Localization System using Microsoft Kinect for Indoor Structures*)

Yuichi TAMURA, Yuki TAKABATAKE, Naoya KASHIMA and Tomohiro UMETANI

Konan University, Kobe 658-8501, Japan

(Received 9 December 2011 / Accepted 7 March 2012)

This paper proposes a localization system using the Microsoft Kinect sensor. It is difficult to accurately measure self-position and self-posture by a red-green-blue (RGB) image sensor, since the accuracy in the depth direction of an image sensor tends to be worse than that in other directions using only an RGB image sensor. The Kinect sensor has both an RGB image sensor and a depth sensor. This sensor can directly calculate the depth value; therefore, the accuracy in the depth direction is better. In the proposed system, natural feature points are detected from an image by a Harris interest operator, and the depth data of these feature points are obtained from the depth sensor. After that, these points are tracked by the template matching method. The camera's position and posture are calculated from the position of these tracked points. Finally, we provide examples of 3D scene reconstruction and estimation results.

Keywords: Kinect, self localization, machine vision, depth sensor, virtual reality

DOI: 10.1585/pfr.7.2406036

1. Introduction

It is useful to provide current location information to a user and many services are provided, for example, in a car navigation system. In these systems, the global positioning system (GPS) is generally used to obtain spatial information. However, GPS is not available inside buildings, factories and power plants. If the localization system can be used indoors, suitable information, such as maintenance manuals, the name, and roles of the device, can be shown to the workers or autonomous vehicle robots can be used. Many studies employing an indoor localization system have been conducted. For example, the indoor messaging system (IMES), the radio frequency identification (RFID) system, and ZigBee were proposed. However, these systems require special equipment. Therefore, localization systems using red-green-blue (RGB) camera(s) have been proposed. These vision-based systems are also classified as a system with (e.g., [1, 2]) and without artificial markers (e.g., [3–5]). The system with artificial markers has high accuracy, but they ruin the scenery. In contrast, the system without markers preserves the scenery, but do not have high accuracy, especially in the depth direction.

To overcome this problem, we propose a localization system using a Microsoft Kinect sensor (Kinect), which has both an RGB image sensor and a depth sensor. Using this sensor, the accuracy in the depth direction can be improved.

2. Localization System

2.1 Preprocessing

Before processing, three preprocessing stages are performed for accuracy. First, the depth value is calibrated. The depth data from the depth sensor of Kinect has a depth misalignment against actual depth data (Fig. 1). This calibration is done only in the center of the RGB image sensor, since the variation of the depth value on the depth sensor is smaller than the measuring accuracy of the depth sensor. Therefore, the depth sensor calibration is conducted using a quadratic equation.

Second, the RGB and depth images are calibrated, since the angle of view of the RGB sensor is wider than that of the depth sensor. Finally, the camera parameters are calibrated. We use Zhang's method [6] to calibrate the camera intrinsic matrix, which consists of camera-specific parameters (e.g., focal length, distortion).

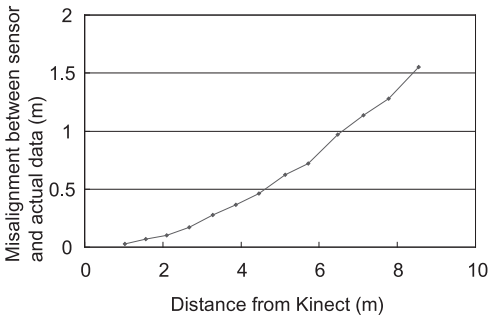

Fig. 1 Misalignment calibration.

2.2 Processing procedure

Our localization system is based on natural feature tracking. Figure 2 depicts the system procedure.

author's e-mail: tamura@konan-u.ac.jp

*) This article is based on the presentation at the 21st International Toki Conference (ITC21).

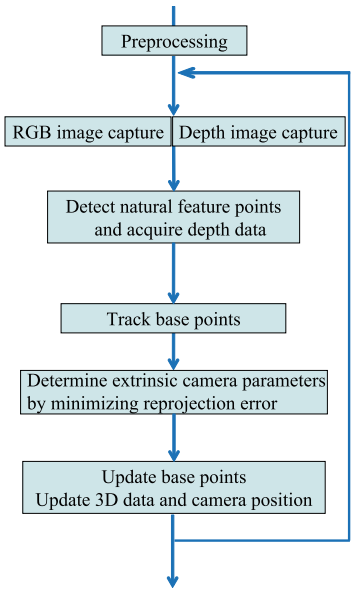

Fig. 2 Processing procedure.

For detecting natural feature points, we use the Harris interest operator [7] represented by (1), (2):

$$\mathbf{C} = \begin{bmatrix} \sum_{S(p)}(\mathrm{d}I/\mathrm{d}x)^2 & \sum_{S(p)}(\mathrm{d}I/\mathrm{d}x \cdot \mathrm{d}I/\mathrm{d}y)^2 \\ \sum_{S(p)}(\mathrm{d}I/\mathrm{d}x \cdot \mathrm{d}I/\mathrm{d}y)^2 & \sum_{S(p)}(\mathrm{d}I/\mathrm{d}y)^2 \end{bmatrix} \tag{1}$$

$$dst(x,y) = \det\mathbf{C}^{(x,y)} - k \cdot (\mathrm{trace}\mathbf{C}^{(x,y)})^2, \tag{2}$$

where x, y is the position of the pixel in the camera coordinate system, and I denotes the intensity of the pixel. $S(p)$ is a region in which Harris operator is estimated. The feature amount $dst(x,y)$ is calculated in $S(p)$ by (2). k is a parameter. If the intensity gradient of an area in an image is large, this feature amount increases. Using this operator, the corner of the image, whose intensity gradient is large, is detected. If natural feature points are detected, the depth value in the camera coordinate system can be measured from the depth sensor. The natural feature point, whose 2D position in the camera coordinate system and 3D position in the world coordinate system is calculated and known, is defined as the "base point". In the first step, the 3D positions of base points cannot be calculated; therefore, some base points, whose actual position is already known, are given. After the first step, the positions of these base points are automatically estimated. Next, the base points are tracked. In this procedure, we employ a template matching method. The estimation function of the template matching method is given by

$$E(u,v) = \sum_{j=0}^{N-1}\sum_{i=0}^{M-1}\left(I^{(u,v)}(i,j) - T(i,j)\right)^2, \tag{3}$$

where $T(i,j)$ is a template image in the previous frame, and $I^{(u,v)}(i,j)$ is a template at (u,v) in the present frame. N and M denote the template height and width, respectively. The template image is a small image cut from the area around the base point. Even if the position of the base point changes, the image around the base point in the present frame will be similar to that in the previous frame. The estimation factor E becomes large if the base point is located in the center of the area. Then, in the template area ($N \times M$ pixels) of the present frame, the Harris interest operator is applied again, and the positions of the base points are finally determined. Using this procedure, the base points are tracked across frames. Next, extrinsic RGB camera (Kinect) parameters are estimated, which include the position and posture information of the camera. The relationship between the position in the camera coordinate system and the world coordinate system is given by

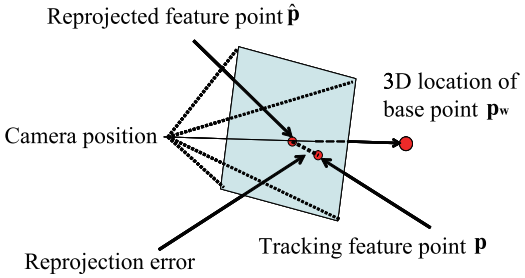

Fig. 3 Reprojection of extrinsic camera parameters.

$$\mathbf{p_c} = \mathbf{A}[\mathbf{R} \quad \mathbf{t}]\mathbf{p_w}, \tag{4}$$

where $\mathbf{p_c} = [u,v,1]^T$ is the position of the natural feature point in the camera coordinate system, and $\mathbf{p_w} = [X,Y,Z,1]^T$ is one of the base points in the world coordinate system using a homogeneous coordinate system. $\mathbf{A}$ is a 3×3 camera intrinsic matrix calculated by Zhang's method. $[\mathbf{R} \quad \mathbf{t}]$ is an extrinsic camera matrix, which consists of rotation $\mathbf{R}$ and translation $\mathbf{t}$ components and a 3×4 matrix. This extrinsic camera parameter is optimized by minimizing the factor R in the following function:

$$R = \sum_{i=1}^{L} \|\mathbf{p_i} - \hat{\mathbf{p}}_\mathbf{i}\|^2, \tag{5}$$

where $\mathbf{p_i}$ is the position of a natural feature point in the camera coordinate system, and $\hat{\mathbf{p}}_\mathbf{i}$ is a reprojected point of a base point. L is the number of detected base points. Figure 3 illustrates this procedure.

Finally, if a base point is out of the frame, this point data is deleted from the list of base points. On the other hand, if a new natural feature point is detected, this point is appended to the list.

2.3 System configuration

This system consists of one Kinect and one PC. These devices are connected via a USB. Figure 4 shows the configuration of Kinect. Kinect has 2 sensors: an RGB sensor and a depth sensor, and an infrared projector that emitts an infrared image pattern. The depth sensor acquires an infrared image and calculates the depth values of pixels from

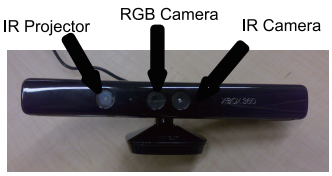

Fig. 4 Microsoft Kinect sensor.

the pattern size and distortion. In this procedure, Kinect can measure the 3D positions of all pixels in the image sensor. In this study, we use OpenNI[1], which is a library for programming natural interaction, and OpenCV[2], which is a library for computer vision.

3. Results and Discussions

In this section, two experimental results are shown. First is the rotational estimation in a room, while the second is translational estimation in a corridor.

3.1 Rotational test

The purpose of this test is to estimate rotational accuracy. If the Kinect sensor does not move and only rotates, Kinect will be back to the same position after a 360° rotation. In this test, Kinect was located at the origin in Fig. 5 and rotated on a turntable. The initial direction of Kinect is negative z-direction, and the initial position is about 1.4 m away from a wall (white board). Then, Kinect is rotated counter-clockwise. The room size is about 7.5 m × 6.5 m. In this estimation, the template size is 20 × 20 pixels, and the seek area (N and M in eq. 3) is 50 pixels. The maximum number of tracking base points is 50. The reconstruction result is shown by a point cloud in Fig. 5. These estimations were repeated 10 times. The average of the measurement error is 25 cm in the x-direction, 17 cm in the y-direction, and 39 cm in the z-direction. The root mean square (RMS) of the error is also 35 cm in the x-direction, 24 cm in the y-direction, and 53 cm in the z-direction.

3.2 Translational test

The purpose of this test is to estimate translational accuracy. In this test, the Kinect sensor is located on a mobile robot that moves along a corridor. Figure 6 shows the result of the estimation in the corridor. The width of the corridor is about 2.4 m. The template size is also 20×20 pixels, and the seek area is 50 pixels. If Kinect accurately detects base points, the shape of the corridor must be straight; however, the result is not straight. The angular misalignment is about 2.7° before reaching the stairs, after which, it is about 10.4°. Next, the accuracy in the y-direction is estimated. The height of Kinect is constant in this test; thus, the value in the y-direction is ideally constant. The result of this estimation is that the RMS in the y-direction is 14 cm.

[1]http://www.openni.org/
[2]http://opencv.willowgarage.com/wiki/

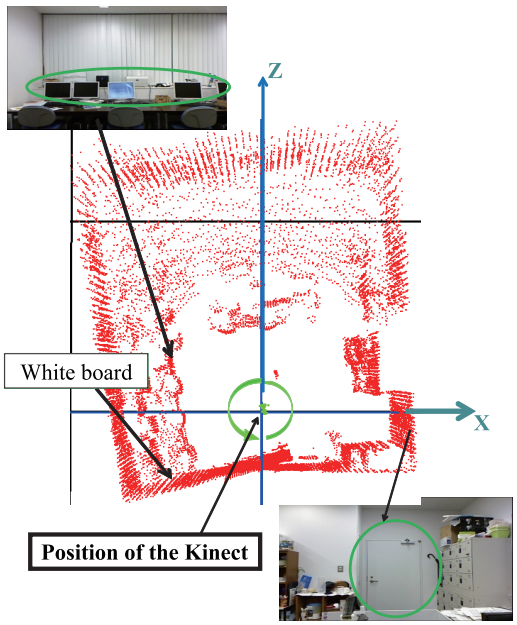

Fig. 5 Rotational result: The Kinect sensor is located at the origin and rotated at 360° on a turntable. The points show the result of the reconstruction.

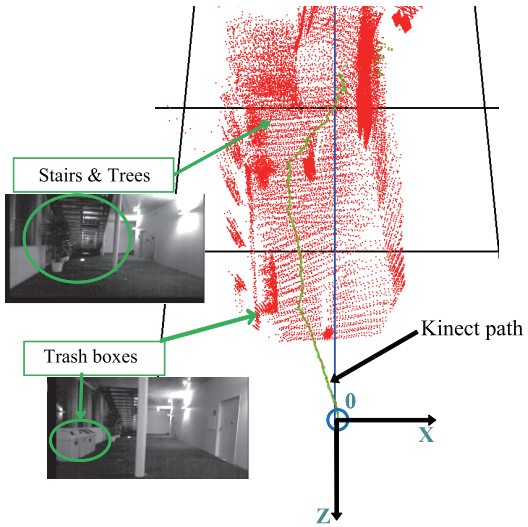

Fig. 6 Translational result: At first, the Kinect sensor is located at the origin and moved along a corridor on a mobile robot. The points show the result of reconstruction, and the line shows the estimation result of the camera (the Kinect) path.

3.3 Discussions

From both estimation results, the accuracy of the proposed system is approximately several dozen centimeters in all directions. It may be improved by accurate calibration between an RGB image sensor and a depth image sensor with preprocessing. For example, if the calibration per pixels is performed in the image plane, the accuracy improves; however, it is difficult to do this and the process consumes a lot of time. In a rotational test, the recon-

structed shape of the white board (Fig. 5) is discontinuous. The accumulating measurement error causes this false result. However, this indicates that if the camera path is assumed to be a closed loop and is optimized by this information, the accuracy will be higher. In a translational test, the shape of the corridor is suddenly curved. This is caused by incorrect recognitions. Natural feature points detected in the corridor are less than those detected in the room. The accuracy depends on the number of natural feature points and the reliability of those points. As the first step in this paper, we used base points, whose absolute position in the world coordinate system is known. If some base points, whose positions are absolutely known, are assigned in the experimental environment, the accuracy can be improved further. However, depending on the environment, it may be impossible to do so and may be quite time consuming.

4. Conclusion

A localization system using a Kinect sensor is proposed. In this system, the natural feature points are detected using an RGB image sensor, and the depth value of these feature points is measured by a depth image sensor. Finally, the position and posture of Kinect are calculated from these 3D position data. Using the depth sensor, the position and posture can be detected more easily and more accurately by the system than by that having only an RGB image sensor.

Acknowledgment

This study was partly funded by a Grant-in-Aid for Scientific Research KAKENHI (22500114) and MEXT, Japan.

[1] L. Naimark and E. Foxlin, Proc. IEEE/ACM Int. Symp. on Mixed and Augmented Reality (2002) p.27.
[2] S. Saito, A. Hiyama, T. Tanikawa and M. Hirose, Proc. IEEE Virtual Reality (2007) p.67.
[3] V. Lepetit, L. Vacchetti, D. Thalmann and P. Fua, Proc. Int. Symp. on Mixed and Augmented Reality (2003) p.93.
[4] I. Gordon and D.G. Lowe, Proc. Int. Symp. on Mixed and Augmented Reality (2004) p.110.
[5] M. Oe, T. Sato and N. Yokoya, Proc. 14th Scandinavian Conf. on Image Analysis (2005) p.171.
[6] Z. Zhang, IEEE Transactions on Pattern Analysis and Machine Intelligence **11**, 22 (2000).
[7] C. Harris and M. Stephens, Proc. 4th Alvey Vision Conf. (1998) p.147.