

A Probabilistic-based Drift Correction Module for Visual Inertial SLAMs

Pouyan Navard, Alper Yilmaz

Photogrammetric Computer Vision Lab, The Ohio State University
(boreshnavard.1, yilmaz.15)@osu.edu

Keywords: SLAM, drift, positioning, dead-reckoning

Abstract

Positioning is a prominent field of study, notably focusing on Visual Inertial Odometry (VIO) and Simultaneous Localization and Mapping (SLAM) methods. Despite their advancements, these methods often encounter dead-reckoning errors that leads to considerable drift in estimated platform motion especially during long traverses. In such cases, the drift error is not negligible and should be rectified. Our proposed approach minimizes the drift error by correcting the estimated motion generated by any SLAM method at each epoch. Our methodology treats positioning measurements rendered by the SLAM solution as random variables formulated jointly in a multivariate distribution. In this setting, The correction of the drift becomes equivalent to finding the mode of this multivariate distribution which jointly maximizes the likelihood of a set of relevant geo-spatial priors about the platform motion and environment. Our method is integrable into any SLAM/VIO method as an correction module. Our experimental results shows the effectiveness of our approach in minimizing the drift error by $10\times$ in long treverses.

1. Introduction

Positioning plays a crucial role across a diverse array of applications within the realm of robotics and platform autonomy. At its core, dead reckoning stands out as a leading solution for determining the position of a mobile platform in environments where GPS signals are unavailable or unreliable. This method relies on Markovian state estimation principles, wherein the current position is inferred based on the previous state. However, due to the incremental nature of positioning updates and the inherent noise present in sensor measurements, dead reckoning-based techniques, including both inertial and visual odometry, are susceptible to accumulating errors over time. Consequently, while dead reckoning offers a valuable means of navigation in GPS-denied scenarios, mitigating error accumulation remains a significant challenge in enhancing the accuracy and reliability of such positioning methods.

Various techniques, including sensor fusion and Bayesian filtering such as the Kalman filter [Bresson et al. \(2015\)](#), have been employed to mitigate drift errors. Earlier vision-based methods like those proposed by [Horn and Schmidt \(1995\)](#); [Srinivasan et al. \(1997\)](#) address dead-reckoning errors, while ORB-SLAM [Mur-Artal et al. \(2015\)](#) offers real-time monocular SLAM functionality for both indoor and outdoor settings. VINS-MONO [Qin et al. \(2018\)](#), integrates images with inertial measurements for relative positioning. Despite incorporating heuristics like loop closure [Williams et al. \(2009\)](#) to reduce errors such as drift, reliability diminishes when errors accumulate excessively over time, leading to trajectory drift from the true position. This challenge persists, particularly in large-scale environments, due to the accumulative nature of Markovian estimation [Kerl et al. \(2013\)](#). There are also some research papaers that study the effect of using GPS in SLAM or VIO setting to improve the positioning accuracy. For example, [Kiss-Illés et al. \(2019\)](#) proposes an augmented version of ORB-SLAM which fuses GPS and inertial data to make the algorithm capable of dealing with low frame rate datasets. [Schleicher et al. \(2009\)](#) studied a real-time hierarchical (topological/metric) SLAM system that is exclusively based on the information provided by both a low-cost,

wide-angle stereo camera and a low-cost GPS in large-scale outdoor urban environments. [Alsayed et al. \(2018\)](#) used a Ensemble Multi Layer Perceptron (EMLP) model designed specifically for 2D likelihood SLAM methods. The EMLP model estimates the errors based on the likelihood distribution which makes it independent of the sensor used.

This paper introduces a probabilistic drift correction module tailored for real-time positioning applications. It can easily be plugged in into various relative positioning methodologies. The module operates under the premise that location estimates obtained from the positioning pipeline are inherently uncertain and thus treated as stochastic variables. At each epoch, we define probability density function over the SLAM's estimates to eventually build a unified multivariate distribution. The module incorporates geospatial priors, as probability distribution, about the characteristic of the platform's motion such as heading, and the surrounding environment into this multivariate framework. Estimating the mode of the multivariate distribution is equivalent to minimizing the accumulated drift error in the SLAM estimate. Section 3 discusses our methodology in detail and section 4 evaluates our proposed method in different scenarios.

2. Related Work

Positioning systems such as VIO [Qin et al. \(2018\)](#); [Forster et al. \(2014\)](#) and SLAM techniques [Mur-Artal et al. \(2015\)](#); [Zhou et al. \(2021\)](#); [Behley and Stachniss \(2018\)](#); [Keitaanniemi et al. \(2023\)](#) usually use camera, Inertial Measurement Unit (IMU) [Bai et al. \(2020\)](#) or other sensor configuration [Rivard et al. \(2008\)](#) to estimate the location. For example, VINS-Mono [Qin et al. \(2018\)](#) is a real-time optimization-based sliding window SLAM framework that leverages from IMU pre-integration with bias correction and loop detection. Usually, these methods suffer greatly from drift error mainly because their estimation approach is based on Markovian assumption [Nobre et al. \(2017\)](#); [Li et al. \(2020\)](#); [Liu and Meng \(2020\)](#); [Li et al. \(2022\)](#); [Zhao et al. \(2020\)](#). For instance [Botterill et al. \(2012\)](#) overcomes accumulating scale drift problem by recognizing certain objects through

the environment to and learning their scales to minimize the scale drift issue. There are also positioning systems that uses Global Navigation Satellite System (GNSS), that are very robust to the drift problem, such as GPS which can perform highly accurate positioning stand-alone or in conjunction with SLAM (Schleicher et al. (2009); Kiss-Illés et al. (2019)). For instance, Hening et al. (2017) introduces a data fusion method employing an Adaptive Extended Kalman filter (AFK) to estimate the velocity and position of a UAV. It combines data from a LIDAR sensor for local position updates via SLAM, corrected by GPS upon the availability of the GPS signal, and incorporates input from an Inertial Navigation System (INS) into the Extended Kalman filter. While GPS offers precise positioning, its signal can be significantly degraded or be denied in obstructed settings which creates Non-Line-Of-Sight (NLOS) or multipath effect for GPS commonly seen in scenarios such as outdoors environment with thick tree canopies, indoor environment or high altitude (Weiss et al. (2011); Øvstedal (2002); Chen and Zhu (2018)).

3. Methodology

Assume a platform is moving and, without the loss of generality, has planar motion. Using SLAM, a set of platform positions denoted by a set of points $\Gamma = \langle (x_0, y_0), (x_1, y_1), \dots, (x_n, y_n) \rangle$ is estimated. The differential geometric representation of the trajectory Γ is generated by taking the first-order derivative of consecutive points:

$$\Delta\Gamma = \langle (dx_1, dy_1), (dx_2, dy_2), \dots, (dx_{n-1}, dy_{n-1}) \rangle.$$

This differential representation removes the dependency on the reference frame selection but contains the drift effect. The geometric differential representation introduces the requirement for a known initial position and orientation as initial conditions to set up a differential system of equations introduced later in the discussion. Using two consecutive epochs from $\Delta\Gamma$, we convert the motion vectors into polar coordinates

$$\phi_t = \arctan2(dy_t, dx_t), \quad (1)$$

$$m_t = \sqrt{dx_t^2 + dy_t^2}, \quad (2)$$

and define angular motion at t as $\alpha_t = \phi_t - \phi_{t-1} \in [-\pi, \pi]$. $m_t \in \mathbb{R}$ in this representation becomes the motion magnitude at t . Due to measurement uncertainty, we assume that at t , platform angular velocity, X_1 , and true magnitude, X_2 are random variables with Gaussian distributions $X_1 \sim \mathcal{N}(\mu_1 = \alpha_t, \sigma_1^2)$ and $X_2 \sim \mathcal{N}(\mu_2 = m_t, \sigma_2^2)$ respectively. These two distributions represent observables in our module.

Spatial knowledge of the environment or object motion, represented by additional random variables, can significantly aid in reducing drift error when integrated into our module. Traversable regions within the spatial coverage, such as roads or walkways, are particularly useful for correcting drift errors and is obtained from sources like OpenStreetMap contributors (2017). These regions are typically represented as vectors in Geospatial Information Systems (GIS), which is rasterized for analysis. However, due to the complexity of the vector shapes, using an explicit function for parametric distribution isn't always feasible. To address this, in our paper, we convert rasterized paths into an implicit function, akin to those used in fluid dynamics (Sussman and Fatemi (1999)) for analyzing moving fronts. This implicit function, depicted in Figure 1, is based on the distance function (\mathcal{D}), yielding a distribution where $X_3 \sim \mathcal{N}(\mu_3 =$

$0, \sigma_3^2)$ where σ_3 modeling positioning uncertainty in the GIS map.



Figure 1. Implicit function representation of vector data through the inverse of the Distance Transform, depicting traversable paths with probabilities favoring center paths and lower values off-center. Maximum allowed distances are truncated akin to road width constraints in GIS-defined traversable paths.

Another example of a useful assumption is motion constraints, such as the heading preservation, H , that is added to the random variable set. This constraint assures that the current heading direction does not change radically between consecutive epochs. The random variable associated with this constraint is defined to follow normal distribution like the previous motion-related random variables $X_4 \sim \mathcal{N}(\mu_4 = 0, \sigma_4^2)$, and it shares the same calculation procedure as in (1).

In our multivariate recipe, each distribution discussed above contains useful beliefs about the environment and the traversed trajectory at time t , providing a practical utility to exploit these priors simultaneously. Considering independence between random variables, the multivariate distribution can be written as:

$$P = \prod_{i=1}^4 P(X_i | \mu_i, \sigma_i) \quad (3)$$

where each term in this equation generates a spatial distribution that is visualized in Fig. 2. Having formed the joint prior distribution, recursive filtering is applied to yield a posterior probability density function. This procedure is statistically equivalent to estimating the position as the distribution mode. Considering the model in 3, the maximization is more tractable when a negative log-likelihood approach converts the problem to a minimization problem. Therefore, (3) becomes:

$$P = - \sum_{n=1}^4 w_i \cdot \ln P(X_i | \mu_i, \sigma_i), \quad (4)$$

where $w_i \in (0, 1]$ is used to weigh the contribution of each constraint depending on their reliability in obtaining the corrected position estimate. The solution to this equation is computed by gradient descent.

4. Experiments

The experiments in this section use VINS-MONO, a visual and inertial SLAM for the relative positioning pipeline, to generate

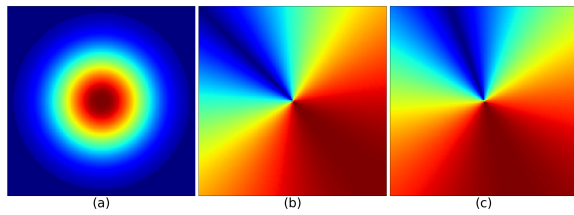


Figure 2. Spatial probabilities at $\Gamma(t)$ where red and blue denote high and low probability values, respectively. Random variables (a) motion magnitude X_2 , (b) motion angle X_1 (c) heading preservation X_4 .

Γ . The data is recorded on a smartphone using the MarsLogger tool [Huai et al. \(2019\)](#) that records images at 30Hz and IMU measurements at 100Hz synced to the same clock source. We collected three scenarios containing trajectories with sharp turns, long straight traverses, and loop closures to test the proposed drift correction module to SLAM.

To resolve for the mode of the proposed multivariate distribution depicted in 4, we used PyTorch [Paszke et al. \(2017\)](#) to perform Stochastic Gradient Decent [Bottou et al. \(1991\)](#) optimization with learning rate of 0.01. For taking the derivative of the distance transform grid depicted in 1, bilinear interpolation is used to estimate the derivatives for non-discrete coordinates where derivative is not directly available from the grid.

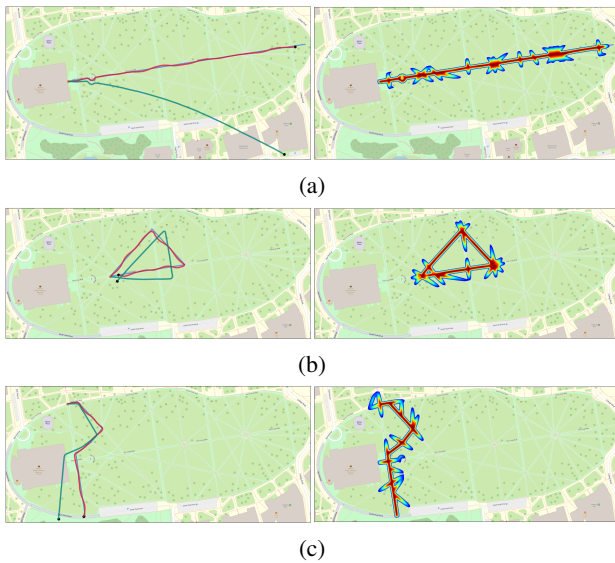


Figure 3. Comparison of reference, proposed, and SLAM trajectories (blue, red, and green) with start and end points highlighted (orange and black), alongside spatial distributions generated by the proposed method on the right had side.

The proposed trajectories in Fig. 3 qualitatively follow the ground truth in all testing scenarios. However, the instability in the proposed trajectories occurs due to the competing effect of having different terms in the mixture model. One can perform an ablation study to choose the correct distribution weights w_i and the model's uncertainty parameters σ_i .

Table 1 compares the closing distances between the estimated trajectories, using our module and SLAM only, and the reference trajectory generated from the GPS readings. The labels in the first column of the table respectively correspond to the labels in Fig. 3. We can see that the drift correction module successfully overcomes the drift error by ten to twenty times

for long traverses. For the loop closure example, the results are similar, yet as observed from the qualitative results, the maximum error is higher.

Table 1. Computed trajectory's closing distance to reference trajectory, reported in meters up to decimeter error, accounting for GPS sensor reliability based on specifications.

Fig 3 part	Our Method (m.)	SLAM only (m.)
(a)	15.8	171.8
(b)	12.7	12.6
(c)	2.3	40.9

5. Conclusion

This paper proposes a differential geometric correction module for the drift error in localization and navigation pipelines, such as SLAM or VIO. The drift error results in significant deviation, especially when the platform traverses large-scale environments. The proposed drift correction module formulates the platform positioning problem as a mode seeking of a multivariate probability distribution generated from differential geometric observations. This strategy and the independence of the module from the SLAM approach make the proposed drift correction to be used by most SLAM algorithms without difficulty. The results for various scenarios show the module successfully corrects the drift.

References

Alsayed, Z., Bresson, G., Verroust-Blondet, A., Nashashibi, F., 2018. 2d slam correction prediction in large scale urban environments. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 5167–5174.

Bai, N., Tian, Y., Liu, Y., Yuan, Z., Xiao, Z., Zhou, J., 2020. A high-precision and low-cost IMU-based indoor pedestrian positioning technique. *IEEE Sensors Journal*, 20(12), 6716–6726.

Behley, J., Stachniss, C., 2018. Efficient surfel-based slam using 3d laser range data in urban environments. *Robotics: Science and Systems*, 2018, 59.

Botterill, T., Mills, S., Green, R., 2012. Correcting scale drift by object recognition in single-camera SLAM. *IEEE transactions on cybernetics*, 43(6), 1767–1780.

Bottou, L. et al., 1991. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, 91(8), 12.

Bresson, G., Aufrère, R., Chapuis, R., 2015. Improving slam with drift integration. *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, IEEE, 2700–2706.

Chen, C., Zhu, H., 2018. Visual-inertial SLAM method based on optical flow in a GPS-denied environment. *Industrial Robot: An International Journal*, 45(3), 401–406.

Forster, C., Pizzoli, M., Scaramuzza, D., 2014. Svo: Fast semi-direct monocular visual odometry. *2014 IEEE international conference on robotics and automation (ICRA)*, IEEE, 15–22.

Hening, S., Ippolito, C. A., Krishnakumar, K. S., Stepanyan, V., Teodorescu, M., 2017. 3d lidar slam integration with gps/ins for uavs in urban gps-degraded environments. *AIAA Information Systems-AIAA Infotech@ Aerospace*, 0448.

- Horn, J., Schmidt, G., 1995. Continuous localization of a mobile robot based on 3D-laser-range-data, predicted sensor images, and dead-reckoning. *Robotics and Autonomous Systems*, 14(2-3), 99–118.
- Huai, J., Zhang, Y., Yilmaz, A., 2019. The mobile ar sensor logger for android and ios devices. *2019 IEEE SENSORS*, IEEE, 1–4.
- Keitaanniemi, A., Rönholm, P., Kukko, A., Vaaja, M. T., 2023. Drift analysis and sectional post-processing of indoor simultaneous localization and mapping (SLAM)-based laser scanning data. *Automation in Construction*, 147, 104700.
- Kerl, C., Sturm, J., Cremers, D., 2013. Dense visual slam for rgb-d cameras. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2100–2106.
- Kiss-Illés, D., Barrado, C., Salamí, E., 2019. GPS-SLAM: An augmentation of the ORB-SLAM algorithm. *Sensors*, 19(22), 4973.
- Li, X., Wang, W., Chen, J., Zhang, X., 2022. DR-SLAM: drift rejection SLAM with Manhattan regularity for indoor environments. *Advanced Robotics*, 36(20), 1049–1059.
- Li, Y., Brasch, N., Wang, Y., Navab, N., Tombari, F., 2020. Structure-slam: Low-drift monocular slam in indoor environments. *IEEE Robotics and Automation Letters*, 5(4), 6583–6590.
- Liu, J., Meng, Z., 2020. Visual slam with drift-free rotation estimation in manhattan world. *IEEE Robotics and Automation Letters*, 5(4), 6512–6519.
- Mur-Artal, R., Montiel, J. M. M., Tardos, J. D., 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, 31(5), 1147–1163.
- Nobre, F., Kasper, M., Heckman, C., 2017. Drift-correcting self-calibration for visual-inertial slam. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 6525–6532.
- OpenStreetMap contributors, 2017. Planet dump retrieved from <https://planet.osm.org> . <https://www.openstreetmap.org>.
- Øvstedal, O., 2002. Absolute positioning with single-frequency GPS receivers. *GPS Solutions*, 5(4), 33–44.
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A., 2017. Automatic differentiation in PyTorch.
- Qin, T., Li, P., Shen, S., 2018. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4), 1004–1020.
- Rivard, F., Bisson, J., Michaud, F., Létourneau, D., 2008. Ultrasonic relative positioning for multi-robot systems. *2008 IEEE International Conference on Robotics and Automation*, IEEE, 323–328.
- Schleicher, D., Bergasa, L. M., Ocaña, M., Barea, R., López, M. E., 2009. Real-time hierarchical outdoor SLAM based on stereovision and GPS fusion. *IEEE Transactions on Intelligent Transportation Systems*, 10(3), 440–452.
- Srinivasan, M. V., Chahl, J. S., Zhang, S.-W., 1997. Robot navigation by visual dead-reckoning: inspiration from insects. *International Journal of Pattern Recognition and Artificial Intelligence*, 11(01), 35–47.
- Sussman, M., Fatemi, E., 1999. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on scientific computing*, 20(4), 1165–1191.
- Weiss, S., Scaramuzza, D., Siegwart, R., 2011. Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics*, 28(6), 854–874.
- Williams, B., Cummins, M., Neira, J., Newman, P., Reid, I., Tardós, J., 2009. A comparison of loop closing techniques in monocular SLAM. *Robotics and Autonomous Systems*, 57(12), 1188–1197.
- Zhao, Y., Smith, J. S., Karumanchi, S. H., Vela, P. A., 2020. Closed-loop benchmarking of stereo visual-inertial slam systems: Understanding the impact of drift and latency on tracking accuracy. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 1105–1112.
- Zhou, L., Koppel, D., Kaess, M., 2021. LiDAR SLAM with plane adjustment for indoor environment. *IEEE Robotics and Automation Letters*, 6(4), 7073–7080.