VOL. 10, NO. 21, NOVEMBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

3D PANORAMA SCENE RECONSTRUCTION USING KINECT CAMERA

Mohd Razali Daud, Nur Afzan Murtadza, M.S. Hendriyawan Achmad and Saifudin Razali
Intrumentation and Control Engineering Research Centre, Fakulti Kejuruteraan Electrik & Elektronik, Universiti Malaysia Pahang, Pekan, Pahang, Malaysia
E-Mail: mrazali@ump.edu.my

ABSTRACT

In this paper, 3D panorama scene is constructed by moving around the Kinect Xbox 360 camera horizontally in indoor environment. The Kinect sensor is used because its price is cheaper than other devices but able to provide bountiful data for image processing purpose. By integrating the Kinect for windows with MATLAB, all computations, programming and processing of this project are done using the MATLAB itself. The overall system undergoes three major sectors which are the Image Acquisition Module, Image Processing and Analysis Module, and Result Processing and Displaying Module. The proposed system uses the latest "Point Cloud Processing" that was introduced in the MATLAB R2015a. Based on the result obtained, the system is able to reconstruct the 3D scene environment via offline and also real-time using the Graphical User Interface (GUI) for ease of use. The online system however, may need further improvement in terms of stabilization. Furthermore, the system is able to function with minimum lighting i.e. dark room or at night.

Keywords: 3D panorama, kinect, point cloud processing, image processing.

INTRODUCTION

3D mapping is expensively needed in many sectors and applications such as for search and rescue (SAR) mission with the major concern of navigations. In the case of fire rescue operation, a detailed map containing heat signal and overall view of the environment is essential to be translated to the fire fighters before heading into the scene. Nowadays, 3D environment is reconstructed with a combination of technique commonly known as SLAM [1] with the help of supporting devices such as mobile camera, LRF, laser scanner, or depth camera. One of the most important steps in SLAM is the 3D scene reconstruction. There are many support and application for 3D reconstruction such as for object recognition, path planning, navigation, localization and data association, etc. As such, there are two popular sensors used to acquire environment data for mapping problem in the market, which are the depth camera and the laser range finder (LRF). LRF takes a longer time to process due to the tilting time of laser scanner when generating 3D depth map. Furthermore, the LRF is an expensive item to use commercially. On the other hand, the most commonly used camera in 3D scene reconstruction is by using stereo camera. It is popular among many users due to its simple coding, software friendly and flexibility. Software friendly mentioned is in regard to the system have been widely used either in C, C++, VB, MATLAB and more. Flexibility of the stereo vision is because it is applicable to indoor and outdoor environment with less limitation compared to Kinect sensor mostly due to the absent of IR sensors. The shortcoming of using the stereo vision system is the extremely complicated pre-calibration. Since there are no IR sensors used and only the presence of RGB camera, extra measurement and calibration are needed to find the disparity formula to get the depth (distance) data. Therefore, correspondence matching issues will arise and requires higher cost as it involves a larger number of hardware [2]. The key point in this project is the 3D scene reconstruction is relied solely on the Microsoft Kinect sensors that consist of CMOS IR sensor for depth sensing, CMOS colour sensor for RGB imaging, a tilting motor and three-axis accelerometer. The low cost depth camera such as the Kinect's sensor image processing capabilities can reconstruct a real-time 3D scene faster and simpler than using the stereo camera [3]. However, Kinect's sensor is not quite as accurate as the LRF. Therefore, this project is inspired by the shortage of the both systems.

SCENE RECONSTRUCTION

The scene reconstruction will be going through three main steps; (1) Image acquisition module. This will be separated into two individual scan that is the depth camera and RGB camera. (2) Image processing and analysis module. In this step, the RGB data and depth information will undergo image alignment including point cloud transformation (3) Result Processing Module. Point cloud processing will then merge the results from the previous module and the 3D scene environment can be reconstructed. The above said steps are depicted in the Figure-1 below.

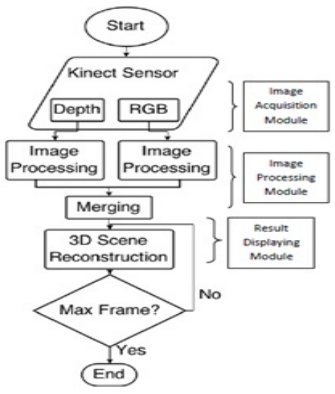

Figure-1. Three steps processing of scene construction.

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

Image acquisition

Images are acquired using Kinect Xbox 360 camera with Kinect SDK acts as the medium to connect the camera to windows. A RGB-D sensor such as the Microsoft Kinect sensor is well-suited for extracting data as it produces a real-time feed of RGB-D measurements at 30 frames per second. The 30 Hz Kinect consisting of CMOS IR sensor, IR emitter and RGB camera provides us with the RGB data and also real-time depth data. The depth or distance is calculated by projecting a speckled pattern from the IR projector and measuring its time of flight (TOF) [4]. Using the principle of TOF, the CMOS sensor will measure the time taken for the IR light to bounce back or reflect of the object [5]. This phenomenon is known as structured light and also known as Kinect's depth by stereo's principle. The principle states that, the projecting and the measuring sensor should be set at different angle. Furthermore, Kinect also uses another principle which is depth by focus whereby the image produced by the Kinect will assume the more blurry the object, the further away it is from the camera [6]. This concept comes from utilizing the dots from the projected pattern that are more spaced out on near object rather than the further object which has more dense dots.

The depth or distance is calculated by projecting a speckled pattern from the IR projector and measuring its time of flight (TOF). Main functionality of the acquisition module is to capture the RGB and depth image per-frame separately and transforming it into single point cloud form. However, the pixel data is not usable in their raw-bit form and must be converted to depth value with the following equation below with raw bit as the depth image output value [7].

To obtain a single point cloud, the pixel value are translated by using the model where (*u*,*v*,*d*) are the coordinates of pixel in the depth image while (*x*,*y*,*z*) are coordinates of 3D in global frame. *fx, fy* as focal length on each direction and *cx, cy* are coordinates of principal point of camera. Camera calibration is in order first to evaluate the focal length and principal point position.

$$d = \frac{1}{-0.002955 \times raw_{bit} + 3.206} \quad (1)$$

$$x = \frac{u - cx}{fx} \times d \quad (2)$$

$$y = \frac{v - cy}{fy} \times d \quad (3)$$

$$z = d \quad (4)$$

$$\begin{bmatrix} x\ colour \\ y\ colour \\ z\ colour \end{bmatrix} = r \begin{bmatrix} x\ depth \\ y\ depth \\ z\ depth \end{bmatrix} + t \quad (5)$$

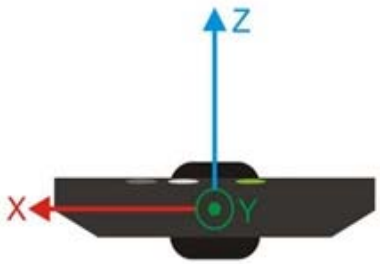

Figure-2. Kinect axes orientation.

Then, the system undergoes the alignment method instead of the Point Cloud Processing. Alignment is the process in which colour information is translated to point clouds where *r* and *t* is the transformation between colour and depth camera to display the 3D scene environment desired.

The colour and depth (distance) data are aligned and transformed into point cloud representation that includes crucial information including the location, point cloud count, X-limit, Y-limit and Z-limit. In addition, the X, Y and Z- axes of the point cloud displayed, are based on the axes orientation aligned to the Kinect sensor by default.

Point cloud processing

Point cloud processing is the latest 3D vision process introduced in MATLABR2015a [8]. It includes a number of functions in handling the point cloud data strictly in ply format such as down sampling, transform, reading, merging and more. However, the point cloud that was acquisitioned must be saved in a cell format before it can undergo the main processing. In relation to this, the collection of point clouds are loaded into the workspace and saved as one big structure named as "data.mat". Next, the properties of the structure must be in cell format to accommodate the large data and information held by each point cloud. By using the *struct2cell* function and transposing the cell, the data's properties is changed so that it is saved horizontally (point cloud per-column) as "OfflineData.mat".

EXPERIMENTAL RESULTS AND DISCUSSION

From the results shown below, the gap or holes in the depth map of Figure-4 in bright lights are higher than the depth map of Figure-3 in darken room. This is due to the correlation and the disparity measurements are affected by the lighting conditions. The laser speckles appeared in low contrast in strong light therefore resulting in outliers and holes in the resulting point cloud [9].

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

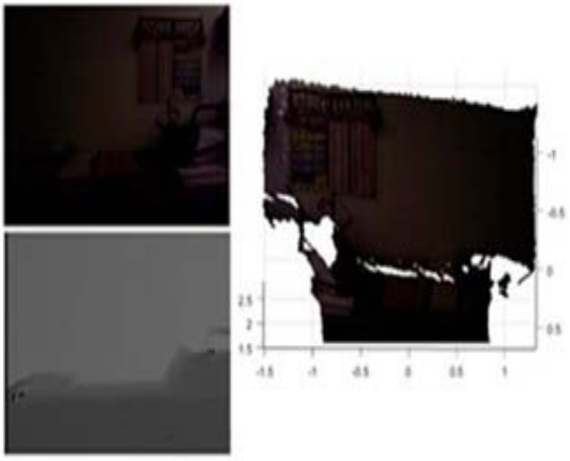

Figure-3. Point cloud output of darken room.

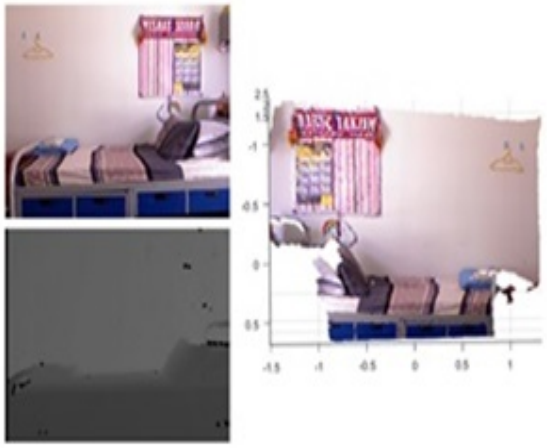

Figure-4. Point cloud output of bright room.

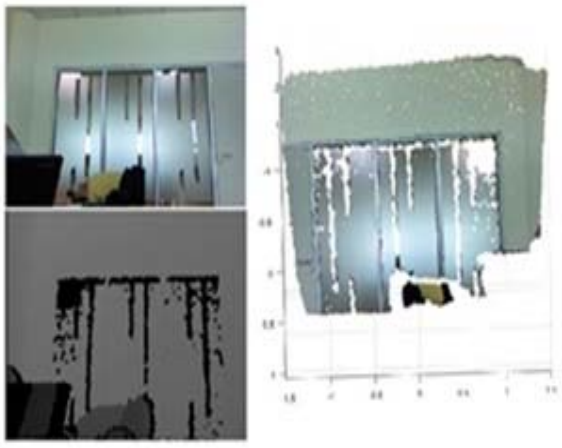

Figure-5. Point cloud output of indoor environment.

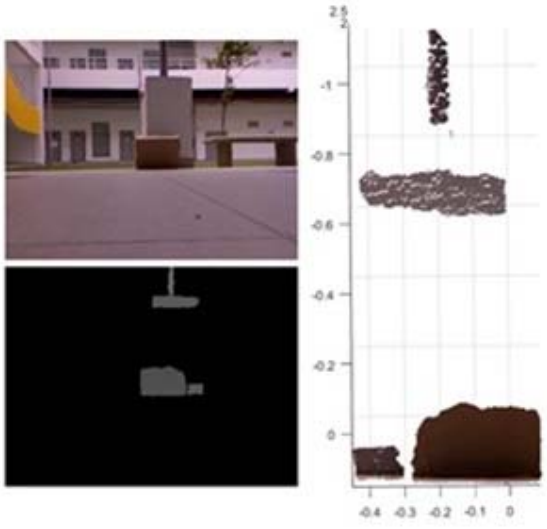

Figure-6. Point cloud output of outdoor environment.

The images shown in Figure-5 and 6 are the results of experiment conducted in two different environments that are indoor and outdoor. The biggest difference here can be seen clearly from the disparity map produced. Figure-5 shows the point cloud output of the system tested inside a room (indoor) with sufficient lighting. Meanwhile, Figure-6 shows the resulting output of the system tested outdoor in the evening. From the above tests, most prominent result here is gap produced in the depth image during outdoor capture. There will be a battle of IR light here which causes the interrupt in the depth map. This is also known as sunlight interference, whereby IR from the sunlight itself is naturally higher than the IR in Kinect sensor used. Thus, the distance / disparity cannot be measured and resulting in large gap in the depth image captured. The Kinect is highly sensitive in outdoor environment with the worst output depth map during noon. Furthermore, in accordance to 24 hours system, as the sun rises, the depth image captures next to nothing. However, as the sun starts to set, the chances of Kinect capturing the depth data starts to increase proving the phenomenon mentioned above.

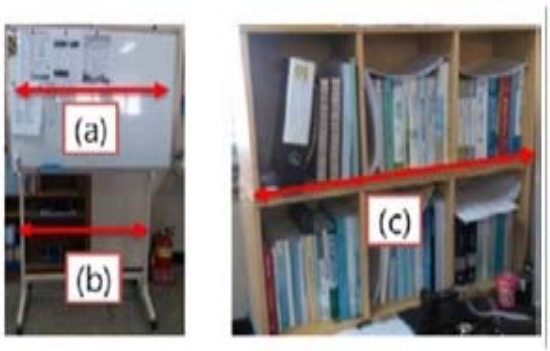

Figure-7. Objects for estimating accuracy.

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

Another test did was for point cloud's axes accuracy, by comparing the estimated length of an object with a real value (known). The tested objects are as shown in Figure-7 and the results are summarized in Table-1 below.

Table-1. Accuracy of point cloud axes.

(mm)	Estimated Length	Real Length	Rate of Error (%)
(a)	420	412	1.94
(b)	365	350	4.3
(c)	870	845	2.95

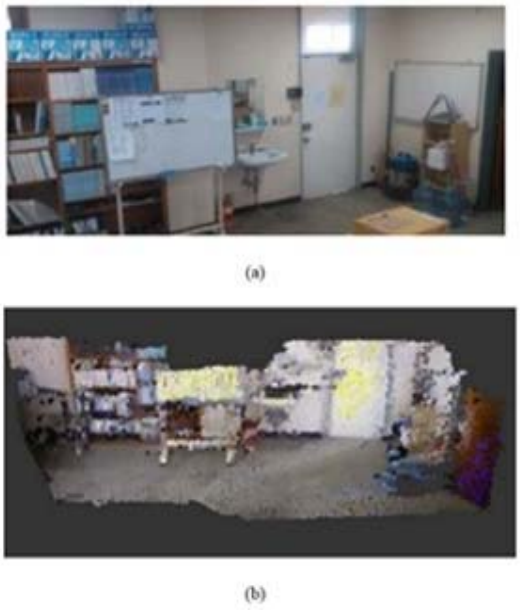

Figure-8. (a) The ground truth (b) Point cloud output.

Figure-8 shows the ground truth and the point cloud output images for the whole system. As can be seen, the system is able to produce a point cloud image almost same as the ground truth image. However, the biggest downside of this method is that it requires a lengthy coding and complex steps for each section.

Following the tests, a few limitations are encountered; properties of the object that may impede on the disparity measurement. Objects with shinier texture tend to cause a disruption in the IR structured light principle more than the dull textured [10].

CONCLUSIONS

Based on the results acquired, the proposed system of "3D Scene Reconstruction Using Kinect Sensor" was successfully developed and tested. It is capable in reconstructing the 3D scene environment via offline and also real-time. Point clouds are transformed and registered using ICP algorithm and stitched based on Matlab point cloud processing.

The integration of Kinect and the software MATLAB were successfully carried out along the pre-development of the reconstruction system. The completed system was tested in several different factors and the accuracy of the 3D map generated was tested by calculating error rate between measured value and the ground truth (real value) of a given object. Furthermore, the system is also compared with another method and the accuracy are calculated.

The system is able to reconstruct the 3D scene environment with an acceptable error of less than 5%. However, due to the limitations of the Kinect, this system are much more suitable for indoor usage. The system also shows that it produced a better output in offline system compared to the online system. The overall results shows that the depth data from the IR sensors and the RGB images of Kinect sensor could be used in reconstructing the 3D scene environment.

REFERENCES

[1] T. Emter and A. Stein. 2012. Simultaneous Localization and Mapping with the Kinect sensor. In: Proceedings of ROBOTIK: 7th German Conference on Robotics. pp. 1-6.

[2] A. Okubo, A. Nishikawa, and F. Miyazaki. 1997. Selective reconstruction of a 3-D scene with an active stereo vision system. In: IEEE International Conference on Robotics and Automation. Vol. 1. pp. 751-758.

[3] E. P. Bonnal. 2011. 3D Mapping of indoor environments using RGB-D Kinect camera for robotic mobile application. Control and Computer Engineering. Politecnico Di Torino. Master Thesis.

[4] How It Works: Xbox Kinect. Available: http://www.jameco.com/jameco/workshop/howitworks/xboxkinect.

[5] H. Jungong, S. Ling, X. Dong, and J. Shotton. 2013. Enhanced Computer Vision With Microsoft Kinect Sensor: A Review. IEEE Transactions on Cybernetics. Vol. 43. pp. 1318-1334.

[6] Y. Wan, J. Wang, J. Hu, T. Song, Y. Bai, and Z. Ji. 2012. A Study in 3DReconstruction Using Kinect Sensor. In: 8th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM). pp. 1-7.

[7] Natalia Neverova, Damien Muselet, A. Trémeau. 2013. $2^1/_2$D scene reconstruction of indoor scenes from single RGB-D images. S. Tominaga, R. Schettini and A. Trémeau (Eds). pp. 281-295. Springer-Verlag Berlin Heidelberg.

[8] MathWorks. Available: http://www.mathworks.com/

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.


[9] N. Neverova, D. Muselet, and A. Trémeau. 2012. Lighting Estimation in Indoor Environments from Low-Quality Images. Computer Vision – ECCV 2012. Workshops and Demonstrations. A. Fusiello, V. Murino, and R. Cucchiara (Eds). Springer Berlin Heidelberg. Vol. 7584. pp. 380-389.

[10] K. Khoshelham and S. O. Elberink. 2012. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications. Sensors. Vol. 12, pp. 1437-1454.