Quasi-globally Optimal and Real-time Visual Compass in Manhattan Structured Environments (Supplementary Material)

Pyojin Kim[1], Haoang Li[2], Kyungdon Joo[3]

***Abstract*— In this supplementary material, we provide the mathematical derivations of how to calculate the candidate intervals explained in the paper a bit more details. Also, we define and visualize the consistency error used in the horizontal inlier lines based nonlinear optimization section. We provide additional experimental results on York Urban [1], ICL-NUIM [2], and our Tello Urban datasets. We present the detailed experimental results for each dataset, and validate the effectiveness of the proposed approach with the geometric analysis on the Gaussian sphere.**

I. Manhattan Mine-and-Stab Algorithm

Let us assume that we have the known vertical dominant direction (VDD) – $\mathbf{v}(\alpha, \beta)$ with respect to the azimuth α and elevation β (see Eq. (3) of the main paper). We first compute an orthogonal basis $\{\mathbf{s}_1, \mathbf{s}_2\}$ of the null space of $\mathbf{v}$. Specifically, $\mathbf{s}_1 = [s_1^x, s_1^y, s_1^z]$ and $\mathbf{s}_2 = [s_2^x, s_2^y, s_2^z]$ satisfy the constraints $\mathbf{s}_1^\top \mathbf{v} = 0$, $\mathbf{s}_2^\top \mathbf{v} = 0$, and $\mathbf{s}_1^\top \mathbf{s}_2 = 0$. We use the known basis $\{\mathbf{s}_1, \mathbf{s}_2\}$ and an unknown rotation angle θ to express a horizontal dominant direction (HDD) – $\mathbf{h}(\alpha, \beta, \theta)$ as follows:

$$\begin{aligned}\mathbf{h}(\alpha, \beta, \theta) = &[s_1^x \cdot \cos(\theta) + s_2^x \cdot \sin(\theta),\\ &s_1^y \cdot \cos(\theta) + s_2^y \cdot \sin(\theta),\\ &s_1^z \cdot \cos(\theta) + s_2^z \cdot \sin(\theta)]^\top\end{aligned} \tag{1}$$

Accordingly, $\{a_i, b_i\}_{i=1}^3$ and $\mathbf{t}$ in Eq. (4) of the main paper can be written as follows:

$$\begin{aligned}a_1 &= s_1^x\\ b_1 &= s_2^x\\ a_2 &= s_1^y\\ b_2 &= s_2^y\\ a_3 &= s_1^z\\ b_3 &= s_2^z\\ \mathbf{t} &= [\cos(\theta), \sin(\theta)]^\top\end{aligned} \tag{2}$$

For each sphere point $\mathbf{p}_i$, we mine its candidate interval based on the candidate region. As shown in Fig. 1, the candidate interval $[\theta_i^{\mathrm{l}}, \theta_i^{\mathrm{r}}]$ of the point $\mathbf{p}_i$ corresponds to the case that the horizontal dominant plane intersects with the candidate region edge $\mathbf{e}_i$. For writing simplification, we denote θ_i by θ hereinafter. Mathematically, the quadratic system has two distinct real solutions that are the coordinates of two plane-edge intersections. We use basic variable substitutions to eliminate the variables y and z of this system, obtaining a quadratic polynomial equation with respect to a single variable x as follows:

$$\lambda_2(\alpha, \beta, \theta) \cdot x^2 + \lambda_1(\alpha, \beta, \theta) \cdot x + \lambda_0(\alpha, \beta, \theta) = 0 \tag{3}$$

where the coefficients $\{\lambda_2, \lambda_1, \lambda_0\}$ are composed of the known α and β as well as the unknown $\cos\theta$ and $\sin\theta$. Therefore, we formulate the case that the horizontal dominant plane intersects with the candidate region edge $\mathbf{e}_i$ as the case that the quadratic polynomial in Eq. (3) has two distinct real roots. We compute the discriminant of this polynomial as $\Delta(\alpha, \beta, \theta) = {\lambda_1}^2 - 4\lambda_0\lambda_2$. In the following, we aim to find the candidate interval with respect to θ where $\Delta(\alpha, \beta, \theta) > 0$.

We first analyze the case that $\Delta(\alpha, \beta, \theta) = 0$. It corresponds to the case that the horizontal dominant plane is tangential to the candidate region edge $\mathbf{e}_i$ as shown in Fig. 1. The original $\Delta(\alpha, \beta, \theta)$ is a quartic polynomial with respect to $\cos\theta$ and $\sin\theta$. We use the power reduction formula [3] to simplify it as follows:

$$\Delta(\alpha, \beta, \theta) = A \cdot \cos(2\theta) + B \cdot \cos(4\theta) + C \cdot \sin(2\theta) + D \cdot \sin(4\theta) + E \tag{4}$$

where the known coefficients $\{A, B, C, D, E\}$ are computed by α and β. Then we substitute $\cos(4\theta) = 2\cos^2(2\theta) - 1$ and $\sin(4\theta) = 2\sin(2\theta)\cos(2\theta)$ into Eq. (4) to transform

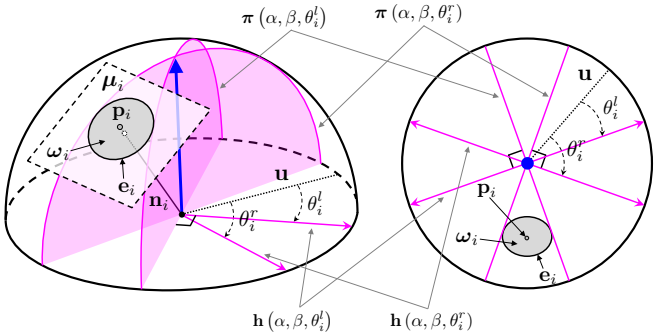


Fig. 1. 3D side view (left) and top orthographic view (right) of the Gaussian sphere with the tracked VDD (blue) and the projected i-th image line (gray dot). The sphere point $\mathbf{p}_i$ (gray dot) lies on the horizontal dominant plane $\boldsymbol{\pi}$. We expand $\mathbf{p}_i$ into the spherical cap $\boldsymbol{\omega}_i$, the candidate region to obtain the candidate interval $[\theta_i^l, \theta_i^r]$ for every image line.

[1]Pyojin Kim is with Department of Mechanical Systems Engineering, Sookmyung Women's University, Seoul, South Korea. {pjinkim}@sookmyung.ac.kr

[2]Haoang Li is with Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, China. {haoang.li.chuk}@gmail.com

[3]Kyungdon Joo is with the Artificial Intelligence Graduate School and the Department of Computer Science, UNIST, Ulsan, South Korea. kdjoo369@gmail.com, kyungdon@unist.ac.kr

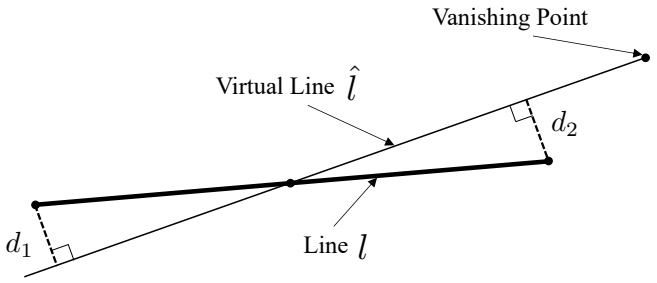


Fig. 2. Consistency error metric d_1 and d_2 from the endpoints of the line l to a virtual line $\hat{l}$ defined by the estimated vanishing point (VP) and the middle point of the l in the image plane.

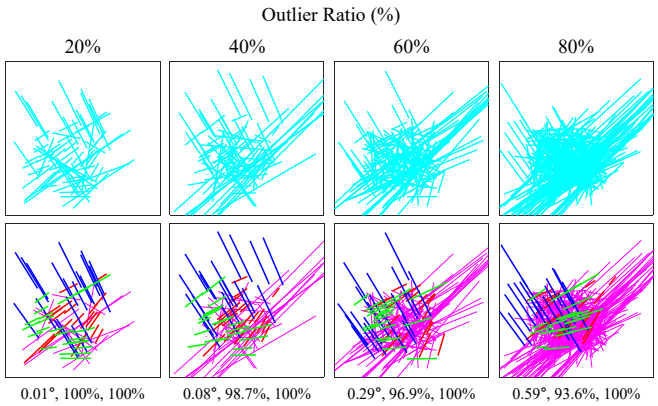


Fig. 3. Study about sensitivity to noise on the synthetic line data. The first row presents the synthetic lines (cyan) mixed with inliers and outliers corresponding to the outlier ratio from 20% to 80%. The second row shows the clustered inlier lines (red, green, and blue) with the inferred MW from the proposed method. The irregular outlier lines (magenta) are well classified as outliers that do not satisfy the MW. The numbers below images are the camera orientation error in degree, precision, and recall [6].

$\Delta(\alpha,\beta,\theta)$ as a polynomial with respect to only $\cos(2\theta)$ and $\sin(2\theta)$. Finally, we use Weierstrass substitution [3], i.e., $\cos(2\theta)=\frac{1-\tan^2\theta}{1+\tan^2\theta}$ and $\sin(2\theta)=\frac{2\tan\theta}{1+\tan^2\theta}$ to simplify $\Delta(\alpha,\beta,\theta)$ as follows:

$$\Delta(\alpha,\beta,\theta)=a\cdot\tan^4\theta+b\cdot\tan^3\theta+c\cdot\tan^2\theta+d\cdot\tan\theta+e \quad (5)$$

$\Delta(\alpha,\beta,\theta)$ in Eq. (5) is a quartic polynomial with respect to $\tan\theta$, and its known coefficients $\{a,b,c,d,e\}$ are computed by α and β. As shown in Eqs. (4) and (5), we transform the discriminant $\Delta(\alpha,\beta,\theta)$ from a trigonometric polynomial with respect to $\{\cos(4\theta),\sin(4\theta),\cos(2\theta),\sin(2\theta)\}$ to a trigonometric polynomial with respect to $\tan\theta$. Accordingly, the polynomial coefficients $\{a,b,c,d,e\}$ in Eq. (5) are expressed by the polynomial coefficients $\{A,B,C,D,E\}$ in Eq. (4) as follows:

$$a=(B-A+E) \quad (6a)$$
$$b=(2\cdot C-4\cdot D) \quad (6b)$$
$$c=(2\cdot E-6\cdot B) \quad (6c)$$
$$d=(2\cdot C+4\cdot D) \quad (6d)$$
$$e=A+B+E \quad (6e)$$

Recall that the coefficients $\{A,B,C,D,E\}$ are expressed by the azimuth α and elevation β.

We solve the real root $\tan\theta$ of the polynomial $\Delta(\alpha,\beta,\theta)$ in Eq. (5) by SVD [4] and then obtain the zero $\theta\in[-\frac{\pi}{2},\frac{\pi}{2}]$. Note that θ has two solutions $\{\theta^{\mathrm{l}},\theta^{\mathrm{r}}\}$ that both correspond to the case of tangency (see Fig. 1). Given $\{\theta^{\mathrm{l}},\theta^{\mathrm{r}}\}$, we aim to find the candidate interval corresponding to the case that $\Delta(\alpha,\beta,\theta)>0$. We compute the midpoints θ^{m} of θ^{l} and θ^{r}. If $\Delta(\alpha,\beta,\theta^{\mathrm{m}})>0$, we treat $[\theta^{\mathrm{l}},\theta^{\mathrm{r}}]$ as the candidate interval. If $\Delta(\alpha,\beta,\theta^{\mathrm{m}})<0$, we treat $[-\frac{\pi}{2},\theta^{\mathrm{l}}]\cup[\theta^{\mathrm{r}},\frac{\pi}{2}]$ as the candidate interval. Our candidate interval computation leads to $O(K)$ complexity where K is the number of image lines. It is noteworthy that the proposed Manhattan MnS utilizes the periodicity of the MW by shifting $90°$ phase of the candidate interval values between $-90°$ and $0°$. It accelerates our parameter search by reducing the search space of θ from $\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$ to $\left[0,\frac{\pi}{2}\right]$, resulting in "quasi-globally" optimal 3-DoF rotation estimate in real-time.

II. Definition of Consistency Error

The initial rotation estimate from the Manhattan MnS focuses on maximizing the number of inlier lines rather than minimizing the consistency error [5] of the inlier lines in the image plane, resulting in suboptimal 3-DoF rotation estimation in terms of the accuracy of the MF rotation. To estimate more accurate 3-DoF camera orientation, we further refine the initial rotation estimate by minimizing the average orthogonal distance with orthogonal and parallel inlier lines. The consistency error represents the orthogonal distance on the image plane in pixel from an endpoint of the image line l to a virtual line $\hat{l}$ defined by the midpoint of l and an estimated VP as illustrated in Fig. 2.

III. Additional Experimental Results

A. Synthetic Line Dataset

We have performed additional experiments to evaluate the sensitivity of the proposed method against noise with synthetic line data. We synthesize several 3D lines aligned to the MW, and project them on the virtual image plane to generate inlier lines satisfying the MW regularities (red, green, and blue) as shown in Fig. 3. We generate each of the 20 image inlier lines along the three mutually orthogonal axes of the MW (red, green, and blue), e.g., the 1-st to 20-th lines are vertical inlier lines, the 21-th to 40-th lines are the horizontal x-axis inlier lines, and the 41-th to 60-th lines are the horizontal y-axis inlier lines, respectively. We perturb the endpoints of these inlier lines by a zero-mean Gaussian noise. Then, we synthesize the irregular outlier lines (magenta) by randomizing their endpoints within the image as shown in Fig. 3. In order for the outlier ratio to be 20%, 40%, 60%, and 80%, the number of the outlier lines is 15, 40, 90, and 240, respectively.

Figure 3 shows the performance of the proposed method with respect to the outlier ratio from 20% to 80%. We fix the noise level of the endpoints as 1 pixel, and estimate the 3-DoF MF orientation with the proposed method where the VDD is given. The proposed method demonstrates high tracking accuracy and robustness despite the high outlier

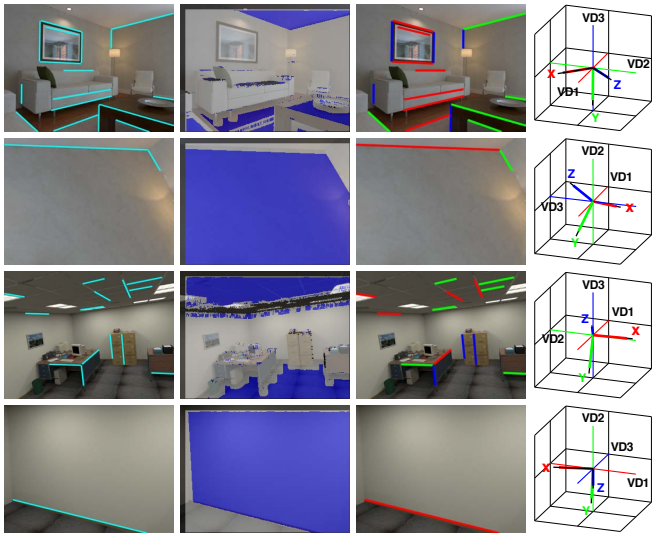


Fig. 4. Representative evaluations on ICL-NUIM [2] dataset. Each row represents a tested frame in the living room and office room datasets. Each column denotes the extracted image lines by LSD [7], tracked vertical dominant direction (typically ground plane), clustered lines in the MW, and the true (black) and estimated (RGB) 3-DoF camera orientation, respectively.

Fig. 5. Representative evaluations on our Tello Urban dataset. We plot the raw image lines (cyan) extracted by LSD [7] and the clustered lines (RGB) for the estimated MF in the image pairs. The top two rows show the image sequences during the indoor flight, and the bottom two rows are the data obtained during the flight in an outdoor urban environment.

ratio of about 80%, resulting in the 0.59 degrees rotational error and the precision is 93.6%.

B. ICL-NUIM Dataset

Fig. 4 shows the detailed experimental results on the ICL-NUIM [2] RGB-D dataset with the proposed approach. The top two rows and bottom two rows in Fig. 4 denote the 3-DoF camera tracking results in the living room and office room, respectively. The first column shows the extracted raw image lines by LSD [7], and the second column visualizes the tracked dominant plane direction (blue) with the surface normals from the depth image sequences. Without loss of generality, the dominant plane does not have to be the ground plane in the indoor RGB-D camera setting. The proposed method continues to track and update the dominant plane that is dominantly observed in the current field of view. The third column is the result of clustering the lines with respect to the estimated Manhattan frame (MF) with the proposed method. In the fourth column, colored thick and thin lines denote the estimated 3-DoF camera frame and the vanishing directions (VDs), and the black lines represent the true pose of the camera. Each of the colored lines and plane in the images corresponds to the VD of the same color.

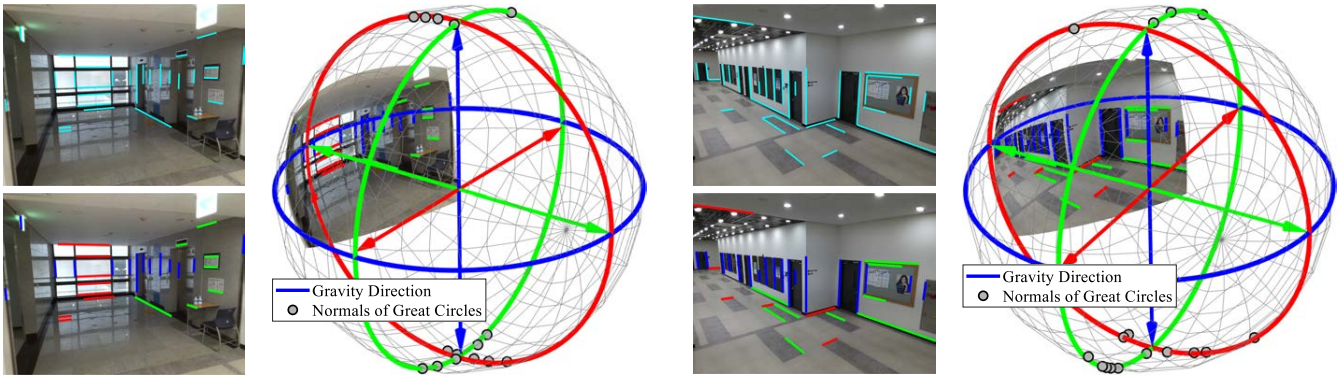
The proposed method can continue tracking the absolute camera orientation stably and accurately as shown in Fig. 4, achieving the average rotation error as 0.26 degrees. In addition, the proposed method can stably track the 3-DoF camera orientation even in a harsh environment with insufficient lines as shown in the second and fourth row of Fig. 4. Theoretically, the proposed method only requires at least a single plane for vertical dominant direction and a single line for horizontal dominant direction to track the MF rotations. The proposed method can stably track the absolute rotations even when the camera sees only a planar surface with little texture by exploiting the minimal sampling (a single line and plane) to recognize structural regularities.

C. Tello Urban Dataset

Fig. 5 and Fig. 6 show the indoor and outdoor experimental results of applying the proposed approach to the RGB image sequences and gravity direction vectors from an IMU obtained from a DJI Tello drone during the flight. We obtain the time-synchronized RGB image and the gravitational direction vector from an IMU through the ROS DJI Tello driver[1], and additionally refine the gravity direction with the vertical image lines (blue) using the data sampling-based approach [8]. In Fig. 5, each image pair consists of the raw (cyan) and clustered (RGB) image lines, and the top two rows and bottom two rows are the results of 3-DoF camera orientation estimation based on the data obtained during the indoor and outdoor flights, respectively. Existing approaches relying on the depth camera cannot operate in such a drone flight environment due to the short effective range of the depth camera and a limited field of view (FoV). The proposed method accurately and precisely clusters all image lines with respect to the estimated Manhattan frame (MF), showing that it can operate like a drift-free visual-inertial compass in a challenging outdoor flight environment.

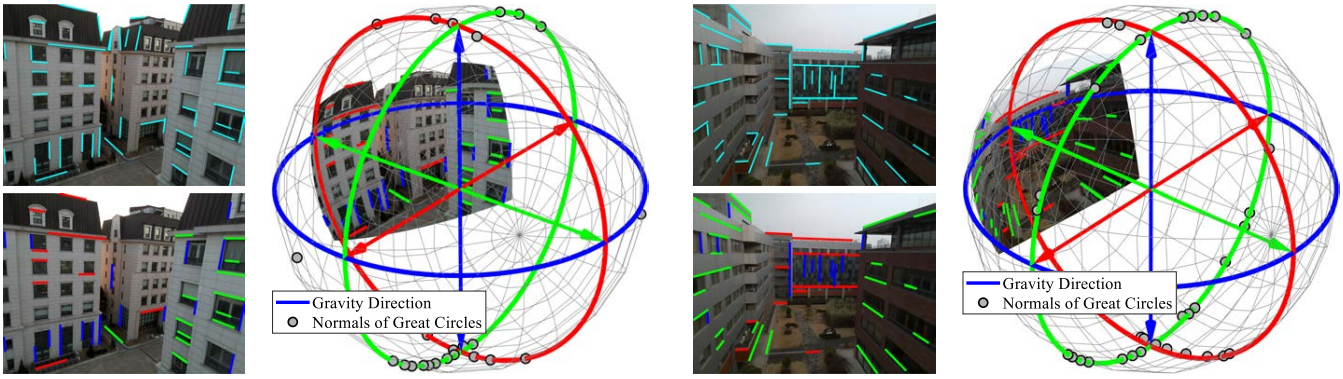
We additionally visualize the geometric relationships between the image lines and the estimated Manhattan frame (MF) on the Gaussian sphere as shown in Fig. 6. The proposed method clusters the image lines correctly in both indoor and outdoor frames, and the normal vectors of the great circles from the image lines (gray dots) are well aligned with respect to the estimated Manhattan frame (MF). We exploit the proposed Mine-and-Stab (MnS) with Manhattan

[1] http://wiki.ros.org/tello_driver



(a) Indoor Frame 9

(b) Indoor Frame 22



(c) Outdoor Frame 32

(d) Outdoor Frame 26

Fig. 6. Representative evaluations on our Tello Urban dataset with the Gaussian sphere domain. In each tested frame, we plot the raw (cyan) and clustered (RGB) image lines with respect to the estimated Manhattan frame (MF). We visualize the estimated Manhattan frame (MF) and the normal vectors of the great circles from the image lines (gray dots) on the Gaussian sphere together. The normal vectors (gray dots) are aligned well on the estimated Manhattan frame (MF) correctly. The proposed Manhattan MnS searches for the optimal horizontal direction (red and green axis) of the MW rotation, achieving the quasi-global optimality in terms of the number of inlier lines in real-time.

constraints to search for the optimal third DoF of the camera orientation, achieving the quasi-global optimality in terms of maximizing the number of inlier lines (gray dots) in real-time.

D. iOS Logger Dataset

We have performed more comparison experiments to evaluate the accuracy and robustness of the proposed method on roll and pitch camera motion as well as pure yaw rotation. We utilize the iPhone XS and custom iOS app[2] to acquire various kinds of data such as an RGB image sequence, gravity direction vector, and Apple ARKit (VIO) 6-DoF camera poses. To evaluate the accuracy of rotation estimation for each principal axis (pitch-yaw-roll), we use the DJI OM4, a 3-axis gimbal controller, to rotate the camera around the three principal axes with little translational motions as shown in Fig. 7 (a,b).

We first rotate the camera for each x-y-z axis of the camera frame (pitch-yaw-roll), and then evaluate the proposed method compared to the ARKit camera poses as shown in Fig. 7 (c). It shows almost similar rotational motion estimation performance between MWMS (magenta) and ARKit (gray) except for the last section of pure yaw motion, the frame index from 2200 to the end. As we expected, roll and pitch motion can be recovered well without drift over time by using the gravity direction [9], so both MWMS and ARKit can track the roll and pitch motions accurately in a drift-free manner. Yaw motion results, the middle row of Fig. 7 (c), show that even though the camera returns to its original position after yaw rotations, the ARKit drifts about 4 degrees, while the proposed method accurately tracks it to almost 0 degrees.

We additionally evaluate the proposed method compared to the ARKit on the 3-DoF complex rotational motion by randomly mixing roll-pitch-yaw motions as shown in Fig. 8. While the camera has yaw rotation continuously, we also add roll and pitch motion at the same time to evaluate the performance of the proposed method. Both MWMS and ARKit estimate the roll and pitch motions well similar to the previous experiment, but only the proposed method can accurately track the yaw motion without drifting at the end. Apple ARKit has a cumulative drift error of over 12 degrees at the end, whereas the proposed method estimates the final orientation of the camera accurately without a drift.

[2]https://github.com/PyojinKim/ios_logger

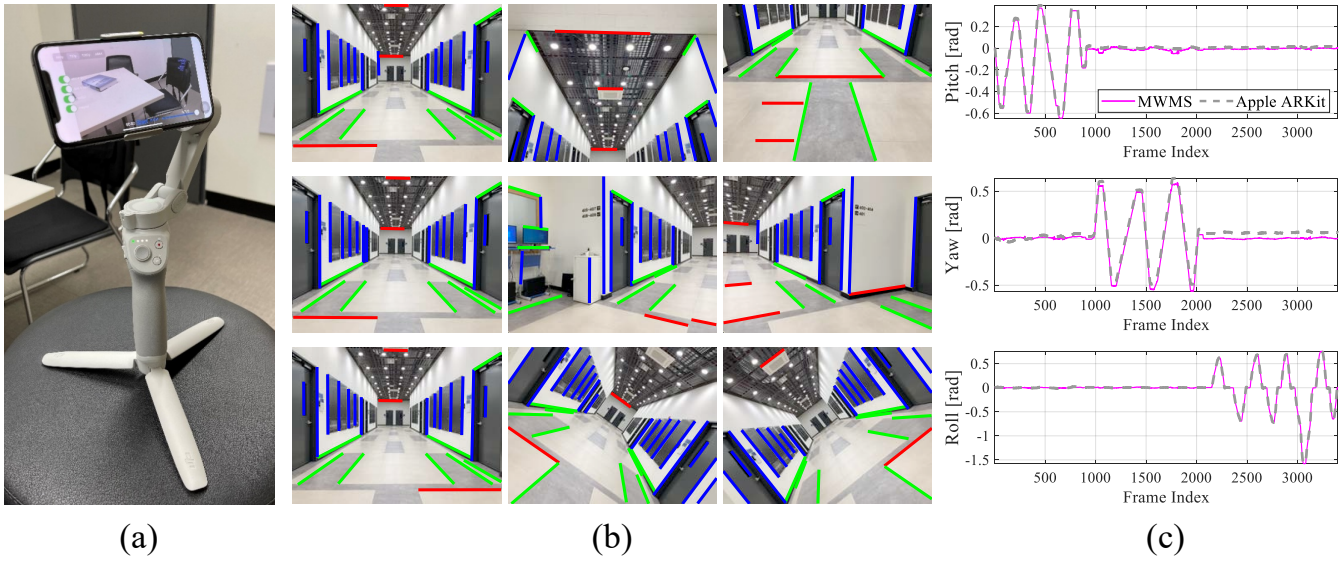


Fig. 7. Pitch-yaw-roll rotational motion sequences. (a) iPhone XS for acquiring RGB image sequences and ARKit camera poses with DJI OM4 smartphone gimbal stabilizer. (b,c) Each row represents the image sequences for pitch-yaw-roll rotational motion and corresponding rotation angle (radian) over time. Clustered lines for the inferred MW with the proposed method are overlaid on the RGB images.

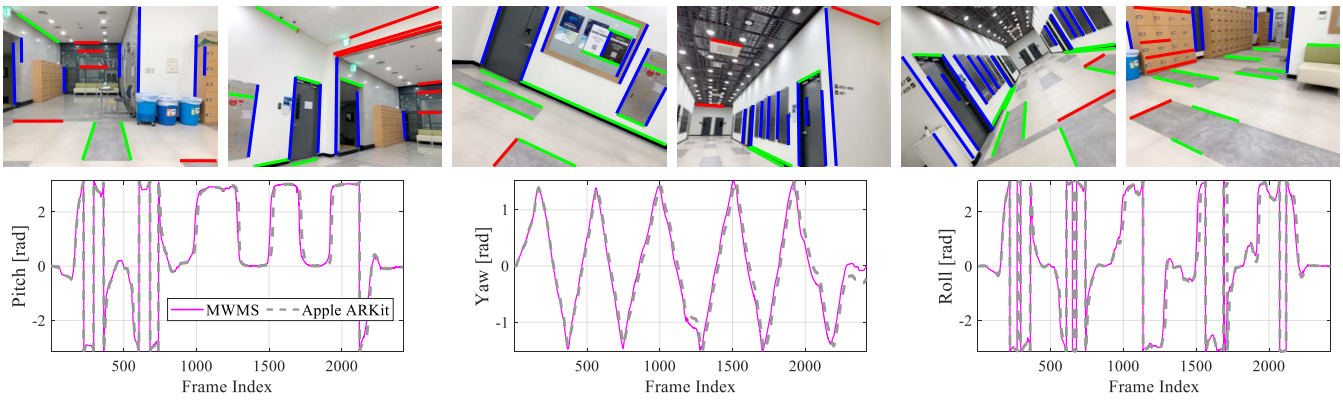


Fig. 8. Mixed roll-pitch-yaw rotational motion sequences. Representative images showing the 3-DoF camera rotations around each principal axis (top row). We cluster a set of image lines satisfying the inferred MW orientations from MWMS. Comparison results of 3-DoF rotational motion estimation for each principal axis (pitch-yaw-roll) from MWMS (magenta) and Apple ARKit (gray), showing that they overlap significantly except the yaw motion (bottom row).

E. York Urban Dataset

The Fig. 9 shows the drift-free 3-DoF camera orientation estimation results with the proposed Manhattan Mine-and-Stab (MnS) approach on the York urban dataset satisfying the Manhattan world (MW). Since the York urban dataset [1] does not provide the gravity direction data (blue in Fig. 9), we obtain the virtual gravity direction vector by utilizing the lines (blue) vertical to the ground plane with the data sampling-based approach [8]. The proposed method can estimate the accurate Manhattan frame (MF) with respect to the camera frame, and it clusters all image lines correctly. Also, as we expected, the normals of the great circles from the image lines (gray dots) are all aligned correctly on the horizontal dominant planes. The proposed method shows high efficiency and accuracy by hybridizing the data sampling and parameter search strategies, achieving the quasi-global optimality in terms of maximizing the number of inlier lines in real-time.

REFERENCES

[1] P. Denis, J. H. Elder, and F. J. Estrada, "Efficient edge-based methods for estimating manhattan frames in urban imagery," in *ECCV*, 2008.

[2] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM," in *IEEE ICRA*, 2014.

[3] W. Beyer, *CRC Standard Mathematical Tables*. CRC Press, 1987.

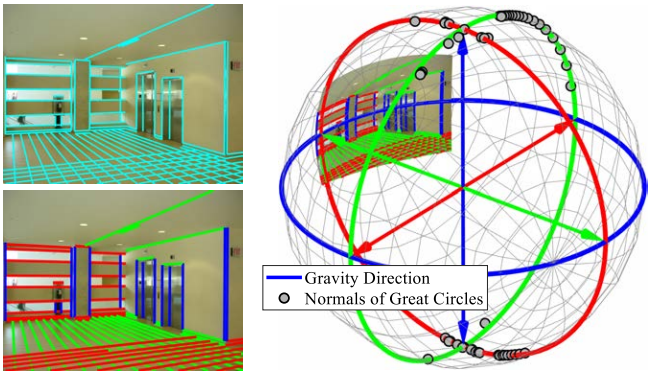
[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2003.

[5] L. Zhang, H. Lu, X. Hu, and R. Koch, "Vanishing point estimation and line classification in a manhattan world with a unifying camera model," *IJCV*, 2016.

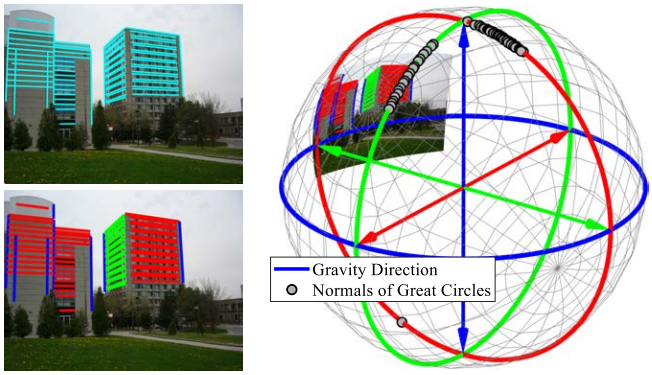
[6] H. Li, P. Kim, J. Zhao, K. Joo, Z. Cai, Z. Liu, and Y.-H. Liu, "Globally optimal and efficient vanishing point estimation in atlanta world," in *ECCV*, 2020.

[7] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "Lsd: A fast line segment detector with a false detection control," *IEEE T-PAMI*, 2008.

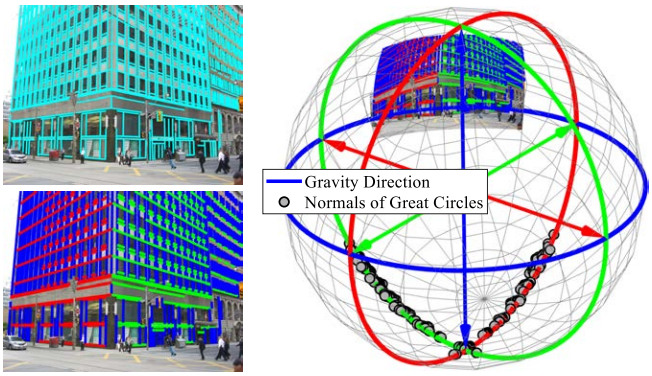
[8] J.-C. Bazin and M. Pollefeys, "3-line RANSAC for orthogonal vanishing point detection," in *IEEE IROS*, 2012.



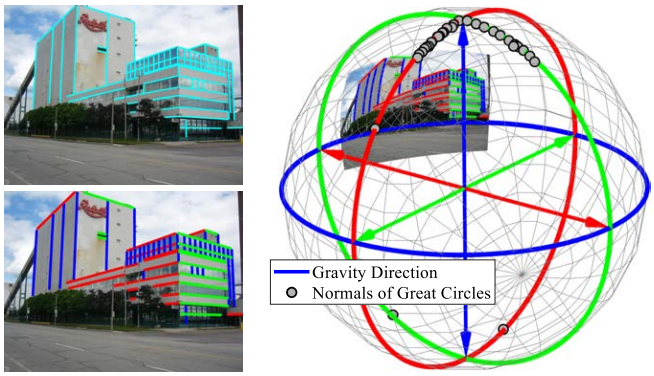
(a) Frame 4



(b) Frame 36



(c) Frame 66



(d) Frame 96

Fig. 9. Representative evaluations on York Urban [1] dataset. In each tested frame, we plot the raw (cyan) and clustered (RGB) image lines with respect to the estimated Manhattan world (MW). We visualize the estimated Manhattan frame (MF) and the corresponding normals of the great circles on the Gaussian sphere. The proposed Manhattan Mine-and-Stab (MnS) can search for the optimal horizontal direction (red and green axis) of the MW rotation, achieving the quasi-global optimality in terms of the number of inlier lines in real-time.

[9] R. C. Leishman, J. C. Macdonald, R. W. Beard, and T. W. McLain, "Quadrotors and accelerometers: State estimation with an improved dynamic model," *IEEE Control Systems Magazine*, 2014.