ICP WITH DEPTH COMPENSATION FOR CALIBRATION OF MULTIPLE TOF SENSORS

*Norishige Fukushima**

Nagoya Institute of Technology, Japan

ABSTRACT

We propose an iterative closest point (ICP) based calibration for time of flight (ToF) multiple depth sensors. For the multiple sensor calibrations, we usually use 2D patterns calibration with IR images. The depth sensor output depends on calibration parameters at a factory; thus, the re-calibration must include gaps from the calibration in the factory. Therefore, we use direct correspondences among depth values, and the calibrating extrinsic parameters by using ICP. Usually, simultaneous localization and mapping (SLAM) uses ICP, such as KinectFusion. The case of multiple sensor calibrations, however, is harder than the SLAM case. In this case, the distance between cameras is too far to apply ICP. Therefore, we modify the ICP based calibration for multiple sensors. The proposed method uses specific calibration objects to enforce the matching ability among sensors. Also, we proposed a compensation method for ToF depth map distortions.

Index Terms — Multiple RGB-D camera, Multiple Kinect, Calibration, ToF, ICP

1. INTRODUCTION

Depth sensors are essential devices for 3D applications. In the applications, extrinsic camera calibration is essential to utilize multiple depth sensors for real-time 3D modeling and 3D video, and free viewpoint TV [1]. Usual cases of camera calibrations of RGB images use a 2D image pattern, which has known geometry, and then the relationship among points in the 2D image and its 3D coordinates introduces the calibration results [2]. Depth sensors, which include time of flight (ToF) [3, 4] and structured light [5, 6], have infrared (IR) cameras; thus, we can also calibrate the sensors by the 2D image pattern on the IR images. Besides, RGB-D sensors, e.g., Microsoft Kinect, Kinect V2, Intel RealSense, and so forth, also have RGB sensors; hence, we can calibrate the sensors by the usual camera calibration in RGB images.

Calibration accuracy is an issue in the depth sensor calibration with 2D patterns. There are stitching gaps in transformed 3D models from the multiple sensors with the calibration results due to depth map distortions. In particular, depth maps of ToF depth sensors have several distortions, such as lens distortion, vignetting, sensor's heating-time, wavelength sensitivity of IR sensor, relative positions of IR emitter and camera, and wiggling [7, 8, 9]. These distortions generate spatial and temporal errors e on the true depth value z for the capturing depth value z_{depth};

$$z_{depth} = z + e(i, j, z, t), \tag{1}$$

where i, j are pixel positions and t is time from the sensor startup. With such distortions, it is hard to recover 3D geometry from the usual the number of image corresponding points. We require tremendous corresponding relations for this problem [10]. Fortunately, tendencies of the distortions are similar in multiple sensors; thus, direct 3D corresponding, such as iterative closest point (ICP) [11, 12], minimizes the gaps straightforwardly.

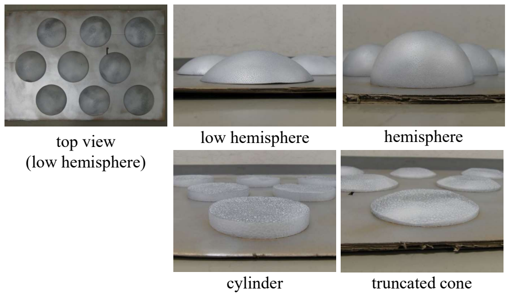

Figure 1. Calibration objects for ICP. Radius of all object is 15 cm. Height of low hemisphere is 3 cm. One of hemisphere is 7.5 cm. One of cylinder is 2 cm. One of truncated cone is 2 cm. Upper-side radius of truncated cone is 10 cm. 9 objects are put on a plane object (60 cm $\times$ 80 cm).

Simultaneous localization and mapping (SLAM) with depth maps utilizes ICP in applications, such as Kinect Fusion [13]. These applications reconstruct 3D models from sequences of depth maps obtained by a single depth sensor. In this condition, the system aligns the depth maps captured around near location by ICP. In the condition of multiple depth sensors, registering depth maps are located at far positions because of a limitation of the number of sensors and interference [14] among IR output of sensors. The condition causes tremendous miss-matching; therefore, the SLAM approach fails registration between point clouds from depth maps for the calibration of the multi-depth sensors. The paper [15] resolves this problem by using a rough alignment method for ICP initialization, but the method has remaining gaps.

In this paper, we propose an ICP based calibration for ToF sensors (Kinect V2). The main contributions of this paper are;

- demonstrating convenient calibration objects for the ICP based calibration,
- showing effective ICP steps for largely distant ToF sensors,
- proposing a depth correction method for the distortion of ToF sensors by combining ICP based calibration with a 2D pattern based one.

2. PROPOSED METHOD

2.1. ICP for registration

The proposed method utilizes specific calibration objects for ICP registration. Figure 1 shows various shapes of the calibration objects. Essential points of the calibration objects are:

1. containing much 3D features or unevenness,
2. suppressing occlusions,
3. having easiness to trim the objects.

Points 1 and 2 have a trade-off relationship; therefore, we prepare four types, which have different height and surface.

In our calibration procedure, we capture a depth map and transform the depth map to point clouds, and then we trim and

*This work was supported by JSPS KAKENHI (JP17H01764).

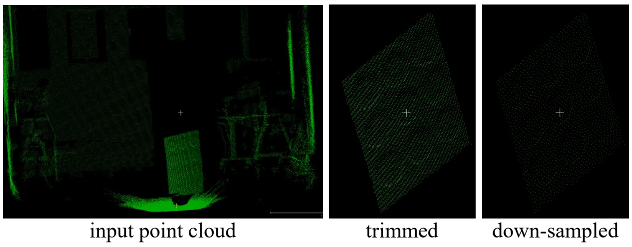

Figure 2. Calibration objects for proposed ICP steps.

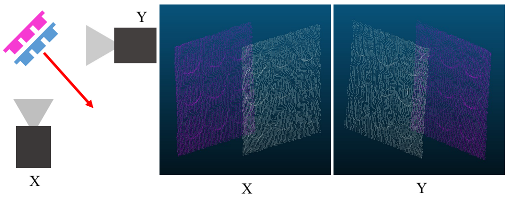

Figure 3. Combining multiple calibration objects.

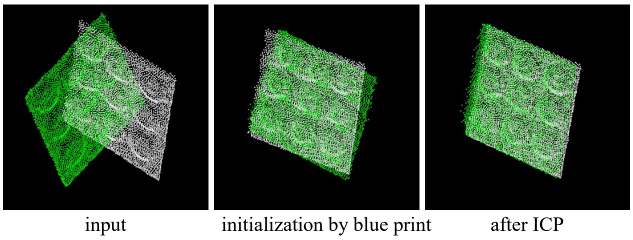

Figure 4. Flow of proposed ICP steps.

down-sample the point clouds (Fig. 2). Next, we iterate capturing and combine multiple point clouds of the calibration objects, which have different positions and orientations (Fig. 3). Finally, we perform our ICP steps. At first, we initialize rotation and translation parameters by using a blueprint of sensors setups. For example, we set 90-degree for rotation and 2-meter radius between two sensors, in this paper. Then, we perform ICP, which is the sparse ICP with point-to-plane matching [12] and is robust to noises. The point-to-plane based ICP requires normal vectors, which are the first order of local features and tend to be noisy; thus, we smooth the point clouds by 3D Gaussian filtering before ICP. The simple filtering had better performance than the edge-preserving refinement filtering [16]. Figure 4 shows the ICP steps.

The proposed method requires specific calibration objects, and also the conventional 2D pattern based calibration requires specific objects. The significant difference between the objects is the accuracy of forming the objects. For 2D pattern object, we need precise printing on hard materials for obtaining certain relationships between the printing object position and imaging position. On the contrary, our calibration patterns do not require such accuracy, because we utilize only correspondences between depth maps. Therefore, we roughly handcraft our calibration objects with cardboards and Styrofoam-objects.

2.2. Correcting distortion of depth map

Depth sensors have distortions in depth maps, and these 3D coordinates also have distortions. ICP directly registers each depth map; thus, the method minimizes these errors by adjusting rotation and translation parameters. For more accurate calibrations, however, removing the depth map distortions is essential. We propose z-compensation to cover following error factors; lens distortion, the relative position of IR emitter and camera, and heating time.

For time-dependent distortion, we had 30-minutes pre-heating time to obtain stable depth maps [7]. For lens distortion, we re-

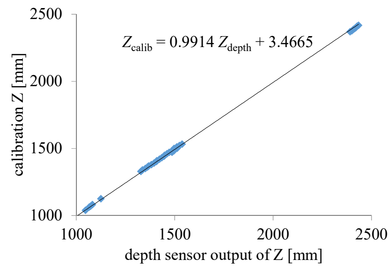

Figure 5. Linear distortion of depth map.

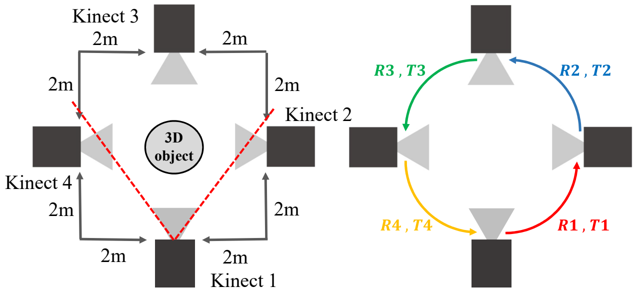

Figure 6. Sensor setup.

calibrate intrinsic parameters by the 2D patterns approach on IR images. We do not use factory provided intrinsic parameters.

Also, the causes of distortions, which are the error of the relative position of IR emitter and camera, and wiggling, generate biases of depth sensor outputs. The critical factor is a twist of calibration coordinates between factory and current calibrations. When we obtain depth maps from an official library, the depth values of the depth map, Z_{depth}, is related to the extrinsic parameter at factory estimation. To obtain the same extrinsic parameters, we need identical objects at identical locations with identical noisy IR images. Reproducing these conditions are impossible; thus, the parameters have slight differences. For estimating the difference, we calibrate intrinsic and extrinsic parameters of a sensor by using the 2D image patterns. With this process, we can obtain 3D coordinates on corners of the 2D image patterns, and also we can obtain depth values from the captured depth map co-located on the calibrating images. Figure 5 shows the difference between depth values on corners computed by the calibration parameter Z_{calib} and Z_{depth}. We can find that the difference has almost linear relationship;

$$Z_{calib} = aZ_{depth} + b. \tag{2}$$

After this calibration, we can utilize the z-compensation parameters of a and b. We correct Z_{depth} values by multiply-and-add with these parameters. The actual distortion has the per-pixel dependency, although, the z-compensation removes the significant errors. The remaining notable distortion is the vignetting-like distortion that is depth values are smaller near boundaries of images. Recovering this distortion is hard; thus, we should discard boundary pixels with some threshold, e.g., 50-pixels.

3. EXPERIMENTAL RESULTS

In our experiment, we surrounded an object with four sensors. Figure 6 shows sensors configuration for experiments. All sensors were located at 90-degree intervals with 2-meter radius. We utilized Kinect V2 as depth sensors.

At first, we compared three approaches; the 2D patterns method, the proposed ICP steps with a non-calibration object (human body), and the proposed ICP steps with our calibration objects. Figure 7 shows each calibration object. In this experiment, we measured re-

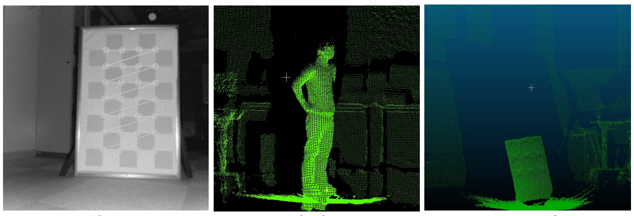

Figure 7. Comparing calibration patterns.

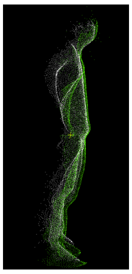

Figure 8. Registration result of body calibration object. Green and white point clouds are captured by different sensors.

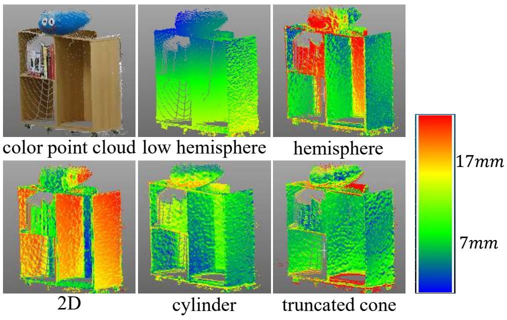

Figure 9. Heat map of registration results.

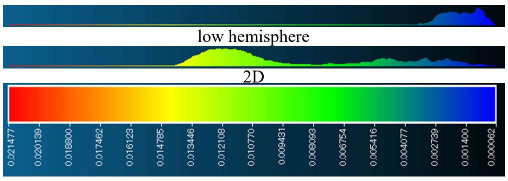

Figure 10. Histogram of registration results.

Table 1. Reprojection error of each calibration pattern. The low hemisphere case has two result, which the lower is twice pattern capturing case, and the higher is once capturing case. The other proposed method is once capturing case.

2D	Body	low hemisphere	hemisphere	cylinder	truncated cone
0.019	0.99	0.011/0.008	0.019	0.012	0.016

projection errors between input point clouds and one around projected point clouds with root mean square error (RMSE).

$$\mathrm{RMSE} = \|\boldsymbol{p} - \boldsymbol{P}_4\boldsymbol{P}_3\boldsymbol{P}_2\boldsymbol{P}_1\boldsymbol{p}\|_2, \tag{3}$$

where $\boldsymbol{p}$ is a vector, which contains 3D coordinates of point clouds, and $\boldsymbol{P}_n = \boldsymbol{R}_n;\boldsymbol{T}_n (n = 1, 2, 3, 4)$ are projection matrices, which are composed from a rotation matrix R_n and a translation vector T_n. $\|\cdot\|$ indicates L2 norm. Note that we used the same point clouds $\boldsymbol{p}$ for evaluations of each calibration method. The point clouds are trimmed subjects, such as dolls, tables, and shelves.

Table 1 shows reprojection errors of each calibration object. The proposed method with the proposed calibration object (low hemisphere) has the best performance. In particular, lower height objects have better performance because higher object generates

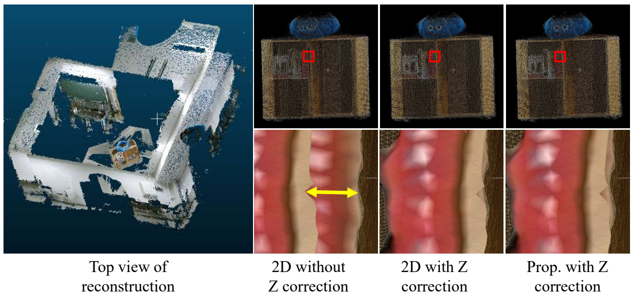

Figure 11. Reconstruction models.

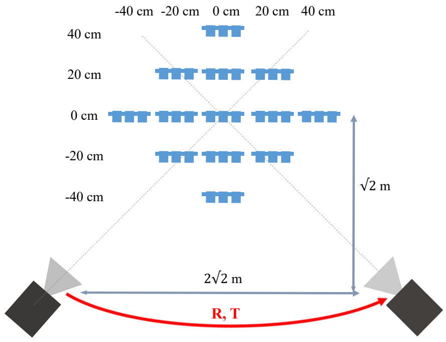

Figure 12. Experimental setup for estimating location dependency.

Table 2. 2D method: Location dependency of RMSE $\times 10^{-4}$. Summations of errors with/without z-compensation are 50.5 and 37.3.

	-40	-20	0	20	-40
40			3.5		
20		4.0	4.6	3.3	
0	4.3	4.2	4.5	3.7	4.0
-20		3.5	3.8	3.8	
-40			3.3		

	-40	-20	0	20	-40
40			2.8		
20		3.0	2.8	2.5	
0	3.3	3.0	3.3	2.7	3.2
-20		2.3	3.1	3.0	
-40			2.3		

Table 3. Proposed method: Location dependency of RMSE $\times 10^{-4}$. Summations of errors with/without z-compensation are 21.4 and 21.1.

	-40	-20	0	20	-40
40			2.5		
20		1.5	1.6	1.4	
0	2.2	1.2	1.2	1.3	1.8
-20		1.5	1.8	1.6	
-40			1.8		

	-40	-20	0	20	-40
40			1.6		
20		1.5	2.1	1.6	
0	1.8	1.5	1.4	1.5	1.5
-20		1.8	1.7	1.5	
-40			1.6		

much-occluding part. Also, the curved surface object is better performance because the object has much entropy for registration. The planer surface object has ambiguity in matching. The human body object, however, cannot achieve better performance than the 2D object. Figure 8 shows the difference between point clouds captured by two sensors for the body pattern. This pattern has significant gaps. Although the human body is easy to setup for calibration, the accuracy of this ICP based calibration is low. This fact shows that the ICP based calibration requires some simple calibration objects. Figure 9 shows various heat maps of each object excepting for the body object. The low hemisphere pattern suppresses overall errors. Figure 11 shows reconstruction models from the calibration results of 2D with/without z-compensation and the proposed method with z-compensation. For modeling, z-compensation is essential. Without z-compensation, the 2D approach generates a gap in the ocher-color texture part. The difference between the 2D with z-compensation and the proposed looks small; however, we can find the difference easily from 3D rendering results with different viewpoints.

The next experiments indicate positional dependences of the proposed and competitive method. Figure 12 shows the experimental condition. Firstly, we calibrated the two sensors by the patterns at the center of the 5 grid positions. Secondly, we located the pattern around the center position and measured the projection error between point clouds the around sensor by using the parameters estimated at the center position. Table 2 and 3 show

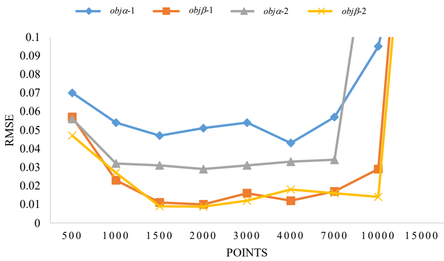

Figure 13. Relation between the number of point cloud and RMSE. obj$\alpha - n$ and $\beta - n$ represent difference 3D calibration patterns, and $-n$ represents the number of capturing times.

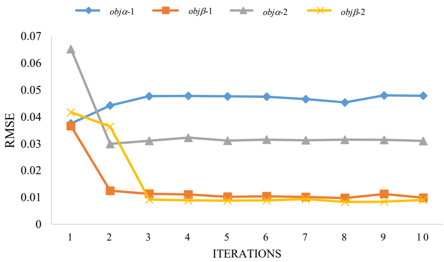

Figure 14. Relation between the number of iterations and RMSE. obj$\alpha - n$ and $\beta - n$ represent difference 3D calibration patterns, and $-n$ represents the number of capturing times.

the positional dependences in RMSE of the 2D pattern and proposed method with/without the z-compensation. Note that we use the 3D calibration object for measuring the RMSE; thus, the error values are lower than the Table 1. The 2D method does not minimize the error at the center, and the z-compensation works well. The proposed method minimizes the error at the center, while the z-compensation boost the error around the center. The total error, however, is reduced by the z-compensation for the proposed method. The proposed z-compensation compensates the error by global parameters of a, b for easiness, but actual errors have per-pixel distortion. The amplitude of the distortion have convex shape around the center of imaging sensor; thus, the z-compensation suppresses the global error, but amplitudes the specific position, i.e., the center position.

The following are detail experiments for setting parameters of the proposed method. Figure 13 shows the relation between the number of point clouds and RMSE. We can find that 1500-4000 points are suitable. In twice capturing case, we should more massively down-sample point clouds. The tremendous point clouds case, i.e., over 7000 points case, the registration generates much miss-matches and ICP does not ensure global minimum solutions; hence, the accuracy becomes low. Note that we had tested for three and four times capturing cases; however, the twice time approach was the peak of our method. Figure 14 shows the relation between the number of iterations and RMSE. We can find that 2 or 3 are enough iterations. Note that objα-1 does not have enough feature points to converge the results. Unlike the usual ICP, the proposed method initializes the position and rotation parameters from the blueprint. This process reduces the number of iterations.

4. CONCLUSION

In this paper, we propose an ICP based calibration method for multiple depth sensors with z-compensation. Experimental results show that the proposed method has higher accuracy than the 2D pattern's calibration approaches.

I thank Yoshiki Kusatsugu for his help of experiments.

5. REFERENCES

[1] Y. Mori, N. Fukushima, T. Fujii, and M. Tanimoto, "View generation with 3d warping using depth information for ftv," *Signal Processing: Image Communication*, vol. 24(1-2), pp. 65–72, 2009.

[2] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22(11), pp. 1330–1334, 2000.

[3] M. Reynolds, J. Doboš, L. Peel, T. Weyrich, and G. Brostow, "Capturing time-of-flight data with confidence," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[4] M. Hansard, Seungkyu Lee, Ouk Choi, and Radu Horaud, *Time of Flight Cameras: Principles, Methods, and Applications*, SpringerBriefs in Computer Science. Springer, 2012.

[5] J. Salvi, J. Pages, and J. Batlle, "Pattern codification strategies in structured light systems," *Pattern Recognition*, vol. 37, pp. 827–849, 2004.

[6] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE MultiMedia*, vol. 19, no. 2, pp. 4–10, 2012.

[7] P. Frsattel, S. Placht, M. Balda, C. Schaller, H. Hofmann, A. Maier, and C. Riess, "A comparative error analysis of current time-of-flight sensors," *IEEE Transactions on Computational Imaging*, vol. 2, no. 1, pp. 27–41, 2016.

[8] A. Kuznetsova and B. Rosenhahn, "On calibration of a low-cost time-of-flight camera," in *Proceedings of European Conference Computer Vision (ECCV) Workshops*, 2015.

[9] O. Wasenmüller and D. Stricker, "Comparison of kinect v1 and v2 depth images in terms of accuracy and precision," in *Proceedings of Asian Conference on Computer Vision (ACCV) Workshops*, 2016.

[10] A. Louie, K. Takeuchi, N. Ito, and A. Kojima, "Multiple plane vivew camera calibration for rgb-d sensor rectification," in *IEICE Technical Report*, 2015.

[11] P. J. Besl and Neil D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[12] S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," in *Proceedings of Eurographics/ACMSIGGRAPH Symposium on Geometry Processing*, 2013, pp. 113–123.

[13] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon, "Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of Annual ACM Symposium on User Interface Software and Technology (UIST)*, 2011.

[14] H. Sarbolandi, D. Lefloch, and A. Kolb, "Kinect range sensing: Structured-light versus time-of-flight kinect," *Computer Vision and Image Understanding*, vol. 139, pp. 1 – 20, 2015.

[15] M. Kowalski, J. Naruniec, and M. Daniluk, "Livescan3d: A fast and inexpensive 3d data acquisition system for multiple kinect v2 sensors," in *Proceedings of International Conference on 3D Vision (3DV)*, 2015.

[16] T. Matsuo, N. Fukushima, and Y. Ishibashi, "Weighted joint bilateral filter with slope depth compensation filter for depth map refinement," in *Proc. International Conference on Computer Vision Theory and Applications (VISAPP)*, 2013.