Large Scale Indoor 3D Mapping Using RGB-D Sensor

Xiaoxiao Zhu[1(✉)], Qixin Cao[1], Hiroshi Yokoi[2], and Yinlai Jiang[2]

[1] State Key Lab of Mechanical Systems and Vibration, Research Institute of Robotics, Shanghai Jiao Tong University, Shanghai, China
{ttl,qxcao}@sjtu.edu.cn

[2] The University of Electro-Communications, Tokyo, Japan
{yokoi,jiang}@hi.mce.uec.ac.jp

Abstract. 3D Mapping using RBG-D sensor is a hot topic in the robotic field. This paper proposes a sub-map stitching method to build map in the large scale indoor environment. We design a special landmark, and place it in the environment. Every sub-map contains those landmarks, and then can be stitched by BA optimization. The result shows that the map error is blow 1 % in a room with the dimensions of 13 m × 8 m.

Keywords: Large scale indoor mapping · RGB-D sensor · Map stitching

1 Introduction

There are many ways to build a 3D model of the indoor environment using RGB-D sensor, such as RGBDSLAM [1, 2] SLAM6D from 3Dtk [3] or KinectFusion [4]. KinectFusion has the highest accuracy and real-time performance of those methods. RGBDSLAM usually runs at 2 FPS and will become slower when more data is captured, while SLAM6D is an offline method. To build a hole-less map, we have to capture many views and also need the real-time feedback of the current model, so the real-time performance is very important. The KinectFusion method runs at 15 fps, so we can building the map by just handing Kinect and scanning all the environment with a real-time feedback. However, the biggest disadvantage of KinectFusion is that, it can only build limited size of map, because this method is memory consuming and especially because it uses a GPU memory. Although simply reducing the resolution of the map can enlarge the mapping area, this will decrease accuracy and make the mapping process unstable. To resolved this limitation, paper [5, 6] propose a GPU-Octree data structure to replace the 3-dims array of the Truncated Signed Distance Function (TSDF) model [7], so it can reduce the demands of GPU memory and keep the accuracy. Paper [8] proposal the Kinectous algorithm, which use the GPU as a ring buffer so can recycle use the GPU. However, those methods all finish the mapping process in one try. And also requires a lot of time to complete the entire map creation process, so reduce the flexibility of the map created in the indoor complex cases difficult to practice.

In this paper, we resolved this problem by a sub-map stitching method. Since the KinectFusion algorithm suit to create the small scale map, so we just use the KinectFusion algorithm to build a sub-map with dimensions of 6 × 3 × 3 m, and the

N. Kubota et al. (Eds.): ICIRA 2016, Part I, LNAI 9834, pp. 313–321, 2016.
DOI: 10.1007/978-3-319-43506-0_27

sub-maps were then stitched into a large map. To help the stitching process, a kind of special landmark will be proposed, which is easy to be detected and localized in the sub-map. The experiment result shows good precision.

2 Procedure of Mapping

2.1 The Main Idea and the Landmark

Since the KinectFusion algorithm will produces a point cloud style sub-map, so the stitching target is the point cloud. The traditional point cloud stitching or alignment problem is solved by ICP [9] method, which is not suitable for this mapping case for three reasons:

1. The traditional ICP method need a good initial for converging. But in the mapping process, the RGB-D sensor is holding by human, so there is no odometer information, then a good initial position cannot be provided for ICP; and
2. The general ICP method is a 3D matching method, but notice that a consistent ground plane is especially important in the indoor environment; and
3. The ICP need the overlap between the two point clouds, and more overlap the more coverage the more precision. However, in the mapping case, more overlap between the sup-maps means more number of sub-map, and will cost more mapping time.

Therefore, we proposed a ground plane consistent sub-map stitching using 3D landmark. As shown in Fig. 1, the 3D landmark is made with balls because the spherical shape is easy to be detected in the point cloud and a small plane for computing the localization data. The landmarks are distinguishable by radius and height.

Fig. 1. Four ball landmarks

2.2 Layout of the Landmarks

Before the mapping process, every sub-map will be placed with four ball landmarks and every two sub-maps have two corresponding ball landmarks, as shown in Fig. 2. For least overlap between sub-maps, the landmark is place near the edge of the sub-map. And to make the recognition of the corresponding relationship easy, beside the landmark's identification, the distance between two landmarks is also used to determine the correspondence of two sub-maps. For example, in sub-map1 and sub-map2, the distance between landmark1 and landmark2 should be distinctive, so landmark3 and landmark4 are the corresponding landmarks.

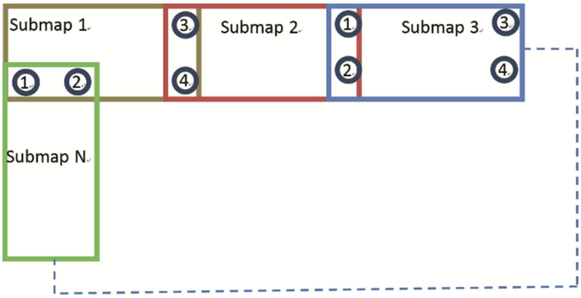


Fig. 2. Illustration of the alignment of the landmarks

2.3 Sub-map Building Using Advanced KinectFusion Algorithm

As mentioned before, the original KinectFusion method is not very stable, especially in some simple environment. So we use an advanced KinectFusion algorithm proposed in our previous work [10]. This advanced KinetFusion has two improvements of KinectFusion algorithm. On one hand, the edge feature points in the environment are matched to improve the robustness, on the other hand, a ground plane point cloud in the model is preset to improve the accuracy. The improved algorithm decreases the modeling error from 4.5 % to 1.5 % in a room of 6 m ×3 m ×3 m. Although the efficiency is influenced, the running speed of the algorithm is still very high, and the user experience during modeling is good.

An example of sub-map is shown in Fig. 3.

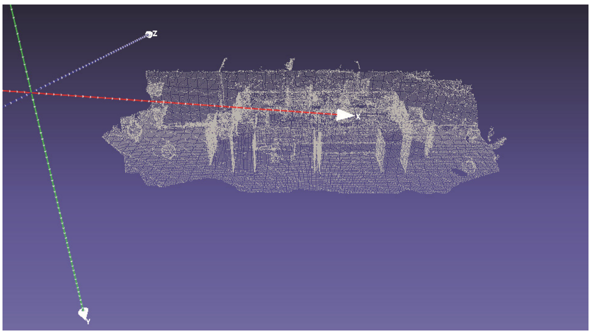


Fig. 3. An example of a sub-map

2.4 Ground Plane Extraction

The ground plane is extracted using the Random sample consensus (RANSAC) method [11], which is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers, and the threshold value is set as 2 cm. A temporary ground plane coordinate system is defined and the sub-map is transformed to the ground coordinate system. Although the ground plane coordinates of the sub-maps are not yet consistent, by doing this the dimension of the sub-map stitching problem is decreased to 2D. The transformation results are shown in Fig. 4, in which the x-y plane is coincident with the ground plane.

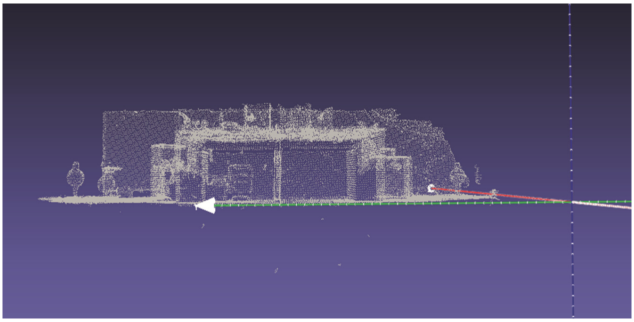


Fig. 4. The result of ground plane extraction and point cloud transformation

2.5 Ball Landmark Extraction and Corresponding Computation

Extracting the balls from the sub-map also uses the RANSAC method, but directly running the RANSAC for the sub-map will fail because the ball is small compared to the sub-map, meaning that the number of outliers is far larger than the inliers. Therefore, the ground plane is first extracted and deleted from the sub-map and the Euclidean cluster method is then used to detect those point clusters having a similar size to the ball;

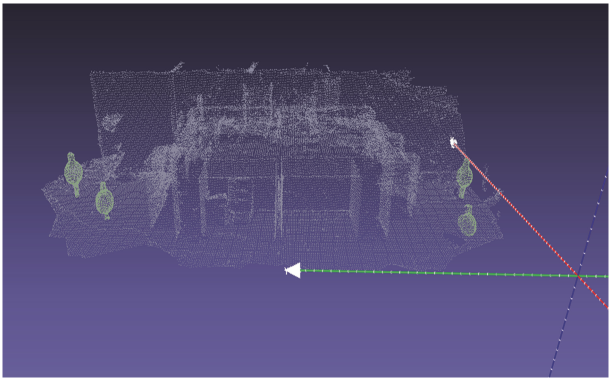


Fig. 5. The result of ball extraction

RANSAC is then run to find balls in every cluster. The result is shown in Fig. 5. After ball extraction, determining the correspondences by ball radius and height and ball-ball distance is straightforward.

2.6 Computation of the Pairwise Transformation

In this step, we intend to compute the transformation between two adjacent sub-maps. There are two correspondence points (the ball centers) and a 2D transformation must be computed. The general computation method for an optimized transformation from two sets of corresponding points is reviewed by [12]. In our case, only two corresponding points are used to compute a 2D transformation, so a simpler way of solving the problem was found. First, the ball points were projected to the x-y plane as in Fig. 6(b) (where c1 and c2 are the centers of the points), then the optimized transformation is the one that makes the two centers coincide and the four points lie in one line (as shown in Fig. 6(c)). This result concerns with the result of the SVD based transform optimized method [13].

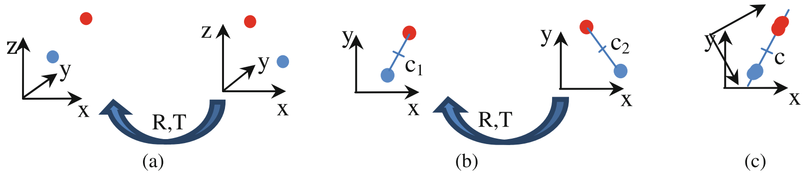


Fig. 6. (a) Illustration of the original problem (b) Projection to the x-y plane decreases the problem to 2D (c) The optimized 2D transformation

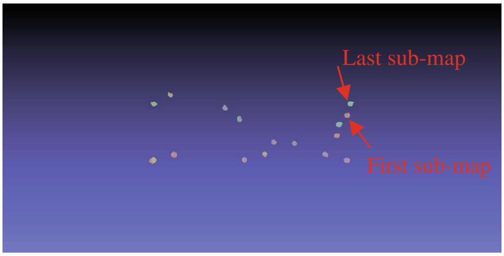
2.7 Loop Closure and BA Optimization

When building the sub-maps, the end map and the first map must be connected, giving a loop closure. Based on a good initial value from the pairwise transformation result, a BA algorithm [14] is used to optimize the transformation between those sub-maps.

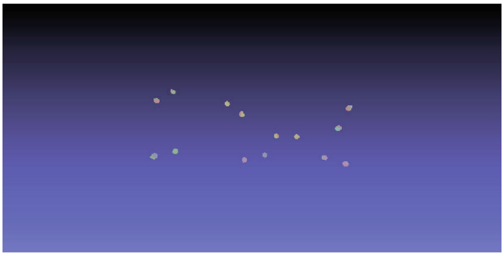
The origin BA algorithm assume there are n 3D points projected into m pictures, let x_{ij} to be the projection of point i in the image j, v_{ij} to be a binary variable, $v_{ij} = 1$ if point i can be seen in image j, otherwise $v_{ij} = 0$. And let a_j to the camera internal and external parameters, let b_i represent the 3D point i in the world. BA algorithm minims the projection error and compute the optimal coordinate of those 3D points and also the transformations between those cameras.

$$\min_{a_j,b_i} \sum_{i=1}^{n} \sum_{j=1}^{m} v_{ij} d(Q(a_j, b_i), x_{ij})^2 \tag{1}$$

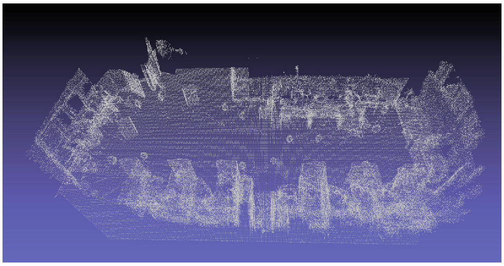
In which the Q is the prediction of the projection of the point i in image j, and d(x, y) represent the Euclidean distance between two 2D points in the image.



(a)



(b)



(c)

(d)

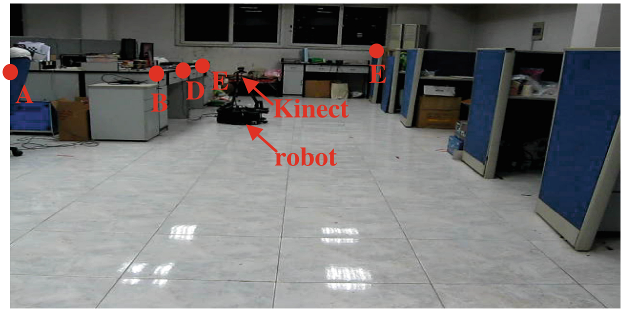
Fig. 7. (a) sub-map stitching before BA optimize (b) balls map after BA optimize (c) Final map after BA optimize (d) Final map without the landmarks

Here, in our case, the b_i is the real 2D coordinates on the ground of the ith landmark($x_i y_i$), and a_j is jth sub-map's 2d pose information include the rotation angle Theta and the transformation component T, v_{ij} can determined by the neighborhood relationship, $Q(a_j, b_i)$ represent the coordinates of landmark i respect to the sub-map j.

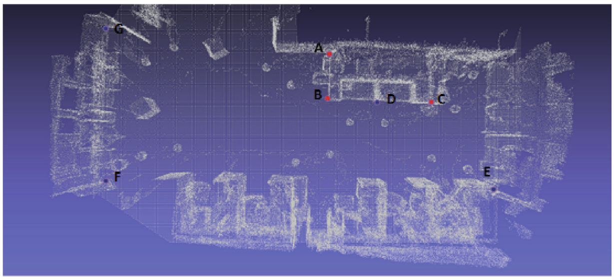
The Fig. 7(a) show the error of initial stitching result using the result of the previous section, show as Fig. 7(b), after the optimization, the ball landmarks in each sub-map are in good agreement. Figure 7(c) and (d) show the final map with and without the landmarks.

3 Experiment

This experiment mainly test the accuracy of the map built by our method. The test lab environment is show as Fig. 8(a), in which several test points were selected shown as the red points and the distance between each other were measured manually. The built 3D map is shown in Fig. 8(b), we import the map into a tool software named Meshlab and also measured the corresponding distance. Then the two types of measured distances are compared with each other, and the result is shown in Table 1. The d_{AB} and d_{BC} show the accuracy inside a sub-map, and the others show the accuracy of the sub-map stitching method. The result shows that map error is 1 % in the room with dimensions of 13 m × 8 m.



(a)



(b)

Fig. 8. (a) The test scene (b) Some points used for distance comparison

Table 1. Distance comparison result

	Real distance (cm)	Map distance (cm)	Absolute error (cm)
d_{AB}	136.5	137	0.5
d_{BC}	278	279.2	1.2
d_{DE}	400.252	394.049	6.20283
d_{DF}	811.758	812.607	0.8493
d_{DG}	822.771	816.12	6.65138
d_{EF}	1089.81	1080.58	9.22867
d_{EG}	1193.04	1181.14	11.9036

4 Conclusion

In this paper, a sub-map stitching based mapping method for building a large scale 3D indoor environment model was presented, with emphasis on the consistency of the ground plane. And by using landmark, the overlap between the sub-maps can be reduced, so make the mapping process The experimental results show that the accuracy the final 3D map is less than 1 % that in a room with dimensions of 13 m × 8 m.

Acknowledgment. We thank the support of China Postdoctoral Science Foundation, No. 2015M571561 and the National Natural Science Foundation of China, No. 61273331.

References

1. Engelhard, N., Endres, F., Hess, J., Sturm, J., Burgard, W.: Real-time 3D visual SLAM with a hand-held RGB-D camera. In: RSS 2010 Workshop on RGB-D Cameras (2010)
2. Henry, P., Krainin, M., Herbst, E., Ren, X., Fox, D.: RGB-D mapping: using depth cameras for dense 3D modeling of indoor environments. In: RSS 2010 Workshop on RGB-D Cameras (2010)
3. Slam6d. Slam6d toolkit. http://slam6d.sourceforge.net/index.html
4. Izadi, S., Newcombe, R.A., Kim, D., Hilliges, O., Molyneaux, D., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A., Fitzgibbon, A.: Kinectfusion: real-time dynamic 3D surface reconstruction and interaction. In: ACM SIGGRAPH 2011 (2011)
5. Zeng, M., Zhao, F., Zheng, J., Liu, X.: Octree-based fusion for realtime 3D reconstruction. In: Graphical Models (2012)
6. Zeng, M., Zhao, F., Zheng, J., Liu, X.: A memory-efficient KinectFusion using octree. In: Hu, S.-M., Martin, R.R. (eds.) CVM 2012. LNCS, vol. 7633, pp. 234–241. Springer, Heidelberg (2012)
7. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: Proceedings of SIGGRAPH 1996, pp. 303–312 (1996)
8. Whelan, T., Kaess, M., Fallon, M., Johannsson, H., Leonard, J., McDonald, J.: Kintinuous: spatially extended kinectfusion (2012)
9. Besl, P.J., McKay, N.D.: A Method for Registration of 3-D Shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992)

10. Zhu, X., Cao, Q., Yang, Y., et al.: An improved kinect fusion 3D reconstruction algorithm. Robot **36**(2), 129–136 (2014)
11. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
12. Eggert, D.W., Lorusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: a comparison of four major algorithms. Mach. Vis. Appl. J. **9**(5), 272–290 (1997)
13. Arun, K.S., Huang, T.S., Blostein, S.D.: Least-squares fitting of two 3-D point sets. IEEE Trans. Pattern Anal. Mach. Intell. J. **5**, 698–700 (1987)
14. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment – A modern synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) ICCV-WS 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000)