S-VIO: Exploiting Structural Constraints for RGB-D Visual Inertial Odometry (Supplementary Material)

Pengfei Gu, Ziyang Meng*

***Abstract*—In this supplementary materials, we provide the details of the structural line clustering. Particularly, detailed derivations of the proposed candidate interval calculation and vanishing point computation method are presented, to provide a comprehensive understanding about the algorithm. Besides, synthetic experiments are carried out to evaluate the accuracy of the proposed line initialization method leveraging the known direction information.**

I. Details of the Structural Line Clustering

As stated in the paper, we implement an improved mined-and-stabbed (MnS) algorithm for the structural line clustering. In particular, the key novelty of our method lies in the candidate interval calculation. Fig. 1 (the same as Fig. 2 in the paper) depicts the process of the candidate interval calculation method. In Fig. 1, μ represents a secant plane which intersects the Gaussian sphere at the edge of the spherical cap ω. Suppose that Q is the intersection point of μ and $\vec{OS}$, and R is the intersection point of π and the edge of ω. Then the edge of ω can be regraded as a circle centered at Q, and the tangent point T between the circle and the horizontal dominant plane can be parameterized by the angle α between $\vec{TQ}$ and $\vec{TR}$. We define two unit base vectors $\vec{b_1} = (\vec{z} \times \vec{n})/||\vec{z} \times \vec{n}||$ and $\vec{b_2} = \vec{n} \times \vec{b_1}$. Then the tangent vector $\vec{v}_1$ of the circle at T and the vector $\vec{v}_2 = \vec{OT}$ can be expressed as follows:

$$
\begin{aligned}
\vec{v}_1 &= \cos(\alpha) \cdot \vec{b}_1 - \sin(\alpha) \cdot \vec{b}_2 \\
\vec{v}_2 &= \cos(\theta) \cdot \vec{n} + \sin(\theta) \cdot \Big(\cos(\alpha) \cdot \vec{b}_2 + \sin(\alpha) \cdot \vec{b}_1\Big)
\end{aligned} \tag{1}
$$

At the tangent point T, the vector $\vec{v}_1$, $\vec{v}_2$ and $\vec{z}$ are co-planar, resulting in the following equation:

$$
(\vec{v}_1 \cdot \vec{z}) \times \vec{v}_2 = 0 \tag{2}
$$

Combining Eq. (1) and Eq. (2), we can obtain:

$$
\begin{aligned}
(a-d) \cdot x^4 + 2a \cdot x^2 + a + d &= 0 \\
a = b_{2,x} \cdot sin(\theta) \cdot b_{1,y} &- b_{2,y} \cdot sin(\theta) \cdot b_{1,x} \\
d = n_x \cdot cos(\theta) \cdot b_{1,y} &- n_y \cdot cos(\theta) \cdot b_{1,x}
\end{aligned} \tag{3}
$$

where $x = tan(\alpha/2)$ is the unknown variable. Note that Eq. (3) provides the closed-form solution. When $d <= a$, there is no solution, corresponding to the case depicted in Fig. 1. Otherwise, Eq. (3) has two opposite roots $\alpha^* = \pm 2{\cdot}\mathrm{arctan}\left(\sqrt{\frac{\mathrm{a+d}}{\mathrm{d-a}}}\right)$. After computing the angle α^*, the bound of the candidate interval can be easily obtained as follows:

$$
\begin{aligned}
\beta &= arctan(-\frac{q_x}{q_y}) \\
\vec{q} &= \vec{n} \cdot cos(\theta) + sin(\theta) \cdot (cos(\alpha^*) \cdot \vec{b}_1 + sin(\alpha^*) \cdot \vec{b}_2)
\end{aligned} \tag{4}
$$

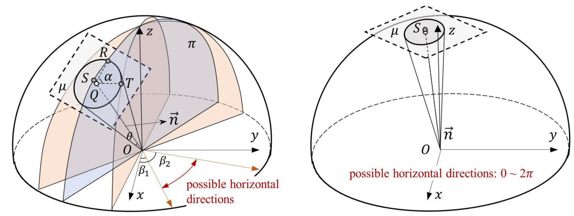

Fig. 1. Two situations for the candidate interval calculation. In most cases (right), the normal vectors (red arrows) of two vertical planes which are tangential to the circle centered at Q constitute the candidate interval for possible horizontal directions. In some cases (left), the sphere point S is too close to the z-axis, which leads to a candidate interval from 0 to 2π.

After obtaining the structural line clusters, we compute the vanishing point for each line cluster and use it to match with the HDD. The computation of the vanishing point is based on the minimization of the following cost:

$$
\mathcal{C} = \sum_i \cos^2(\gamma_i) = \sum_i \left(n_i^T v\right)^2 = v^T \sum_i \left(n_i n_i^T\right) v \tag{5}
$$

where v is a unit vector representing the vanishing point, n_i is the unit normal vector of the i-th line feature and γ_i is the angle between v and n_i. Taking v as the eigenvector corresponding to the smallest eigenvalue of $\sum_i \left(n_i n_i^T\right)$ gives the global minimum of Eq. (5).

II. Evaluation of Direction-aided Line Initialization

Previous works generally use the intersection of two planes to estimate the initial position of the line feature in the world/reference frame [1], [2]. In this section, we preform a synthetic experiment to compare the accuracy of the proposed direction-aided line initialization method with the commonly used plane-intersection-based method. Particularly, we implement the n-view line initialization method proposed in [3] as the representation of the traditional plane-intersection-based method. Compared with the two-view line initialization method generally used in many odometry systems, the n-view approach avoids the degeneracy problem by aggregating multiple views' measurements. This n-view method is then followed by an intersection of the estimated line and the projection

*Corresponding author: Ziyang Meng.
The authors are with the Department of Precision Instrument, Tsinghua University, Beijing 100084, China (e-mail: gpf20@mails.tsinghua.edu.cn; ziyangmeng@mail.tsinghua.edu.cn).

TABLE I
THE CONFIGURATION FOR THE SYNTHETIC LINE ENVIRONMENT

configuration		[x, y, z] / m	[yaw, pitch, roll][1] / deg
A	camera 1	-5, 0, 0	0, 0, 0
	camera 2	-5, 2, 0	0, 0, 0
	camera 3	-5, -2, 0	0, 0, 0
	camera 4	-5, 0, 2	0, 0, 0
	camera 5	-5, 0, -2	0, 0, 0
B	camera 1	-5, 0, 0	0, 0, 0
	camera 2	-5, 2, 0	45, 0, 0
	camera 3	-5, -2, 0	-45, 0, 0
	camera 4	-5, 0, 2	0, 45, 0
	camera 5	-5, 0, -2	0, -45, 0

1: Here the rotation adopts the ZYX Euler angles representation.

TABLE II
THE MAE [CM] OF LINE INITIALIZATION IN THE SYNTHETIC EXPERIMENT

configuration	A			B		
& noise std.	$0.1°$	$0.5°$	$1.0°$	$0.1°$	$0.5°$	$1.0°$
[3]	0.93	4.66	9.53	0.93	4.66	9.52
ours	0.80	4.09	8.72	0.81	4.12	8.79

plane to calculate the initial two-parameter expression for the structural line features, which is also the approach adopted by the structural VIO systems [4], [5]. Both methods are implemented in C++ and tested on a desktop computer with an Intel Core i9 (3.6GHz) CPU and 32 GB RAM.

As shown in Table I, we generate a synthetic environment which contains (a) 10000 lines generated randomly in terms of both the direction and the position but with a constraint that the distance of the line and the origin of the world frame must be lower than 2 meters, (b) five fixed cameras with known poses. In the experiment, we project the lines onto the image plane of five cameras and perturb the normal vector of the projected line feature by the zero-mean Gaussian noise. Then the two methods are used to calculate the initial two-parameter expression of these line features. It should be noted that the direction of these lines are known for both methods, and all five views' measurements are used in these two methods to initialize the line. After the line initialization completes, we compute the distance between the estimated line and the true line to reflect the accuracy of the line initialization algorithm.

Table II reports the mean absolute error (MAE) of the distance between the estimated line and the true line. Under the lowest noise level (i.e. $0.1°$), both methods can produce accurate line estimation results very close the true value, which demonstrates the correctness of both methods. Also, it is observed that the proposed direction-aided method obtains a smaller error than the traditional n-view plane intersection-based method does under different noise levels. This shows the effectiveness of our method for structural line initialization. Besides, we also record the average computation time of the two methods. The proposed method consumes 0.18 ms for a single line initialization from five views, slightly slower than the traditional method does (0.12 ms).

REFERENCES

[1] Y. He, J. Zhao, Y. Guo, W. He, and K. Yuan, "Pl-vio: Tightly-coupled monocular visual–inertial odometry using point and line features," *Sensors*, vol. 18, no. 4, 2018.

[2] Q. Fu, J. Wang, H. Yu, I. Ali, F. Guo, Y. He, and H. Zhang, "Pl-vins: Real-time monocular visual-inertial slam with point and line features," *arXiv:2009.07462*, 2020.

[3] S. J. Lee and S. S. Hwang, "Elaborate monocular point and line slam with robust initialization," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1121–1129.

[4] H. Wei, F. Tang, Z. Xu, C. Zhang, and Y. Wu, "A point-line vio system with novel feature hybrids and with novel line predicting-matching," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8681–8688, 2021.

[5] H. Wei, F. Tang, Z. Xu, and Y. Wu, "Structural regularity aided visual-inertial odometry with novel coordinate alignment and line triangulation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 613–10 620, 2022.