Fusion of Camera, IMU, and Speedometer for Localization of Autonomous Vehicles

Chang-Ryeol Lee[O], Kuk-Jin Yoon
Gwangju Institute of Science and Techonology
crlee@gist.ac.kr, kjyoon@gist.ac.kr

Abstract

Visual-inertial odometry (VIO) for autonomous vehicles provides ego-motion estimates with the help of an inertial measurement unit (IMU). However, VIO in large-scale outdoor environments is limited in its ability to estimate translational motion owing to forward motion degeneracy. In this paper, we propose an approach to estimate the ego-motion of a vehicle using a camera, an IMU, and a speedometer. The speed measurement model is incorporated into the Bayesian VIO framework, and a state re-initialization is applied based on the speed measurements. Experiments using the public KITTI dataset show the superiority of the proposed method compared to conventional VIO and stereo-based visual odometry. Furthermore, the proposed method achieves about 20 Hz frame rates for real-time autonomous driving.

1. Introduction

Odometry, which estimates the 6-DOF ego-motion, is a crucial technology for many computer vision and robotics applications. Over the last decade, visual-inertial odometry (VIO), which uses a monocular camera in conjunction with an inertial measurement unit (IMU), has been extensively studied for robotic navigation and autonomous driving in GPS-denied environments, such as urban, military, underwater, and indoor areas. Unlike monocular visual odometry, VIO provides scale information and produces trajectories with less drift compared to wheel odometry and inertial odometry. This is particularly advantageous in lightweight mobile systems owing to its compactness while maintaining a high level of performance.

Studies on the use of VIO in autonomous vehicles have been recently conducted because cars are typically equipped with IMUs. However, even with the help of inertial measurements, VIO estimates are subject to scale drift in large-scale environments owing to forward motion degeneracy and the existence of distant feature points in translation estimations, as illustrated in [7]. In this study, we observed that the position drift of VIO estimates begins from drifts in the velocity estimates. Velocity estimates have a large influence on the position estimates because they are closely related to the state prediction process. Furthermore, previous VIO methods for autonomous driving require a large number of computations because they use heavy feature descriptors for feature detection and matching, such as SIFT, as in [5]. When VIO uses the FAST and KLT algorithms for fast feature detection and matching, it reduces the number of computations, but the scale drift of the translation estimates significantly worsens. Figure 1 shows that VIO provides scale-drifted velocity estimates while moving forward.

To handle this problem, we incorporate a speedometer, which is included in most cars. The speed measurements essentially originate from the speed sensors of the wheels. Therefore, this study is a multimodal approach with three heterogeneous sensors: a monocular camera, an inertial sensor, and wheel speed sensors.

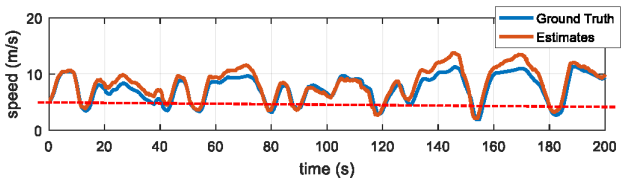


Fig. 1 Speed estimates of VIO with FAST and KLT for sequence 1 used in our experiments. When the speed reaches greater than 5 m/s, the vehicle mostly moves forward during this sequence.

Using these heterogeneous sensors, we estimate the 6-DOF egomotion online in real-time, while removing the scale drift. Our method is based on the Bayesian filtering framework in which vehicle pose states are predicted based on inertial measurements and are updated using images and speed measurements. Furthermore, using speed measurements, we re-initialize the states when a vehicle stops because the VIO estimator can diverge when a vehicle restarts after stopping, owing to a rapid change in motion.

2. Related works

The fusion of visual and inertial measurement data has been theoretically validated, and various state estimation techniques have been previously applied [1]. Such methods typically require heavy computational power because of the estimation of 3D landmarks. As a result, the real-time use of VIO has become important, and an efficient algorithm has therefore been studied [2]. Furthermore, sliding-window approaches have been proposed for VIO [5], and optimization-based VIO methods have been studied [6] to improve the accuracy and robustness.

3. Method

We estimate the ego-motion of vehicles using a speedometer and an IMU, which are commonly installed in vehicles, along with a monocular camera. The states of the vehicle poses $\mathbf{x}$ are estimated using 6-DOF inertial measurements $\mathbf{u}_{imu}$, camera measurements $\mathbf{z}_{cam}$, and speed measurements $\mathbf{z}_{speed}$ as

$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{z}_k, \mathbf{u}_k) &= p(\mathbf{z}_k \mid \mathbf{x}_k)\, p(\mathbf{x}_k \mid \mathbf{x}_{1:k-1}, \mathbf{u}_k) \\ &= p(z_{speed,k} \mid \mathbf{x}_k)\, p(\mathbf{z}_{cam,k} \mid \mathbf{x}_k)\, p(\mathbf{x}_k \mid \mathbf{x}_{1:k-1}, \mathbf{u}_{imu,k}). \end{aligned} \tag{1}$$

Based on the Bayesian inference described above, our algorithm formulates a nonlinear system model $f_{imu}(\cdot)$ that describes the prediction of state vectors, and two nonlinear measurement models $h_{cam}(\cdot)$ and $h_{speed}(\cdot)$ that describe the relations between the state vectors and measurements as

$$\begin{aligned} \mathbf{x}_k &= f_{imu}(\mathbf{x}_{k-1}; \mathbf{u}_{imu,k}) + \mathbf{q}_{imu}, \\ \hat{\mathbf{z}}_{cam,k} &= h_{cam}(\mathbf{x}_k) + \mathbf{r}_{cam}, \\ \hat{\mathbf{z}}_{cam,k} &= h_{speed}(\mathbf{x}_k) + \mathbf{r}_{speed}. \end{aligned} \tag{2}$$

where $\mathbf{q}_{imu,k-1}$ is process noise, and $\mathbf{r}_{imu,k}$ and $\mathbf{r}_{speed,k}$ are measurement noises.

3.1 State Vector

Our state vector $\mathbf{x} \in \mathbb{R}^{17}$ is constructed as follows

$$\mathbf{x} = \begin{bmatrix} {}^{W}_{I}\mathbf{p}_{1:3}^{\mathrm{T}} & {}^{W}_{I}\mathbf{q}_{1:3}^{\mathrm{T}} & {}^{W}_{I}\mathbf{v}^{\mathrm{T}} & \mathbf{b}_a & \mathbf{b}_g & b_s \end{bmatrix}^{\mathrm{T}}, \tag{3}$$

where $\mathbf{p} \in \mathbb{R}^3$ is the position of the IMU, $\mathbf{q} \in \mathbb{R}^4$ is the orientation of the IMU, the subscript 1:3 indicates the index of three consecutive frames, and $\mathbf{v} \in \mathbb{R}^3$ is the velocity of the IMU. $\mathbf{b}_a \in \mathbb{R}^3$ is the bias in the acceleration measurements, $\mathbf{b}_g \in \mathbb{R}^3$ is the bias in the angular velocity measurements, and $b_s \in \mathbb{R}$ is the bias in the speed measurements.

3.2 System Model: IMU

The system model $f_{imu}(\cdot)$ is formulated based on second-order motion dynamics using acceleration and angular velocity measurements $\{\mathbf{a}_m, \mathbf{w}_m\} \subset \mathbf{u}_{imu}$ as

$$f_{imu}(\mathbf{x}_{k-1}) = \begin{bmatrix} {}^{W}_{I}\mathbf{p}_{k-1} + {}^{W}_{I}\mathbf{v}_{k-1}\Delta T + {}^{W}_{I}\mathbf{a}_{m,k-1}\dfrac{\Delta T^2}{2} \\ {}^{W}_{I}\mathbf{v}_{k-1} + {}^{W}_{I}\mathbf{a}_{m,k-1}\Delta T \\ {}^{W}_{I}\mathbf{q}_{k-1} + \Omega(-{}^{W}_{I}\mathbf{w}_{m,k-1}\dfrac{\Delta T}{2}){}^{W}_{I}\mathbf{q}_{k-1} \end{bmatrix}, \tag{4}$$

where ΔT is the time interval between the inertial measurements and the quaternion kinematic function: $\Omega : \mathbb{R}^3 \to \mathbb{R}^{4\times4}$, which explains the variation in IMU orientation. Here, the process noise vector qimu is modeled using a zero-mean Gaussian distribution $\mathrm{N}(0, \mathbf{Q}_{imu})$, and $\mathbf{Q}_{imu}$ is the process noise covariance.

3.3 Measurement Model: Speedometer

The speedometer provides a scalar measurement $z_{speed} \in \mathbb{R}$ of the vehicle velocity, and contains biases and additive Gaussian noises. Therefore, the measurement model $h_{speed}(\cdot)$ is formulated as

$$h_{speed}(\mathbf{x}_k) = \left\| {}^{W}_{I}\mathbf{v} \right\| - b_s \tag{5}$$

where rspeed is modeled using a zero-mean Gaussian distribution $\mathrm{N}(0, R_{speed})$, and R_{speed} is the measurement noise covariance of the speedometer. Although the speedometer and IMU have different coordinates, the difference between speeds at both coordinates is negligible because they are rigidly connected.

3.4 Measurement Model: Camera

The camera measurement $\mathbf{z}_{cam} \in \mathbb{R}^{4M}$ is generated from the corresponding feature points of three consecutive frames, where M represents the number of the corresponding feature points. $\mathbf{z}_{cam}$ is constructed by stacking $\mathbf{z}_{cam,i}$ M number of times. The camera measurement $\mathbf{z}_{cam,i} \in \mathbb{R}^4$ from the i-th feature points is composed of two epipolar constraints between two frames and the tracked feature points of the third frame:

$$\mathbf{z}_{cam,i} = \begin{bmatrix} 0 \\ 0 \\ u_3 \\ v_3 \end{bmatrix}, \; h_{cam,i}(\mathbf{x}) = \begin{bmatrix} \mathbf{m}_1^{\mathrm{T}}\mathbf{F}_{12}\mathbf{m}_2 \\ \mathbf{m}_2^{\mathrm{T}}\mathbf{F}_{23}\mathbf{m}_3 \\ \mu\left(\left(u_1\mathbf{T}_1 + v_1\mathbf{T}_2 + \mathbf{T}_3\right)\mathbf{l}_2\right) \end{bmatrix}, \tag{6}$$

where $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_3 \in \mathbb{R}^3$ are the corresponding feature points of the three consecutive frames in normalized camera coordinates. $\mathbf{l}_2 \in \mathbb{R}^3$ is a line in the 2nd frame, which is perpendicular to the epipolar line between the first and second frames. $\mathbf{F}_{12}$, $\mathbf{F}_{23}$, and $\mathbf{T}_{i=\{1,2,3\}}$ are fundamental matrices and a trifocal tensor of three consecutive frames computed from states, respectively. $\mu(\cdot)$ represents the camera projection function to the image coordinates. The measurement noise $\mathbf{r}_{cam,i} \in \mathbb{R}^4$ is modeled using a zero-mean Gaussian distribution, $\mathrm{N}(0, \mathbf{R}_{cam,i})$.

3.5 Zero Speed Re-Initialization

When a vehicle departs after stopping, the states sometimes diverge because of the dramatic changes in motion. To handle this, we detect the instances of re-initialization using the speed measurements as a clue to determine if the vehicle has stopped. We then re-initialize the states and state error co-variances while the vehicle stops.

4. Experimental Results

An ideal experiment setup would utilize raw measurements obtained from a camera, an IMU, and a speedometer within an actual vehicle. Unfortunately,

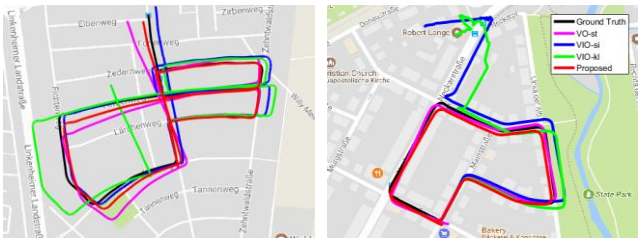
however, there are currently no publicly available autonomous vehicle datasets providing speed measurements. Therefore, we conducted quantitative and qualitative evaluations of the proposed method using the public KITTI dataset [3]. Although the KITTI dataset does not provide speed measurements of a vehicle, it contains velocity measurements of the inertial navigation system, allowing us to generate the speed measurements by adding Gaussian noise and bias to the magnitude of the velocity measurements. Furthermore, as a well-known autonomous driving benchmark, the KITTI dataset makes it easy to conduct a performance comparison with other existing methods. The methods used for comparison are near-online odometry methods that utilizes two or three frames to estimate the ego-motion of a vehicle: stereo visual odometry (VO-st [4]), visual-inertial odometry with SIFT and KLT (VIO-si [5], VIO-kl). Table 1 shows the errors in the rotation and translation estimates for each sequence. Essentially, VIO-si provides more accurate rotation and translation estimates than VO-st owing to the high-quality matching capabilities of SIFT. However, VIO-kl for real-time operation produces less accurate estimates than VIO-si, and shows large-scale drift in the translation estimates. Our method dramatically reduces the scale drift of VIO-kl with errors of less than 1 %, and incurs smaller rotation errors than VIO-kl despite using KLT for feature matching. In particular, our method does not diverged when starting after stopping as in sequence 7 unlike VIO-si,-kl. Moreover, it provides more accurate rotation estimates than VO-st. We qualitatively compared our method to the other odometry methods. Figure 2(a) shows that the estimated trajectory of our method is closest to the ground truth trajectory. Note that we do not use bundle adjustments based on local maps or loop closures. Figure 2(b) illustrates the divergence of VIO-si and -kl odometry when starting after a period of stopping. As the figure indicates, our method does not diverge, and instead provides accurate trajectory estimates resulting from the zero speed re-initialization. The VO-st approach provides consistent ego-motion estimates utilizing depth information;. however, the accumulated rotation estimates are reflected in the trajectory comparison with the ground truth. Additionally, we evaluated the operating times of the compared methods in our experiments. This comparison was conducted using a PC with a 4.0 GHz Intel i7 CPU. We compared VIO-si, VIOkl, and the proposed method. The VIO-si method was implemented using MATLAB/C++, and both the proposed method and VIO-kl were implemented using C++. We also implemented the camera measurement model of the proposed method using multi-processing (mp) for boosting the real-time operation. As Table 2 shows, VIO-si takes too much time for feature matching; however, VIO-kl and the proposed method do not, owing to the use of FAST and KLT. Regarding outlier rejection, VIO-kl and the proposed method were found to be slower than VIO-si because they remove large residual features after updating all features, as in [7], instead of RANSAC. However, the operating time of the outlier rejection and state update can be reduced to one-half using multi-processing. Finally, we achieved about 20 H frame rates on the KITTI dataset.

Table 1: Quantitative comparison with other odometry methods using the public KITTI dataset.

Seq.	Distance (m)	Rotation error(deg/m)				Translation error(%)			
		VO-st	VIO-si	VIO-kl	Ours	VO-st	VIO-si	VIO-kl	Ours
1	2203.8	0.0059	0.0008	0.0026	0.0020	2.69	1.27	8.18	0.94
2	1229.8	0.0055	0.0010	0.0030	0.0026	0.85	1.69	1.45	0.85
3	4205.7	0.0068	0.0027	0.0087	0.0032	2.02	1.61	11.40	0.65
4	1707.4	0.0052	0.0008	0.0028	0.0003	4.91	2.54	0.85	0.26
5	5068.1	0.0051	0.0004	0.0015	0.0015	4.11	2.00	7.77	1.00
6	3720.7	0.0065	0.0007	0.0026	0.0028	2.29	1.74	10.04	0.64
7	665	0.0084	0.0488	0.1262	0.0049	2.99	42.19	38.89	0.71
Total (1-6)	18135.5	0.0059	0.0011	0.0038	0.0022	2.93	1.80	8.05	0.76

Table 2. Timing comparison of each method.

	Feature matching	State prediction	Outlier rejection	State update
VIO-si	1980 ms	3 ms	15 ms	45 ms
VIO-kl	12 ms	1 ms	65 ms	47 ms
Proposed	12 ms	1 ms	55 ms	33 ms
Proposed-mp	12 ms	1 ms	20 ms	16 ms



(a) (b)

Fig. 2 Qualitative comparison of ego-motion estimates using the public KITTI dataset: (a) sequence 1 and (b) sequence 7

5. Conclusion

In this paper, we proposed a fast and accurate odometry method that incorporates a camera, an IMU, and a speedometer. Experiment results on the public KITTI dataset show the superiority of the proposed method compared to stereo-based visual odometry and visual-inertial odometry.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. NRF-2015R1A2A1A01005455).

Reference

[1] Huang, G. P., Mourikis, A. I., and Roumeliotis, S. I.: 'Observability-based Rules for Designing Consistent EKF SLAM Estimators', International Journal of Robotics Research, 2010, 29, (5), pp. 502-528.

[2] Weiss, S. and Siegwart, R.: 'Real-Time Metric State Estimation for Modular Vision-Inertial Systems', IEEE International Conference on Robotics and Automation,

[3] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R.: 'Vision meets Robotics: The KITTI Dataset', International Journal of Robotics Research, 2013, 31,(11), pp. 1231-1237

[4] Geiger, A., Ziegler, J., and Stiller, C.: 'Stereoscan: Dense 3D reconstruction in real-time', IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, Jun 05 - 09 2011, pp. 963-968.

[5] Hu, J.S. and Chen, M.Y.: 'A sliding-window visual-IMU odometer based on tri-focal tensor geometry', IEEE International Conference on Robotics and Automation, Hong Kong, China, May 31 - Jun 07 2014, pp. 3963-3968.

[6] Leutenegger, S., Furgale, P., Rabaudy, V. Chli, M., Konoligez, K. and Siegwart, R.: 'Keyframe-Based Visual-Inertial SLAM Using Nonlinear Optimization', The International Journal of Robotics Research, 2015, 34, (3), pp. 314-334.

[7] Mur-Artal, R., Tard, J. D.: 'ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras, and RGB-D Cameras', IEEE Transactions on Robotics, 2017, PP, (9), pp. 1-8.