path planning in an indoor environment. Using a stereo camera mounted on a humanoid robot, the system recognizes the floor and detects obstacles via plane extraction without any a priori information of the surrounding space. The disadvantage of this method is that the environment needs to contain enough texture. In [17], the researchers also use stereo vision guidance for a humanoid robot. The main goal is to make this robot walking up stairs and crawling underneath obstacles. This is achieved by using scan-line grouping in order to segment planes in the environment. The key contribution of this paper is the extraction of height information that is used for path planning and navigation. However, it is mentioned that the success of stereo camera systems significantly depends on the level of texture since stereo vision relies on the horizontal disparity in order to create 3D images. Another way of getting a 3D reconstructed map of an environment is apply a 3D laser sensor with a hemispherical field [18] or use an IR sensor in combination with a single camera [19]. These last two sensors

are responsible for a different part of the mobile robot navigation system: The vision camera is used for planning the closest path to the target, whereas the IR sensors will help to avoid static and dynamic obstacles. The goal will be hit as the path is divided into intermediate steps.

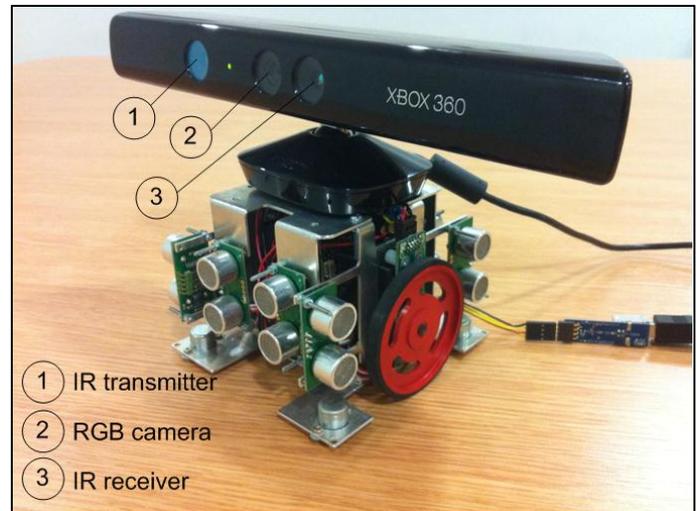


Figure 1. THE KINECT™ SENSOR & THE KCLBOT

This paper integrates the Microsoft Kinect™ to the simultaneous localization and mapping (SLAM) technique using not only 2D data (RGB images) but also depth information (RGB-D). Further, this system aims autonomously for a pre-defined audio signal. Without any a priori knowledge about the surrounding environment, the mobile robot traverses its own planned path and navigates to the source. Followed by a section that describes the vision device and calibration, the paper introduces the autonomous mobile robot KCLBOT, as depicted in Fig. 1. Next, an algorithm for polynomial-based nonholonomic path planning and obstacle avoidance is presented. Experimental results prove the stability and robustness of this approach.

2. RGB & RGB-D VISUAL IMAGE CAPTURE

The RGB and RGB-D capturing device was launched in the UK early November 2010. The vision devices, that are located on a horizontal line, are connected to a small base with a motorized tilt mechanism. The Kinect™ consists of an RGB camera, depth sensor and multi-array microphone (Fig. 1). This chapter describes the functionality and ability of the device as well as the calibration.

2.1 The Microsoft Kinect™ Sensor

The RGB images obtained by the color CMOS camera have 8-bit resolution (640×480 pixels). An extracted RGB image can be seen in Fig. 2(a). The CMOS sensor that will receive the IR light from the transmitter provides input for the depth map with 11-bit resolution (320×240 pixels). However in

this paper an 8-bit resolution (640×480 pixels) will be extracted (Fig. 2(b)). The principle of the Kinect™ sensor is as follows: Between the IR transmitter, sending out structured light, and receiver is a small angle. Also, the IR sensor should be provided with a band-pass filter in order to capture the IR light only. Using triangulation the depth can be recalculated.

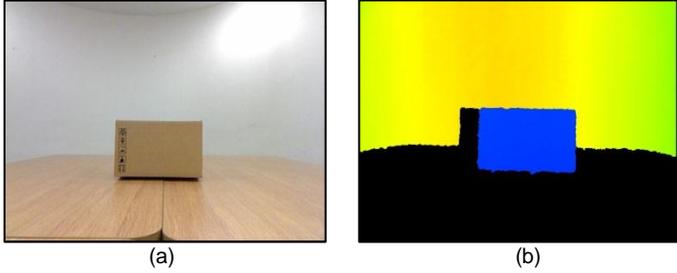


Figure 2. (a) RGB AND (b) RBG-D IMAGE CAPTURE

Fig. 2 (a) and (b) present the RGB and RGB-D images captured by the sensor, respectively.

2.2 Sensor Calibration

The two CMOS cameras are calibrated using the widely known pinhole camera model. Regarding the extrinsic parameters, the RGB camera will be used as the world coordinate frame, so that the depth sensor needs to be translated by -25mm in y -direction. The intrinsic matrix $\mathbf{M}_{intrinsic}$ is described by the focal length f_x and f_y and the principle point p_x and p_y , so that everything adds up to the following camera matrix:

$$\mathbf{M}_{intrinsic} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

In order to consider non-linear effects, the intrinsic matrix has to be multiplied with the radial distortion vector \mathbf{r}_d :

$$\mathbf{r}_d = \begin{bmatrix} d_x(x^2 + y^2)^3 + d_y(x^2 + y^2)^2 + d_z(x^2 + y^2) + 1 \\ d_x(x^2 + y^2)^3 + d_y(x^2 + y^2)^2 + d_z(x^2 + y^2) + 1 \\ 1 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} X/Z \\ Y/Z \end{bmatrix} \quad (3)$$

where, X , Y and Z is a point in the camera reference frame. For the RGB camera, the intrinsic parameters are:

$$\mathbf{M}_{intrinsic,RGB} = \begin{bmatrix} 522.82 & 0 & 320.61 \\ 0 & 521.63 & 242 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$\mathbf{d}_{RGB} = \begin{bmatrix} 1.12 \\ -0.91 \\ 0.26 \end{bmatrix} \quad (5)$$

and for the IR sensor:

$$\mathbf{M}_{intrinsic,RGB-D} = \begin{bmatrix} 557.14 & 0 & 304.44 \\ 0 & 556.75 & 229.18 \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$\mathbf{d}_{RGB-D} = \begin{bmatrix} -0.2 \\ 0.54 \\ -0.48 \end{bmatrix} \quad (7)$$



Figure 3. RGB IMAGE WITH INTRINSIC CALIBRATION

The RGB-D image shows a certain RGB color sequence going from close to deep. As z increases, the order is as follows: Magenta (1, 0, 1), Blue (0, 0, 1), Cyan (0, 1, 1), Green (0, 1, 0), Yellow (1, 1, 0), Red (1, 0, 0), where $R, G, B \in [0, 1]$. This can be written in cylindrical-coordinate representations by calculating the hue, saturation and lightness value in the HSV color space. The three equations are given by [20]:

$$H = \begin{cases} 0^\circ, & \text{if } R = G = B \\ 60^\circ \left(0 + \frac{G - B}{\max(R, G, B) - \min(R, G, B)} \right), & \text{if } \max(R, G, B) = R \\ 60^\circ \left(2 + \frac{B - R}{\max(R, G, B) - \min(R, G, B)} \right), & \text{if } \max(R, G, B) = G \\ 60^\circ \left(4 + \frac{R - G}{\max(R, G, B) - \min(R, G, B)} \right), & \text{if } \max(R, G, B) = B \end{cases} \quad (8)$$

$$S = \begin{cases} 0, & \text{if } R = G = B \\ \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)}, & \text{otherwise} \end{cases} \quad (9)$$

$$V = \max(R, G, B) \quad (10)$$

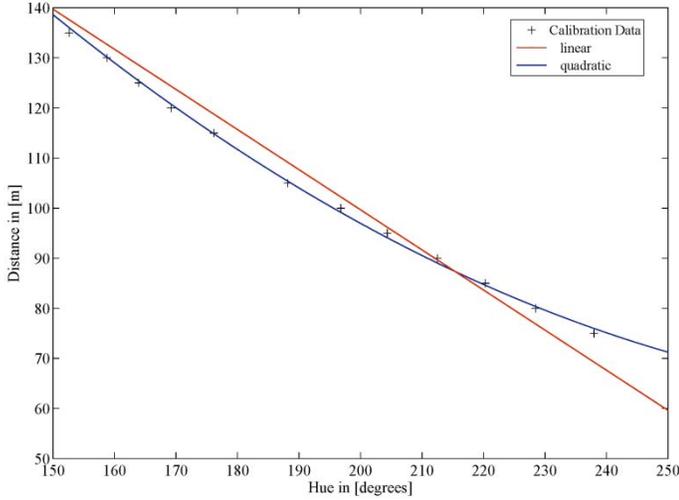


Figure 4. QUADRATIC INTERPOLATION BETWEEN DISTANCE AND HUE VALUE

Fig. 4 plots the distance $d = [70,170]$ in [cm] against the hue value H in $[\circ]$. Unlike a linear approximation, a quadratic equation describes the ratio between the distance and the hue value more accurate:

$$d = 0.0032H^2 - 1.9492H + 35926 \quad (11)$$

Having Equation (11) allows for the computation of distance based on quadratic relationship to Hue.

2.3 HSV-Distance Results

Since the IR depth sensor is calibrated for a distance between 70cm and 170cm, tests are taken within this interval. Four samples can be seen in Fig. 9. The obstacle in the middle of the image is located $d=80, 95, 110$ and 162 cm from the KinectTM. Table 1 shows the translation from the RGB color space to the HSV color space. As mentioned before, the hue value is of special interest because this is related to the distance d by the quadratic Equation (11). Using this interpolation, the distance can be calculated. Compared with the measured distance, these is an average error of 1.01% .

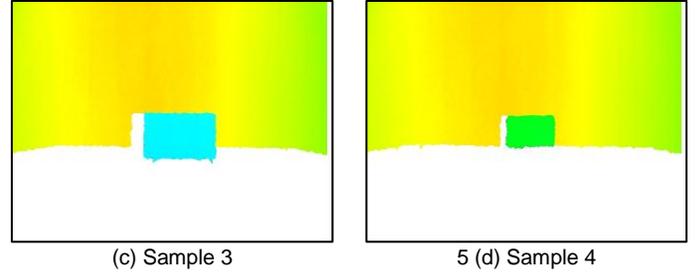
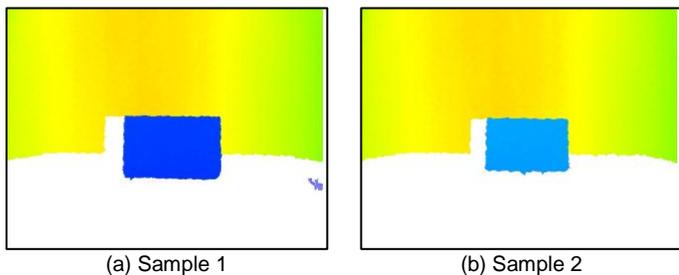


Figure 5. FOUR SAMPLES FOR DISTANCE CALCULATION

Table 1. RGB-D SAMPLE DISTANCE ESTIMATION

Sample	R	G	B	Hue in $[\circ]$	Distance d in [cm]
1	0.00	0.24	1.00	225.88	81.67
2	0.01	0.64	1.00	202.06	95.60
3	0.00	0.98	1.00	181.41	110.60
4	0.00	1.00	0.12	127.29	162.81

Implementing the calibrated quadratic distance equation for d , Equation (11), Table 1 and Fig. 5 (a), (b), (c), and (d) present the expected distance valuation of the tracked obstacle.

2.4 3D Reconstruction using a RGB & RGB-D Sensor

From the 3D data gained from the RGD and RGB-D sensor, it is possible to generate a point cloud. The point cloud includes a description of the alignment of surfaces specified by a 3-tuple in order to reconstruct a polygonal mesh. These points are referred to as vertices if they are to be used as corners. Furthermore, the data supplies information about the RGB values for each point. Fig. 6(a) shows a view along the positive x-axis. It can be clearly distinguished between the background and obstacle. In Fig. 6(b) this view has been pitched by 45° .



Figure 6. 3D RECONSTRUCTION (a) FRONT VIEW AND (b) ROTATED BY 45°

The processed 3D reconstruction, Fig. 6 (a) and (b), provides the mobile robot with an environment map for path planning.

3 THE KCLBOT: AN AUTONOMOUS MOBILE ROBOT

The KCLBOT [21] is a non-holonomic two wheeled mobile robot. The mobile robot is built around the specifications for 'Micromouse Robot' and the 'RoboCup' competition. These specifications contribute to the mobile robot's form factor and size. This mobile robot holds a complex

electronic system to support on-line path planning, self-localization, and even simultaneous localization and mapping (SLAM), which is made possible by the onboard sensor array.

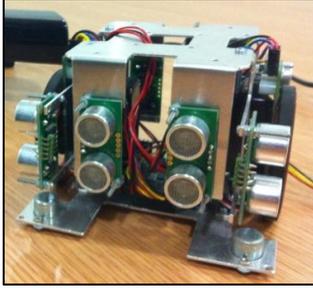


Figure 7. THE KCLBOT: A NONHOLONOMIC MOBILE ROBOT

A suitable autonomous mobile robot is required as a platform for the Microsoft Kinect™ sensor. Fig. 7 presents the KCLBOT which is the platform used to support the sensor array.

3.1 Mobile Robot Configuration

In the maneuverable classification of mobile robots [22], the vehicle is defined as being constrained to move in the vehicle's fixed heading angle. For the vehicle to change maneuver configuration, it needs to rotate about itself. As the vehicle traverses on a two dimensional plane both left and right wheels follow a path that moves around the instantaneous center of curvature at the same angle, which can be defined as ω , and thus the angular velocity of the left and right wheel rotation is deduced as follows:

$$\dot{\theta}_L = \omega \left(icc_r - \frac{L}{2} \right) \quad (12)$$

$$\dot{\theta}_R = \omega \left(icc_r + \frac{L}{2} \right) \quad (13)$$

Where L is the distance between the centers of the two rotating wheels and the parameter icc_r is the distance between the mid-point of the rotating wheels and the instantaneous center of curvature. Using the velocities Equations (12) and (13) of the rotating left and rights wheels, $\dot{\theta}_L$ and $\dot{\theta}_R$ respectively, the instantaneous center of curvature, icc_r and the curvature angle, ω can derived as follows:

$$icc_r = \frac{L(\dot{\theta}_R + \dot{\theta}_L)}{2(\dot{\theta}_R - \dot{\theta}_L)} \quad (14)$$

$$\omega = \frac{(\dot{\theta}_R - \dot{\theta}_L)}{L} \quad (15)$$

Using Equations (14) and (15), two singularities can be identified. When $\dot{\theta}_R = \dot{\theta}_L$, the radius of instantaneous center of

curvature, icc_r tends towards infinity and this is the condition when the mobile robot is moving in a straight line. When $\dot{\theta}_R = -\dot{\theta}_L$, the mobile robot is rotating about its own center and the radius of instantaneous center of curvature, icc_r , is null.

When the wheels on the mobile robot rotate, the quadrature shaft encoder returns a counter tick value; the rotation direction of the rotating wheel is given by positive or negative value returned by the encoder. Using the numbers of tick counts returned, the distance travelled by the rotating left and right wheel can be deduced in the following way:

$$d_L = \frac{L_{ticks} \pi D}{L_{res}} \quad (16)$$

$$d_R = \frac{R_{ticks} \pi D}{R_{res}} \quad (17)$$

Where L_{ticks} and R_{ticks} depicts the number of encoder pulses counted by left and right wheel encoders, respectively, since the last sampling, and where D is defined as the diameter of the wheels. With resolution of the left and right shaft encoders L_{res} and R_{res} , respectively, it is possible to determine the distance travelled by the left and right rotating wheel, d_L and d_R . This calculation is represented in Equations (16-17).

3.2 Self-localization via a Dual Shaft Encoder Configuration

By using the quadrature shaft encoders that accumulate the distance travelled by the wheels, a form of position can be deduced by deriving the mobile robot's x , y Cartesian position and the maneuverable vehicle's orientation φ , with respect to time. The derivation starts by defining and considering $s(t)$ and $\varphi(t)$ to be function of time, which represents the velocity and orientation of the mobile robot, respectively. The velocity and orientation are derived from differentiating the position form as follows:

$$\frac{dx}{dt} = s(t) \cdot \cos(\varphi(t)) \quad (18)$$

$$\frac{dy}{dt} = s(t) \cdot \sin(\varphi(t)) \quad (19)$$

The change in orientation with respect to time which was defined in Equation (15) and can be described as follows:

$$\frac{d\varphi}{dt} = \omega = \frac{\dot{\theta}_R - \dot{\theta}_L}{b} \quad (20)$$

When Equation (20) is integrated, the mobile robot's angle orientation value $\varphi(t)$ with respect to time is achieved. The mobile robot's initial angle of orientation $\varphi(0)$ is written as φ_0 and is represented as follows:

$$\varphi(t) = \frac{(\dot{\theta}_r - \dot{\theta}_l)t}{b} + \varphi_0 \quad (21)$$

The velocity of the mobile robot is equal to the average speed of the two wheels and this can be incorporated into Equations (18) and (19), which is depicted as follows:

$$\frac{dx}{dt} = \frac{\dot{\theta}_r + \dot{\theta}_l}{2} \cos(\varphi(t)) \quad (22)$$

$$\frac{dy}{dt} = \frac{\dot{\theta}_r + \dot{\theta}_l}{2} \sin(\varphi(t)) \quad (23)$$

The next step is to integrate equations (22) and (23) to the initial position of the mobile robot, which is depicted as follows:

$$x(t) = x_0 + \frac{L(\dot{\theta}_r + \dot{\theta}_l)}{2(\dot{\theta}_r - \dot{\theta}_l)} \left(\sin\left(\frac{(\dot{\theta}_r - \dot{\theta}_l)t}{b} + \varphi_0\right) - \sin(\varphi_0) \right) \quad (24)$$

$$y(t) = y_0 + \frac{L(\dot{\theta}_r + \dot{\theta}_l)}{2(\dot{\theta}_r - \dot{\theta}_l)} \left(\cos\left(\frac{(\dot{\theta}_r - \dot{\theta}_l)t}{b} + \varphi_0\right) - \cos(\varphi_0) \right) \quad (25)$$

Equations (24) and (25) describe the mobile robot's position, where $x(0) = x_0$ and $y(0) = y_0$ are the mobile robot's initial positions. The next step is to represent Equations (21), (24) and (25) in terms of the distances that the left and right wheels have traversed, which are defined by d_r and d_l . This can be achieved by substituting $\dot{\theta}_r$ and $\dot{\theta}_l$ (in Equations (21), (24) and (25)) for d_r and d_l , respectively, and also dropping the time constant t to achieve the following:

$$\theta = \frac{d_r - d_l}{2} + \theta_0 \quad (26)$$

$$x(t) = x_0 + \frac{L(d_r + d_l)}{2(d_r - d_l)} \left(\sin\left(\frac{(d_r - d_l)t}{b} + \varphi_0\right) - \sin(\varphi_0) \right) \quad (27)$$

$$y(t) = y_0 + \frac{L(d_r + d_l)}{2(d_r - d_l)} \left(\cos\left(\frac{(d_r - d_l)t}{b} + \varphi_0\right) - \cos(\varphi_0) \right) \quad (28)$$

By implementing Equations (26) to (28), they provide a solution to the relative position of a maneuverable mobile robot. This might offer a possible solution to the self-localization problem but is subject to accumulative drift of the position and orientation with no method of re-alignment. The accuracy of this method is subject to the sampling rate of the data accumulation, such that if small position or orientation changes are not recorded then the position and orientation will be erroneous.

4. POLYNOMIAL-BASED NONHOLONOMIC PATH PLANNING AND OBSTACLE AVOIDANCE

This part concentrates on finding a path for the KCLBOT from its initial configuration as described by (x_0, y_0, φ_0) to a final one (x_1, y_1, φ_1) . The nonholonomic constraint has to be satisfied and the three dimensional final configuration space has to be reached with two controls only. The paper adopts a polynomial approach to the path planning while obstacle avoidance is realized by using the higher order of the polynomials. The vertices and edges of the KCLBOT as well as those of the obstacles are enclosed in simple shapes such as circles or squares.

To achieve the task of path planning detailed information about the traversable space and location of potential obstacles is required. Using the RGB-D image a localization map is produced for the path planning and obstacle avoidance.

4.1 RGB-D Image to 2D Environment Mapping

Before the autonomous mobile robot can complete any path planning or path following tasks, it requires sufficient information about the environment that it will be traversing. To provide the mobile robot with this information the detail from the RGB-D camera is used to make plot of the terrain, plotting the un-obstructed space the mobile robot can utilize.

Before the RGB-D image can be used, the noise resolved as black pixels in the range of #E4 E1 C0h to #FF FF FFh needs to be removed from the image. This is achieved by converting the RGB-D image to gray scale [23]. This process is carried out to protect natural colors in the #E4 E1 C0h to #FF FF FFh range. In the RGB color model, a color image can be represented by the following intensity function:

$$I_{RGB} = (F_R, F_G, F_B) \quad (29)$$

From Equation (29), F_R is the intensity of the pixel (x,y) in the red channel, F_G is the intensity of pixel (x,y) in the green channel, and F_B is the intensity of pixel (x,y) in the blue channel. Using only the brightness information the color image can be transformed into a gray scale image [23].

$$I_{GS} = 0.33F_R + 0.5F_G + 0.166F_B \quad (30)$$

Where Equation (30) presents the equation that converts a color pixel to a gray scale pixel.

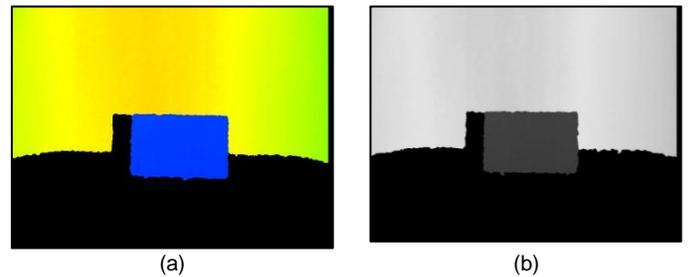


Figure 8. RGB-D (a) TO GRAY SCALE (b) CONVERSION

After the image has been converted to gray scale, as depicted in Fig. 8, the black pixels are filtered out of the image.

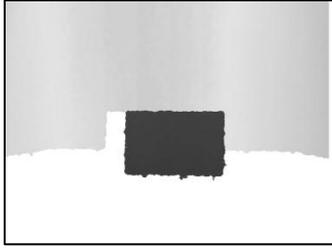


Figure 9. GRAY SCALE FILTERED IMAGE

Once the image has been stripped from the black noise pixels, as depicted in Fig. 9, the color detail is required for mapping the traversable environment.

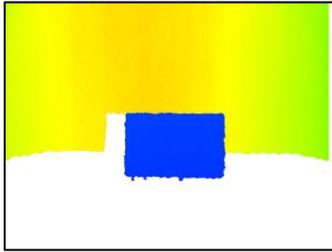


Figure 10. COLOR REMAPPING ON FILTERED IMAGE

The RGB-D depth color information from Fig. 8 (a) is remapped onto the gray scale filtered image; the result is presented in Fig. 10. Using the HSV [24] cylindrical-coordinate representation of points in an RGB color model, the image is rotated by 90° , resulting in an image of a topological view of the traversable space.

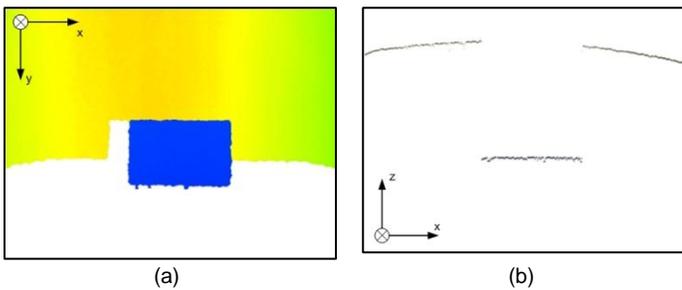


Figure 11. REMAPPED RGB-D FILTERED IMAGE ROTATION

The rotation of the RGB-D image, Fig. 11 (a), results in detailed localization mapping information, presented in Fig. 11 (b), that the mobile robot can use for path planning.

4.2 Obstacle Avoidance: A Polynomial Approach

Two independently driven wheels are used to drive the mobile robot vehicle. It is assumed that the system moves at a low speed and the ground provides enough friction force. So

the two driven wheels do not slip sideways. The velocity of any point on the wheel axis is normal to this axis. This leads to following constraint equation:

$$\dot{x} \sin(\varphi) - \dot{y} \cos(\varphi) + \dot{\varphi}L = 0 \quad (31)$$

Where L represents the width of the robot.

The above equation is a nonholonomic constraint involving velocities and, as is well known, it cannot be integrated analytically to result in a constraint between the configuration variables of the platform, namely, x , y , and φ . Also, the configuration space of this system is three-dimensional while the velocity space is two-dimensional.

The nonholonomic constraint can be written in the form of

$$\begin{aligned} u &= x \sin(\varphi) - y \cos(\varphi) \\ v &= L - x \cos(\varphi) - y \sin(\varphi) \\ \omega &= \dot{\varphi} \end{aligned}$$

If we choose functions f and g as follows:

$$\omega = f(t), u = g(t), v = -du/d\omega \quad (32)$$

and select the functions f and g to be fifth and third order time polynomials, we can obtain the trajectory with obstacle avoidance. Details can be found in [25].

6. CONCLUSION & DISCUSSION

This paper presents the utilization of the Microsoft Kinect™ Sensor to support the SLAM methodology by exploiting the RGB and RGB-D images for mapping, sensing, locating, and modeling. Before any image processing is possible the image inputs are calibrated to acquire the image in a pinhole model with intrinsic calibration. Using a HSV cylindrical-coordinate mapping space, a quadratic distance estimation model is presented to resolve the estimation of a potential obstacles distance from the sensor. Using the RGB and RGB-D images a 3D reconstruction method is presented for environment modeling. The KCLBOT, an autonomous nonholonomic maneuverable mobile robot, is used as a platform for the sensor to capture the experimental images. The mobile robot is self-localizing using the quadrature shaft encoders to resolve orientation and planar position. The mobile robot is provided with an environment overview map by the presented RGB-D image rotation method. This mapping information is applied to the polynomial based obstacle avoidance and path planning approach.

The experimental images demonstrate the cost effectiveness of this off the shelf sensor array. The results show the effectiveness to produce a 3D reconstruction of an environment and the feasibility of using the Microsoft Kinect™ sensor for mapping, sensing, locating, and modeling, that enables the implementation of SLAM on this type of platform.

REFERENCES

- [1] Amutha, B., Ponnaivaikko, M., November 2009, "Mobile Assistant as a Navigational Aid for Blind Children to identify Landm," *International Journal of Recent Trends in Engineering*, 2(3), pp. 152-154.
- [2] Gordon, S., Pang, S., Nishioka, R., Kasabov, N., and Yamakawa, T., 2009, "Vision Based Mobile Robot for Indoor Environmental Security," *Proc. 15th International Conference on Neural Information Processing of the Asia-Pacific Neural Network Assembly*, Springer-Verlag, Berlin Heidelberg, pp. 962-969.
- [3] Mori, K., Sato, M., Sonoda, T., and Ishii, K., "Toward realization of swarm intelligence," *Proc. 7th Postech-Kyutech Joint Workshop on Neuroinformatics*.
- [4] Scholtz, J., Young, J., Drury, J.L., and Yanco, H.A., "Evaluation of Human-Robot Interaction Awareness in Search and Rescue," *Proc. 2004 IEEE International Conference on Robotics and Automation*, pp. 2327-2332.
- [5] Davids, A., 2002, "Urban Search and Rescue Robots: From Tragedy to Technology," *IEEE INTELLIGENT SYSTEMS Histories and Futures*, pp. 81-83.
- [6] DeSouza, N. G., and Kak, A.C., 2002, "Vision for Mobile Robot Navigation: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2), pp. 237-267.
- [7] Chen, Z., and Birchfield, S.T., "Qualitative Vision-Based Mobile Robot Navigation," *Proc. 2006 IEEE International Conference on Robotics and Automation*, pp. 2686-2692.
- [8] Li, M.-H., Hong, B.-R., Cai, Z.-S., Piao, S.-H., and Huang, Q.-C., 2007, "Novel indoor mobile robot navigation using monocular vision," *Engineering Applications of Artificial Intelligence*, 21, pp. 485-497.
- [9] Santosh, D., Achar, S., and Jawahar, C.V., "Autonomous Image-based Exploration for Mobile Robot Navigation," *Proc. 2008 IEEE International Conference on Robotics and Automation*, pp. 2717-2722.
- [10] Hwang, C., and Shih, C., March 2009, "A Distributed Active-Vision Network-Space Approach for the Navigation of a Car-Like Wheeled Robot," *IEEE Transactions on Industrial Electronics*, 56(3), pp. 846-855.
- [11] Gaspar, J., Winters, N., and Santos-Victor, J., 2000, "Vision-Based Navigation and Environmental Representations with an Omnidirectional Camera," *IEEE Transactions on Robotics and Automation*, 16(6), pp. 890-898.
- [12] Adorni, G., Mordonini, M., Cagnoni, C., Sgorbissa, A., "Omnidirectional stereo systems for robot navigation," *Proc. 2003 Conference on Computer Vision and Pattern Recognition Workshop*, pp. 1-7.
- [13] Lui, W. L. D., and Jarvis, R., 2010, "Eye-Full Tower: A GPU-based variable multibaseline omnidirectional stereovision system with automatic baseline selection for outdoor mobile robot navigation," *Robotics and Autonomous Systems*, 58, pp. 747-761.
- [14] Mejias, L., Campoy, P., Mondragon, I., and Doherty, P., 3-5 September 2007, "Stereo Vision-Based Navigation for an Autonomous Helicopter," *6th IFAC Symposium on Intelligent Autonomous Vehicle*.
- [15] Muraka, A., and Kuipers, B., "A Stereo Vision Based Mapping Algorithm for Detecting Inclines, Drop-offs, and Obstacles for Safe Local Navigation," *Proc. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1646-1653.
- [16] Sabe, K., Fukuchi, M., Gutmann, J.S., Ohashi, T., Kawamoto, K., and Yoshigahara, T., "Obstacle Avoidance and Path Planning for Humanoid Robots using Stereo Vision," *Proc. 2004 IEEE International Conference on Robotics and Automation*, pp. 592-597.
- [17] Gutmann, J.-S., Fukuchi, M., and Fujita, M., 2008, "3D Perception and Environment Map Generation for Humanoid Robot Navigation," *The International Journal of Robotics Research*, 27(10), pp. 1117-1134.
- [18] Ryde, J., and Hu, H., "3D Laser Range Scanner with Hemispherical Field of View for Robot Navigation," *Proc. 2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 891-896.
- [19] Singh, N. N., Chatterjee, A., Chatterjee, A., and Rakshit, A., 2011, "A two-layered subgoal based mobile robot navigation algorithm with vision system and IR sensors," *Measurement*, in press.
- [20] Smith, A. R., "Color gamut transform pairs," *Proc. 5th Annual Conference on Computer Graphics and Interactive Techniques* pp. 12-19.
- [21] Georgiou, E., 2010, "The KCLBOT Mobile Robot," www.KCLBOT.com.
- [22] Campion, G., Bastin, G., D'Andrea-Novell, B., 1996, "Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots," *IEEE Transactions on Robotics and Automation*, 12(2), pp. 47-62.
- [23] Kumar, T., and Verma, K., 2010, "A Theory Based on Conversion of RGB image to Gray image," *International Journal of Computer Applications*, 7(2), pp. 7-10.
- [24] Joblove, G., and Greenberg, D., "Color spaces for computer graphics," *Proc. 5th Annual Conference on Computer Graphics and Interactive Techniques*.
- [25] Papadopoulos, E., Poulakakis, I., and Papadimitriou, I., 2002, "On Path Planning and Obstacle Avoidance for Nonholonomic Platforms with Manipulators: A Polynomial Approach," *The International Journal of Robotics Research*, 21(4), pp. 367-383.