A 3D reconstruction method based on RGB-D camera

Fan Zhu a, Shaobin Li b, Xiaohan Guo c

School of Information and Communications Engineering Communication University of China Beijing, China

a874429256@qq.com, b shaobin@cuc.edu.cn, c han2724@126.com

Abstract. 3D surface reconstruction with realistic feeling is a hot research field in computer graphics. In order to reconstruct 3D models with high precision, a 3D reconstruction method is proposed in this paper. Based on the BundleFusion reconstruction algorithm, this method uses color images and depth images of indoor scenes obtained by Intel's realsenseD435i depth camera for rapid 3D reconstruction. An improved ORB feature detection algorithm is used for image feature extraction, and the efficiency of several classical feature detection algorithms in image matching is compared. Experiments show that this method can effectively generate 3D models with higher precision and better sense of reality.

Keywords: 3D reconstruction, RGBD camera, feature extraction, The ORB algorithm

1. Introduction

3D Reconstruction is the process of rebuilding a 3D model of a given object or scene based on its 3D information. It is an important research direction in computer graphics. Accurate and reliable 3D dynamic geometric reconstruction has important application value in face restoration, game and animation, virtual reality and augmented reality. Early 3D reconstruction techniques usually use 2D images as input to reconstruct 3D models of scenes. However, limited by the input data, the reconstructed 3D model is usually incomplete and less realistic. The emergence of laser scanner greatly improves the accuracy of 3D reconstruction, but the equipment is often valuable. With the popularization of depth sensor represented by Kinect depth camera, 3D reconstruction technology based on RGB-D data is constantly developing. At present, many researchers have carried out related researches on RGB-D data-based 3D reconstruction depth data processing algorithm, skeleton registration optimization, 3D reconstruction algorithm and texture mapping. The 3D reconstruction method based on RGB-D camera has low cost and simple configuration, but the depth distance acquired by the camera is limited, and the depth data is noisy at the edge of the object, which also poses challenges to 3D reconstruction.

At present, most of the 3D reconstruction methods based on depth sensor are mainly aimed at static objects. Kinect Fusion [1] algorithm is taken as the representative. This method uses ICP iteration to estimate camera pose and construct TSDF point cloud Fusion algorithm, which can better reconstruct the 3D model of static objects. This algorithm is the first to achieve real-time rigid body reconstruction based on cheap consumer cameras. It is more suitable for the reconstruction of small indoor scenes or fixed objects. To solve the problem that the reconstruction space of KinectFusion algorithm is limited, Whelan et al. made the Kintinuous[2] based on the KinectFusion algorithm, adding loopback detection...
and loopback optimization. The TSDF model can be updated following the motion of the camera for large-scale scene reconstruction. Voxel Hashing [3] came out in 2013, with an algorithm that simply splits the surface of a scene captured by the camera into voxels instead of the entire space to be rebuilt, thereby saving video memory. The algorithm is widely known for its open source, fast model update and low memory consumption. InfiniTAM[4] algorithm proposed by Oxford University is also an improvement of voxel hash algorithm. The Surfel model is another popular 3D representation model used in real-time 3D reconstruction. ElasticFusion[5-6] algorithm uses the Surfel model. This algorithm focuses on the accuracy of drawing construction and adds the prediction and display of point light sources in the environment. But you can only display the final result as a point cloud. In 2017, BundleFusion[7] proposed by Stanford University is also a very noteworthy algorithm in the field of dense 3D indoor reconstruction. The core of this method is an effective global pose optimization algorithm which works in cooperation with large-scale real-time 3D reconstruction framework. High precision 3D model can be built in real time. In the real time and the precision of the model has a good performance.

2. The BundleFusion

BundleFusion algorithm proposed by Stanford University in 2017 plays a very important role in dense 3D reconstruction based on RGB-D camera. This algorithm has remarkable performance in the level of precision and real-time modeling. The method in this paper is mainly based on the algorithm.

At the core of BundleFusion is an effective global pose optimization algorithm that works in conjunction with a large-scale real-time 3D reconstruction framework. The specific flow of the algorithm is shown in Figure 1. In each frame, the pose optimization is continuously run and reconstructed according to the newly calculated pose estimation. This approach is not strictly dependent on temporal coherence, allowing free-form camera paths, instantaneous repositioning, and frequent re-access to the same scene area. This makes our method robust to sensor occlusion, fast frame-to-frame motion and featureless region.

This method takes the RGB-D stream collected by a commercial depth camera as input. To achieve global alignment, perform a sparse and then dense global pose optimization: use a set of sparse feature mappings to obtain coarse global alignment, since sparse features inherently provide loop closure detection and relocation. This alignment is then optimized by optimizing density luminosity and geometric consistency. Sparse correspondence is established by pairing scale invariant feature transformation (SIFT) feature correspondence between all input frames. That is, the SIFT keyword detected will match all previous frames and carefully switch to remove the mismatch to avoid an incorrect loop close.

In order to make real-time global pose alignment easier to handle, this method uses filtered frames to perform layered local to global pose optimization. In the first hierarchy, each successive N frame forms a block, which is optimized for local pose taking into account the contained frames. At the second hierarchy level, all blocks are interconnected and globally optimized. This is similar to hierarchical submaps; However, when all frames are available, instead of parsing the global connection in this method, blocks are formed based on the current time window. Note that this is the only time assumption in the method: there is no time dependency between the blocks.

![Figure 1. BundleFusion algorithm flow.](image-url)
This layered two-phase optimization strategy reduces the number of unknowns per optimization step and ensures that the approach is applicable to large scenarios. Alignment at the two levels is expressed as an energy minimization problem in which the filtered sparse correspondence and dense luminosity and geometric constraints are taken into account. In order to solve this highly nonlinear optimization problem at two levels, this method uses a fast data parallel GPU solver to solve this problem.

Voxel Hashing algorithm was used to realize the real-time reconstruction of the large scene. Continuous changes in the optimized global posture require continuous updates to the global 3D scene representation. Unlike the Voxel Hashing algorithm, BundleFusion allows for symmetric dynamic recombination of RGB-D frames, where pose changes need to be mapped to a global TSDF model. In order to update the pose of the frame with improved estimation, a new real-time anti-integral step is used to remove the RGB-D image previously fused into the TSDF model, and then it is re-integrated according to the optimized new pose. Thus, the 3D model will continue to improve as more RGB-D frames and reconstructed pose estimates become available (for example, if the loop is closed).

3. Feature Point Extraction Algorithm

3.1. Feature detection

Feature extraction and matching of images are the foundation for 3D reconstruction. Now the commonly used Feature detection algorithm is the SIFT (Scale Invariant Feature Transform) algorithm proposed by David Lowe in 1999[8], which was further developed and improved in 2004[9]. SIFT algorithm has the biggest advantage of scale invariance, image translation, rotation, affine transformation between the case can still carry out feature matching. SIFT algorithm extracted rich feature points, suitable for accurate image matching, but high time complexity, poor real-time. BAY H et al. [10] improved SIFT algorithm in 2006 and proposed SURF (Speeded Up Robust Features) algorithm. SURF algorithm inherited the advantages of invariant scale of SIFT algorithm and greatly improved the execution efficiency. In 2006, ROSTEN E and DRUMMOND T [11] proposed FAST algorithm, which can judge feature points by checking whether corner points exist in image blocks. The algorithm is simple and avoids solving the second derivative, so the algorithm is faster than SURF and SIFT algorithms. RUBLEE E et al. [12] proposed the ORB algorithm in 2011. This algorithm adopted the visual information feature point detection and description method, combined with the advantages of FAST algorithm for feature point detection, added the orientation information of FAST feature, and improved the problem that FAST algorithm was not able to detect the orientation of feature points. The feature point description part uses the BRIEF [13] feature descriptor based on the binary comparison of pixel points, and improves its sensitivity to image noise and lack of rotation invariance.

Accurate and fast feature extraction and matching algorithm is an important prerequisite for real-time 3D reconstruction. In this paper, several common feature detection algorithms are compared in feature matching.

The experiment used the two frames in Figure 2, using SIFT, SURF, and ORB to perform feature extraction and matching on the two frames under different conditions, and used RANSAC to eliminate false matches. The three algorithms are compared by matching time and matching rate. Table 1 is the matching points and matching time of the original image, and Table 2 is the matching result when scene_r is added with 10% salt and pepper noise. Table 3 is the matching results under scene_r rotation conditions.

![Figure 2. Images used in the experiment](image)
Table 1. Compare The Time And Match Points

|       | Time/s | scene_1 | scene_1 | Matching point |
|-------|--------|---------|---------|----------------|
| SIFT  | 0.689  | 704     | 703     | 430            |
| SURF  | 0.370  | 316     | 320     | 234            |
| ORB   | 0.042  | 492     | 495     | 203            |

Table 2. Comparison of Matching Points Under Noise

|       | Time/s | scene_1 | scene_1 | Matching point |
|-------|--------|---------|---------|----------------|
| SIFT  | 0.951  | 704     | 1790    | 295            |
| SURF  | 0.389  | 316     | 973     | 189            |
| ORB   | 0.05   | 492     | 500     | 213            |

Table 3. Rotation robustness matching point comparison

| Rotation Angle/degree | 0 | 45 | 90 | 135 | 180 | 225 | 270 |
|-----------------------|---|----|----|-----|-----|-----|-----|
| SIFT                  | 430          | 484 | 370 | 483 | 394 | 486 | 488 |
| SURF                  | 234          | 189 | 201 | 183 | 213 | 182 | 183 |
| ORB                   | 203          | 298 | 323 | 283 | 320 | 183 | 297 |

From the experimental data, it can be concluded that ORB is the fastest algorithm among the three methods, while SIFT performs best in most cases. But SIFT does not perform as well as ORB and SURF when the rotation Angle is proportional to 90 degrees. When the input image is noisy, the performance of ORB ORB is better. It can also be seen from the results that there is a disadvantage of ORB results, that is, the extracted feature points are too concentrated and mostly appear on the objects in the center of the image, while the results of SURF and SIFT are distributed on the image.

Since the 3D reconstruction system requires high matching speed and the collected images are affected by noise, the ORB algorithm should be chosen according to the experimental results, because it has very fast calculation speed and performs well in noisy images.

3.2. An improved ORB algorithm

Targeted at ORB algorithm extraction feature points are too concentrated, there will be cluster this problem, ORB-SLAM2[14] on the original ORB algorithm was improved to improve the uniformity of feature distribution improved algorithm and the original ORB algorithm is basically consistent, the main improvement points in the FAST corner point extraction step.

At the time of corner extraction, each layer of the constructed image pyramid is firstly divided into a grid of 30*30pixels in size. Then, FAST corners can be extracted from each grid separately. If appropriate corners cannot be extracted, FAST thresholds can be lowered to ensure that some FAST corners can also be extracted from areas with weak texture. Through this step, a large number of FAST corners are extracted. However, too much quantity will also affect the subsequent matching, resulting in too much computation and reduced speed. Therefore, it is also necessary to screen the extracted corner points, and the selected feature points should be of good quality and distributed as evenly as possible. The screening method is based on quadtree, diagonal points are selected, and finally N FAST corners are uniformly selected.

The operation of filtering is as follows: first, quadtree is used to represent the key points extracted in practice. After the constructed quadtree is obtained, it needs to be split, and after splitting a parent node, four children nodes of the same size are obtained. The process of splitting also pays attention to the attribution of the key points within the parent node, in this case, to which child node it lies after splitting,
and let it belong to that child node. The condition to stop the splitting of quadtree is that the number of leaf nodes of the split quadtree is larger than the number of key points required previously set or the number of leaf nodes of the split tree does not change. When a leaf node has more than one key point, which one of the best quality is selected as the representative. So, at this point, the critical selection is done.

The improved ORB algorithm was verified by experiments and compared with SIFT and OpenCV's own ORB algorithm. First, a scene map of the laboratory was taken, and then 1000 features were extracted with these three different algorithms. From the result figure, it can be clearly seen that feature points in pictures using SIFT algorithm are distributed in the whole picture, while feature points extracted by ORB algorithm in OpenCV are relatively concentrated, with the phenomenon of pushing and pushing, mainly focusing on areas with changes, while for flat areas, there are almost no feature points. Compared with the feature points extracted by the ORB algorithm in OpenCV, the improved ORB algorithm is more evenly distributed, with feature points distributed throughout the picture. In terms of extraction time, the feature extraction time of the improved ORB algorithm was 10.24ms, while that of the ORB algorithm in OpenCV was 9.11 ms. There's not much difference. But both are much faster than the Sift algorithm. This shows that the improved ORB algorithm has achieved good results.
4. Experimental Results and Analysis

The whole 3D reconstruction system framework in this paper is based on the structure of the Bundle Fusion [9] algorithm. The input part of the system is RGB-D data, and the data is acquired by using RGB-D camera to collect the aligned color and depth data streams. Here, a 30Hz sequence image with a resolution of 1280x720 is output by realsenseD435i. The output part of the 3D reconstruction system is the accurate and complete 3d scene model reconstructed. The scene mentioned here can be an area, a room, or even a whole house.

The experiment in this chapter is an engineering 3D reconstruction system experiment. In order to make the experiment go on smoothly, it is necessary to build the hardware system environment and software system environment, and configure the properties well. In this experiment, Intel realsenseD435i sensor was used for data collection of indoor scenes. The hardware system environment configuration is shown in Table 4, and the software system environment configuration is shown in Table 5.

| Project          | Type                        |
|------------------|-----------------------------|
| CPU              |                             |
| GPU              |                             |
| RAM              | 16GB                        |
| RGBD sensor      | realsenseD435i              |

| Project                        | Name/Version               |
|--------------------------------|----------------------------|
| Operating System               | Windows10                  |
| IDE                            | VS2013                     |
| Programming language           | C++                        |
| realsenseD435i Driver          | Intel real sense SDK 2.0   |
| Computer open source vision library | OpenCV                   |
| Cross-platform open source point cloud library | PCL                      |

The apt0 dataset provided by BundleFusion was chosen for this experiment, resulting in a 3D model as shown in Figure 4.
It can be seen from the diagram of the 3D model that the reconstructed model has obvious outline, clear edge, clear detail, high precision and good sense of reality.

Prior to this, Real SenseD435i camera has been used to collect images of a 12-square-meter indoor scene, and the data set has been verified by reconstruction experiments. The reconstructed results are shown in Figure 5.
Figure 5 is the scanning result of part of the living room scene, and Figure 6 is the scanning and reconstruction result of a fixed sofa. The result of the self-made data set is better in some details, such as the objects on the desktop are reconstructed, such as the folds on the sofa cushion surface are reconstructed, but the overall reconstruction effect is not satisfactory. There's a gap between the reworking of BundleFusion's data sets.

5. Conclusion

With the continuous development of The Times, whether it is related to computer vision technology, or deep information acquisition equipment, computer performance and other hardware have been greatly improved. Therefore, 3D reconstruction technology is becoming more and more mature with higher accuracy and has been widely used in many fields. 3D reconstruction technology is shining brightly in many aspects of modern scientific research and daily application.

This paper uses a 3D reconstruction method based on RGB-D sensor, and the reconstruction objects are mainly indoor scenes. On the basis of BundleFusion reconstruction algorithm, 3D reconstruction of indoor scenes is realized through data collection with depth camera, feature extraction and matching, pose optimization and surface generation. Experiments show that this method can reconstruct 3D objects well.

Acknowledgments

This paper was supported by National Key R&D Program of China(2018YFB1404103).

References

[1] Newcombe R A, Izadi S, Hilliges O, et al. KinectFusion: Real-time dense surface mapping and tracking[C]/Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on. IEEE, 2011: 127-136.
[2] Whelan T,Kaess M,Fallon M,et al.Kintinous:Spatially Extended KinectFusion [J].Robotics & Autonomous Systems,2012,69(C):3-14
[3] Nießner M, Zollhöfer M, Izadi S, et al. Real-time 3D reconstruction at scale using voxel hashing[J]. ACM Transactions on Graphics (TOG), 2013, 32(6): 169.
[4] Kahler O, Adrian Prisacariu V, Yuheng Ren C, et al. Very high frame rate volumetric integration of depth images on mobile devices[J]. IEEE Transactions on Visualization and Computer Graphics, 2015, 21(11): 1241-1250.
[5] Whelan T,Leutenegger S,Moreno R S,et al.ElasticFusion:Dense SLAM Without A Pose Graph[C]/Robotics:Science and Systems.2015
[6] Whelan T,Salas-Moreno R F,Glocker B,et al.ElasticFusion:Real-time dense SLAM and light source estimation[J].International Journal of Robotics Research,2016.
[7] Dai A , Matthias Nießner, Michael Zollhöfer, et al. BundleFusion: real-time globally consistent 3D reconstruction using on-the-fly surface re-integration[J]. ACM Transactions on Graphics, 2017, 36(4):1.
[8] Lowe D G. Object Recognition from Local Sale-in-variant Features[C]. International Conference on Computer Vision, 1999: 1150-1157.
[9] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2):91-110.
[10] Viola P.Jones M. Rapid Object Detection Using a Boosted Cascade of Simple Features[J]. Computer Vision and Pattern Recognition, 2001: 511-518
[11] E. Rosten , T. Drummond,Fusing points and lines for high performance tracking[C]. Proceedings of International Conference on Computer Vision New York,USA:ACMPress,2005:1508-1511.
[12] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: an efficient alternative to SIFT or SURF[C], in International Conference on Computer Vision, 2011, pp. 2564–2571.
[13] Calonder,M,Leptit.V,Fua.P.BRIEF:Binary Robust Independent Elementary Features [C]. ECCV, 2010:778-792.
[14] Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System[J]. Robotics IEEE Transactions on, 2015, 31(5):1147-1163.

[15] Bloesch M, Czarnowski J, Clark R, et al. CodeSLAM - Learning a Compact, Optimisable Representation for Dense Visual SLAM[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018.

[16] Tateno K, Tombari F, Laina I, et al. CNN-SLAM: Real-Time Dense Monocular SLAM with Learned Depth Prediction[J]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017:6565-6574