Visual Odometry with Drift-Free Rotation Estimation Using Indoor Scene Regularities

Pyojin Kim[1]
rlavywls@snu.ac.kr
Brian Coltin[2]
brian.j.coltin@nasa.gov
H. Jin Kim[1]
hjinkim@snu.ac.kr

[1] Department of Mechanical and Aerospace Engineering, ASRI, Seoul National University, Seoul, South Korea

[2] SGT, Inc., NASA Ames Research Center, Moffett Field, California, USA

Abstract

We propose a hybrid visual odometry algorithm to achieve accurate and low-drift state estimation by separately estimating the rotational and translational camera motion. Previous methods usually estimate the six degrees of freedom camera motion jointly without distinction between rotational and translational motion. However, inaccuracy in the rotation estimate is a main source of drift in visual odometry. We design a hybrid visual odometry algorithm which separately estimates the rotational and translational motion to achieve improved accuracy and low drift error. To improve the accuracy of rotational motion estimation, we exploit orthogonal planar structures, such as walls, floors, and ceilings, common in man-made environments. We track orthogonal frames with an efficient SO(3)-constrained mean-shift algorithm, resulting in drift-free rotation estimates. Based on the absolute camera orientation, we newly propose a way to compute the translational motion by minimizing the de-rotated reprojection error with the tracked features. We compare the proposed algorithm with other state-of-the-art visual odometry methods and demonstrate an improved performance and lower drift error.

1 Introduction

Visual odometry (VO) and visual simultaneous localization and mapping (V-SLAM) estimate the motion of a camera from a sequence of images. While V-SLAM constructs a surrounding map and localizes the position of the camera within the constructed map simultaneously, VO only estimates the current position of the camera by accumulating the motions between each image frame. They are fundamental components for many emerging applications, from autonomous cars and unmanned aerial vehicles (UAVs) to augmented and virtual reality. Although VO has lower drift than conventional wheel odometry which is affected by wheel slip in uneven terrain [21], substantial research has focused on minimizing VO drift without any V-SLAM techniques (*i.e.*, loop closure, 3D mapping). VO techniques can be categorized into indirect [9, 19, 28] and direct [6, 7, 8, 14] methods. Direct methods use raw-image pixel values to estimate the six degrees of freedom (DoF) camera motion, while

© 2017. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

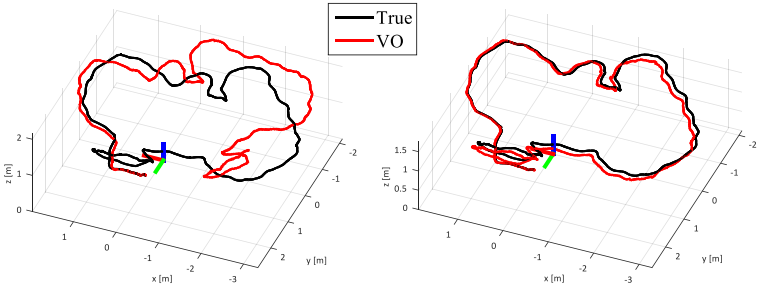


Figure 1: Trajectory estimation from feature-based VO compared to the ground truth. In (a), the rotational and translational motion are estimated jointly with [28]. In (b), we assume that the rotational motion is known, and VO only estimates the translational motion. We confirm that drift of the rotation estimate is the main source of position inaccuracy in VO.

indirect methods use higher-level features detected from the images [7]. Although it is well-known that the main source of the VO drift is inaccurate rotation estimation [20, 25, 29] as shown in Figure 1, most VO approaches do not focus on rotational motion estimation and still estimate rotation and translation together.

We propose a new hybrid visual odometry method which separately estimates the rotation and translation for accurate state estimation in man-made environments. First, the proposed method estimates absolute, drift-free rotation by exploiting orthogonal structures in man-made environments with depth camera to eliminate the main source of positioning inaccuracy. Next, we find the optimal translation by minimizing the de-rotated reprojection error. This algorithm is drift-free in rotation, but requires an orthogonal structure (*i.e.*, inside a building). Extensive evaluation results show that the proposed method produces a low drift error and superior motion estimation to existing indirect and direct VO methods.

2 Related Work

In the last decade, VO and V-SLAM have been extensively studied in robotics and computer vision communities to lower the rate of VO drift. From the vast literature in VO and V-SLAM, we review the latest VO research showing high accuracy and some studies specifically aimed at increasing the accuracy of rotational motion estimation.

VO research can be categorized into direct and indirect methods. Direct methods, which directly exploit the image brightness, have received recent attention for their improved accuracy and robustness to images with little texture with the help of improved hardware. Many direct VO algorithms [8, 14, 16] have been proposed to minimize the photometric error between image frames. But they also suffer from the accumulation of VO drift, for reasons such as irregular illumination changes [15]. In [7], the illumination problem is handled by considering a full photometric calibration model to reduce VO drift error due to light changes. [8] utilizes both direct and feature-based method showing high VO accuracy, and constructs temporary maps at keyframes to reduce VO drift error.

Visual feature-based methods, the most widely used indirect methods, show successful motion estimation results [5, 9, 28]. Using a stereo camera, [9] estimates motion on an autonomous car. In [5], low VO drift error is achieved with careful selection of stable features.

[28] accurately estimates motion by jointly minimizing constraints from both an RGB and a depth image. Some researches [13, 27] have separately estimated the rotational motion, which contributes the most to drift, by using distant feature points and epipolar geometry, without assuming the environment is orthogonal (the "Manhattan World (MW)" assumption) [4]. Our approach requires the MW assumption, but drastically reduces drift thanks to drift-free rotation estimation.

Some recent studies have focused on accurate rotation estimation in structured environments. From MW surface normal vectors, [25] estimates rotational motion based on the maximum a posteriori (MAP) inference of the local Manhattan frame in real-time on a GPU. [30] decouples rotation and translation to estimate absolute orientation by tracking the Manhattan frame (MF) with a mean shift algorithm. However, this method suffers from a translation error that increases rapidly over time, as the translational motion is computed by aligning 1D density distribution of the point cloud. In [1], the absolute attitude (roll and pitch angles) is estimated based on vanishing points (VP) detection, and translation estimation is performed with a 2-point algorithm for catadioptric vision. While [17] jointly estimates accurate camera orientation and VPs without the Manhattan world assumption, it only estimates rotation, not translation. We propose a new hybrid visual odometry algorithm which achieves low drift and high accuracy by first estimating drift-free rotational motion and then computing translational motion separately. This approach relies on the MW assumption, and only applies to orthogonal environments such as buildings.

3 Proposed Method

We present a new hybrid visual odometry algorithm which separately estimates rotational and translation motion to achieve low drift error and higher accuracy. The proposed method is shown in Algorithm 1 in detail. To minimize the effect of the drift in the rotation estimates, which is the main source of the VO drift error [20, 25], drift-free rotational motion is estimated by tracking the Manhattan frame with a SO(3)-manifold constrained mean shift algorithm using density of surface normal vectors as shown in Figure 2. Given the absolute camera orientation, we estimate the translational motion by minimizing the de-rotated reprojection error with the tracked features. By combining the estimated 3-DoF rotational and 3-DoF translational motion, the entire 6-DoF camera motion can be tracked. The overall moving trajectory can be obtained by concatenating the frame-to-frame motion estimation results incrementally.

3.1 Rotational Motion Estimation

3.1.1 Surface Normal Vector Extraction

We pre-process the depth image D_k with a simple box filter to remove noise in the raw depth data. The unit surface normal vectors $\mathbf{n}_k$ on the unit sphere $\mathbb{S}^2$ can be computed using the cross product from the two tangential vectors, which are tangential to the local surface at the 3D points in the point cloud. They can be easily calculated between the left and right neighboring pixels for the u-direction (horizontal), and between the up and down neighboring pixels for the v-direction (vertical) in the point cloud. To reduce noise data in the two maps of tangential vectors, we compute the average u- and v-tangential vectors within a certain neighborhood. To perform this smoothing process efficiently, we generate integral images of the two tangential vectors and calculate the average u- and v-tangential vectors with only $2\times4\times3$ memory access regardless of the size of the smoothing area [12].

Algorithm 1 Rotation and Translation Estimation From Frame $k-1$ to Frame k

Input: Greyscale Images I_k, I_{k-1}; Depth Images D_k, D_{k-1}
Output: Rotation $\mathbf{R}_{k,k-1}$; Translation $\mathbf{T}_{k,k-1}$

```
extract surface normal vectors n_k from D_k
repeat
    project n_k into each tangential plane of MF axes using Eqs. (1) and (2)
    perform Gaussian mean shift algorithm using Eq. (3)
    back onto the unit sphere using Eqs. (4) and (5)
    project R̂_{c_k M} onto the SO(3) manifold using Eq. (7)
until R_{c_k M} converges
R_{k,k-1} ← R_{c_k M} · R_{M c_{k-1}}
track feature points from I_k
derive residual vectors of all tracked features using Eq. (9)
find optimal T* using Eq. (10)
T_{k,k-1} ← T*
```

It is very important to extract accurate and reliable surface normal vectors since the density distribution of surface normal vectors obtained from the depth image directly affects the accuracy of rotational motion estimation in MW.

3.1.2 Tracking Manhattan Frame

The core of the proposed MF tracking is that we compare and track orthogonality of planar structures, which is called the Manhattan frame, observed from the current camera viewpoint. We track the orthogonal Manhattan frames with a SO(3)-manifold constrained mean-shift algorithm, an approach for finding the mode, for a dominant axis given a set of surface normal vectors on the unit sphere $\mathbb{S}^2$ under the assumption that the MF does not change too much between the frame-to-frame motion.

We express the Manhattan world frame with respect to the camera frame as a 3D rotation matrix $\mathbf{R}_{\mathrm{cM}} = \begin{bmatrix}\mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3\end{bmatrix} \in \mathrm{SO}(3)$ where each column $\mathbf{r}_j$ denotes the x-, y-, and z-axis of the dominant MF expressed in the camera frame. We assume that the previous rotation of MF $\mathbf{R}_{\mathrm{c}_{k-1}\mathrm{M}}$ is known, and it is used as initialization point to find the current unknown orientation of MF $\mathbf{R}_{\mathrm{c}_k\mathrm{M}}$. We transform the current surface normal vectors $\mathbf{n}_k$ expressed in the camera frame into $\mathbf{n}'_k$ expressed in the MF:

$$\mathbf{n}'_k = \mathbf{R}^{\top}_{\mathrm{c}_k\mathrm{M}}\,\mathbf{n}_k \tag{1}$$

We alternately project the surface normal vectors $\mathbf{n}'_{k_j}$ into the tangential planes of the x-, y-, and z-axis of the MF to compute a mean shift. The index k_j indicates relevant normal vectors inside a conic section of the j-th dominant axis in the MF. We apply the Riemann logarithmic map to represent proper distances in the tangential plane given by:

$$\mathbf{m}'_{k_j} = \frac{\sin^{-1}(\lambda)\,\mathrm{sign}(\mathbf{n}'_{k_j,z})}{\lambda}\begin{bmatrix}\mathbf{n}'_{k_j,x}\\ \mathbf{n}'_{k_j,y}\end{bmatrix} \tag{2}$$

$$\text{where } \lambda = \sqrt{\mathbf{n}'^2_{k_j,x} + \mathbf{n}'^2_{k_j,y}}$$

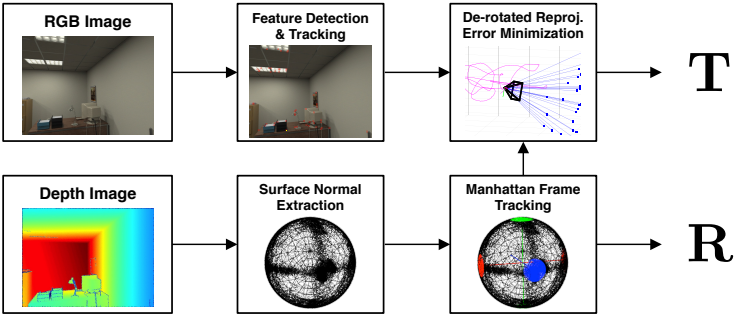


Figure 2: Overview of the proposed algorithm that separately estimates rotational and translational motion. A drift-free camera rotation (**R**) is estimated by tracking Manhattan frame with surface normal vectors. Given the absolute camera orientation, a de-rotated reprojection error is minimized for estimating translational motion (**T**) with the tracked features.

where $\mathbf{m}'_{k_j}$ means the two-dimensional coordinate position in the tangential plane. We perform the mean shift algorithm with a Gaussian kernel in the tangential plane:

$$\mathbf{s}'_j = \frac{\sum e^{-c\|\mathbf{m}'_{k_j}\|^2}\mathbf{m}'_{k_j}}{\sum e^{-c\|\mathbf{m}'_{k_j}\|^2}} \tag{3}$$

where c is the width of the kernel, defined by the user. We apply the Riemann exponential map to transform the mean shift result back to the unit sphere from the tangential plane:

$$\mathbf{s}_j = \overline{\begin{bmatrix} \frac{\tan(\|\mathbf{s}'_j\|)}{\|\mathbf{s}'_j\|}\mathbf{s}'^{\top}_j & 1 \end{bmatrix}}^{\top} \tag{4}$$

where the $\overline{x}$ means x normalized. The estimated j-th dominant axis expressed in the MF is converted with respect to the camera frame to obtain the updated direction vector:

$$\hat{\mathbf{r}}_j = \mathbf{R}_{\mathrm{c}_k\mathrm{M}}\,\mathbf{s}_j \tag{5}$$

Repeating for all dominant axes in the MF, we obtain the updated rotation matrix:

$$\hat{\mathbf{R}}_{\mathrm{c}_k\mathrm{M}} = \begin{bmatrix}\hat{\mathbf{r}}_1 & \hat{\mathbf{r}}_2 & \hat{\mathbf{r}}_3\end{bmatrix} \tag{6}$$

However, $\hat{\mathbf{R}}_{\mathrm{c}_k\mathrm{M}}$ violates the orthogonality constraint because each axis in the MF is updated independently by the mean shift algorithm in Eq. (3). Thus, we project $\hat{\mathbf{R}}_{\mathrm{c}_k\mathrm{M}}$ onto the SO(3) manifold to satisfy the orthogonality constraint using singular value decomposition (SVD):

$$\mathbf{R}_{\mathrm{c}_k\mathrm{M}} = \mathbf{U}\mathbf{V}^{\top} \tag{7}$$

$$[\mathbf{U},\mathbf{D},\mathbf{V}] = \mathrm{SVD}(\begin{bmatrix}\lambda_1\hat{\mathbf{r}}_1 & \lambda_2\hat{\mathbf{r}}_2 & \lambda_3\hat{\mathbf{r}}_3\end{bmatrix})$$

where λ is a weighting factor of how certain the observation of a direction is [30]. The above procedure (lines 2 to 7 of Algorithm 1) is repeated until the change in the estimated rotation of MF is very small. In this manner, we obtain an absolute and drift-free estimate of the MF orientation. Note that at least two orthogonal planes must be visible to track the MF.

3.1.3 Dominant Manhattan Frame

We find the most dominant MF, *i.e.*, the three dominant planar axes in an environment, to estimate drift-free rotational motion. We employ the mean shift clustering algorithm to initialize the estimation of the rotational motion. We first perform the MF tracking 100 times from a random initial rotation, in the manner explained previously. We cluster these 100 MF tracking results and perform histogram-based non-maximum suppression. The most frequent MF is selected as the dominant initial MF. For more details, see [30]. The dominant MF only needs to be found on the first frame.

3.2 Translational Motion Estimation

3.2.1 Feature Extraction

Feature points used in translational motion estimation are extracted using the Good Features to Track corner detector [23]. Bucketing is utilized to spread them uniformly across the entire image domain and reduce the number of features. We maintain the number of features between 150 and 200 in practice. The points are tracked in the next image frame using KLT feature tracker [2].

3.2.2 De-rotated Reprojection Error Minimization

The core of the estimation of the translational motion is that the translational movement of the camera can be obtained by minimizing de-rotated reprojection error given the absolute camera orientation. Previous approaches [9, 28] jointly minimized the original reprojection error, which leads to higher drift over time.

We model the mathematical relationship between the camera motion and i-th tracked 3D feature point as follows:

$$\mathbf{X}_i^k = Z_i^k \bar{\mathbf{X}}_i^k = \mathbf{R}\mathbf{X}_i^{k-1} + \mathbf{T} \tag{8}$$

where $\mathbf{X}_i^k = [X_i^k, Y_i^k, Z_i^k]^\top$ is the coordinates of the feature in the camera frame at time step k, and $\bar{\mathbf{X}}_i^k = [\bar{X}_i^k, \bar{Y}_i^k, 1]^\top$ is the normalized term of $\mathbf{X}_i^k$ where the z component is one. The rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{T}$ form an SE(3) rigid body transformation [18]. Eq. (8) contains three rows. Substituting the expression Z_i^k in the third row into the first and second rows, we can derive two equations for the tracked feature point as follows:

$$\begin{aligned} r_{i,1}(\mathbf{T}) &= (\mathbf{R}_1 - \bar{X}_i^k \mathbf{R}_3)\mathbf{X}_i^{k-1} + \mathbf{T}_1 - \bar{X}_i^k \mathbf{T}_3 = 0 \\ r_{i,2}(\mathbf{T}) &= (\mathbf{R}_2 - \bar{Y}_i^k \mathbf{R}_3)\mathbf{X}_i^{k-1} + \mathbf{T}_2 - \bar{Y}_i^k \mathbf{T}_3 = 0 \end{aligned} \tag{9}$$

where $\mathbf{R}_h$ and $\mathbf{T}_h$, $h \in \{1,2,3\}$ are h-th rows of $\mathbf{R}$ and $\mathbf{T}$ respectively. There are two residual terms per feature. Because we already know the absolute camera orientation by tracking MF in the previous section, the residual terms in Eq. (9) are only a function of the translational camera motion. The optimal translational camera motion that minimizes the residual vectors of all tracked feature points can be obtained by solving the following optimization problem:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} \sum_{i=1}^{M} \left[(r_{i,1}(\mathbf{T}))^2 + (r_{i,2}(\mathbf{T}))^2 \right] \tag{10}$$

where M is the number of tracked features. Eq. (10) can be solved by the Levenberg–Marquardt (LM) algorithm [11]. Note that textures and brightness in the images should be sufficient for reliable feature detection and tracking. Otherwise, the inaccurate translational motion estimation caused by wrong correspondence can degrade the overall 6-DoF camera motion estimation.

Experiment	Relative Pose Error (m/s)				Absolute Trajectory Error (m)				Final Drift Error (%)				Length (m)	# of frame
	Proposed	DEMO	DVO	MWO	Proposed	DEMO	DVO	MWO	Proposed	DEMO	DVO	MWO		
lr kt0	**0.031**	0.077	0.048	0.084	**0.052**	0.145	0.237	0.317	**4.546**	13.293	7.084	32.359	2.74	502
lr kt1	0.021	**0.020**	0.023	0.100	**0.042**	0.143	0.065	0.589	**1.744**	13.231	3.151	30.158	2.05	951
lr kt2	**0.031**	0.090	0.084	0.052	**0.064**	0.616	0.502	0.130	**0.934**	11.813	8.819	1.363	8.42	881
lr kt3	**0.052**	0.076	0.068	0.090	**0.096**	0.282	0.409	0.373	**1.798**	5.327	4.629	3.826	5.47	554
of kt0	**0.014**	0.063	0.106	×	**0.048**	0.338	0.371	×	**1.295**	5.947	10.090	×	6.53	1507
of kt1	**0.014**	0.054	0.045	0.263	**0.052**	0.371	0.357	1.092	**1.103**	9.197	8.912	25.250	6.72	965
of kt2	**0.015**	0.079	0.065	0.047	**0.061**	0.311	0.229	0.087	**1.577**	7.586	4.696	2.776	4.47	467
of kt3	**0.009**	0.030	0.052	0.155	**0.030**	0.176	0.304	1.312	**0.425**	2.514	5.358	24.161	7.82	1240

Table 1: Evaluation Results on ICL-NUIM Benchmark

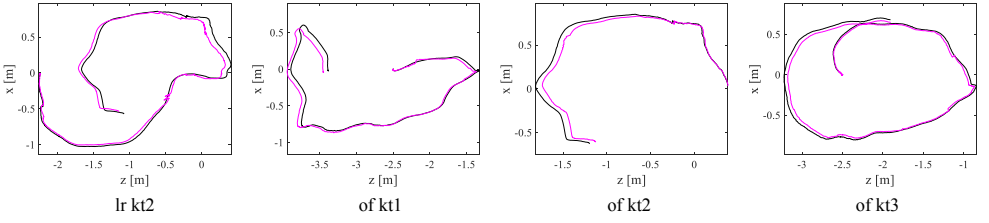


Figure 3: Some motion estimation results of the proposed algorithm in the ICL-NUIM dataset. The magenta and black lines represent the estimated and the ground truth trajectory, respectively.

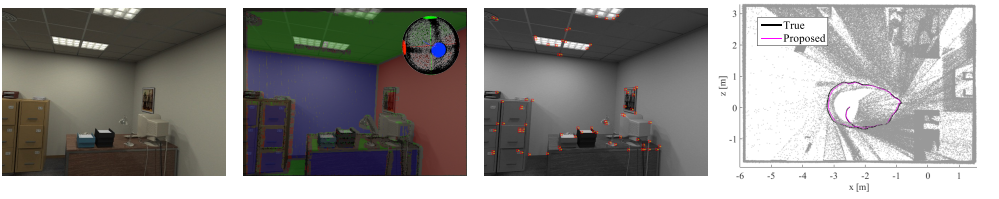
4 Evaluation

We evaluate the effectiveness of the proposed VO algorithm on the publicly available ICL-NUIM benchmark [10] and author-collected RGB-D datasets in man-made environments. For the performance comparison, we provide the motion estimation results of the proposed and other VO algorithms: indirect, direct, and hybrid VO methods, namely DEMO [28], DVO [14], and MWO [30] respectively. DEMO and DVO are the visual odometry methods which estimate the rotational and translational motion jointly, while MWO decouples the estimation of the rotational and translational motion like the proposed VO method.

4.1 Tests with ICL-NUIM Datasets

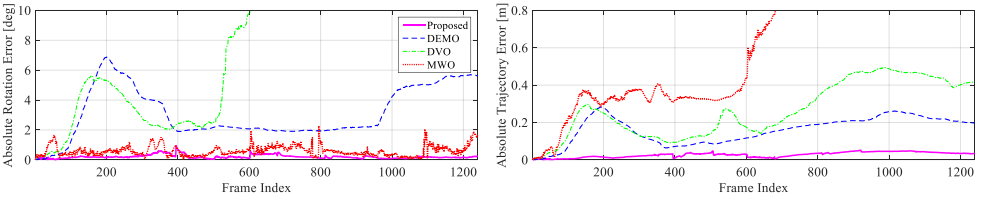
We first test the proposed algorithm with the ICL-NUIM dataset [10]. It consists of a collection of handheld RGB-D camera sequences within synthetically generated living room and office. Although the synthetic RGB and depth images are captured at 30 Hz in virtual environments, typically observed noise existing in the actual camera image is reproduced well by modeling sensor noise in both RGB and depth data. The proposed method and other VO baselines are applied to all living room and office datasets. To perform quantitative analysis, three types of error metrics are selected: root mean square error (RMSE) of the relative pose error (RPE), absolute trajectory error (ATE) as in [26], and the final drift error divided by the total traveling distance.

Table 1 shows the performances of the proposed and other VO algorithms. The smallest error value in each tested dataset for each error metric is highlighted in bold type. Figure 3 shows some motion estimation results of the proposed algorithm compared to the ground truth trajectory. In most cases, the proposed algorithm shows better performance in terms of relative pose error compared to other VO algorithms. In all cases, we observe that the proposed method generates the lowest absolute trajectory error and final drift error compared to other VO baselines. The final drift error of the proposed method is 1.68% on the average,



(a) (b) (c) (d)

Figure 4: Figures (a) and (b) show the original image from 'of kt3' and the plane segmentation results from the extracted MF. The inferred MF orientation is drawn in the top right corner of (b). In (c), the tracked features for estimating translational motion in the proposed method are marked in red. Figure (d) shows the reconstructed trajectory and consistent point cloud on 'of kt3'.



(a) (b)

Figure 5: (a) The rotation matrix errors of each image frame for the proposed and other VO algorithms. The proposed method shows the lowest rotation error compared to other VO algorithms. (b) The translational error between the estimated and ground truth position, showing the superior performance of the proposed method.

while DEMO, DVO, and MWO are 8.61%, 6.59%, and 17.13%, respectively. We also evaluate the proposed algorithm with other VO methods on TUM RGB-D dataset [26], resulting in similar experimental values: 4.59% of the proposed VO, while DEMO, DVO, and MWO are 16.82%, 9.61%, and 22.45% averagely. The main reason for the improved results is that the proposed algorithm can estimate drift-free rotational motion over time. But other VO algorithms are cumulative in the drift of the rotation estimates, which is the main source of position inaccuracy, showing a big difference from the ground truth trajectory at the end.

Excerpts from 'of kt3' in the ICL-NUIM dataset, including Manhattan world scene segmentation result, the inferred MF orientation, and the tracked features are shown in Figure 4. The Manhattan frame in the office room is successfully tracked in Figure 4 (b), resulting in the accurate estimation of the drift-free camera orientation. The tracked features marked red in Figure 4 (c) are used to minimize de-rotated reprojection error for translational motion estimation. In Figure 4 (d), the estimated and the ground truth trajectory overlap significantly, thus, consistent point cloud can be reconstructed based on the motion estimation results of the proposed method.

The strength of the proposed method becomes clear when analyzing the dataset 'of kt3' in terms of the absolute rotation error as shown in Figure 5 (a). The absolute rotation error of the proposed method and MWO does not increase over time due to drift-free Manhattan frame tracking method. The absolute rotation error in the DEMO and DVO methods, however, continues to increase over time, resulting in positioning inaccuracy at the end. The average rotation error of the proposed method is 0.21 degree whereas DEMO, DVO, and MWO are 3.15, 8.12, 0.59 degree respectively. Although the proposed method and MWO employ the similar Manhattan frame tracking method for rotation estimation, different results are

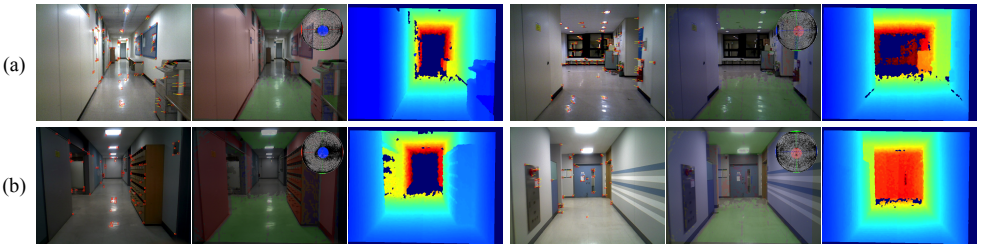


Figure 6: Example images from the author-collected RGB-D dataset. The tracked features and Manhattan frame are overlaid on top of the RGB images from the single-loop (a) and multiple-loop (b) sequences. Corresponding depth images from the RGB-D camera are shown with the colorscale.

obtained due to the different algorithms for extracting surface normal vectors from a depth image.

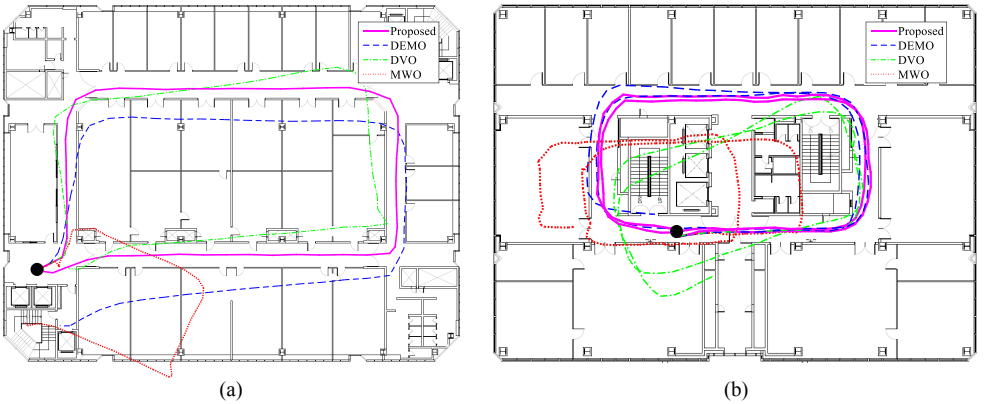
Figure 5 (b) shows the superior performance of the proposed method in terms of absolute trajectory error (ATE), which includes not only rotational but also translational motion error. Although the drift-free rotation estimation is performed in MWO, increases of absolute trajectory error are shown in the graph due to incorrect estimation of translational motion in MWO. While the accumulated rotational error in DEMO and DVO makes the overall position estimation inaccurate, the proposed method shows the lowest growth rate of the absolute trajectory error given the accurate *drift-free* rotation estimates.

4.2 Tests with Author-collected RGB-D Datasets

We want to demonstrate that the proposed algorithm works well in the everyday indoor environments which generally satisfy Manhattan world assumption. So, we recorded our own datasets with an Asus Xtion Pro Live RGB-D camera capable of providing RGB and depth images at 30 Hz with 640 $\times$ 480 resolution. Figure 6 shows the example RGB and depth images captured in corridors of the buildings satisfying Manhattan world constraint. For evaluating the final drift error of the proposed and other VO algorithms without the ground truth data, the images are collected along the carefully designed trajectories where the beginning and end points are at the same place. We also overlap the estimated trajectories on the floorplan of the buildings to check the consistency of the proposed and other VO methods.

Figure 7 (a) shows the evaluation results along about 105 meters long single loop trajectory where we start and end at the same place. DEMO and DVO methods cannot meet the starting and end points due to drift of the rotation estimates accumulated at the corners of the loop. The starting and end points of the trajectory estimated with the proposed method coincide at the black circle. The high consistency of the proposed method is also observed in Figure 7 (b), which is about 127 meter long trajectory consisting of multiple loops in the same place and left turns of 90 degrees. Although MWO can estimate the drift-free rotational motion, inaccuracy in translational motion estimation causes inconsistent results. The overlapping estimated trajectory with the proposed method shows the most consistent path while other estimated trajectories gradually diverge from the initially estimated loop. Please refer to the video clips submitted with this paper showing more details about the experiments.[1]

[1] Video available at https://youtu.be/sC3iiaxBhdw



(a) (b)

Figure 7: Motion estimation results with the proposed algorithm compared to other VO methods on the author-collected RGB-D dataset in a single-loop (a) and multiple-loop (b) sequences. We start and end at the same place in the datasets marked in black circle to check loop closing in the proposed VO algorithm.

5 Conclusion

We have presented a low-drift visual odometry algorithm that separately estimates the rotational and translational motion. For reducing drift of the rotation estimate, which is the main source of position inaccuracy in visual odometry algorithms, the Manhattan frame tracking is performed to estimate the absolute camera orientation. Given drift-free rotation estimates in MW, translational motion is estimated by minimizing de-rotated reprojection with the tracked features. This approach enables accurate and low-drift motion estimation results of the proposed VO algorithm in man-made indoor environments. Our approach assumes Manhattan world environments; future work should consider more general and relaxed environments, such as Atlanta World (AW) [22] and Mixture of Manhattan Frames (MMF) [24].

Acknowledgements

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (2014R1A2A1A12067588) and the Lockheed Martin Corporation and ASRI (Automation and Systems Research Institute) in Seoul National University.

References

[1] Jean Charles Bazin, Cédric Demonceaux, Pascal Vasseur, and IS Kweon. Motion estimation by decoupling rotation and translation in catadioptric vision. *Computer Vision and Image Understanding*, 114(2), 2010.

[2] Jean-Yves Bouguet. Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm. *Intel Corporation*, 5(1-10):4, 2001.

[3] Andrew I Comport, Ezio Malis, and Patrick Rives. Real-time quadrifocal visual odometry. *The International Journal of Robotics Research*, 29(2-3), 2010.

[4] James M Coughlan and Alan L Yuille. Manhattan world: Compass direction from a single image by bayesian inference. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2. IEEE, 1999.

[5] Igor Cvišić and Ivan Petrović. Stereo odometry based on careful feature selection and tracking. In *Mobile Robots (ECMR), European Conference on*. IEEE, 2015.

[6] Jakob Engel, Jörg Stückler, and Daniel Cremers. Large-scale direct SLAM with stereo cameras. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015.

[7] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[8] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014.

[9] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3D reconstruction in real-time. In *Intelligent Vehicles Symposium (IV)*. IEEE, 2011.

[10] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *Robotics and automation (ICRA), 2014 IEEE international conference on*. IEEE, 2014.

[11] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[12] Dirk Holz, Stefan Holzer, Radu Bogdan Rusu, and Sven Behnke. Real-time plane segmentation using RGB-D cameras. In *Robot Soccer World Cup*. Springer, 2011.

[13] Michael Kaess, Kai Ni, and Frank Dellaert. Flow separation for fast and robust stereo odometry. In *Robotics and Automation (ICRA), International Conference on*. IEEE, 2009.

[14] Christian Kerl, Jürgen Sturm, and Daniel Cremers. Robust odometry estimation for RGB-D cameras. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013.

[15] Pyojin Kim, Hyon Lim, and H Jin Kim. Robust visual odometry to irregular illumination changes with RGB-D camera. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015.

[16] Sebastian Klose, Philipp Heise, and Alois Knoll. Efficient compositional approaches for real-time robust direct visual odometry from RGB-D data. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013.

[17] Jeong-Kyun Lee and Kuk-Jin Yoon. Real-time joint estimation of camera orientation and vanishing points. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[18] Yi Ma, Stefano Soatto, Jana Kosecka, and S Shankar Sastry. *An invitation to 3-D vision: from images to geometric models*, volume 26. Springer Science & Business Media, 2012.

[19] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31 (5), 2015.

[20] Clark F Olson, Larry H Matthies, Marcel Schoppers, and Mark W Maimone. Stereo ego-motion improvements for robust rover navigation. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2. IEEE, 2001.

[21] Davide Scaramuzza and Friedrich Fraundorfer. Visual odometry [tutorial]. *IEEE robotics & automation magazine*, 18(4), 2011.

[22] Grant Schindler and Frank Dellaert. Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1. IEEE, 2004.

[23] Jianbo Shi and Carlo Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*. IEEE, 1994.

[24] Julian Straub, Guy Rosman, Oren Freifeld, John J Leonard, and John W Fisher. A mixture of Manhattan frames: Beyond the Manhattan world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[25] Julian Straub, Nishchal Bhandari, John J Leonard, and John W Fisher. Real-time Manhattan world rotation estimation in 3D. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015.

[26] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012.

[27] Jean-Philippe Tardif, Yanis Pavlidis, and Kostas Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008.

[28] Ji Zhang, Michael Kaess, and Sanjiv Singh. Real-time depth enhanced monocular odometry. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014.

[29] Yi Zhou, Laurent Kneip, and Hongdong Li. Real-time rotation estimation for dense depth sensors in piece-wise planar environments. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016.

[30] Yi Zhou, Laurent Kneip, Cristian Rodriguez, and Hongdong Li. Divide and conquer: Efficient density-based tracking of 3D sensors in Manhattan worlds. In *Asian Conference on Computer Vision*. Springer, 2016.