IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Visual SLAM with Drift-Free Rotation Estimation in Manhattan World

Jiacheng Liu and Ziyang Meng

***Abstract*—This paper presents an efficient and accurate simultaneous localization and mapping (SLAM) system in man-made environments. The Manhattan world assumption is imposed, with which the global orientation is obtained. The drift-free rotational motion estimation is derived from the structural regularities using line features. In particular, a two-stage vanishing points (VPs) estimation method is developed, which consists of a short-term tracking module to track the clustered line features and a long-term searching module to generate abundant sets of VPs candidates and retrieve the optimal one. A least square problem is constructed and solved to provide refined VPs with the clusters of structural line features every frame. We make full use of the absolute orientation estimation to benefit the whole SLAM process. In particular, we utilize the absolute orientation estimation to increase the localization accuracy in the front end, and formulate a linear batch camera pose refinement problem with the known rotations to improve the real time performance in the back end. Experiments on both synthesized and real-world scenes reveal results with high-precision in the real time camera pose estimation process and high-speed in pose graph optimization process compared with the existing state-of-the-art methods.**

***Index Terms*—SLAM, Localization, Visual-Based Navigation.**

I. Introduction

Visual SLAM (V-SLAM) problem consists of estimating the body pose while constructing a map of an unknown environment simultaneously. V-SLAM is a fundamental tool for augmented/virtual reality and autonomous vehicles such as micro air vehicles (MAVs), self-driving cars, and automatic guided vehicle systems (AGVs).

In recent years, V-SLAM has been extensively investigated by two popular frameworks, i.e., optimization-based one and filter-based one. It is also shown that the optimization-based method is more suitable for V-SLAM application due to its robustness and superior accuracy [1]. The dominant optimization-based method includes a front end and a back end. The front end serves as an odometry, and it first extracts a set of image features such as points [2], lines [3]–[10], planes [11], [12] and etc. Then, the front end estimates the rotational and translational camera motion in real time, and simultaneously reconstructs the 3D landmarks from feature correspondence. The back end is responsible for map maintenance, e.g., new landmarks creation, landmarks culling, and keyframe culling [2]. Besides, it helps to reduce the accumulated estimation drift of camera pose and 3D map through a batch bundle adjustment (BA), detects the loop closure and corrects it by solving a large-scale non-linear optimization problem. Although different approaches are proposed for general scenes, three notable issues cannot be ignored: (i) the main inaccuracy source of camera pose estimation in the front end is from the rotation estimation error [13]; (ii) the majority of existing approaches induce accumulated drift after running a period of time due to the lack of global measurement such as GPS, and the only solution is to rely on loop closure to correct the drift; (iii) to correct the loop closure, existing strategies construct and solve a global BA problem or pose-graph optimization problem, which are non-linear, non-convex and time-consuming. Moreover, most solvers refine the estimation in an iterative way, e.g., Gauss-Newton method or Levenberg-Marquardt method, and therefore may result in the convergence to the local minima given a bad initial guess [14].

On the other hand, most of the applications in practice work on particular scenes like man-made environments at most time. The man-made environments exhibit strong structural regularity with most parts of surrounding environments can be modeled as a box world with three mutual orthogonal predominant directions according to the Manhattan world assumption [15]. Each Manhattan world has a single frame, which is denoted by Manhattan frame (MF), and the algorithms for inferring MF have been studied extensively in the computer vision community. The MF can be inferred using RGB-D camera by clustering the surface normals as orthogonally-coupled ones [16], [17], assuming that each plane is perpendicular to one of the axes of MF. Besides, the MF can also be inferred by estimating the vanishing point (VP) [18]–[21], which is the intersection of the image projections of the predominant parallel lines in Manhattan world. In addition, it can be inferred by the joint statistical analysis of both line features and surface normals [12].

Motivated by the above observations, we make use of the estimation of MF in this paper to provide global estimation of camera orientation for the considered 3D V-SLAM problem. This accordingly contributes to reduce the estimation drift in the front end and reduce computation burdens in the back end. Specifically, the main contributions of this paper are as follows.

J. Liu and Z. Meng are with Department of Precision Instrument, Tsinghua University, Beijing 100084, China. Corresponding author: Ziyang Meng. Emails: liu-jc18@mails.tsinghua.edu.cn (J. Liu), ziyangmeng@tsinghua.edu.cn (Z. Meng).

- A fast and accurate two-stage MF inference approach is proposed. In particular, the VPs are detected every few frames and the clustered structural line features are tracked to estimate the VPs for every ordinary frame.
- A structural regularity awared camera pose estimation method is introduced in the front end, which utilizes the absolute orientation estimation to improve the localization accuracy.
- A robust and linear batch camera poses refinement strategy is proposed with the further utilization of the drift-free orientation in the back end.

II. Related Work

We next review the related works on the study of SLAM system with the focus on leveraging the structural regularity to improve system performance. In particular, existing works can be classified into two major categories based on whether or not the rotational and translational camera motion estimations are separately obtained.

In the first place, methods that use line features as landmarks, especially structural line features have become popular in recent years since the use of pure point features has their limitations in challenging environments. Overall, these methods do not decouple rotational and translational motion estimations but just introduce line features into SLAM system. In man-made environments, the abundant structural regularities, e.g., parallelism, orthogonality and coplanarity provide effective geometric constraints, which is prone to lead to a more accurate and robust camera pose estimation.

In particular, the authors of [5] design an efficient VPs estimation algorithm by applying an inertial-aided RANSAC method, and develop a tightly-coupled visual inertial odometry (VIO) using VPs to increase the localization accuracy based on an extended Kalman filter (EKF) framework. Besides, the authors of [7] present an EKF-based V-SLAM algorithm using the structural line segments of buildings as landmarks, which adopts the dominant direction of MF to parameterize the 3D map lines. Based on this result, the authors of [8] develop an EKF-based VIO system considering the Atlanta world model [22] which contains multiple local MFs sharing a common vertical axis. In this EKF framework, the different heading directions of local MFs are added in the state vector, such that it is more robust for complex man-made environments. Taking the geometric information of orthogonality, parallelism and coplanarity into consideration, a back-end optimization strategy is proposed in [9], which helps to refine the estimation results from [3] that uses both points and line segments features.

In the second place, the structural regularity provides ability to produce global orientation estimation by inferring the MF and helps to solve the SLAM problem using rotation-known methods. The absolute orientation estimations directly eliminate accumulated error of camera pose and simplify the pose-graph optimization. In particular, the authors of [12] present a visual odometry (VO) system by detecting the MF utilizing both line segments and planes in RGB-D images. Given the rotational estimation from MF tracking, the translational camera motion is calculated by minimizing the de-rotated reprojection error in an iterative way based on point features tracked by optical flow. Based on this VO method, the authors of [11] develop a linear SLAM system, which utilizes the mutual orthogonal plane features in the environment as landmarks and estimates the positional pose of the plane features and camera under a filter-based framework. The authors of [10] develop an Atlanta frame inference method by clustering image line features, and they exploit the directional constraints for SLAM optimization through a reprojection error-based cost function of the endpoint-to-line distance. Ref. [23] builds a 2D SLAM system that exploits relative feature measurements to recover the robot heading and then solve a linear estimation problem over robot and feature positions. However, this 2D SLAM system is only verified by simulation with a robot carrying a range bearing sensor. Furthermore, the authors of [24] present a 2D SLAM system by fusing the measurement datum from wheel encoder, camera, lidar and IMU. The global heading is obtained through detecting the structural cues using a ceiling facing camera, which assists in fusing the relative pose estimation from other sensors. While in the back end, it performs loop closure detection and refines the orientation estimation and localization estimation separately.

In this paper, we adopt a two-stage MF tracking method to first estimate the global orientation from structural line features, and then decouple the rotational and translational motion estimation. Compared with the line-based methods that use the line features directly as landmarks, the proposed algorithm has lower computation burden and complexity. Besides, compared with the existing rotation-known methods, it is notable that the proposed system is a complete 3D SLAM system using a single stereo camera. In addition, both the real-time camera pose estimation process and the back-end pose graph optimization process benefit from the global orientational estimation.

III. System Overview

Throughout this paper, we use $\{\mathbf{R}_{iw}, \mathbf{t}_{iw}\}$ to represent the camera pose of the ith frame relative to the world frame, where the rotation matrix $\mathbf{R}_{iw} \in SO(3)$ represents the camera orientation and the translation vector $\mathbf{t}_{iw} \in \mathbb{R}^3$ represents the position. We also use three unit vectors $\{\mathbf{d}_k^w\}_{k=1}^3$ to represent the three mutually orthogonal dominant directions, and $\mathbf{MF}^w = [\mathbf{d}_1^w|\mathbf{d}_2^w|\mathbf{d}_3^w]$ to represent the global MF. Besides, $\{\mathbf{d}_k^i\}_{k=1}^3$ are defined as the observed directions of $\mathbf{MF}^i = \left[\mathbf{d}_1^i|\mathbf{d}_2^i|\mathbf{d}_3^i\right]$ in the ith frame.

Our SLAM system takes the stereo camera as the sensor input. The overall framework is depicted in Fig. 1, which adopts ORB-SLAM2 as its basic architecture [2]. In particular, both a front end and a back end are included. The front end serves as a real-time VO, while the back end are launched to perform time-consuming processes including local mapping and loop closing only when a new keyframe is inserted.

We next specify each module of the proposed SLAM system. In the first place, the front end is responsible to provide ego-motion estimation and keyframe decision for

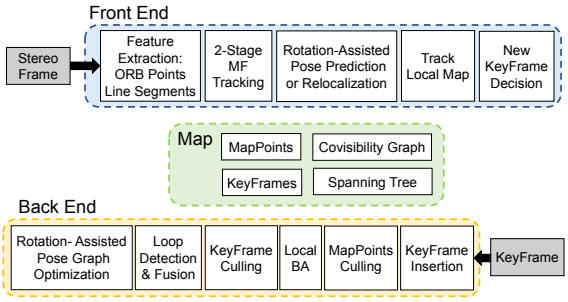


Fig. 1. The framework of the proposed SLAM system.

each newly added frame. The point features and line features are extracted in parallel. The extracted line segments are then utilized to estimate the MF in current frame by a two-stage method, which provides an absolute rotational motion estimation and returns the VPs with the corresponding clustered line features. Given the global orientation, the camera pose is initially predicted in an efficient and accurate way by tracking previous frame with the assistance of the known rotation, or it is relocalized when tracking is failed due to abrupt movements. Based on the result of the above initial estimation, the camera motion is then rectified with the matched landmarks from the local map. The decision on the new keyframe is executed at the end of the front end. When current frame is decided to be a new keyframe, the observed 3D points will be recovered.

In the second place, local mapping process and loop closing process are implemented in the back end. The stored map consists of 3D map points, a set of keyframes, a covisibility graph and a spanning tree. Specifically, the covisibility graph is maintained to link two keyframes with a number of covisible landmarks, and a minimum spanning tree connecting all keyframes is derived from the covisibility graph. The local mapping consists of map point culling, local BA and keyframe culling. Particularly, the map point culling module is carried out to release the storage burden and retain map points with high quality. Based on this result, the local BA optimizes the poses of the newly inserted keyframe and all the local keyframes that are connected to it in the covisibility graph, together with all the 3D landmarks observed by these keyframes. Then, a keyframe culling policy is applied to remove the low quality keyframes. Last but not the least, when a new keyframe is inserted, the loop detection module searches for loop candidate keyframes based on their visual descriptor derived by the technique of bags of words (BoW) [25]. Once a loop is detected, the relative poses of the current keyframe and loop keyframes are estimated, and the duplicated map points in these keyframes are fused. Finally, a rotation known pose graph optimization is conducted to correct the accumulated drift. Also note that we adopt the same strategy to perform local mapping and loop detection as ORB-SLAM2 [2].

IV. Methodology

A. Two-Stage MF Tracking

In this section, we describe the details of the two-stage MF tracking process applied in this paper. Using a single stereo camera, the MF tracking is achieved by VPs estimation from detected line segment features in image. Note that VPs can be computed given the line clustering results, and the line clustering can be obtained in turn if the VPs are estimated. Based on this fact, we design a robust two-stage MF tracking method which essentially consists of a long-term searching module and a short-term tracking module. The long-term module performs time-consuming searching, which helps to modify the results from short-term tracking timely and eliminate the influence of scene changes, while the time-saving short-term module keeps tracking the clustered lines within several frames. Therefore, the combination of these two modules guarantees an efficient and accurate global orientation estimation.

We first extract line segment features by means of line segment detector (LSD) algorithm [26], which provides good repeatability and subpixel precision results. To deal with various scenes and increase the robustness of line segment matching in the MF inference problem, we merge the line segments belonging to the same straight line which may be divided to several short parts. Besides, the line segments with comparative short length are filtered out.

The long-term searching module first takes the extracted line segments as input and estimates three orthogonal VPs with corresponding line feature clusters in image plane. Then the dominant directions of MF are recovered by a least square algorithm given the line feature clustering results.

In particular, the method proposed by [18] is adopted to provide the initial VPs estimation. Gaussian sphere is used here to represent the parameter space of rotation. As is shown in Fig. 2, a 3D line is projected onto the Gaussian sphere as a great circle, and two great circles of parallel lines intersect at a direction, which is regarded as a candidate VP direction ($\mathbf{v}_1$). In practice, a polar grid that spans the latitude and longitude is built on a half Gaussian sphere with a size of 90×360 and an accuracy of 1°. Each pair of two line segments contributes a score to the polar grid cell that the intersection belongs to:

$$score = \|\boldsymbol{l}_1\| \times \|\boldsymbol{l}_2\| \times \sin(2\theta), \tag{1}$$

where $\|\boldsymbol{l}_1\|$ and $\|\boldsymbol{l}_2\|$ represent the length of two line segments in pixel and θ is the angle between them.

Taking the mutual orthogonal constraints into consideration, it generates 360 evenly distributed candidates VPs ($\{\mathbf{v}_2^i\}_{i=1}^{360}$) in the great circle that is perpendicular to the first VP direction ($\mathbf{v}_1$). Finally, the third candidates VPs directions ($\{\mathbf{v}_3^i\}_{i=1}^{360}$) is calculated by the cross product of each pair of $\mathbf{v}_1$ and $\{\mathbf{v}_2^i\}_{i=1}^{360}$. From the abundant sets of VPs candidates, the one with the highest score is retrieved as the optimal one, where the score of a VPs hypothesis set is the sum score of three polar grid cells that belong to the three corresponding VP directions. Readers are suggested to refer [18] for more details about this scored strategy.

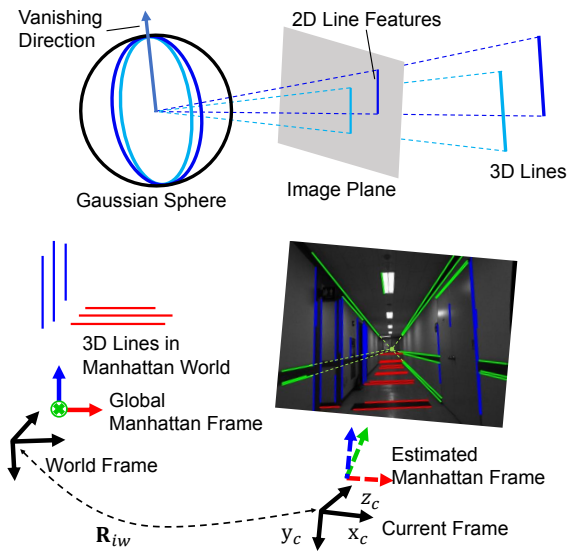


Fig. 2. The MF is modeled from the mutual orthogonal 3D lines in the environment, whose projection in image can be clustered into three groups (shown in different colors) with three corresponding orthogonal VPs. A vanishing direction can be estimated by two great circles in Gaussian sphere projected from two parallel lines. Given the inferred MF in current frame and that in global frame, the rotational motion is obtained.

Given the optimal estimated set, line clustering can be obtained easily by checking the angular difference between itself and the line determined by its midpoint and the VP in image. A line segment is determined as a structural line if the angular difference is smaller than a given threshold (2° in this paper), otherwise discarded as outlier.

We next estimate the accurate MF in current frame given at least two line clusters with abundant liners. In particular, a dominant direction $\mathbf{d}$ is solved through a least square problem:

$$\mathbf{S}^\top \mathbf{d} = \mathbf{0}, \tag{2}$$

where $\mathbf{S} \in \mathbb{R}^{3\times n}$ and n is the number of lines in this cluster. Each column $\mathbf{s}_j$ in $\mathbf{S}$ represents a normal vector of a great circle in the Gaussian sphere [19]. Since the great circle is the intersection between the Gaussian sphere and the plane passing through the jth line feature in image plane and current camera center, its normal vector can be obtained as $\mathbf{s}_j = \mathbf{K}^\top \boldsymbol{l}_j$, where $\mathbf{K}$ is the known camera intrinsic matrix and $\boldsymbol{l}_j$ is the equation of the jth line. In particular, $\boldsymbol{l}_j$ can be calculated by the cross product between two endpoints of the jth image line in homogeneous coordinates. Finally the solved directions by (2) are orthogonalized to represent $\{\mathbf{d}_k^i\}_{k=1}^3$ and $\mathbf{MF}^i$ in the ith frame.

The short-term tracking module efficiently uses the clustering results from last frame to simplify the estimation process by calculating the VPs from known line clustering. Particularly, line band descriptor (LBD) [27] is employed to find correspondence between line features, which provides binary descriptor based on the local appearance of selected line segments. We propose to find pairwise matched line features between consecutive frames taking both the line band descriptor and geometric constraints into consideration. Specifically, we take advantage of the useful geometrical information including orientation and disparity of endpoints to check the matching results, and filter out the mismatching lines while preserving the computational efficiency.

The successfully matched line features make up the initial clusters and are utilized to calculate the initial VPs estimation by solving a least square problem. Then, more line features are clustered by checking the angular difference with the initial estimated VPs. The final VPs are determined by solving a least square problem given the updated line clusters as in the long-term module.

Given $\mathbf{MF}^i$ and the MF in global frame $\mathbf{MF}^w$, the rotational motion is estimated by:

$$\mathbf{R}_{iw} = \mathbf{MF}^i \cdot (\mathbf{MF}^w)^{-1} = \mathbf{MF}^i \cdot (\mathbf{MF}^w)^\top. \tag{3}$$

Typically, in the initialization of the SLAM process, we set the first frame contains more than two line clusters with enough inliers as the world frame and set the MF estimated in this initial frame as the global one ($\mathbf{MF}^w$). It is noted that the combination of long-term searching module and short-term tracking module guarantees the consistency of MF estimation. If tracking is lost, we align the MF in current frame with the global MF using the method in [19] that provides rotation estimation with consistency.

B. Rotation-Assisted Ego-Motion Estimation

Traditionally, the camera ego-motion estimation is solved by minimizing the cost function based on reprojection error in feature-based SLAM system, i.e., it is a BA problem that optimizes the 6-DoF frame-to-frame camera motion:

$$\{\mathbf{R}_{iw}, \mathbf{t}_{iw}\} = \arg\min_{\mathbf{R},\mathbf{t}} \sum_{j\in\chi} \rho(\|\mathbf{x}_j - \pi(\mathbf{R}\mathbf{P}_j + \mathbf{t})\|^2_{\Sigma_j}), \tag{4}$$

with the robust Huber cost function ρ and the covariance matrix Σ_j that represents the weight. In fact, the data association is performed between each pair of point feature $\mathbf{x}_j = [u_j, v_j]^\top \in \mathbb{R}^2$ in image plane and the matched 3D map point $\mathbf{P}_j \in \mathbb{R}^3$ in world coordinate, with a set of successful matches χ. The definition of the projection function π is:

$$\pi\left(\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}\right) = \begin{bmatrix} f_x \frac{P_x}{P_z} + c_x \\ f_y \frac{P_y}{P_z} + c_y \end{bmatrix}, \tag{5}$$

where the focal length (f_x, f_y) and principal point (c_x, c_y) belong to camera intrinsic parameters.

It is obvious that the known global orientation estimation $\hat{\mathbf{R}}_{iw}$ helps to improve the accuracy and convergence speed of solving the translational motion in BA problem as:

$$\mathbf{t}_{iw} = \arg\min_{\mathbf{t}} \sum_{j\in\chi} \rho(\|\mathbf{x}_j - \pi(\hat{\mathbf{R}}_{iw}\mathbf{P}_j + \mathbf{t})\|^2_{\Sigma_j}). \tag{6}$$

In addition, the availablity of the global orientation estimation simplifies the original non-linear optimization problem into a linear one:

$$\mathbf{t}_{iw} = \arg\min_{\mathbf{t}} \sum_{j\in\chi} \|\omega_j \mathbf{e}_j(\mathbf{t})\|^2, \quad \text{with} \quad \mathbf{e}_j(\mathbf{t}) = ((\hat{\mathbf{R}}_{iw}\mathbf{P}_j)^{(3)} + \mathbf{t}^{(3)}) \begin{bmatrix} \frac{u_j - c_x}{f_x} \\ \frac{v_j - c_y}{f_y} \end{bmatrix} - \begin{bmatrix} (\hat{\mathbf{R}}_{iw}\mathbf{P}_j)^{(1)} + \mathbf{t}^{(1)} \\ (\hat{\mathbf{R}}_{iw}\mathbf{P}_j)^{(2)} + \mathbf{t}^{(2)} \end{bmatrix}, \tag{7}$$

where we refer $[\cdot]^{(k)}$ as the kth row of a vector and ω_j represents the weight that is associated to the scale of the detected keypoint [2]. Since (7) is quadratic in $\mathbf{t}$, we can rewrite it as:

$$\min_{\mathbf{t}} \|\mathbf{A}_t\mathbf{t} - \mathbf{b}_t\|^2, \tag{8}$$

where the known matrix $\mathbf{A}_t$ and vector $\mathbf{b}_t$ are built from the data association. Since (8) is a linear least square problem, its solution can be found by solving the normal equation:

$$(\mathbf{A}_t^\top \mathbf{A}_t)\mathbf{t} = \mathbf{A}_t^\top \mathbf{b}_t. \tag{9}$$

In practice, we apply the widely-used ORB method [28] to obtain point feature detection and binary description. In the considered BA formulation, the robust Huber cost function reduces the error produced by outliers, while the iterative process helps to remove outliers and find estimation from almost no outliers in the final iteration.

To deal with outliers, we conduct a RANSAC framework instead of solving (9) directly. Since each matched feature provides two independent constraints on $\mathbf{t}_{iw}$ in (9), the RANSAC method samples two features at each iteration and finds the maximum inlier set from all measurements after iterations. Then the translational motion estimation is obtained by solving (9) on the inlier set.

In summary, the global orientation estimation benefits the translational motion estimation in two ways, i.e., the rotation-assisted BA method which solves the BA problem with the known global rotation estimation and the RANSAC based framework that is solved through a linear least square problem.

C. Rotation-Assisted Pose Graph Optimization

In the back end, the loop detection is conducted on each newly added keyframe, which is achieved by comparing the BoW vector formed from ORB features between frames. Once the loop closure is validated, the relative motions between current frame and its corresponding loop keyframes are obtained. Typically, the accumulated error can be reduced by performing an optimization on the closed pose graph. In this paper, we conduct an optimization on the essential graph [2]. The essential graph regards all the keyframes as nodes and preserves a strong network of them. Three types of connections are contained in the essential graph, i.e., the edge connecting the loop, the basic spanning tree, and the other edges with high covisibility. Since the use of stereo images makes scale observable, there is no need to deal with scale drift. Traditionally, the nonlinear pose graph optimization is conducted on SE(3) poses which minimizes the cost function [2]:

$$\min_{\{\mathbf{T}_i\}\in SE(3)} \sum_{(i,j)\in\chi_{EG}} \|log_{SE(3)}\left(\bar{\mathbf{T}}_{ij}\mathbf{T}_{jw}\mathbf{T}_{iw}^{-1}\right)\|^2, \tag{10}$$

where $log_{SE(3)}$ is the log mapping in SE(3) and χ_{EG} represents all edges belonging to the essential graph with $\bar{\mathbf{T}}_{ij}$ the corresponding measurement in a certain edge. To simplify this issue, we formulate a linear pose graph optimization problem given the obtained drift-free orientation estimation:

$$\min_{\{\mathbf{t}_i\}\in\mathbb{R}^3} \sum_{(i,j)\in\chi_{EG}} \|\mathbf{t}_j - \hat{\mathbf{R}}_j\hat{\mathbf{R}}_i^\top \mathbf{t}_i - \bar{\mathbf{t}}_{ji}\|^2, \tag{11}$$

where $\bar{\mathbf{t}}_{ji}$ is the corresponding measurement of the translation part and $\hat{\mathbf{R}}_i, \mathbf{t}_i$ the abbreviations of $\hat{\mathbf{R}}_{iw}, \mathbf{t}_{iw}$.

Since (11) is quadratic in each $\mathbf{t}_i$, we can rewrite it by rearranging the translational motion $\mathbf{t}_i$ of all keyframes into a single vector $\mathbf{p}$ and building the known matrix $\mathbf{A}_p$ and vector $\mathbf{b}_p$ from drift-free orientation estimations and relative motion measurements on the essential graph respectively:

$$\min_{\mathbf{p}} \|\mathbf{A}_p\mathbf{p} - \mathbf{b}_p\|^2, \tag{12}$$

whose solution can be obtained by solving the normal equation:

$$(\mathbf{A}_p^\top \mathbf{A}_p)\mathbf{p} = \mathbf{A}_p^\top \mathbf{b}_p. \tag{13}$$

Based on the solution of (13), we refine the position estimation of all keyframes.

V. Experiment Evaluation

In this section, we validate the capabilities of the proposed system by conducting experiments on both synthesized and real-world scenarios. In particular, we evaluate the accuracy and efficiency compared with state-of-the-art approaches. All these experiments have been run on a desktop with Intel Core i7-6700 @ 3.40GHz and 8 GB RAM.

A. Synthesized Scenes

To validate the proposed rotation-assisted ego-motion estimation method, we generate a 30 m × 30 m × 4 m four-side fence, including 100 lines (80 vertical, 20 horizonal) and 400 points. A stereo camera moves in a 6-DoF ellipse trajectory with a periodic change in the pitch and roll angles inside the fence and is always facing towards it. The stereo camera has a baseline of 0.1 m, and the size of each camera is 640 × 480 pixels with the focal length of 350 pixels and the principle point located at the center of image. A total of 600 sequential images with 1-pixel Gaussian measurement noise are generated by projecting the synthesized fence to the camera, as is depicted in Fig. 3.

Since there is no available descriptor for each line segment, we only use the long-term searching method to detect the VPs in the simulated experiment. It is efficient enough for the simulation because there is a small quantity of line features in image and the complexity of exhaustive searching is low. Then, we evaluate three methods for comparison: (i) the rotation-assisted BA method (abbreviated as R-BA)

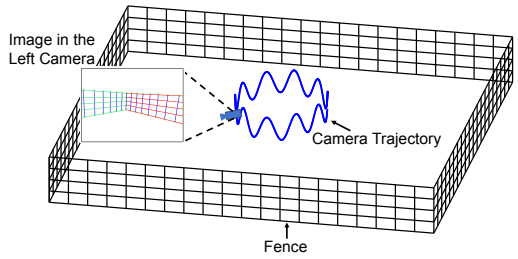


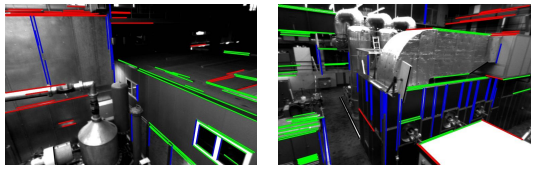
Fig. 3. A stereo camera moves inside a four-side fence to generate synthesized frames. The line segments in image are clustered as three groups and used to estimate the VPs.

TABLE I
RMSE (m) IN SIMULATION WITH DIFFERENT CONDITIONS

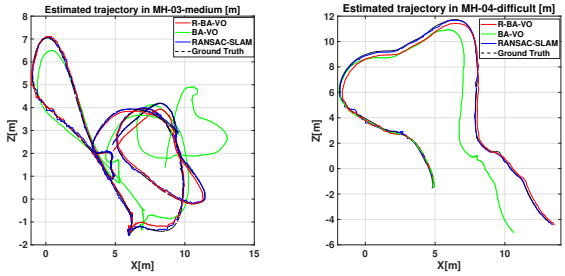
Conditions	RANSAC	R-BA	BA
Pixel noise only	0.071	**0.048**	0.514
Rotational noise	0.147	**0.137**	0.514
Reduced lines	0.108	**0.096**	0.864
Mismatch	**0.078**	0.104	0.562
Rotational noise and mismatch	**0.149**	0.158	0.562

which solves the BA problem with the known global rotation estimation; (ii) the RANSAC-based framework that decouples translation from rotation and solves it through a linear least square problem; (iii) the typical BA-based method that uses the estimation from last frame as an initial guess. In practice, we solve the BA problem (4) and (6) using the Levenberg–Marquardt method implemented in g2o [29], and solve the normal equation (9) using Singular Value Decomposition (SVD) method.

To investigate the robustness of the rotation-assisted ego-motion estimation method, we disturb the orientation estimation with 1-deg Gaussian noise on all three rotation axes to validate the sensitivity of the proposed methods to the orientation estimation, and we reduce the lines in the simulated fence to explore the robustness of the MF tracking approach to the lack of lines in the environment. Besides, we mismatch 20% of the point features from their corresponding 3D landmarks to validate the efficiency of handling outliers. Experiments are conducted on these different conditions using the above three methods to evaluate the performance. The absolute translation root-mean-square error (RMSE) [31] of the trajectory compared with the ground truth for over 10 runs is calculated to provide a quantitative analysis, which is presented in Table I. In summary, the simulation results show that with the utilization of global rotation estimation, the RANSAC-based method and the R-BA method outperform the typical BA-based one. In particular, the RANSAC-based method performs better than the R-BA method does in the case with a large number of outliers. Besides, both of the rotation-assisted ego-motion estimation accuracy become worse with noisy rotation estimation since the rotational noise brings error to the mapping process and in turn perturbs the translational estimation. In addition, sometimes there are only two observable structural directions with 8 vertical lines and 3 horizonal lines in the case with reduced lines, which is challenging and also brings error to the rotation estimation.



(a) Structural regularity in the machine hall



(b) MH-03-medium (c) MH-04-difficult

Fig. 4. The scenes in the machine hall and the experiment results in this man-made environment. (a) The perceived structural regularity in the machine hall. (b) The estimated trajectory of RANSAC-based SLAM method (abbreviated as RANSAC-SLAM), R-BA based VO method (abbreviated as R-BA-VO) and typical BA-based VO method (abbreviated as BA-VO) in the MH-03-medium sequence. (c) The estimated trajectory in the MH-04-difficult sequence.

B. Real-World Scenes

We conduct experiments to test our methods on EuRoC dataset [30] and it3f dataset [6], which have been proposed as benchmarks for SLAM problems.

1) EuRoC Dataset: The EuRoC dataset [30] are collected by a flying MAV, which contains 11 stereo sequences recorded in two rooms with different scenarios and one industrial environment in large scale. These videos in man-made environment contain different challenging scenarios due to illumination changes and drastic motion of the drone. On the other hand, the windows, tubes, and other background in the machine hall provide abundant structural line features as is shown in Fig. 4(a). Therefore, we test the proposed rotation-assisted ego-motion estimation methods on part of the image sequences recorded in the machine hall.

We evaluate the RANSAC-based method and R-BA based method both in the VO pipeline (the front end in Fig. 1) and in the SLAM pipeline (both the front end and the back end in Fig. 1). A VO using the typical BA-based method, a VO using the typical BA-based method with the estimated rotation as initial guess and ORB-SLAM2 [2] are all tested for comparation. Table II shows the absolute translation RMSE [31] of the compared VO methods and SLAM methods, and the estimated trajectories of different methods in the MH-03-medium and MH-04-difficult sequences are presented in Fig. 4(b) and Fig. 4(c). The results indicate that the VO methods utilizing the proposed rotation-assisted ego-motion estimation outperform the typical BA-based one and the one with the estimated rotation as initial guess. Between the two proposed rotation-assisted ego-motion estimation methods, the R-BA-based one performs better than the RANSAC-based one does. It is reasonable since the drift-free rotation estimation provides the global constraints and avoids the

TABLE II
RMSE IN THE EuRoC MAV DATASET

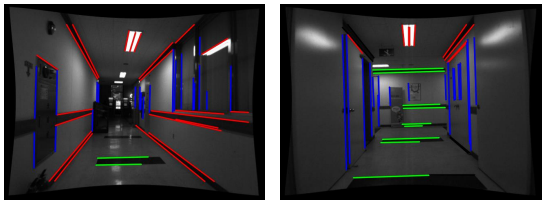
Sequences	MH01	MH02	MH03	MH04	MH05
Methods	The Absolute Translation RMSE (m)				
R-BA-VO	**0.113**	**0.067**	**0.168**	**0.138**	**0.073**
RANSAC-VO	0.117	0.068	0.169	0.139	0.074
BA-VO	0.308	0.135	1.260	0.821	0.990
BA-VO w/ initial guess	0.294	0.129	0.266	0.767	0.874
R-BA-SLAM	0.104	0.049	0.072	**0.073**	0.068
RANSAC-SLAM	0.104	0.051	0.079	0.092	0.086
ORB-SLAM2	**0.022**	**0.044**	**0.042**	0.082	**0.056**
	The Absolute Rotation RMSE (deg)				
MF-Tracking	**0.44**	0.93	**0.60**	**1.28**	**0.75**
BA-VO	2.01	2.60	9.54	7.36	6.46
ORB-SLAM2	0.54	**0.83**	0.92	1.35	0.97

accumulated error in camera pose estimation. Besides, the RANSAC-based and R-BA based SLAM methods achieve similar but slightly worse accuracy than ORB-SLAM2 does, due to the massive presence of noisy lines that do not have any structural information in some scenes. In addition, the rotation estimation method that tracks the MF outperforms the typical BA-based VO and ORB-SLAM2 in most of the tested sequences in term of absolute rotational error as is listed in table II. However, it is noteworthy that the typical BA-based method will be adopted when MF tracking is failed in some specific frames.

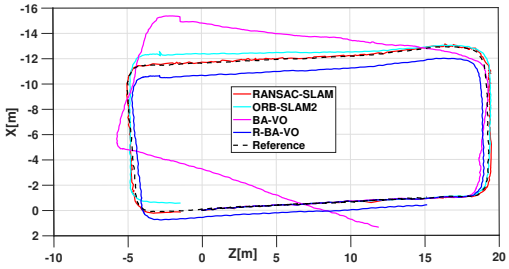
2) It3f Dataset: Experiments are also conducted on it3f dataset [6], which has an undistorted and rectified stereo image sequence recorded in an indoor environment. In particular, it is collected by a wheel robot moving flatly in an office building with abundant architectural elements including narrow corridors and low textured walls as shown in Fig. 5(a). In this sequence, the robot moves in a rectangular and comes back to the original position after a period of time, with a trajectory of 88 m in length. To provide a quantitative analysis, we regard the final camera trajectory estimation using the stereo version of ORB-SLAM2 as the reference trajectory, which is solved by the global BA after loop closure.

We first conduct experiments of the VO systems that use the RANSAC and R-BA based estimation methods respectively. Besides, the SLAM systems that use these two rotation-assisted methods are tested before the loop is closed. ORB-SLAM2 [2] and the typical BA-based VO are tested for comparation. The experiments show that the RANSAC and R-BA based methods provide pose estimation with similar accuracy when applied in both VO and SLAM systems. In particular, the tested VO and SLAM systems with assistance of global orientation estimation produce less accumulated error than the typical BA-based VO and ORB-SLAM2 do, and the illustrative results are shown in Fig. 5(b).

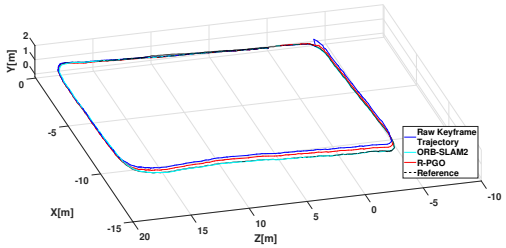
Despite the reliable motion estimation in XZ plane using the rotation assisted method, there is still an accumulated error in the vertical direction, as is illustrated by trajectory with blue line in Fig. 5(c). With the utilization of rotation-assisted ego-motion estimation method, the error in the vertical direction is the main source of accumulated error, and it can be eliminated by pose graph optimization in the back end. We conduct experiments to evaluate the



(a) Corridor and Low-textured wall



(b) Comparative results of ego-motion estimation



(c) Comparative results of pose graph optimization

Fig. 5. Some challenging scenarios such as corridor and wall with low-texture in the man-made environment of it3f dataset. The VO and SLAM systems with assistance of global orientation estimation produce less accumulated error than other methods do. And the proposed rotation-assisted pose graph optimization (abbreviated as R-PGO) efficiently correct the accumulated error in Y direction that is obvious in the raw keyframe trajectory.

TABLE III
COMPARATIVE TIMING RESULTS [ms] OF EACH PROCESS

Settings	Dataset	EuRoC		it3f	
	Resolution	752 × 480		640 × 480	
	Method	Ours	[2]	Ours	[2]
Front End	Feature Extraction	23.47	13.58	18.93	10.38
	MF Tracking	12.02	-	10.15	-
	Rotation-Assisted BA	2.33	-	2.27	-
	(RANSAC)	(5.93)	-	(5.70)	-
	BA	-	2.24	-	4.28
	Track Local Map	11.26	9.42	9.30	9.38
	Others	25.24	24.22	24.43	22.08
	Total	74.32	49.46	65.08	46.12
	(RANSAC)	(77.92)	-	(68.51)	-
Back End	Local Mapping	183.24	142.48	256.33	262.12
	Loop Detection & Fusion	184.13	176.41	177.38	196.08
	Pose Graph Optimization	24.61	342.35	27.23	656.37

proposed rotation-assisted pose graph optimization method with a comparison of that used in the stereo version of ORB-SLAM2. Given the generated essential graph, we find the translational pose estimations by solving the normal equation using the Cholesky decomposition method, while ORB-SLAM2 conducts the 6-DoF pose graph optimization using the Levenberg–Marquardt method implemented in g2o [29]. The illustrative results are presented in Fig. 5(c), where

the accumulated error of the trajectory is efficiently corrected using the proposed rotation-assisted pose graph optimization method (abbreviated as R-PGO). The trajectory optimized by our method is close to the reference trajectory and the result from ORB-SLAM2, with an absolute translation RMSE of 0.104 m. On the other hand, it costs 27 ms to perform the R-PGO while ORB-SLAM2 costs 656 ms to perform the nonlinear pose graph optimization.

3) Timing Consumption Results: Table III presents the evaluated time consumption results of each process in both the front end and the back end. Although the line feature detection and MF tracking processes increase the cost time, the whole system can still perform in nearly real time. Besides, the proposed rotation-assisted pose graph optimization strategy is more efficient than the nonlinear pose graph optimization method adopted in ORB-SLAM2.

VI. Conclusions

In this paper, we propose an accurate and efficient stereo visual SLAM system leveraging the structural regularity in man-made environments. The Manhattan world assumption is considered and a two-stage MF tracking method including a long-term VPs searching strategy and a short-term structural line tracking strategy is presented to produce global orientation estimation. In the front end, the drift-free rotational estimation is utilized to improve the accuracy of translational ego-motion prediction. In particular, the rotation-assisted BA method and the RANSAC-based least square method are designed. Besides, an efficient rotation-assisted pose graph optimization method is introduced in the back end to eliminate the accumulated error by solving a least square problem. Experiments on both synthetic environment and benchmark datasets show that the proposed system is capable of leveraging the abundant structural information to provide an efficient and accurate estimation. In the future, we will conduct more experiments on the real urban scenes with large trajectories.

References

[1] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Visual SLAM: Why filter?" *Image Vis. Comput.*, vol. 30, no. 2, pp. 65–77, 2012.

[2] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, 2017.

[3] R. Gomez-Ojeda, F. Moreno, D. Scaramuzza, and J. González-Jiménez, "PL-SLAM: a stereo SLAM system through the combination of points and line segments," *IEEE Trans. Robot.*, vol. 35, no. 3, pp. 734–746, 2019.

[4] X. Zuo, X. Xie, Y. Liu and G. Huang, "Robust visual SLAM with point and line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1775–1782.

[5] F. Camposeco, M. Pollefeys, "Using vanishing points to improve visual-inertial odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 5219–5225.

[6] G. Zhang, J. H. Lee, J. Lim, and I. H. Suh, "Building a 3-D line-based map using stereo SLAM," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1364–1377, 2015.

[7] H. Zhou, D. Zou, L. Pei, R. Ying, P. Liu, and W. Yu, "Structslam: Visual slam with building structure lines," *IEEE Trans. Veh. Tech.*, vol. 64, no. 4, pp. 1364–1375, 2015.

[8] D. Zou, Y. Wu, L. Pei, H. Ling and W. Yu, "StructVIO: Visual-Inertial Odometry With Structural Regularity of Man-Made Environments," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 999–1013, 2019.

[9] H. Li, J. Yao, J. Bazin, X. Lu, Y. Xing and K. Liu, "A Monocular SLAM System Leveraging Structural Regularity in Manhattan World," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 2518–2525.

[10] H. Li, Y. Xing, J. Zhao, et al, "Leveraging structural regularity of Atlanta world for monocular SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 2412–2418.

[11] P. Kim, B. Coltin, J. Kim H, "Linear RGB-D SLAM for planar environments," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 333–348.

[12] P. Kim, B. Coltin, J. Kim H, "Low-drift visual odometry in structured environments by decoupling rotational and translational motion," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 7247–7253.

[13] J. Straub, N. Bhandari, J. J. Leonard and J. W. Fisher, "Real-time manhattan world rotation estimation in 3D," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 1913–1920.

[14] L. Carlone, R. Tron, K. Daniilidis and F. Dellaert, "Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4597–4604.

[15] J. M. Coughlan and A. L. Yuille, "Manhattan world: Compass direction from a single image by Bayesian inference," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 941–947.

[16] J. Straub, O. Freifeld, G. Rosman, J. J. Leonard and J. W. Fisher, "The Manhattan Frame Model—Manhattan World Inference in the Space of Surface Normals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 235–249, 2018.

[17] K. Joo, T. Oh, J. Kim and I. S. Kweon, "Robust and Globally Optimal Manhattan Frame Estimation in Near Real Time," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 682–696, 2019.

[18] X. Lu, J. Yao, H. Li, and Y. Liu, "2-line exhaustive searching for real-time vanishing point estimation in Manhattan world," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2017, pp. 345–353.

[19] J. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment," *Int. J. Robot. Res.*, vol. 31, no. 1, pp. 63–81, 2012.

[20] J. Bazin and M. Pollefeys, "3-line RANSAC for orthogonal vanishing point detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 4282–4287.

[21] H. Li, J. Zhao, J. Bazin, et al, "Quasi-Globally Optimal and Efficient Vanishing Point Estimation in Manhattan World," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1646–1654.

[22] G. Schindler and F. Dellaert, "Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2004, pp. 203–209.

[23] S. Agarwal, V. Shree and S. Chakravorty, "RFM-SLAM: Exploiting relative feature measurements to separate orientation and position estimation in SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 6307–6314.

[24] S. Agarwal, K. S. Parunandi and S. Chakravorty, "Robust Pose-Graph SLAM Using Absolute Orientation Sensing," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 981–988, 2019.

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

[26] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, 2010.

[27] L. Zhang and R. Koch, "An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency," *J. Vis. Commun. Image Represent.*, vol. 24, no. 7, pp. 794–805, 2013.

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.

[29] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3607–3613.

[30] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC microaerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[31] A. Geiger, P. Lenz, and R.Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.