High Quality 3D Reconstruction of Indoor Environments using RGB-D Sensors

Jun Wang*, Shoudong Huang*, Liang Zhao*, Janet Ge†, Sean He*, Chengqi Zhang*, Xiangyu Wang‡
*Faculty of Engineering and IT, University of Technology, Sydney, Australia
{Jun.Wang,Shoudong.Huang,Liang.Zhao,Xiangjian.He,Chengqi.Zhang}@uts.edu.au
†School of Built Environment, University of Technology, Sydney, Australia
XinJanet.Ge@uts.edu.au
‡School of Built Environment, Curtin University, Perth, WA 6845, Australia
Xiangyu.Wang@curtin.edu.au

***Abstract*—High-quality 3D reconstruction of large-scale indoor scene is the key to combine Simultaneous Localization And Mapping (SLAM) with other applications, such as building inspection and construction monitoring. However, the requirement of global consistency brings challenges to both localization and mapping. In particular, significant localization and mapping error can happen when standard SLAM techniques are used when dealing with the area of featureless walls and roofs. This paper proposed a novel framework aiming to reconstruct a high-quality, globally consistent 3D model for indoor environments using only a RGB-D sensor. We first introduce the sparse and dense feature constraints in the local bundle adjustment. Then, the planar constraints are incorporated in the global bundle adjustment. We fuse the point clouds in a truncated signed distance function volume, from which the high quality mesh can be extracted. Our framework leads to a comprehensive 3D scanning solution for indoor scene, enabling high-quality results and potential applications in building information system. The video of 3D models reconstructed by the method proposed in this paper is available at https://youtu.be/DWMP4YfeNeY.**

I. Introduction

Simultaneous Localization And Mapping (SLAM) has been studied for decades. Various solutions have been proposed for the emerging sensors and various robot platforms, such as RGB-D cameras, event cameras, drones and underwater robots [1]. As pointed out in [1], although different SLAM techniques are now available, robustness in perception still requires further enhancement for practical applications. In this paper, we focus on the use of RGB-D sensor for SLAM in indoor complex scene. The proposed method provides the localization for the sensor, and simultaneously reconstructs a dense globally consistent 3D model utilizing the RGB-D information.

Many algorithms dealing with the RGB-D SLAM systems have been proposed in literature [2]–[5], most of which are focused on the real time localization and sparse or dense mapping. A dense planar SLAM is proposed in [6], where surfel is utilized to represent the map. In the process of detecting planar elements, surfel is splited into planar or non-planar set and five rules are defined to merge the surfels. However plane is only used for map visualization, and the constraints between planes are ignored. Each plane is organized into 3-level hierarchy relationship [1]. In the first level, features are coplanar. Child-plane and parent-plane are also coplanar in the second level. More complex relationships between different planes are explored in the third level. All the relationships are coded into constraints in the optimization. Elastic fusion [7] attempts to construct a globally consistent map by applying embedded deformation to the selected nodes. But the deformation works not so well when the scale of indoor scene increases, especially when the scene include roofs, walls, and ground. [8] proposed a globally consistent 3D reconstruction method, but manual labeling is involved and the output is point cloud instead of high-quality mesh. For long sequences of images, the convergence is also a problem [8], [9].

In this paper, we focus on reconstructing a globally consistent model in complex indoor scene in an efficient manner. As shown in Figure 1, the method consists of three processing steps. First, we apply sparse pose estimation using the visual features extracted from the incoming RGB images, and utilize Iterative Closest Point (ICP) with the local point clouds. Then, a planar constraint bundle adjustment is applied to the pose sequence. Based on the optimized poses, we integrate all the depth frames into a Truncated Signed Distance Function (TSDF) volume. The output of the approach is a globally consistent point cloud and a high-quality mesh.

The highlights of this approach is twofold. First, we proposed a new framework to generate globally consistent 3D model using RGB-D scanning. The planar constraint is simply embedded in the bundle adjustment in an efficient way, which makes the algorithm be able to handle texture-less environment. Second, we integrate all the noisy depth images into several TSDF volumes. High-quality mesh then can be extracted using a modified marching cube algorithm.

This paper is organized as follows. Section II introduces each part of our approach. We evaluate our system in Section III using data collected by a RGB-D sensor. Then, we conclude our approach in Section IV and future work is also proposed in this section.

II. Methodology

This section provides the details of the proposed method.

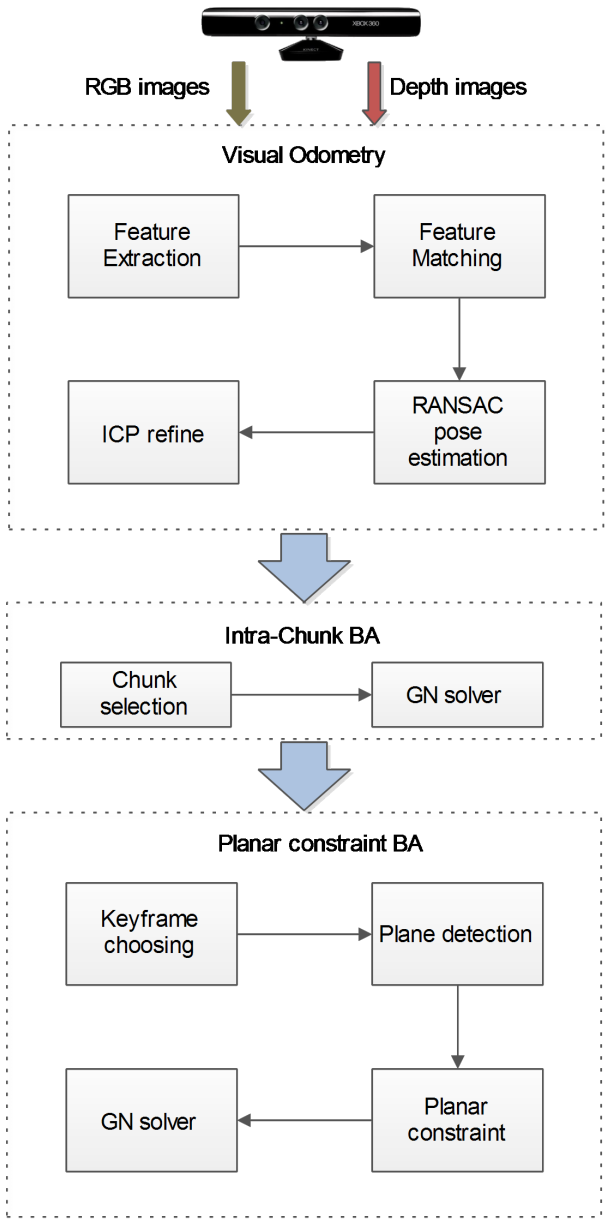


Fig. 1: The framework of the proposed globally consistent 3D reconstruction.

A. Pair-wise visual odometry

In our system, it is necessary to localize the sensor in real-time. The pose of the RGB-D camera can be estimated using visual odometry by exploring the geometric relationship between the visual features and the local 3D point cloud.

Many visual feature descriptors has been proposed in the literature, all of which attempted to achieve scale-invariant, illumination-invariant and so on. Scale-Invariant Feature Transform (SIFT) [10], Speeded Up Robust Features (SURF) [11] and Oriented fast and Rotated Brief (ORB) [12] are extensively used in many feature related applications. Recently, ORB has been proved to be more accurate and efficient than the other two methods, especially in the indoor SLAM problem [13]. ORB uses a binary descriptor, which makes the matching more efficient than float descriptors such as SIFT.

Relative pose of the two frames can be calculated using 3 points. However, the result may be inaccurate since the data from the camera is error-prone. The error in relative pose estimation comes from the limitation of the camera itself, the pre-estimated intrinsic parameters, or the calibration between the infrared and the RGB cameras. So we use RANdom SAmple Consensus (RANSAC) to estimate a robust solution from the noisy data. First, we randomly choose 3 points from the ORB matched features and compute the relative pose using the 3D points. Then, we separate all the points into inliers and outliers based on the solution. The above two steps are repeated until we get a satisfied solution or the maximum iterations is reached.

Though we can get a rough solution of the relative poses between pair-wise frames, the RGB-D camera give us more abundant information about the 3D scene. We apply ICP to refine the relative pose, using the above solution as an initial value. The two dense point clouds are regarded as rigid body transformation and we want to minimize the distance between the points in the first frame and the corresponding plane (point with normal) in the second frame as:

$$E_{icp} = \sum_{i=1}^{N} ||(Rp_i + t - q_i)n_i||_2 \tag{1}$$

where R is the rotation matrix, t is the translation vector, and n_i is the normal vector. Note that, we have to do the RANSAC step to obtain a rough relative pose as an initial guess, as the ICP usually fails when it is initialized using identity matrix as rotation, especially when the camera moves fast or shakes. In practice, it is common to scan the same object several times, for example step forward and backward scanning. Moreover, in a loopy environment, we will get more links relating to these revisit, which can dramatically reduce the accumulated error and drifts. Similar/same images cannot be identified by just using the low-level ORB features, fortunately DBoW2 [14] can efficiently retrieve the similar images by searching in the coded database. DBoW2 constructs a vocabulary tree from the extracted ORB binary features to discrete them. The vocabulary tree can then speed up the verification of geometry correspondences.

B. Bundle adjustment

In the previous section, we obtain a sequence of pose estimations, and many links between them. A graph will be constructed, where poses and links are regarded as nodes and edges respectively. However these nodes and edges will compose a giant graph. In formulation every observation will form one term in the objective function, moreover the Jacobian matrix calculates the partial derivatives for every variable in each observation equation. This problem is very hard to solve due to limited memory and computation resources.

We apply a chunk-splitting strategy to the giant graph and solve it in a hierarchical way [15]. The strategy is based on the fact that the drift in a short range of sequence is very small and can be regarded as accurate links in a global view. In each chunk, we apply the local bundle adjustment to optimize the poses and feature locations. Then a representative keyframe is chosen in each chunk. A global bundle adjustment will be applied on the selected keyframes based on a reasonable assumption that there are only rigid transformations between any two chunks.

1) Intra-chunk optimization: Each chunk is built on the sequence of N_{chunk} images (in our experiments, N_{chunk} is set to 30 frames). In Equation (2), we minimize the sum of the squared distances of reprojective pixels and the distances between 3D points. The optimal solution is found when the minimal error is obtained using a Gauss-Newton (GN) solver.

$$\boldsymbol{E}_{intra-chunk} = \sum_{c} \sum_{p \in s(c)} (||p_i^c - KT_w^c P_i^w||^2 + ||T_c^w K^{-1} p_i^c - P_i^w||^2) \tag{2}$$

where p_i is the pixel in cth camera frame, with the corresponding feature in feature set $s(c)$ of cth frame, P_i^w is the 3D position of corresponding feature in world frame, T_w^c is the transformation between world frame and camera frame, T_c^w is the inverse of the transformation, and K is the intrinsic matrix.

2) Keyframe selection: After the intra-chunk optimization, the poses are relatively precise in each chunk. Then, one keyframe is chosen in each chunk in order that the keyframe can represent a chunk in rigid transformation. So we choose a keyframe as representative as possible and defined a simple principle to measure it, as described in (3).

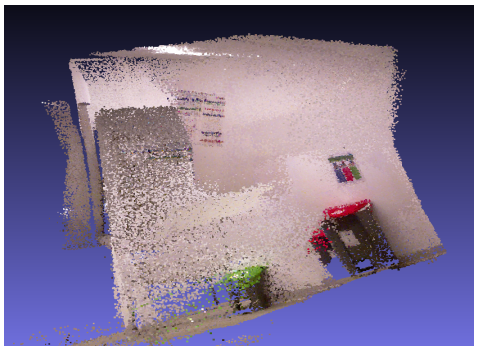
$$\boldsymbol{max}(w_1 N_i^f + w_2 B_i^f) \tag{3}$$

where N_i^f is the number of the features, B_i^f is the bounding box of all the features in the frame, and w_1 and w_2 are the corresponding weights.

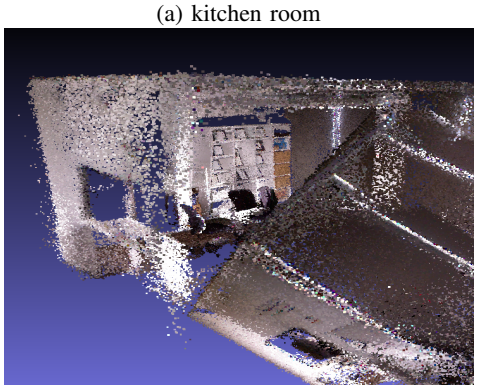
3) Global optimization: The global optimization of the keyframes is critical for the whole framework as it will register all chunks together and is also the most challenging part. Many algorithms use the sparse features in optimization, as the dense images provide too many pixels to handle. However the RGB images provide many valuable information which should not be ignored. In this paper, sparse and dense registration methods are combined for finding a more robust solution. As the value changes a lot in color space when illumination varies, we compare the difference in image gradient space, In Equation (4), we transform the points in image G_i to image G_j, and find the corresponding gradients in image G_j.

$$\boldsymbol{E}_{dense} = \sum_{i,j}^{N_{img}} \sum_{k}^{N_{pixels}} (||G_i(\pi(P_{i,k}) - G_j(\pi(T_j^{-1} T_i P_{i,k})))) \tag{4}$$

where G is the gradient image, π is the perspective projection, and $P_{i,k}$ is the 3D location associating with the kth pixel in image G_i. Ideally, every pixel in image G_i should be transformed to image G_j and find all the corresponding pixels. However, it is too much computation burden of this and adjacent pixel are usually redundant. We resize the image to a much smaller one, for example 64 by 48, and calculate the gradients on these new images.



(a) kitchen room



(b) meeting room

Fig. 2: 3D reconstruction results using bundle adjustment without any planar constraints.

C. Planar constraint bundle adjustment

A rough optimization result is obtained after the previous step. As shown in Figure 2, we do not get a globally consistent 3D model. So we should add new constraints to the optimization process. We noticed that there are many obvious structures in a common indoor environment, for example, the walls are always vertical, and the roof and the ground are parallel. In previous section, the features on these planes are treated as standalone ones, and the geometric relationships between them are ignored. In this section, we will explore the potential features on these planes.

First, we describe how to detect planar elements in the RGB-D images. It is hard to detect structure in a single image, pop up SLAM [16] proposed a method to detect/separate ground and walls, however the algorithm cannot deal with more complex scenes. The RGB-D camera makes the task much easier by providing a corresponding depth image.

Hole filling: There are many holes and noises on the depth image, a bilateral filter with a small window is applied to the

raw image. We define the plane as Equation (5):

$$ax + by + cz + d = 0 \tag{5}$$

where $[a, b, c]^T$ is the normal of the plane, d is the distance from the origin to the plane.

RANSAC plane fitting: A RANSAC algorithm is applied to detect the planar structure in the current frame: (1) random select 3 points; (2) calculate the parameters a_i,b_i,c_i,d_i using the three points; (3) calculate the distance to other points and support points; (4) repeat (1) to (3) until the best plane is found.

We suppose that the first frame contains plane from the ground, so we can roughly figure out the walls and roof by comparing the normal and location of the plane. The features on the ground and roof has no constraint in x and y directions, but the variance in z direction is usually strictly to L_z (we set L_z to 5cm in our experiments). For the features on the wall, the variance in the normal direction should be considered. We set L_n on the on the normal direction for walls. All the constraints can be adjusted to any values according to the specific indoor environment. We can further define the new objective function:

$$\boldsymbol{E} = \boldsymbol{E}_{sparse} + \boldsymbol{E}_{dense} + \sum_{k=1}^{N} \sum_{i,j}^{f_k} \Psi_{planar}(f_i, f_j) \tag{6}$$

where f_i and f_j are the 3D points on the same plane and

$$\boldsymbol{E}_{sparse} = \sum_{c} \sum_{p \in s(c)} ||p_i^c - KT_w^c P_i^w||^2 \tag{7}$$

$$\Psi_{planar}(f_i, f_j) = \begin{cases} 0, & |(f_i - f_j)| \leq N_{constraint} \\ |f_i - f_j|, & |(f_i - f_j)| > N_{constraint} \end{cases} \tag{8}$$

where $N_{constraint}$ is the corresponding constraint of along the coordinate such as L_z or L_n. The first term in Equation (6) is to minimize the difference between image pixels and projected feature points, the second term is the same as that defined in Equation (4). The third term is for the constraints between the features on each plane. As in Equation (8), the energy equals to the distance if the distance between the two point features exceed the constraints, and the function equals to 0 if the distance does not exceed the constraints.

D. TSDF integration

The planar constraint bundle adjustment provides us a precise sequence of poses, this section will discuss how to create a precise surface from the point clouds. The simple stack of point clouds will result in a much redundant point cloud, and the noise in the depth image will remain in the final results. To get a smooth and noise-free surface/map, we apply TSDF [17] to manage our map. As shown in Figure 4, TSDF is a 3D volume, with every grid denoting the distance from this grid to the closest surface in current frame. TSDF also maintain a separate weight volume to roughly measure uncertainty. In initialization, the TSDF value grids are set to N_i and the TSDF weight grids are set to zero. In updating, the new TSDF value and weight value are calculated using Equation (9) and (10) as:

$$T_i(x) = \frac{W_{i-1}(x)T_{i-1}(x) + w_i(x)t_i(x)}{W_{i-1} + w_i(x)} \tag{9}$$

$$W_i(x) = min(W_{i-1}(x) + w_i(x), W_{max}) \tag{10}$$

where $T_i(x)$ is the TSDF value and $W_i(x)$ is the weight volume, t and w are from the new observation. The weight values gradually increase, as new observation occurs. In this way, we can get an appropriate fusion surface by merging all the depth images.

As shown in Figure 4, the surface lies on the change point where the value change from negative to positive or the other way around. A marching cube algorithm is applied to the TSDF grid. For a fast extraction, all kinds layout of the surface in a single grid is enumerated and saved in a lookup table. All the surface will be extracted using only one traverse. Note that we initialize the grid value to $N_i = 1$ and the value behind the surface is set to negative, it accidentally obey the rule where the surface lies. So we delete the changing point grids where the value change from negative to N_i to avoid false surface extraction.

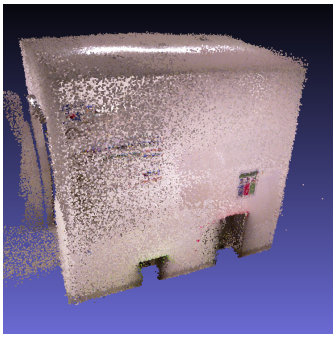
III. EXPERIMENTS

The approach is implemented on a desktop computer with an Intel Xeon CPU, and GeForce GTX 970 GPU. There are four separate threads in the system: odometry estimation, plane detection, loop closure detection, and graph optimization. The grid unit in TSDF fusion is 2cm and there are 300 by 300 by 300 grids in one volume. The chunk size in bundle adjustment is set to 30 frames. We split the TSDF volume when the indoor scene exceed 6 meters to save memory, and the overlap is set to 6cm.

We collect the test data in a kitchen room and a meeting room besides our office. There are kitchen wares, a refrigerator, tables, rubbish bins in the kitchen room. The light on the roof is challenging for the RGB camera, as it leads to dramatical changing in the intensity of images. There are many chairs, a bookshelf, a television and a big table in the meeting room. All the space, including grounds, walls, and roofs, in the rooms are scanned for the purpose of testing the extremely challenging case using our proposed method.

We compared the reconstruction result using bundle adjustment with and without planar constraints. As shown in Figure 2, the point cloud align well in local frames, but it is not globally consistent. After the planar constraint bundle adjustment, we get a globally consistent 3D model as shown in Figure 3 and Figure 5.

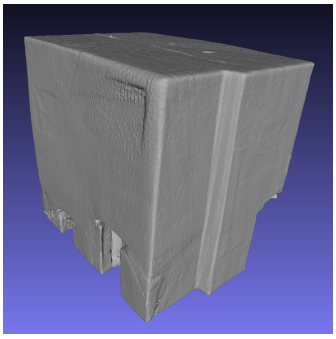
Figure 3a shows the whole point cloud of the kitchen room, which is down-sampled from the depth frames by applying all the transformations using the accurate poses from the planar constraint bundle adjustment. For a better visualization, we delete two walls in front of the kitchen table, as shown in Figure 3b. Figure 3c shows a detailed mesh of the kitchen room, which is extracted from the TSDF volume using the



(a) Point cloud of the kitchen room

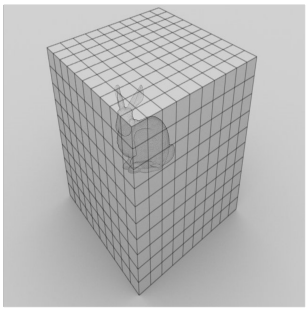
(b) Point cloud of the kitchen room with a wall deleted

(c) High-quality mesh of the kitchen room in a zoomed view

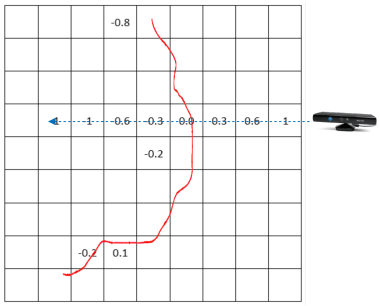


(d) High-quality mesh of the kitchen room in a bird's eye view

Fig. 3: 3D reconstruction results of a kitchen room using proposed planar constraint bundle adjustment.



(a) TSDF volume



(b) TSDF values around a surface denoted by a red line

Fig. 4: TSDF volume and a 2D slice showing how values are assigned

modified marching cube algorithm. As shown in Figure 3d the global structure of the kitchen room are reconstructed with high-quality mesh.

Figure 5 shows a dataset from a meeting room. Note that there are black holes around the door, as the algorithm cannot deal with glasses at this moment. The global structure of both datasets are well reconstructed using our approach, and the details of different objects are also remained in the fusion result.

We compared our model with designed Building Information Modeling (BIM), as shown in Figure 6. Though we use a commodity noisy sensor, our algorithm can produced a matched global consistent 3D model. So our proposed method can be applied to applications in construction monitoring or evaluation.

IV. Conclusion and Future Work

In this paper, we propose a planar constraint bundle adjustment, for globally consistent, high-quality 3D reconstruction in complex indoor scene using RGB-D camera. It is especially suitable for low-texture environment as it utilizes planes as constraints in the optimization. We first generate a rough 3D model using hierarchical intra-chunk and keyframe bundle adjustment. Then, planar elements are detected in the selected keyframes. The planar constraints are incorporated into the global optimization. A TSDF volume integrates all the depth images to reduce noise and generates a high-quality mesh model.

In the experiments of the kitchen room and meeting room, which are the common indoor scene, the results show good performance of the proposed approach, while other RGB-D SLAM algorithms cannot obtain such high-quality model in a room-scale scene.

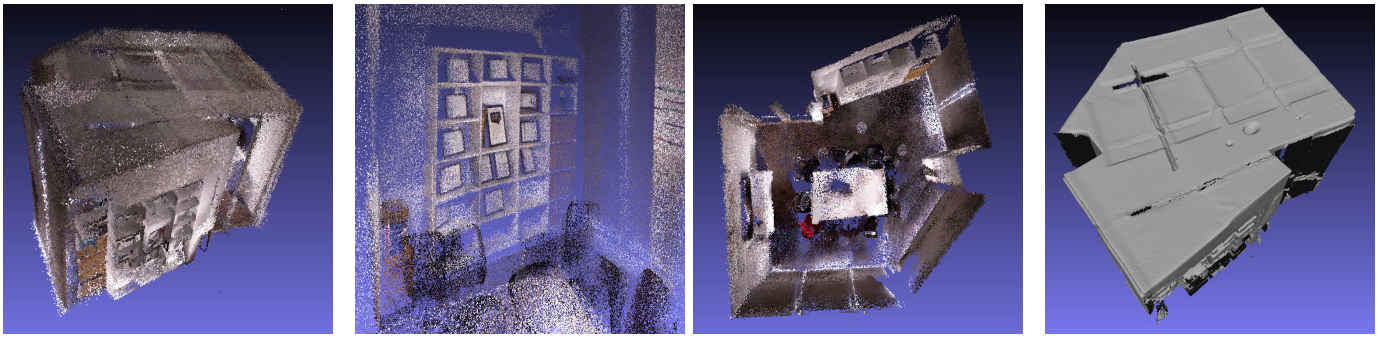
In the future, we would like to formulate edge and plane as landmarks in a unified framework. The nearly perfect 3D models show high potential in applications in the field of BIM [18]. Our 3D model may provide the information about the construction progress at a specific time. The construction progress can be monitored by combining our reconstructed 3D model and BIM, to compare the geometric difference between the two models, and further decisions can thus be made. We would like to test our algorithm in real construction site in the near future.

Acknowledgment

The authors would like to thank Faculty of Engineering and IT at University of Technology Sydney for the support through the Data Arena Research Exhibit Grant.

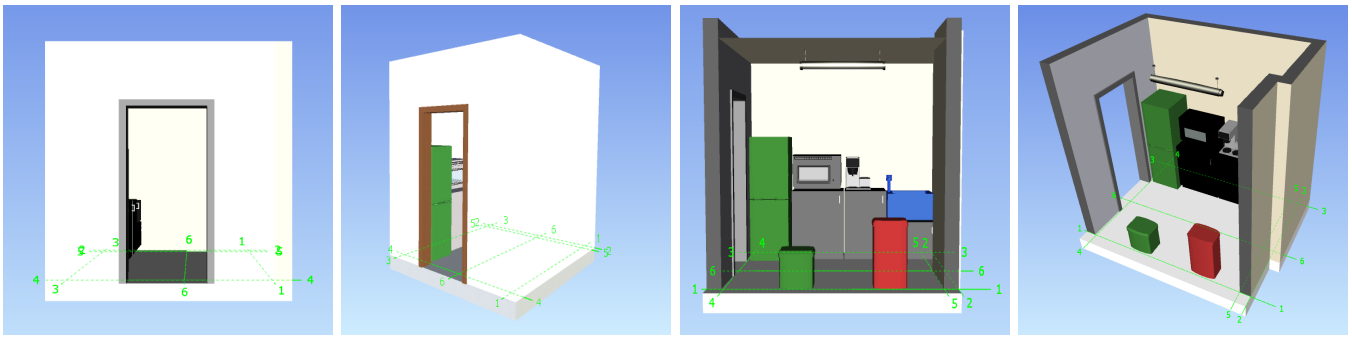
References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. D. Reid, and J. J. Leonard, "Simultaneous localization and mapping: Present, future, and the robust-perception age," *CoRR*, vol. abs/1606.05830, 2016. [Online]. Available: http://arxiv.org/abs/1606.05830



(a) Point cloud of the meeting room (b) Point cloud of the book shelf in the meeting room (c) Point cloud of the meeting room with the roof delteted (d) High-quality mesh of the meeting room in a bird's eye view

Fig. 5: 3D reconstruction results of a meeting room using proposed planar constraint bundle adjustment.



(a) Front view of the BIM model. (b) Side view of the BIM model. (c) Some details of the BIM model with one wall hided. (d) Some details of the BIM model with one wall and ceiling hided.

Fig. 6: BIM model of the kitchen room.

[2] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the rgb-d slam system," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1691–1696.

[3] G. Hu, S. Huang, L. Zhao, A. Alempijevic, and G. Dissanayake, "A robust rgb-d slam algorithm," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 1714–1719.

[4] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.

[5] T. Whelan, M. Kaess, H. Johannsson, M. Fallon, J. J. Leonard, and J. McDonald, "Real-time large-scale dense rgb-d slam with volumetric fusion," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 598–626, 2015.

[6] R. F. Salas-Moreno, B. Glocken, P. H. Kelly, and A. J. Davison, "Dense planar slam," in *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*. IEEE, 2014, pp. 157–164.

[7] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "Elasticfusion: Dense slam without a pose graph," *Proc. Robotics: Science and Systems, Rome, Italy*, 2015.

[8] J. Xiao, A. Owens, and A. Torralba, "Sun3d: A database of big spaces reconstructed using sfm and object labels," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1625–1632.

[9] T. Zhang, K. Wu, J. Song, S. Huang, and G. Dissanayake, "Convergence and consistency analysis for a 3-d invariant-ekf slam," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 733–740, April 2017.

[10] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International journal of computer vision*, vol. 74, no. 1, pp. 59–73, 2007.

[11] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[12] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. IEEE, 2011, pp. 2564–2571.

[13] R. Mur-Artal, J. Montiel, and J. D. Tardós, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[14] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[15] A. Dai, M. Nießner, M. Zollöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration," *arXiv preprint arXiv:1604.01093*, 2016.

[16] S. Yang, Y. Song, M. Kaess, and S. Scherer, "Pop-up slam: Semantic monocular plane slam for low-texture environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, October 2016.

[17] D. Werner, A. Al-Hamadi, and P. Werner, "Truncated signed distance function: experiments on voxel size," in *International Conference Image Analysis and Recognition*. Springer, 2014, pp. 357–364.

[18] F. Bosche and C. T. Haas, "Automated retrieval of project three-dimensional cad objects in range point clouds to support automated dimensional qa/qc," *Information Technologies in Construction*, vol. 13, pp. 71–85, 2008.