(Room 107) 22 June 2016, 16:00 - 16:30 PM

Analysis of 3D Reconstruction System Using Hand-held RGB-D Camera

Yo-Sung Ho[1]
[1]*School of Information and Communications, Gwangju Institute of Science and Technology (GIST), Gwangju, Republic of Korea*

Abstract

3D reconstruction is the process of capturing the shape and appearance of a real scene. There are several categories such as active, passive, rigid, non-rigid, tuntable, hand-object intraction and surrounding capture. It can be applied to the various fields such as E-commerce, virtual reality, augmented reality, human-computer interaction and so on [1].
Currently, hand-held RGB-D cameras such as Microsoft Kinect, Asus Xtion Pro, Occipital Structure sensor, Google Tango and Intel Realsense are widely available at reasonable prices. Therefore, many users can easily generate their own 3D content using 3D reconstruction.
In this paper, we represent a framework and evaluation of a 3D reconstruction system using a hand-held RGB-D camera. The overall procedure of the 3D reconstruction system is shown in Fig. 1.

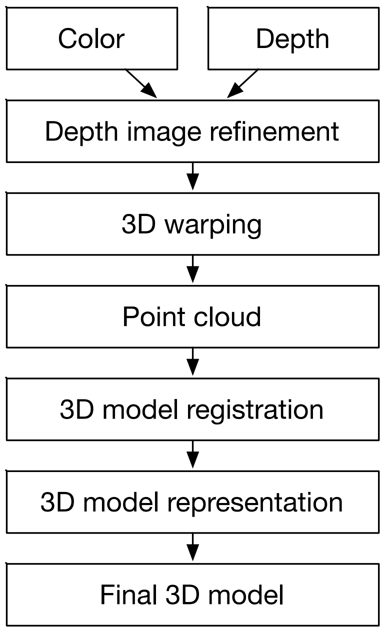

Fig. 1. Procedure of the 3D reconstruction system

First, we acquire color and depth images from an RGB-D camera and refine the raw depth image using a joint bilateral filter to fill the hole regions. After depth image refinement, we generate a point cloud by 3D warping and calculate a transformation (rotation and translation) for a pair of frames in the temporal domain using the iterative closest points (ICP) algorithm.
The ICP is the core process in the 3D reconstruction system since the accuracy of the transformation determines the quality of the reconstructed 3D model. There are several conventional methods such as point-to-point and point-to-plane methods [2].
Representation of the 3D model in the virtual space is also an important issue. Each frame from the RGB-D camera contains roughly valid 250,000 points in case of 640×480 images. For example, if there are depth frames lasting one minute, 450,000,000 points would be required; 250,000 point×30 fps×60 sec. = 450,000,000 points/min. Therefore, it is necessary to use a more concise representation of the model than that. There are several breakthroughs such as a

truncated signed distance function, surfels representation and octomap [3-5]. Finally, we can obtain the reconstructed 3D model which we can observe at arbitrary viewpoints.
There are two main measures to evaluate the accuracy of the 3D reconstruction system: relative pose error (RTE) and absolute trajectory error (ATE) [6]. The RTE measures the local accuracy of the trajectory over a fixed time interval, therefore, it is more suitable for evaluating a visual odmetry stystem. The ATE calculates the global consistency by comparing the absolute distances between the estimated and the ground truth trajectory, thus, it is useful for for simultaneous localization and mapping systems. Nevertheless, such two metrics are strongly correlated and both measures internally exploit the root mean squared error.
Figure 2 shows the 3D reconstructed model using a part of methods explained above from color and depth images. We used various open source library such as opencv, opengl and libicp [7]. However, there are drift problem which is not exactly matched to the actual 3D model because of the accumulated error generated as time goes by. Therefore, we will additionally apply a bundle adjustment to this project to adjust the camera parameters obtained from ICP [8].

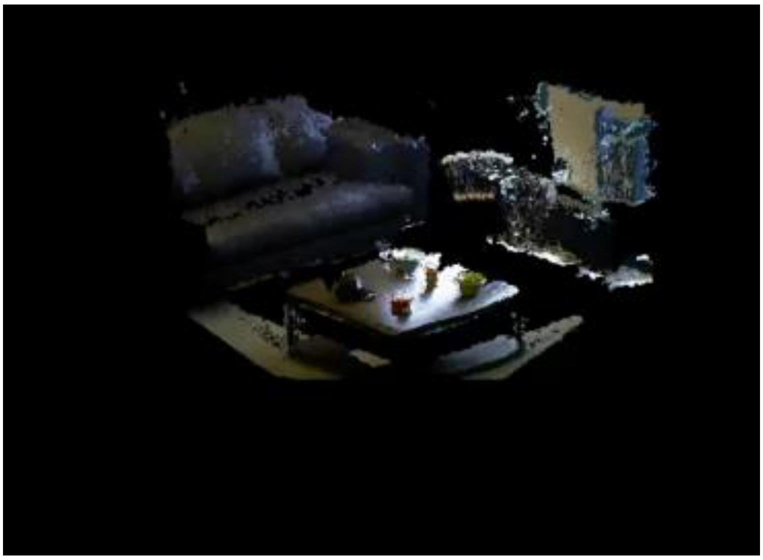

Fig. 2. The 3D reconstructed model

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (2011-0030079)

References

1. Dong-Won Shin, Yo-Sung Ho, "Implementation of 3D Object Reconstruction using a Pair of Kinect Cameras," Asia-Pacific Signal and Information Processing Association (APSIPA), vol. 10, no. 4, pp. FA1-5.5(1-4), Dec. 2014.
2. K. S. Arun, T. S. Huang, and S. D. Blostein, “Least-Squares Fitting of Two 3-D Point Sets,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, no. 5, pp. 698–700, Jan. 1987.
3. B. Curless and M. Levoy, “A Volumetric Method for Building Complex Models from Range Images,” SIGGRAPH, pp. 303–312, 1996.
4. H. Pfister, M. Zwicker, J. van Baar, and M. H. Gross, “Surfels: surface elements as rendering primitives,” SIGGRAPH, pp. 335–342, 2000.
5. K. M. Wurm and A. H, “OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems,” International Conference on Robotics and Automation, 2010.

6. J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, “A benchmark for the evaluation of RGB-D SLAM systems,” International Conference on Intelligent Robots and Systems, pp. 573–580, 2012.
7. http://www.cvlibs.net/software/libicp/
8. http://ceres-solver.org/