Egomotion Estimation Under Planar Motion with an RGB-D Camera

Xuelan Mu(✉), Zhixin Hou, and Yigong Zhang

School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, People's Republic of China
{117106010701,zxhou}@njust.edu.cn

Abstract. In this paper, we propose a method for egomotion estimation of an indoor mobile robot under planar motion with an RGB-D camera. Our approach mainly deals with the corridor-like structured scenarios and uses the prior knowledge of the environment: when at least one vertical plane is detected using the depth data, egomotion is estimated with one normal of the vertical plane and one point; when there are no vertical planes, a 2-point homography-based algorithm using only point correspondences is presented for the egomotion estimation. The proposed method then is used in a frame-to-frame visual odometry framework. We evaluate our algorithm on the synthetic data and show the application on the real-world data. The experiments show that the proposed approach is efficient and robust enough for egomotion estimation in the Manhattan-like environments compared with the state-of-the-art methods.

Keywords: Egomotion estimation · Indoor scene · RGB-D camera · Planar motion · Visual odometry

1 Introduction

Egomotion estimation is an intensively discussed issue in computer vision, which aims at understanding the six-degree-of-freedom (6-DoF) transformation (three for the rotation and three for the translation) of the visual sensor with reference to the input sequence of images. It has drawn a lot of attentions in numerous applications such as augmented reality, motion control and autonomous navigation [3,4,14,20]. The term visual odometry (VO) was originally presented in the work [18] in 2004, which is the process of evaluating the egomotion of an agent such as mobile robot with only the input of a single or multiple cameras mounted to it. The approaches of VO can be classified into two major categories: one is the optical flow method based on pixel information [2,5,15], and the other is the vision-based method [8,22]. Compared with the first one, the vision-based indirect method is much more robust because of its use of discernible feature points from image. Hence, this paper focuses on the study of feature-based egomotion estimation method which belongs to the second one.

For a calibrated camera, it needs only five points to estimate the 6-DOF pose between two consecutive views [17] while seven or eight points [6] are needed if

Z. Cui et al. (Eds.): IScIDE 2019, LNCS 11935, pp. 68–79, 2019.
https://doi.org/10.1007/978-3-030-36189-1_6

the camera is not calibrated. Specifically, in the case of a 99-percent probability of success and a set of data with half-rate outliers, the linear 8-point essential matrix algorithm [7] requires about 1177 samples whereas the 5-point essential matrix algorithm [11,13,23] only needs 145 trials. Therefore, finding an approach with minimal points to meet the real-time requirements is necessary. To reduce the amount of needed point correspondences between frames, some reasonable hypotheses combining additional sensor data are needed. Consequently, based on the availability of low-cost RGB-D cameras, we mainly investigate the motion of the mobile robot in indoor environments where at least one plane is present in the scene. The RGB-D camera is mounted rigidly on a mobile robot and the robot is always under planar motion because of the flat indoor floor. So the pitch and roll angles remain constant during the whole process. Assuming the roll and pitch angles of the camera as known, we correct the RGB-D camera through rotating the generated point clouds so that the values of the roll and pitch angles are approximately equal to zeros. In this case, we just need to calculate a three-degree-of-freedom egomotion estimation problem, which consists of two horizontal translations and one yaw angle.

Despite of its many developments, egomotion estimation still remains somewhat challenging to be efficient and robust in structural and low-texture scenes (e.g., wall or hallway). To solve this problem, several SLAM systems using high geometric characteristics such as lines and planes have been proposed recently [1,9,10,12]. While, Kim et al. [10] estimated the drift-free rotation by applying a mean-shift algorithm with the surface normal vector distribution. However this method required at least two orthogonal planes for demonstrating superior rotation estimation. Kaess [9] introduced a minimal representation with planar features using a hand-held RGB-D sensor and presented a relative plane formulation, which improved the convergence for faster pose optimization. While this method required plane extraction and matching at each frame to construct optimization function, and additional odometry sensors were utilized to perform plane matching, which increased the complexity for VO system.

In contrast to these methods, in this paper we obtain a rough plane-segmentation result using only RGB-D frame, which is fast to meet the real-time application instead of segmenting the scenes into very precise planar regions. We directly estimate the subsequent egomotion through extracting the normal in each plane from the inverse-depth-induced histograms. We will take different strategies according to the prior knowledge about the 3D scenarios. The major contributions of this work are twofold: First, if there exists at least a vertical plane, we realize the pose and location estimation with the normal of the vertical plane and one point correspondence, which is called the direction-plus-point algorithm. Second, if the plane orientation is completely not available, we propose an efficient 2-point minimal case algorithm for the homography-based method to estimate the egomotion. Compared with the classical 5-pt essential matrix estimation [17], our method just needs two matched points between views instead of five, which speeds up the process of iterative optimization for egomotion estimation. At last, we evaluate our algorithm on both synthetic and real datasets.

The rest of the paper is organized as follows. Section 2 describes two efficient algorithms for estimating the egomotion under the weak Manhattan-world assumption. Section 3 presents the performance of our solutions on synthetic and real data and compares with the classical method in a quantitative evaluation. Finally, in Sect. 4, conclusions are drawn.

2 Ego-Motion Estimation

In VO, compared with the estimation of translation which is relatively simple, the estimation of rotation deserves more attention. The majority of experimental errors are derived from the inaccurate estimation of rotation. Thus, reducing the accumulated error caused by rotation can greatly improve the performance of the algorithm. In this work, if there exists at least one vertical plane in the scene, we estimate the motion by decoupling rotation and translation so that a drift-free rotation estimation can be derived from the alignment of local Manhattan frames. Based on the accurate rotational motion, we can obtain a robust estimation of translation with 1-point RANSAC approach. In addition, we propose a new 2-pt minimal-case algorithm with the simplified motion model while no vertical planes are known.

2.1 The Plane-Plus-Point Algorithm

We design a fast plane segmentation method through using only the depth image based on a RGB-D camera, where we extract the inverse depth induced horizontal and vertical histograms to detect planes instead of segmenting the huge point clouds directly. We view the whole indoor scenery as a composition of one or several local Manhattan structures. We can recognize at least one local Manhattan coordinate frame according to the detected vertical planes at any time. And the pose estimation is simplified when knowing the vertical direction in the scenes. Then through aligning the measured vertical direction with the camera coordinate system, the y-axis of the camera is parallel to the vertical planes while the x-z-plane of the camera is parallel to the ground plane (illustrated in Fig. 1). This alignment can make relative motion reduce to a 3-DOF motion, which includes 1 DOF of the remaining rotation and 2 DOF of the translation (i.e., a 3D translation vector up to scale).

In general, we can detect only one ground plane, but multiple vertical planes which correspond to different Descartes coordinate frames may be obtained at the same time. Therefore, we need to distinguish the dominant Manhattan frame from the minor ones based on the specified Descartes coordinate frames attached to the walls. For each Descartes coordinate frame, we calculate its evaluation score according to the area of all the vertical planes in the RGB image whose normal (in the depth image) is approximately parallel or perpendicular to each other. Among them we choose the frame whose score is the largest as the dominant local Manhattan frame.

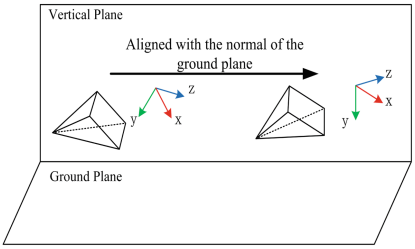

Fig. 1. Alignment of the camera with the normal of the ground plane.

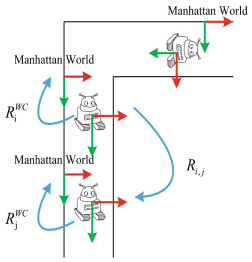

Fig. 2. Rotation between two successive times.

Hence, the drift-free rotation between two successive views will be obtained as soon as the dominant Manhattan frame has been determined at each moment. As shown in Fig. 2, assuming the robot is in the same Manhattan structure (frame) denoted at two different times t_i and t_j, we can estimate only one rotation $R_{i,j}$ of C_i with respect to C_j:

$$R_{i,j} = (R_i^{wc})^T R_j^{wc} \tag{1}$$

where C_i and C_j are respectively the camera coordinate frame at the continuous time t_i and t_j and where R_i^{wc} and R_j^{wc} denote the rotations of C_i and C_j with respect to the local Manhattan coordinate.

Knowing the rotation information based on Manhattan world constraints, the translation estimation needs to be obtained through other algorithms such as a 1-point RANSAC method. We detect and match corner points from two successive RGB images and get the 3D points through the depth image. Then the translation T is estimated with one 3D point correspondence:

$$T = P^{'} - (R_{i,j})^T P \tag{2}$$

where P and $P^{'}$ are respectively the current 3D points and the previous 3D points.

2.2 The 2-Point Homography-Based Algorithm

In the real-world indoor environments the vertical plane is not always available. In this case, the plane-plus-point algorithm which is proposed in the previous section does not work. Therefore we propose a new 2-point minimal case algorithm for the homography matrix based method, and we do a local refinement between the current and previous plane according to the Manhattan assumption to eliminate the drift if the vertical plane is detected again. Given $p_i = [x_i, y_i, 1]^T$ and $p_j = [x_j, y_j, 1]^T$, which are points on the ground plane in the first and second camera coordinate frames, the homography constraint is defined as:

$$\sigma p_j = H p_i \tag{3}$$

With

$$H = R - \frac{t}{d} N^T \tag{4}$$

Where σ is a scale factor, R and t are the rotation matrix and translation vector respectively, and N is the normal vector of the 3D plane and d is the distance from the camera to the corresponding plane. Giving two 3D points in the world coordinate, they will uniquely define a virtual vertical plane and the unit normal vector of this virtual plane with respect to the i_{th} view is $n = [n_x, 0, n_z]^T$ (if they are not vertically aligned). Let $\frac{t}{d} = [t_x\ 0\ t_z]^T$. The homography induced by this virtual vertical plane can be written as:

$$H = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} - \begin{bmatrix} t_x \\ 0 \\ t_z \end{bmatrix} \begin{bmatrix} n_x \\ 0 \\ n_z \end{bmatrix}^T, \tag{5}$$

$$= \begin{bmatrix} \cos\theta - n_x t_x & 0 & \sin\theta - n_z t_x \\ 0 & 1 & 0 \\ \text{-}\sin\theta - n_x t_z & 0 & \cos\theta - n_z t_z \end{bmatrix}. \tag{6}$$

There are four unknown elements of a 3×3 homography matrix in (6), therefore this matrix can be parametrized as:

$$H = \begin{bmatrix} h_1 & 0 & h_2 \\ 0 & 1 & 0 \\ h_3 & 0 & h_4 \end{bmatrix}. \tag{7}$$

In order to eliminate the scalar factor in (3), we use cross product instead and obtain:

$$p_j \times H p_i = 0. \tag{8}$$

Combining (7) and (8), we have the following relation:

$$\begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \times \begin{bmatrix} h_1 & 0 & h_2 \\ 0 & 1 & 0 \\ h_3 & 0 & h_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = 0. \tag{9}$$

It gives us three linear equations, but cross product can also be expressed as a skew-symmetric matrix product, and the rank of the skew-symmetric $[p_j]_\times$ is two, only two linearly independent equations are achieved. By choosing the first two (9) can be rearranged into:

$$\begin{bmatrix} x_i y_j & y_j & 0 & 0 \\ 0 & 0 & x_i y_j & y_j \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} = \begin{bmatrix} x_j y_i \\ y_i \end{bmatrix}. \tag{10}$$

One point correspondence gives two constrains and $[h_1, h_2, h_3, h_4]$ can be uniquely determined by two point correspondences if they are not vertically

aligned. Note that, for different two point correspondences, the parameters $[d, n_x, n_y]$ of the plane which is determined by these two points are not the same. Once $[h_1, h_2, h_3, h_4]$ is obtained from two point correspondences, let's consider the following relations:

$$\begin{cases} n_x t_x = cos\theta - h_1, \\ n_z t_x = sin\theta - h_2, \\ n_x t_z = -sin\theta - h_3, \\ n_z t_z = cos\theta - h_4, \end{cases} \tag{11}$$

By multiplying the first equation by the forth one and multiplying the second equation by the third one, we obtain:

$$\begin{cases} n_x n_z t_x t_z = (cos\theta - h_1)(cos\theta - h_4), \\ n_x n_z t_x t_z = (sin\theta - h_2)(-sin\theta - h_3), \end{cases} \tag{12}$$

The left part of the two equations in (12) is identical. Therefore, the following relation can be obtained by associating the right parts:

$$(h_1 + h_4)\cos\theta + (h_2 - h_3)\sin\theta + h_2 h_3 - h_1 h_4 - 1 = 0, \tag{13}$$

with:

$$\sin^2\theta + \cos^2\theta = 1. \tag{14}$$

Using (13) and (14) we can compute $\cos\theta$ and $\sin\theta$, which have two possible solutions. The rotation of the camera motion can directly be derived from the $\cos\theta$ and the $\sin\theta$.

Then the normal vector can be obtained by dividing both sides of the first equation of (12) by the second one:

$$\frac{n_x}{n_z} = \frac{cos\theta - h_1}{sin\theta - h_2}, \tag{15}$$

with:

$$n_x^2 + n_z^2 = 1. \tag{16}$$

Finally, the translation up to scale is given by:

$$t = d\left[\frac{cos\theta - h_1}{n_x} \; 0 \; \frac{cos\theta - h_4}{n_z}\right]^T. \tag{17}$$

3 Experiments

To evaluate the performance of the proposed egomotion estimation method, both the synthetic data and the real-world data are used for the experiments. The synthetic data with ground truth is used to compare our 2-point algorithm with another minimal solution, the 5pt-essential method [17]. The real-world datasets are provided with two scenes, one for the laboratory building and the other for the dormitory building. Each scene that satisfies weak Manhattan constrains is captured by using a robot mounted with a Kinect v2.

3.1 Test with Synthetic Data

The synthetic data sets are generated in the following setup. The scene contains of 500 randomly sampled 3D points totally and the focal length of the camera is set to 1000 pixels with a field of view of 50°. The average distance from the first camera to the scene is set to 1 and the base line between two cameras is set to be 15% of the average scene distance. Since we focus on the indoor robot motion estimation, the second camera is rotated around $y-$axis with the relative rotation angle varying from $-15°$ to $15°$. The translation is set parallel to the ground and the moving direction is set into two situations, along the x-axis (sideways) and along the z-axis (forward), respectively. This is similar to Nister's test scene in [19], which has been used in [21].

As the estimated translation is up to a scalar factor, we compare the angle between the ground-truth and estimated translation vector. The errors are defined as follows:

$$\begin{cases} \xi_R = |\theta_g - \theta_e|\,, \\ \xi_t = arccos((t_g^T t_e)/(\|t_g\|\|t_e\|)), \end{cases} \tag{18}$$

where ξ_R is the rotation error and ξ_t is the translation error, the errors are similar to [17]. The θ_g, t_g denote the ground-truth rotation angle and translation, and θ_e, t_e are the corresponding estimated rotation angle and translation vector, respectively.

We evaluate each algorithm under the image noise (corner location) with a different standard deviation and the increased $(Roll, Pitch)$ noise. The noise can be considered as the error of the normal of the ground plane. In our experiments, we assume that there are enough feature points lying on the ground. Half of them are randomly generated on the ground plane and the rest are in the 3D space above the ground plane. We use the least square solution with all the inliers and plot the mean value of 1000 trials with different points and different transformations.

Figures 3 and 4 shows the results of the 5pt-essential matrix algorithm and our 2pt-homography method. The experiments show that our 2pt-homography matrix algorithm outperforms the state-of-art 5pt-essential method, in terms of the rotation error and the translation error. It appears that the 5pt-essential method is more sensitive to $(Pitch, Roll)$ noise while our 2pt-homography matrix algorithm is more robust. Notice, when we use this algorithm for real application, a two point RANSAC method can be used to reject outliers and the final solution is given by the least square method with all the inliers.

3.2 Performance with Real Data

In order to show the efficiency of the proposed frame-to-frame visual odometry framework, several real datasets taken in two different indoor corridor-like environments using an RGB-D camera mounted on a robot have been collected, as

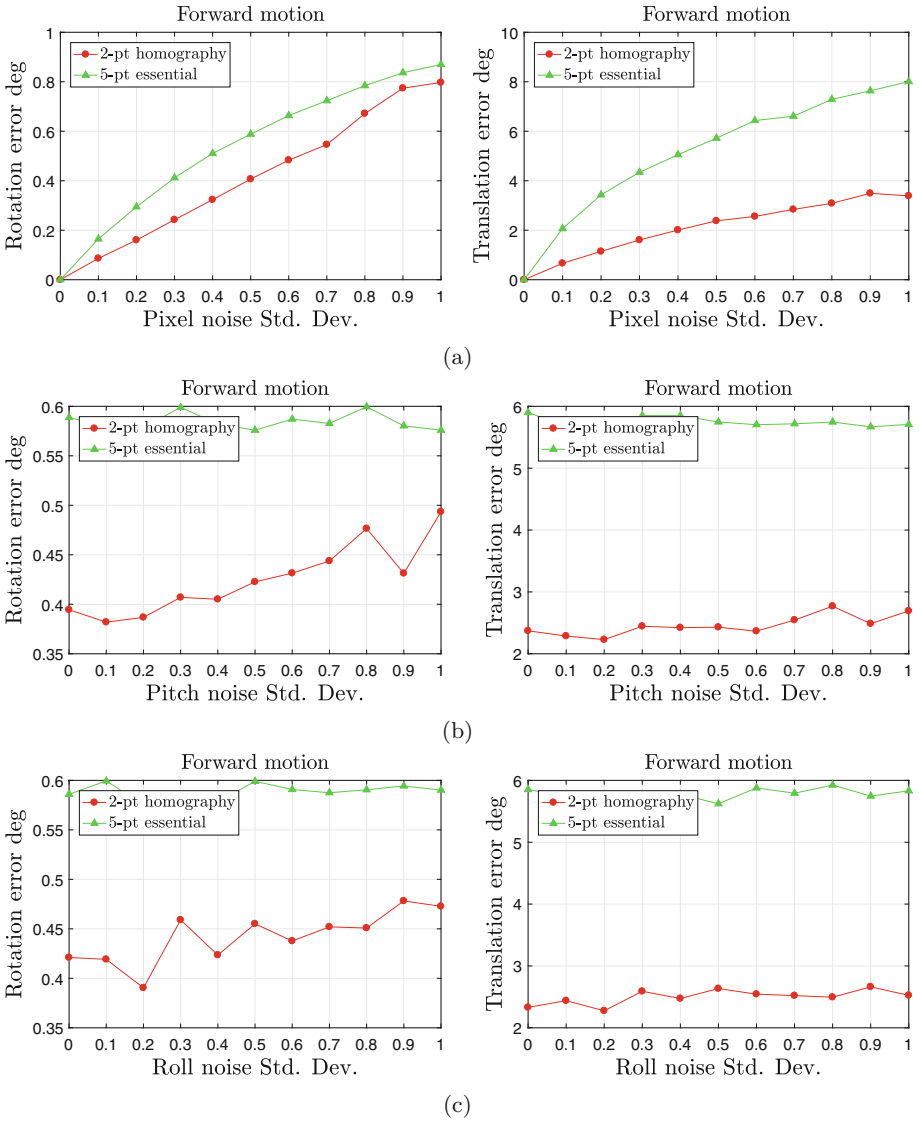

Fig. 3. Rotation and translation error for forward motion. Comparing the 5pt-essential matrix algorithm with our 2pt-homography method. Left column: Rotation error, right column: Translation error. (a) is with varying image noise. (b) is with increased Pitch noise and 0.5 pixel standard deviation image noise. (c) is with increased Roll noise and 0.5 pixel standard deviation image noise.

shown in Fig. 5. The scenes are full of low textured walls and the image resolution used is 960×540 pixels. All the experiments are run at 10 FPS on an Intel Core i5-4460 desktop computer with 3.20 GHz CPU, without GPU acceleration.

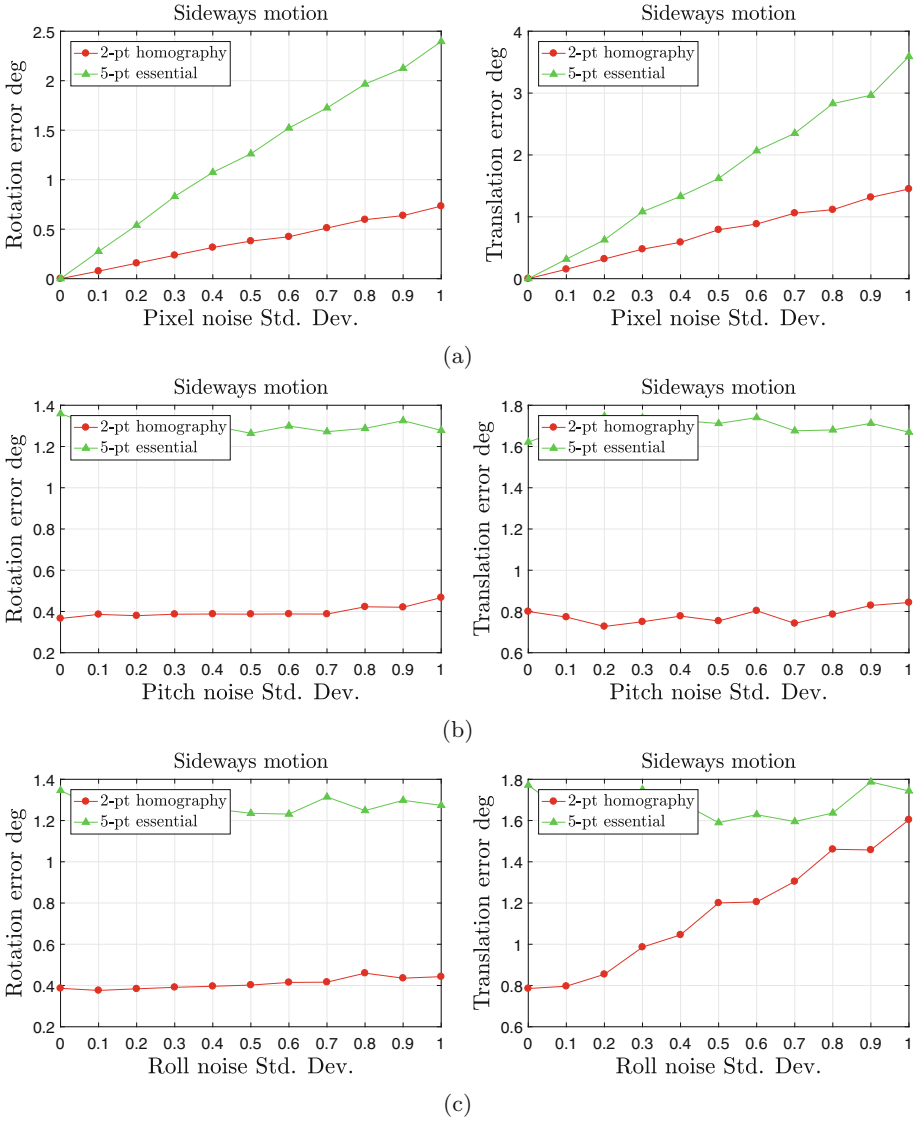

Fig. 4. Rotation and translation error for sideways motion. Comparing the 5pt-essential matrix algorithm with our 2pt-homography method. Left column: Rotation error, right column: Translation error. (a) is with varying image noise. (b) is with increased Pitch noise and 0.5 pixel standard deviation image noise. (c) is with increased Roll noise and 0.5 pixel standard deviation image noise.

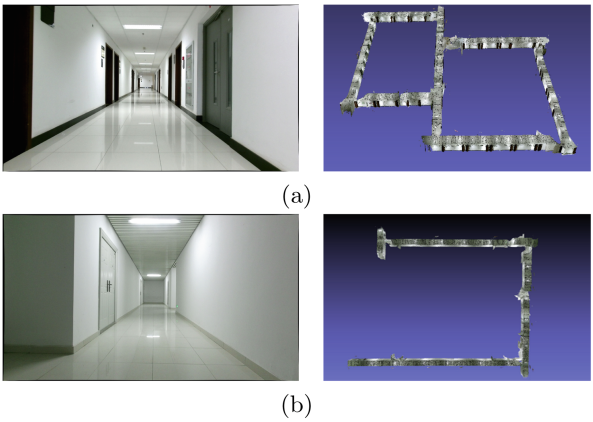

Fig. 5. (a) The laboratory building. (b) The dormitory building. First column: Example images in the school building. Second column: Reconstructed point cloud.

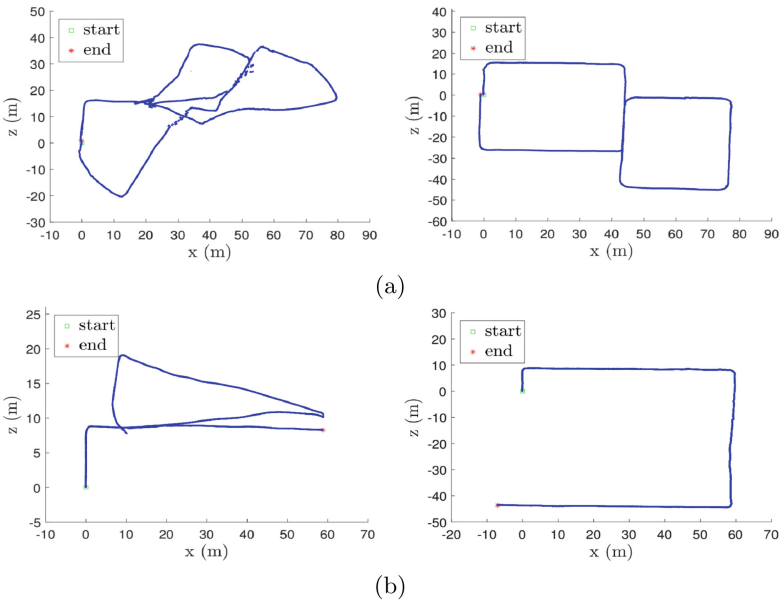

Fig. 6. Comparison between the ORB-RGBD SLAM and our proposed frame-to-frame visual odometry framework. (a) Results on the laboratory building. (b) Results on the dormitory building. First column: Trajectories of ORB-RGBD SLAM. Second column: Trajectories of our method.

We perform a comparison with the state-of-the-art ORB-SLAM2 [16]. As can be seen in Fig. 6, the ORB-SLAM2 fails to complete the entire image sequence because of lacking of features in low-textured walls and then do a false relocation. While our method achieves better results, the performance of our algorithm, using only frame-to-frame camera pose estimation, can be comparable to that of the algorithm with some non-linear refinement or loop closure detection algorithms. The overall errors of our proposed method the laboratory building and the dormitory building are only 1.21% and 1.33% respectively.

4 Conclusion

In this paper, a new method for accurate egomotion estimation of the Manhattan Frame from a single RGB-D image of indoor scenes is proposed. The proposed method differs from previous algorithms by using directions and points to estimate the pose jointly. It firstly detects vertical planes from a large number of RGB-D datasets if at least one vertical plane is available. The normal of the vertical plane is obtained directly based on the inverse-depth induced histograms and we estimate the pose through a novel 3-DOF VO. Secondly, we propose a new minimal-case algorithm to estimate the egomotion if the plane orientation is completely unknown. Finally, we propose a frame-to-frame visual odometry framework based on our algorithms. Experiments with synthetic data and real data validate that the proposed methods are comparable or even superior to the state-of-the-art algorithms while maintaining a high efficiency under planar motion. Our method is currently tested in indoor sceneries with an RGB-D camera. In future work, we will try to implement the proposed algorithm with other sensors and possibly extend to different environments.

References

1. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Comput. Vis. Image Underst. **110**(3), 346–359 (2008)
2. Bergmann, P., Wang, R., Cremers, D.: Online photometric calibration of auto exposure video for realtime visual odometry and SLAM. IEEE Robot. Autom. Lett. **3**(2), 627–634 (2018)
3. Cao, Z., Sheikh, Y., Banerjee, N.K.: Real-time scalable 6DOF pose estimation for textureless objects. In: 2016 IEEE International conference on Robotics and Automation (ICRA), pp. 2441–2448. IEEE (2016)
4. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 611–625 (2018)
5. Forster, C., Zhang, Z., Gassner, M., Werlberger, M., Scaramuzza, D.: SVO: semidirect visual odometry for monocular and multicamera systems. IEEE Trans. Robot. **33**(2), 249–265 (2017)
6. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2000)
7. Hartley, R.I.: In defence of the 8-point algorithm. In: Proceedings of IEEE International Conference on Computer Vision, pp. 1064–1070. IEEE (1995)

8. Hu, H., Sun, H., Ye, P., Jia, Q., Gao, X.: Multiple maps for the feature-based monocular SLAM system. J. Intell. Robot. Syst. **94**(2), 389–404 (2019)
9. Kaess, M.: Simultaneous localization and mapping with infinite planes. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 4605–4611. IEEE (2015)
10. Kim, P., Coltin, B., Kim, H.J.: Visual odometry with drift-free rotation estimation using indoor scene regularities. In: 2017 British Machine Vision Conference (2017)
11. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In: BMVC, vol. 2, p. 2008 (2008)
12. Le, P.H., Košecka, J.: Dense piecewise planar RGB-D SLAM for indoor environments. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4944–4949. IEEE (2017)
13. Li, H., Hartley, R.: Five-point motion estimation made easy. In: 18th International Conference on Pattern Recognition (ICPR 2006), vol. 1, pp. 630–633. IEEE (2006)
14. Li, S., Calway, A.: Absolute pose estimation using multiple forms of correspondences from RGB-D frames. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 4756–4761. IEEE (2016)
15. Matsuki, H., von Stumberg, L., Usenko, V., Stückler, J., Cremers, D.: Omnidirectional DSO: direct sparse odometry with fisheye cameras. IEEE Robot. Autom. Lett. **3**(4), 3693–3700 (2018)
16. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. IEEE Trans. Robot. **33**(5), 1255–1262 (2017)
17. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Trans. Pattern Anal. Mach. Intell. **26**(6), 0756–777 (2004)
18. Nistér, D., Naroditsky, O., Bergen, J.: Visual odometry. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 1, pp. I. IEEE (2004)
19. Nistér, D., Schaffalitzky, F.: Four points in two or three calibrated views: theory and practice. Int. J. Comput. Vis. **67**(2), 211–231 (2006)
20. Rubio, A., et al.: Efficient monocular pose estimation for complex 3D models. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 1397–1402. IEEE (2015)
21. Saurer, O., Vasseur, P., Boutteau, R., Demonceaux, C., Pollefeys, M., Fraundorfer, F.: Homography based egomotion estimation with a common direction. IEEE Trans. Pattern Anal. Mach. Intell. **39**(2), 327–341 (2016)
22. Sun, H., Tang, S., Sun, S., Tong, M.: Vision odometer based on RGB-D camera. In: 2018 International Conference on Robots & Intelligent System (ICRIS), pp. 168–171. IEEE (2018)
23. Ventura, J., Arth, C., Lepetit, V.: Approximated relative pose solvers for efficient camera motion estimation. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8925, pp. 180–193. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16178-5_12