


Article

Accurate Collaborative Globally-Referenced Digital Mapping with Standard GNSS

Lakshay Narula ^{1,*} , J. Michael Wooten ², Matthew J. Murrian ², Daniel M. LaChapelle ² and Todd E. Humphreys ²

¹ Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712, USA

² Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712, USA; jmwooten@utexas.edu (J.M.W.); matthew.murrian@utexas.edu (M.J.M.); daniel.m.lachapelle@utexas.edu (D.M.L.); todd.humphreys@mail.utexas.edu (T.E.H.)

* Correspondence: lakshay.narula@utexas.edu

Received: 16 June 2018; Accepted: 26 July 2018; Published: 28 July 2018



Abstract: Exchange of location and sensor data among connected and automated vehicles will demand accurate global referencing of the digital maps currently being developed to aid positioning for automated driving. This paper explores the limit of such maps' globally-referenced position accuracy when the mapping agents are equipped with low-cost Global Navigation Satellite System (GNSS) receivers performing standard code-phase-based navigation, and presents a globally-referenced electro-optical simultaneous localization and mapping pipeline, called GEOSLAM, designed to achieve this limit. The key accuracy-limiting factor is shown to be the asymptotic average of the error sources that impair standard GNSS positioning. Asymptotic statistics of each GNSS error source are analyzed through both simulation and empirical data to show that sub-50-cm accurate digital mapping is feasible in the horizontal plane after multiple mapping sessions with standard GNSS, but larger biases persist in the vertical direction. GEOSLAM achieves this accuracy by (i) incorporating standard GNSS position estimates in the visual SLAM framework, (ii) merging digital maps from multiple mapping sessions, and (iii) jointly optimizing structure and motion with respect to time-separated GNSS measurements.

Keywords: vehicle localization; SLAM; sensor fusion

1. Introduction

Localization is one of the primary operations that connected and automated vehicles must perform, both to navigate from one location to another and to interact with each other and with their surroundings within a mapped environment. Satellite-based navigation sensors have historically been the unrivalled sensor of choice for navigating from source to destination. However, the high-reliability sub-50-cm precision demanded by automated vehicles for lane-keeping and other applications, especially in urban areas, has significantly changed this landscape [1]. In most automated vehicles being developed, the Global Positioning System (GPS)/Global Navigation Satellite System (GNSS) is relegated to a secondary sensor whose role is to loosely constrain (within a few meters) the primary localization sensors, usually camera(s) and/or LiDAR, to a global reference frame when building a digital map. The vehicles then locate themselves to decimeter accuracy within this digital map.

Automated driving does not necessarily demand sub-50-cm agreement between the coordinates of a given point in the digital map and the coordinates of the same point in a well-defined global

reference frame. Rather, local self-consistency and accurate localization within the digital map is of greater importance. However, consistency of the digital map with a global coordinate frame is likely to become a pre-requisite for cooperative automated driving. Automated driving can be made more efficient (e.g., by platooning) and safe if basic safety information such as vehicle location, velocity, intent, etc. are shared among neighboring agents through vehicle-to-vehicle (V2V) and/or vehicle-to-everything (V2X) communication [2,3]. In many cases, such information can be perceived independently by each agent via its exteroceptive sensors. But in other situations—for example at a blind corner or during a left turn maneuver—full situational awareness will require external sensors—on other vehicles or on infrastructure—that wirelessly communicate data to the ego-vehicle. If all collaborating vehicles navigate within the same digital map, exchange of information can be performed with sub-50-cm precision [4,5], even if the map itself is only globally accurate to a few meters. However, it is unlikely that automated vehicles from different manufacturers will rely on a common digital map. Consequently, the precision of the exchanged vehicle position is lower-bounded by the disagreement on the coordinates of the same physical location between different maps. Thus, exchange of accurate vehicle pose among vehicles, as well as other associated high-level information such as sensor data in the vehicle's body frame, will demand consistency among, or translation between, different digital maps.

Standard code-phase-based GNSS position measurements, such as those provided by all mass-market GNSS receivers, may be biased by as much as 3–5 meters on any given mapping session. Maps anchored by these measurements may not exhibit lane-level consistency with each other. One possible solution is to create digital maps with decimeter-accurate carrier-phase differential GNSS (CDGNSS) systems [6]. However, at current prices, such systems can only be installed on a limited fleet of specialized mapping vehicles. Precise point positioning (PPP) techniques offer a low-cost alternative to CDGNSS, but the frequent cycle-slipping experienced in urban areas impedes the convergence of PPP techniques [7].

While mapping with a specialized fleet is feasible for urban areas, it is time-consuming and cumbersome to create and maintain maps of entire continents. Thus, a key enabler for large-scale up-to-date maps will be enlisting the help of the very consumer vehicles that need the map to build and update it. Consumer vehicles will likely be equipped with low-cost consumer-grade sensor suites. Accordingly, this paper explores the accuracy limit of globally-referenced mapping involving collaborating consumer vehicles whose sense of global position is based on standard code-phase-based GNSS receivers. Key parameters in this exploration are the asymptotic averages of the error sources that impair code-phase-based GNSS positioning: receiver thermal noise, satellite clock and orbit errors, ionospheric and tropospheric modeling errors, and multipath. One or more vehicles navigating through a digital map over time make multiple time-separated GNSS measurements of the same location. If these vehicles collaboratively update the map over multiple sessions, then the GNSS errors are averaged across all sessions with appropriate weighting.

Are the GNSS errors at every map location—including deep urban locations—asymptotically zero-mean, or, on the contrary, do location-dependent biases persist in averages of time-separated standard GNSS measurements? Such is the question this paper seeks to address. To this end, it describes and demonstrates a stereo-camera-based digital mapping pipeline called GEOSLAM (globally-referenced electro-optical simultaneous localization and mapping) that achieves the accuracy limit of digital mapping with standard GNSS. GEOSLAM combines standard GNSS position estimates with a visual SLAM system in a tightly-coupled architecture.

To achieve the accuracy limit, GEOSLAM merges and jointly optimizes maps over multiple sessions with time-separated GNSS position estimates. This paper details the techniques GEOSLAM invokes to smoothly transition between unmapped and previously-mapped regions, consistently fusing current and prior maps without the need for a six degrees-of-freedom (6-DoF) pose from an inertial navigation system (INS). GEOSLAM enables multi-agent collaborative mapping by storing and rendering its map in a global

frame of reference, such as the World Geodetic System 1984 or the International Terrestrial Reference Frame. Multi-session operation of GEOSLAM is demonstrated using camera and GNSS data collected in a moderate urban environment, and the accuracy of global localization with the multi-session GEOSLAM map is assessed with respect to CDGNSS-based ground truth.

2. Previous Work

Improving the accuracy of maps by averaging GPS/GNSS tracks has been explored previously using a variety of approaches. An early effort, detailed in [8], proposed the precise determination of lane centerlines by clustering and averaging the GNSS tracks of probe vehicles. The accuracy of the estimated centerline was assessed in terms of the spread of GNSS tracks, assuming, without analysis, that the error was zero-mean at every location. Likewise, [9] invoked the zero-mean assumption without examination. More recently, [10] proposed vehicle lane determination via PPP on a rural road under open-sky conditions. The current paper aims to perform localization at a similar accuracy level, but in urban and suburban areas and with the aid of a digital mapping sensor.

Minimizing the difference between GNSS measurements and the assigned map coordinates of locations visited multiple times by probe vehicles has been a common feature of the seminal works on map-based precise localization in urban environments for automated driving [4,5], but no analysis of the accuracy of the resulting map in the global coordinate system was provided.

The effect of multipath on measured pseudoranges was studied extensively for various signal types in [11]. However, this study was done under open-sky conditions with a static survey-grade antenna, hardly representative of a mass-market receiver in an urban environment. The effect of multipath from a single large reflector on a static receiver was studied in [12], where it was shown that the position solution of the receiver was significantly biased when tracking multipath-afflicted signals. However, no details were provided about the tracking strategy and navigation filter. Multipath characterization for a dynamic receiver in an urban area was performed in [13], but the study was carried out for a single run through the test area, and many of the important error sources (e.g., atmospheric modeling errors), were assumed to be negligible. A detailed study on the distribution of code-phase and Doppler offsets of the multipath components from individual satellites in a dynamic urban setting was carried out in [14]. However, the error was characterized as the combined distribution of code phase delays over the entire duration of the run, which marginalizes over the temporally- and physically-local biases. On the contrary, this paper explores the errors in the position domain for repeated sessions through a given realization of an urban corridor.

Other GNSS error sources such as errors in modeling of ionospheric [15] and tropospheric [16] delay have been studied extensively over many decades, and their long-term error characteristics have also been reported in the literature. However, the impact of these errors on the asymptotic statistics of code-phase-based GNSS position estimates has not been previously presented.

To the authors' best knowledge, despite the apparent simplicity of the problem, no prior work has studied the long-term statistics of GNSS errors in an urban environment representative of the conditions to be encountered by consumer vehicles creating digital maps anchored by code-phase-based GNSS positioning. One of this paper's primary contributions is to address this gap in the literature.

Sensor fusion of visible-light cameras and GNSS has been extensively studied [17–28]. Some of these works [26–28] have proposed visual odometry as a replacement for, or an augmentation of, the traditional GNSS/INS architecture. Visual data from cameras are exploited to perform dead reckoning in a visual odometry pipeline, wherein an important distinction from the current paper is that the 3D map points do not persist after a window of time has elapsed—that is, no map of feature points is maintained.

Clearly, such an approach does not allow improvement of the 3D map point positions over multiple mapping sessions.

In [25], the relative change in position between two image frames is first estimated based on time-differenced GNSS carrier phase measurements. The metric-accurate GNSS-derived change in position is exploited to initialize the otherwise unobservable scale in a monocular visual SLAM system. However, GNSS measurements are not incorporated any further once the absolute scale has been initialized. Unlike the current paper, the visual SLAM map is rendered in the arbitrary SLAM coordinate frame since only the relative change in position, and not the absolute position, was estimated based on GNSS measurements.

The vision-GNSS fusion in the current paper is closely aligned with the bi-objective bundle adjustment (BA) optimization techniques previously reported in [17,18,20,22–24]. In [17,18,20,22], the traditional visual SLAM reprojection cost function is jointly minimized along with a GNSS position error term. The methods proposed in [23,24] are also similar, but guarantee that the visual reprojection cost after incorporation of the GNSS term is not significantly greater than the visual-only case. However, none of these works showed significant empirical evidence of their efficacy on real-world vehicle data sets. Furthermore, collaborative mapping or multi-session improvement of the map was not discussed.

Collaborative multi-agent mapping, without GNSS aiding, has also been extensively discussed in the literature [29–35]. Some of these proposed solutions require significant overlap in the field-of-view of the agents, or require that the relative pose transformation between the agents be known a priori [30,32]. Other solutions, such as in [31,33–35], enable collaboration by performing data association between non-concurrent mapping sessions where the relative pose transformation between the agents is unknown. The multi-session strategy employed in this paper is similar, but with an important distinction: none of the previous works on collaborative mapping have incorporated GNSS measurements in the map-building process. Without global referencing, the problem of data association between non-concurrent sessions becomes intractable. With no estimate of the pose for the mapping platform in relation to the existing map, data association must be attempted against the entire map. It is easily observed that such data association will become infeasible when scaled to city- or country-wide maps. The current paper proposes rendering and storage of digital maps in a global coordinate frame, such that a new mapping session can readily estimate its approximate pose in relation to the prior map, and perform data association on a small segment of the prior map that is expected to be in view of the vision system.

The work presented in [36] is perhaps the most closely related to the current paper. In [36], a particular stretch of roadway is mapped 25 times with a low-cost sensor setup. However, [36] assumes, without detail, the availability of a lane-level accurate low-cost positioning module that provides the full 6-DoF pose for the mapping platform. This greatly simplifies the ensuing data association and mapping pipeline. No mention is made of the general setting of the roadway being mapped (open-sky highway, urban canyon, etc.), and while the accuracy of the mapped traffic signs is adequately reported, localization within the map is not discussed, presumably since lane-level accurate positioning is already available. Meanwhile, the current paper only assumes the availability of a meter-level accurate code-phase-based GNSS receiver that provides 3D position estimates. Global localization accuracy of a vehicle operating within the multi-session map is presented as the key performance indicator.

3. GNSS Error Analysis

3.1. Low-Cost GNSS in Urban Areas

Low-cost multi-GNSS receiver manufacturers have recently announced the development and release of low-cost multi-frequency multi-GNSS receivers [37]. Accordingly, the analysis in this section considers a vehicular platform equipped with a multi-frequency multi-GNSS receiver capable of tracking both code and carrier phase of GNSS signals.

Development of an extensive dense reference network in support of CDGNSS consumer vehicular positioning in urban areas, as suggested in [38], could be an expensive affair. PPP is a low-cost alternative to CDGNSS that requires only a sparse network of reference stations across the globe, but is not considered a viable option for urban GNSS positioning in this paper because the constant cycle slips and outages experienced in urban areas [6] make it difficult for PPP's float carrier phase ambiguity estimates to converge [7], in which case PPP degrades to code-phase positioning accuracy.

While convergence of PPP carrier-phase ambiguities may be infeasible in urban areas, a partial PPP solution that exploits precise satellite orbits and clocks, as well as ionospheric and tropospheric corrections, can certainly improve the accuracy of code-phase-based GNSS position estimates. Since connected and automated vehicles will perform enjoy network connectivity, this paper assumes the availability of such GNSS corrections. Thus, the kind of GNSS errors assessed in this section lie between those corresponding to the two extremes of standard standalone code-phase positioning and PPP. This type of GNSS positioning, hereafter referred to as enhanced code-phase positioning, exploits both code and carrier phase or frequency tracking, but, as opposed to PPP, does not attempt to estimate a quasi-constant float carrier phase ambiguity, making it suitable for urban applications.

3.2. Pseudorange Measurement

The pseudorange measurement at receiver R from satellite S_i is modeled as

$$\begin{aligned}\rho_i(t_R) &= h_i[\mathbf{x}(t_R), I_{\rho_i}(t_R), T_i(t_R), t_R] + w_{\rho_i}(t_R) \\ &= \Delta r_i + c[\delta t_R(t_R) - \delta t_{S_i}(t - \delta t_{\text{TOF}_i})] + I_{\rho_i}(t_R) + T_i(t_R) + w_{\rho_i}(t_R),\end{aligned}\quad (1)$$

where

$$\mathbf{x}(t_R) \triangleq \begin{bmatrix} \mathbf{r}_R(t_R) \\ \delta t_R(t_R) \end{bmatrix}$$

is the state of the receiver, comprising the receiver position, $\mathbf{r}_R(t_R)$, at the time of the signal receipt event, t_R , and the receiver clock bias, $\delta t_R(t_R) = t_R - t$, with respect to true time t . The nonlinear measurement model is denoted by h_i ; ρ_i denotes the measured pseudorange to S_i ; c denotes the speed of light in vacuum; $\delta t_{S_i}(t) = t_{S_i} - t$ denotes the satellite clock bias with respect to t ; δt_{TOF_i} denotes the time-of-flight of the signal from S_i , as an increment in true time; I_{ρ_i} and T_i denote the ionospheric and tropospheric delay experienced by the signal from S_i , respectively; $w_{\rho_i} \sim (\mu_{w_i}, \sigma_{w_i}^2)$ denotes the sum of measurement thermal noise, multipath interference, non-line-of-sight (NLOS) delay, and other unmodeled errors; and Δr_i denotes the true range between R and S_i , given as

$$\Delta r_i = \|\mathbf{r}_R(t_R) - \mathbf{r}_{S_i}(t_R - \delta t_R(t_R) - \delta t_{\text{TOF}_i})\|,$$

where \mathbf{r}_{S_i} is the satellite position at the signal transmit event. Note from (1) that the receiver clock bias component of the state contributes identically to all pseudorange measurements.

Taking n_z pseudorange measurements $\{\rho_i\}_{i=1}^{n_z}$ and predictions \bar{I}_{ρ_i} and \bar{T}_i for each measurement, R estimates its state by solving a nonlinear least squares problem based on (1). First, it linearizes the measurement model in (1) about an initial guess of its state $\bar{\mathbf{x}}(t_R) = [\bar{\mathbf{r}}_R^T(t_R) \bar{\delta t}_R(t_R)]^T$ and the modeled atmospheric delays:

$$\rho_i \approx h_i(\bar{\mathbf{x}}, \bar{I}_{\rho_i}, \bar{T}_i, t_R) + \underbrace{\left[\frac{\partial h_i}{\partial \mathbf{x}} \right]_{\mathbf{x}=\bar{\mathbf{x}}}}_{H_i} (\mathbf{x} - \bar{\mathbf{x}}) + \bar{I}_{\rho_i} + \bar{T}_i + w_{\rho_i},$$

with $\tilde{I}_{\rho_i} = I_{\rho_i} - \bar{I}_{\rho_i}$ and $\tilde{T}_i = T_i - \bar{T}_i$. Representing all n_z measurements in matrix form yields

$$\begin{bmatrix} \rho_1 \\ \vdots \\ \rho_{n_z} \end{bmatrix} = \begin{bmatrix} h_1(\bar{\mathbf{x}}, \bar{I}_{\rho_1}, \bar{T}_1, t_R) \\ \vdots \\ h_{n_z}(\bar{\mathbf{x}}, \bar{I}_{\rho_{n_z}}, \bar{T}_{n_z}, t_R) \end{bmatrix} + \begin{bmatrix} H_1 \\ \vdots \\ H_{n_z} \end{bmatrix} (\mathbf{x} - \bar{\mathbf{x}}) + \begin{bmatrix} \tilde{I}_{\rho_1} \\ \vdots \\ \tilde{I}_{\rho_{n_z}} \end{bmatrix} + \begin{bmatrix} \tilde{T}_1 \\ \vdots \\ \tilde{T}_{n_z} \end{bmatrix} + \begin{bmatrix} w_1 \\ \vdots \\ w_{n_z} \end{bmatrix}$$

or

$$\boldsymbol{\rho} = \mathbf{h}(\bar{\mathbf{x}}, \bar{\mathbf{I}}, \bar{\mathbf{T}}, t_R) + \mathbf{H}(\mathbf{x} - \bar{\mathbf{x}}) + \tilde{\mathbf{I}} + \tilde{\mathbf{T}} + \mathbf{w}. \quad (2)$$

Rearranging measured and modeled quantities on the left-hand side to get the standard form for a linearized measurement model yields

$$\mathbf{z} \triangleq \boldsymbol{\rho} - \mathbf{h}(\bar{\mathbf{x}}, \bar{\mathbf{I}}, \bar{\mathbf{T}}, t_R) + \mathbf{H}\bar{\mathbf{x}} \implies \mathbf{z} = \mathbf{H}\mathbf{x} + \tilde{\mathbf{I}} + \tilde{\mathbf{T}} + \mathbf{w}. \quad (3)$$

The i th row of the measurement sensitivity matrix \mathbf{H} is

$$H_i \approx \left[\frac{\bar{\mathbf{r}}_R^T(t_R) - \bar{\mathbf{r}}_{S_i}^T(t_R - \delta t_R - \delta t_{\text{TOF}_i})}{\|\bar{\mathbf{r}}_R(t_R) - \bar{\mathbf{r}}_{S_i}(t_R - \delta t_R - \delta t_{\text{TOF}_i})\|}, 1 \right].$$

By solving (3) for \mathbf{x} , updating $\bar{\mathbf{x}}$, and iterating until convergence, R obtains its state estimate $\hat{\mathbf{x}}(t_R)$:

$$\hat{\mathbf{x}}(t_R) \triangleq \begin{bmatrix} \hat{\mathbf{r}}_R(t_R) \\ \hat{\delta t}_R(t_R) \end{bmatrix}.$$

For dynamic applications such as vehicle tracking, the state $\mathbf{x}(t_R)$ is typically augmented to include the time derivatives of $\mathbf{r}_R(t_R)$ and $t_R(t_R)$, and the measurement model typically assumes direct measurement of apparent Doppler frequency.

3.3. Error Sources

The major sources of error in the estimates $\hat{\mathbf{r}}_R$ and $\hat{\delta t}_R$ are as follows:

3.3.1. Thermal Noise

Measurement thermal noise at the receiver is one of the components of w_{ρ_i} in (1). The effect of thermal noise can be accurately modeled as a white Gaussian random variable with zero mean and standard deviation σ_T . For the pseudorange measurement, σ_T is typically between 10–30 cm, depending on the signal carrier-to-noise ratio, signal bandwidth, and receiver tracking bandwidth [39]. Estimation of the receiver state from multiple appropriately-weighted measurements with independent thermal-noise errors, and processing such measurements over time through a filter based on the modeled dynamics of the receiver, renders negligible the position-domain effects of uncorrelated zero-mean thermal noise. As a result, thermal noise is not a major contributor to the asymptotic accuracy of a digital map.

3.3.2. Satellite Orbit and Clock Errors

Satellite orbit and clock errors manifest in the modeled satellite position $\bar{\mathbf{r}}_{S_i}$ and the modeled satellite clock bias $\bar{\delta t}_{S_i}$. The International GNSS Service (IGS) provides orbit and clock models for GNSS satellites. The predicted *ultra rapid* orbits and satellite clocks have an accuracy of ~ 5 cm and ~ 3 ns, respectively [40]. These may add up to ~ 1 m of combined pseudorange model error for a given satellite. The 17-h retroactively-available *rapid* orbits and satellite clock models are accurate to ~ 2.5 cm and ~ 75 ps

RMS errors, respectively [40], adding up to less than 5 cm of RMS error in the modeled pseudorange for a given signal. Since the orbit and clock parameters are fit to measurements made at IGS analysis centers, the errors in the estimated parameters must be asymptotically zero-mean by design of the estimator. For post-processing applications such as mapping, it is reasonable to assume the availability of *rapid* orbit and satellite clock products, and thus the asymptotic average position errors due to errors in modeled satellite position and clock bias can be reduced to a sub-5-cm level.

3.3.3. Ionospheric Modeling Errors

The code-modulated GNSS signal propagates slower through the ionosphere as compared to vacuum due to the slightly-greater-than-unity *group* index of refraction for this atmospheric layer. The excess group delay is given as

$$\Delta\tau_g = \frac{40.3 \cdot TEC}{cf^2},$$

where *TEC* is the total electron content in electrons/m² and *f* is the frequency of the propagating signal. At the GPS L1 frequency centered at 1575.42 MHz, the excess ionospheric group delay is roughly 16.24 cm per TECU (1 TECU \triangleq 10¹⁶ electrons/m²). If not modeled, the ionospheric delay can lead to ranging errors greater than 15 m.

The ionospheric delay can be estimated via an ionosphere model or, in case of a multi-frequency receiver, eliminated via a combination of multiple-frequency pseudorange measurements. The latter technique does not require any external aiding, but the formation of the ionosphere-free combination exacerbates pseudorange noise, including any biases due to tracking of multipath signals. Compensating for ionospheric delay with the aid of an ionosphere model is applicable to both single- and multi-frequency receivers. It relies on accurate delay modeling based on ionospheric measurements at permanent GNSS reference stations, such as those that form the IGS network. While both methods have their merits, the analysis in this section considers corrections from an ionospheric model, and thus will not be relevant to applications where the ionosphere-free combination is applied. Note that those applications would likely experience worse multipath errors than the ones presented later, requiring a separate multipath analysis along the lines of Section 3.3.5.

Ionospheric model accuracy was studied comprehensively in [15]. The method in [15] generates unambiguous carrier-phase measurements from a global distribution of permanent receivers to compute the true slant total electronic content (STEC) for each satellite, and compares the model prediction for a number of models with the ground truth. In [41], the same authors compared PPP convergence times when applying different ionospheric correction models. This section extends the analysis in [15,41] to examine whether there exist long-term position-domain biases in enhanced code-phase positioning.

The post-fit residuals for multiple regional and global ionospheric models, computed as described in [15], were graciously made available by the same authors for the year 2014. These residuals were computed for GPS signals as observed at about 150 reference stations around the globe at 5 min intervals.

To observe the position-domain effect of the ionospheric modeling errors in isolation, this section neglects all other error sources, reducing the linearized measurement model in (3) to

$$z = Hx + \tilde{I}.$$

Historical GPS satellite almanacs can be combined with the timestamps from the residuals data to obtain the measurement sensitivity matrix *H* at each epoch for each station. With an elevation-dependent measurement covariance matrix *R*, the error in the weighted least-squares solution due to errors in ionospheric modeling is

$$\hat{x} - x = \left(H^T R^{-1} H \right)^{-1} H^T R^{-1} \bar{I}.$$

Figure 1 summarizes the results for ionospheric corrections obtained from the IGS global ionospheric map (GIM). Each of the arrows in Figure 1 points in the direction of the position bias in the east-north plane, as estimated over 12 months of data from 2014 (more than 800,000 samples per station). The magnitude of the horizontal position bias is depicted by the color of the arrow according to the scale shown on the right. Interestingly, there is a clear trend of southward bias in the position error for most stations in the northern hemisphere, and a mild trend of northward bias in the position error for stations in the southern hemisphere. A numerical summary of the IGS GIM position bias is presented in Table 1, along with a similar analysis for the Wide Area Augmentation System (WAAS) ionospheric corrections available for the contiguous United States (CONUS) region. As reported in [15], the WAAS model was found to exhibit a significantly smaller RMS error in ionosphere TEC estimates when compared to the IGS GIM; however the long-term position bias due to WAAS corrections is similar to or worse than those for the IGS model.

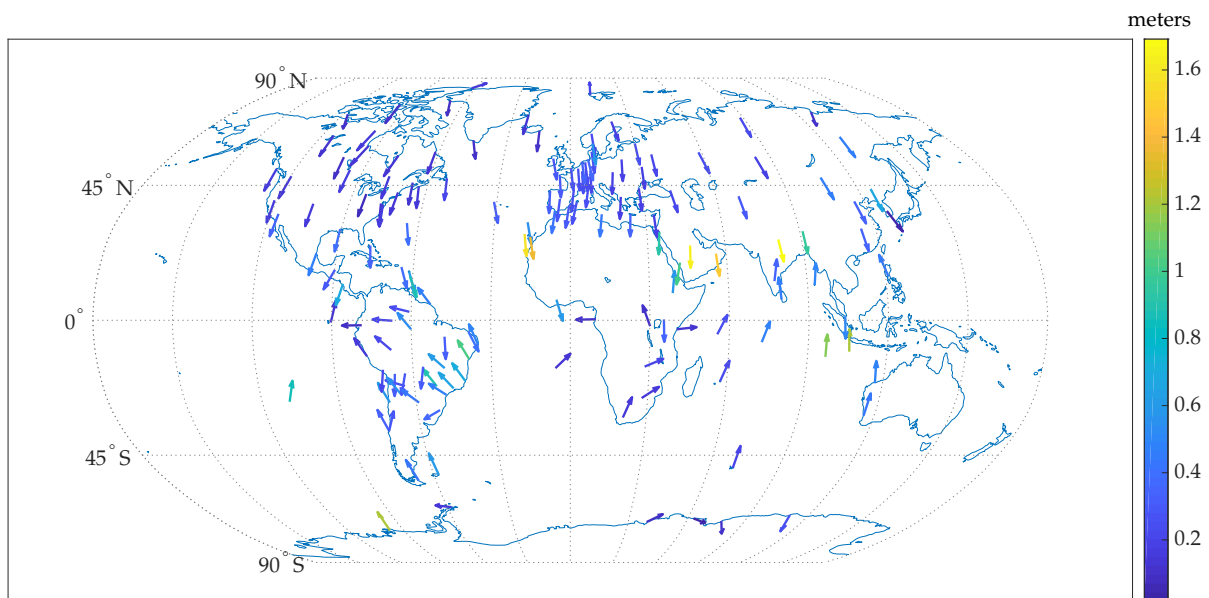


Figure 1. Direction and magnitude (the latter represented by color, in meters) of the long-term average horizontal position error due to errors in the delay estimates provided by the IGS GIM. Note that the meridians are curved outwards due to projection of the spherical map, and that arrows parallel to the curved meridians point directly south or north.

Another global ionospheric model, the Fast PPP model [41], was also studied as above. Fast PPP natively models the ionosphere as a two-layered shell, but is also made available in the standard one-layer IONEX (ionosphere-map exchange) format [15] for dissemination. The results presented in Table 1 represent the IONEX version of Fast PPP. In comparison with the IGS corrections, it is clear that the Fast PPP IONEX GIM corrections result in substantially unbiased long-term position errors at the global test locations. However, it must be conceded that the results in Table 1 are best-case results, as they are based on data from the same permanent reference stations used to constrain the model.

Table 1. Long-term average position error due to ionospheric model errors (ϕ denotes station latitude). IGS: International Global Navigation Satellite System (GNSS) Service; PPP: precise point positioning; WAAS: Wide Area Augmentation System; CONUS: contiguous United States; IONEX: ionosphere-map exchange format.

Ionosphere Model	Region	East (m)	North (m)	Up (m)
IGS	$\phi \geq 30^\circ$	0.0107	-0.2129	0.6733
	$30^\circ > \phi > -30^\circ$	-0.0651	-0.0692	1.5467
	$\phi \leq -30^\circ$	0.0237	0.2450	0.3355
WAAS	CONUS	-0.0048	-0.2916	-0.1248
Fast PPP IONEX	$\phi \geq 30^\circ$	-0.0042	-0.0099	-0.0122
	$30^\circ > \phi > -30^\circ$	-0.0390	0.0013	-0.3053
	$\phi \leq -30^\circ$	-0.0325	-0.0087	0.0309

To understand the reason behind the systematic biases with IGS corrections, note that any ionospheric modeling bias that identically affects all satellites does not have any impact on the accuracy of the GNSS position solution, as this common error is absorbed in $\hat{\delta}t_R$. Rather, position-domain biases arise from the azimuthal- and elevation-dependence of ionosphere model errors. From analysis of the spatial distribution of post-fit residuals, it was found that appreciable azimuthal and elevation residual gradients persist in the IGS ionospheric corrections. These gradients are represented graphically in Figure 2 for one representative station from the northern hemisphere (station code: EUSK, latitude: $50^\circ 40' 26.87''$, longitude: $6^\circ 45' 48.72''$) and one representative station from the southern hemisphere (station code: VACS, latitude: $-20^\circ 17' 48.47''$, longitude: $57^\circ 29' 13.79''$). The post-fit residuals are binned in azimuth and elevation and the average value in each bin is denoted by the color of the representing disc. The size of the disc denotes the number of samples of post-fit residuals available in each bin. Due to the inclination angle of the GPS satellite orbits, the angular distribution of satellites at any given latitude is non-uniform.

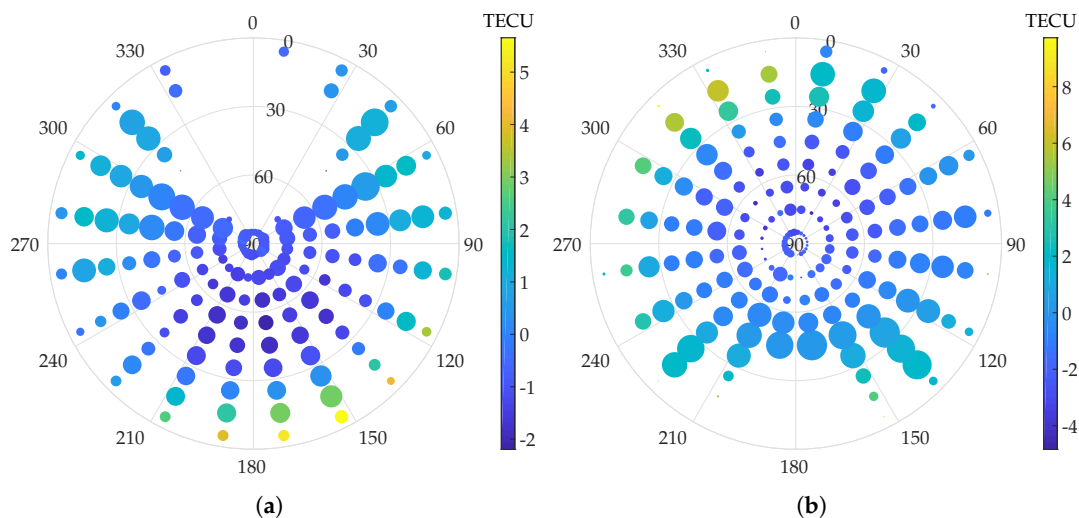


Figure 2. Azimuth and elevation dependence of post-fit IGS global ionospheric map (GIM) residuals. (a) A representative station from the northern hemisphere. (b) A representative station from the southern hemisphere. The average residual error (in TECU) is denoted by the color of the disc. The size of the disc indicates the number of samples of post-fit residuals available in each bin.

From Figure 2, it is clear that the elevation gradients in the ionospheric residuals are pronounced. A subtle azimuthal gradient also exists, mainly along the north-south direction. Such spatial non-uniformity, coupled with the non-uniform satellite angular distribution, may be the reason for the observed persistent position biases. While the elevation gradients are consistent for stations at all locations, the azimuthal gradients appear to invert along the north-south direction between the northern and southern hemisphere. This is likely the reason for the opposite direction of the average horizontal position bias in the northern and southern hemispheres.

Such persistent position-domain biases due to inaccurate ionospheric modeling have not been previously reported in the literature, and are a rather remarkable result. While some single-frequency PPP (SF-PPP) techniques eliminate the ionospheric delays based on the GRAPHIC combination [42], many other techniques that rely solely on ionospheric corrections from GIMs have been shown to achieve 30 cm 95% accuracy in the east-north plane after convergence, with sub-10-cm bias [43], seemingly contradicting the results here. The key difference is that the SF-PPP methods involve estimation of a float carrier ambiguity term for each satellite arc. A portion of the systematic biases in the GIM estimates is likely absorbed in these states of the estimator, thereby attenuating the position biases in the east-north plane. For instance, the SF-PPP technique in [43] is based on the phase-adjusted pseudorange algorithm [44], wherein the ambiguity term for each satellite, physically an unknown constant, is in fact iteratively estimated with small but non-zero process noise. In such an estimator, the ambiguity term can absorb slowly time-varying systematic biases. In other SF-PPP techniques, the ionospheric correction term is explicitly included as a state to be estimated, and the estimates from GIM are applied as pseudo-observations [45,46]. Once again, decimeter-level biases in the GIM estimates of the ionospheric delay may not necessarily appear in the final reported position accuracy of the SF-PPP method. Of course, such absorption of biases in augmented states is not undesirable. However, for the case of urban vehicular positioning, convergence of SF-PPP is a concern due to carrier phase cycle slipping, as discussed earlier. In an enhanced code-phase-based receiver, the high variance of the code noise leads to poor observability of the decimeter-level horizontal position bias due to ionospheric modeling errors. Thus, ionospheric biases are not often estimated in a code-phase-based GNSS estimator.

Another factor of note is that 2014 was a maximum in the 11-year solar activity cycle, and thus the IGS GIM accuracy may have been worse than usual over this period of time.

In conclusion, persistent decimeter-level biases in the east-north plane and meter-level biases in the vertical direction can arise when ionospheric delay corrections are sourced from the IGS GIM, or similar, even under ideal open-sky conditions. More advanced models of the ionosphere with more accurate slant TEC measurements may achieve better results. Elimination of the ionospheric delay based on the ionosphere-free combination is another option, but tends to worsen multipath-induced position errors. If corrections from some ionosphere model lead to unbiased position errors, then for globally-referencing digital maps by averaging GNSS measurements over many sessions it is advisable to avoid the combination of multi-frequency signals.

3.3.4. Tropospheric Modeling Errors

In the troposphere, or more generally the neutral atmosphere, the index of refraction departs from unity much less than in ionosphere at GNSS frequencies, causing a delay of ~ 2.4 m at zenith. The index of refraction in the troposphere is non-dispersive, and thus cannot be estimated using multiple-frequency signals. The tropospheric delay is obtained from models of the climatological parameters (temperature, pressure, and water vapor pressure) along the propagation path.

State-of-the-art tropospheric models [16] fit a small number of location- and day-of-year-dependent coefficients to climatological data from numerical weather models (NWMs) to model the zenith

tropospheric delay. The zenith delay is mapped to any elevation angle using mapping functions [47]. Similar to the ionospheric models, the tropospheric mapping functions may introduce a differential azimuth- and elevation-dependent error. For empirically-derived mapping functions such as VMF1 [47] and GMF [48], the mean error at lowest elevation of 5° has been shown to be under 50 mm (this value is typically reported as 10 mm station height error, which is approximately one-fifth of the delay error at lowest elevation [47]). As a result, this paper assumes that time-averaged tropospheric model errors would introduce sub-5-cm errors in the position domain, and would thus not impede asymptotically accurate collaborative mapping in both horizontal and vertical components at the several-decimeters level.

3.3.5. Multipath Error

In ideal circumstances, each signal received from an overhead satellite arrives only along the least-time path. In practice, however, this so-called line-of-sight (LOS) component is accompanied by other components due to signal diffraction and single- or multiple-signal reflections off surrounding surfaces and obstacles (e.g., the glass facade of a nearby building, poles, trees, etc.). The complex baseband representation of the N signal components received from a particular satellite at a particular frequency and code is

$$r(t) = \sum_{i=0}^{N-1} A_i(t)C[t - \tau_i(t)] \exp[j\theta_i(t)],$$

where A_i is the amplitude of the i th component, $C(t)$ is the GNSS code modulation, $\tau_i(t)$ is the delay of the i th signal component relative to an unobstructed LOS signal, and $\theta_i(t)$ is the beat carrier phase of the i th component. The combination of multiple components distorts the received signal and causes errors in the pseudorange and phase measurements.

Unlike the study of ionospheric modeling errors, for application in urban mapping, multipath errors cannot be characterized with data from survey stations with a clear view of the sky. This section considers a simulation approach for scalable analysis of multipath tracking errors in an urban environment. The objective of this study was to inspect the presence of persistent biases caused by multipath due to the surrounding structure in the navigation solution averaged over multiple sessions

Scenario Setup

The present simulation study was based on the open-access Land Mobile Satellite Channel Model (LMSCM) [49], itself based on extensive experimentation with a wideband airborne transmitter at GNSS frequencies in urban and suburban environments. First, an urban corridor was simulated stochastically following the procedure described in [50]. The corridor was composed of buildings, trees, and poles. Some of the important parameters for the generation of the scene are summarized in Table 2, and a part of the scene realization is shown in Figure 3. Multi-GNSS satellite trajectories were generated at randomly-selected times based on GPS and Galileo satellite almanac data. An average of 25 satellites were available above an elevation mask of 5° , consistent with modern multi-GNSS receivers. The satellites were assumed to be stationary over the simulation period of 60 s. Navigation solution errors were computed over 1000 60-s sessions.

Table 2. Some urban scenario parameters.

Distance from road center to buildings	24 m	Distance from road center to vehicle	5 m
Mean distance between road center and trees	20 m	Antenna height	2 m
Mean building width	30 m	Building width standard deviation	25 m
Mean building height	40 m	Building height standard deviation	20 m
Probability of gap between buildings	0.5	Mean gap width	30 m
Mean distance between trees	60 m	Mean distance between poles	25 m

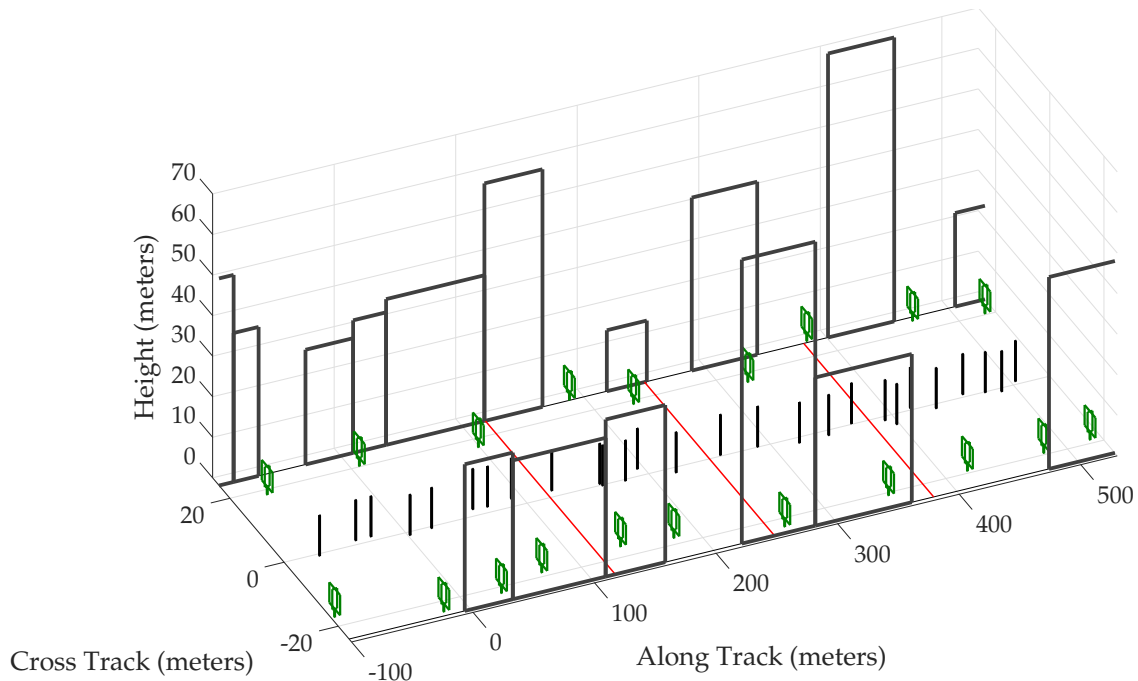


Figure 3. Initial segment of the simulated urban corridor. Red lines across the road denote the positions where the vehicle is momentarily stopped.

The vehicle trajectory was kept consistent across all 1000 driving sessions to avoid decorrelation of multipath error due to variable receiver motion. The trajectory was parametrized by its speed and heading, as described in [50]. The vehicle started at the zero coordinate on the along-track axis, and traveled in the positive direction, which was assumed to be aligned with the local north. The simulated trajectory was 60 s long and simulated a vehicle in stop-and-go traffic executing one 90° right turn, as shown in Figure 4. The vehicle traveled roughly 430 m and faced eastwards at the end of the trajectory. The three low-speed intervals, marked with red line segments in Figure 3, are expected to present severe multipath effects since multipath errors decorrelate slowly, and thus tend to reinforce one another within the navigation filter, when the vehicle moves slowly.

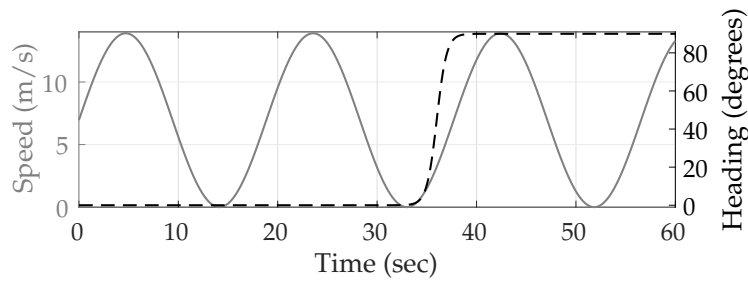


Figure 4. Vehicle speed (solid line) and heading (dashed line) simulating stop-and-go motion with a 90° right turn.

Multipath Simulation

The LMSCM generates power, delay, and carrier phase for N LOS and echo signals. The interaction of the LOS with the simulated obstacles is governed by deterministic models for attenuation, diffraction, and delay. The LOS components of the combined signal, denoted $r_{\text{LOS}}(t)$, may be composed of multiple components due to signal diffraction. These components are modeled as

$$r_{\text{LOS}}(t) = \sum_{i=0}^{N_{\text{LOS}}-1} A_i(t) C[t - \tau_i(t)] \exp[j\theta_i(t)].$$

In the special case of an unobstructed LOS signal, $N_{\text{LOS}} = 1$, $A_0(t) = 1$, $\tau_0(t) = 0$, and

$$\theta_0(t) = \frac{\|\mathbf{r}_R(t) - \mathbf{r}_S(t)\| \cdot 2\pi}{\lambda} + \gamma_0,$$

where λ denotes the wavelength and γ_0 is a constant due to phase initialization in the satellite and receiver [51].

The LMSCM generates the $N - N_{\text{LOS}}$ NLOS echoes stochastically based on satellite azimuth and elevation, receiver dynamics, and general characteristics of the scene (e.g., an *urban car* scenario). This stochastic procedure might not be representative of multipath over multiple sessions through the same urban corridor, where certain echoes might persist over different sessions. To address this limitation, the LMSCM was augmented by the present authors to generate one- and two-bounce deterministic reflective NLOS echoes off the simulated buildings, and a one-bounce NLOS echo off the ground surface. These three additional reflective NLOS echoes, denoted $r_{\text{DET}}(t)$, were added to $r(t)$ and are modeled as

$$r_{\text{DET}}(t) = \sum_{i=N}^{N+2} b_i(t) A_i(t) C[t - \tau_i(t)] \exp[j(\theta_i(t) + \theta'_i(t))],$$

where $b_i(t) \in \{0, 1\}$ denotes whether the surrounding geometry supports the reflective echo. Since these reflections are expected to be the stronger than other diffracted and multiple-bounce NLOS echoes, the amplitudes $A_i(t)$, $i \in \{N, N+2\}$ for reflective echoes were drawn from the distribution of the strongest echo generated stochastically by the LMSCM at each epoch. By experiment, this distribution was found to be log-normal with $20 \log_{10}(A_i) \sim \mathcal{N}(-22, 5)$, $i \in \{N, N+2\}$. The delays for the reflective echoes are given as

$$\tau_i(t) = \frac{\|\mathbf{r}'_R(t) - \mathbf{r}_S(t)\| - \|\mathbf{r}_R(t) - \mathbf{r}_S(t)\|}{c}, \quad i \in \{N, N+2\},$$

where $\mathbf{r}'_R(t)$ is the position of the imaginary *image* antenna [52] about the reflecting plane (building or ground). Similarly, the carrier-phase of the reflective echoes is computed geometrically as

$$\theta_i(t) = \frac{\|\mathbf{r}'_R(t) - \mathbf{r}_S(t)\| \cdot 2\pi}{\lambda} + \gamma_0, \quad i \in \{N, N+2\}.$$

A random carrier-phase offset $\theta'_i(t) \in [0, 2\pi)$ was added at the reflection point every time a new reflective echo was spawned to simulate the material-specific phase offset introduced by the reflection process.

Receiver

A receiver simulator was developed to account for the mediating effects that a receiver's tracking loops and navigation filter have on multipath-induced position errors in a receiver's reported position solution. The simulated receiver tracks the combination of all N_{LOS} line-of-sight signals and $N+2 - N_{\text{LOS}}$ multipath echoes for a given signal. If $R(\tau)$ denotes the correlation function of the GNSS signal's spreading code, then the multipath delay error in the tracked code phase, relative to unobstructed LOS, is given as the solution to [52]

$$0 = S_{\text{coh}}(\tau) = \sum_{i=0}^{N+2} A_i \cos(\theta_i - \theta_c) \times \left[R\left(\tau - \tau_i + \frac{d}{2}\right) - R\left(\tau - \tau_i - \frac{d}{2}\right) \right],$$

where θ_c is the tracked carrier-phase of the combined signal:

$$\theta_c = \text{atan2} \left(\sum_{i=0}^{N+2} A_i R(\tau_c - \tau_i) \sin(\theta_i), \sum_{i=0}^{N+2} A_i R(\tau_c - \tau_i) \cos(\theta_i) \right).$$

The parameter d is the early-to-late correlator spacing in the receiver. It is well-known that a wide-bandwidth receiver with narrow correlator spacing mitigates the effect of multipath [52]. To this end, the receiver considered in this simulation implements $d = 0.1$. It must be mentioned that $R(\tau)$ was implemented as the correlation function for GPS L1 C/A identically for all the simulated signals. Modernized GNSS signals have better multipath mitigation characteristics [11], but this behavior was not included in the simulation.

Another important observation is that when the LOS signal is strong as compared to the echo signals, the time derivative of the tracked carrier-phase is equal to the Doppler frequency of the LOS signal, which changes smoothly in accordance with the motion between the satellite and the receiver. However, when the LOS signal is comparable to or weaker than other rapidly-decorrelating echoes, the combined carrier-phase is uniformly random. In a GNSS receiver, the phase lock loop's phase-lock indicator indicates whether a sufficiently strong LOS signal is available, enabling carrier lock [6]. The simulator's phase-lock indicator is asserted only if (1) the tracked Doppler frequency does not deviate significantly from a second-order polynomial, and (2) the strongest received echo is attenuated by more than 25 dB with respect to an unattenuated signal.

Navigation Filter

At each epoch, n_z multipath-free, ionosphere-free, and troposphere-free simulated pseudorange measurements were combined with corresponding simulated multipath tracking delay errors and fed to a navigation filter that estimates the receiver state. The navigation filter implemented in this paper is an extended Kalman filter (EKF) with a nearly constant velocity motion model following [53]. The standard details of the EKF are omitted for brevity.

The effect of multipath tracking on the navigation solution is strongly dependent on the receiver's multipath rejection scheme. Two schemes are explored here. The first is a hypothetical ideal multipath rejection scheme that excludes all signals for which the LOS signal has a smaller-than-10-dB advantage over its multipath echoes. The second scheme implements a normalized innovation squared (NIS) test to reject multipath signals based on measurement innovations [53]. At the $(k + 1)$ th measurement update step, the difference between the predicted and observed measurement vector, called the innovation and denoted $\nu(k + 1)$, is squared and normalized by its covariance, which is the sum of the measurement covariance matrix, $R(k + 1)$, and the propagated state covariance transformed through the measurement sensitivity matrix, $H(k + 1)P(k + 1|k)H(k + 1)^T$. In the absence of multipath tracking errors, the resulting NIS statistic is chi-squared distributed with n_z degrees of freedom. If the NIS statistic exceeds a chosen threshold, then the signal with the largest normalized innovation is dropped. This continues until the NIS statistic falls below the threshold or the number of remaining signals drops to a preset minimum number of required signals.

Simulation Results

Figure 5 shows the mean position error in the east, north, and up directions over 1000 sessions for the two multipath rejection schemes mentioned previously. From Figure 5a, it can be seen that sub-20 cm average error is achievable with hypothetical ideal multipath exclusion. A closer look at Figures 4 and 5a reveals that the decimeter-level sinusoidal position error trend, initially in the north direction and later in the east direction, in fact corresponds with the along-track accelerations of the vehicle that were not adequately tracked by the nearly-constant-velocity-model-based navigation filter.

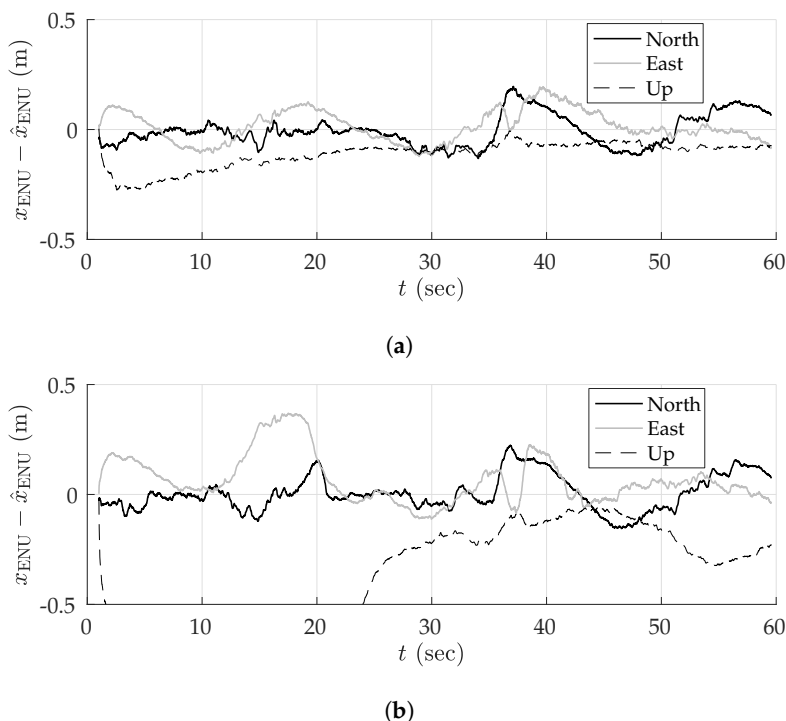


Figure 5. Mean position error in the east-north-up (ENU) frame over 1000 sessions due to multipath. (a) Ideal multipath exclusion. (b) Normalized innovation statistic (NIS)-based multipath exclusion. The black, gray, and dashed-black lines represent the error in the east, north, and up directions, respectively. The up error in the bottom panel reached a maximum magnitude of 1.75 m.

Figure 5b shows that the NIS test based exclusion of signals was able to approach the performance of ideal exclusion in the horizontal plane, save for the first stationary period where the vehicle was moving at low speed between buildings on both sides. The average vertical position error was much worse, growing as large as 1.75 m in magnitude.

To determine whether the average errors shown in Figure 5 are in fact persistent biases, a study of the standard deviation of position errors was conducted. The standard deviation of the average errors in east, north, and up directions was computed for disjoint averaging ensembles of size 1, 2, 4, 8, 16, 32, 50, and 100 sessions taken from the total of 1000 simulated sessions. For instance, 125 disjoint ensembles of eight sessions were selected, and the position errors were averaged over the eight sessions in each set. The standard deviation of the eight-session-averaged errors was then computed across the 125 ensembles. In the case of an averaging ensemble with only a single session (i.e., no averaging), the computed standard deviation is simply the measured standard deviation of the position error across all 1000 simulated runs. In the case of averaging over 100 sessions, the standard deviation is computed based on 10 disjoint averaging ensembles of 100 sessions each.

Note that because the simulation study was based on the same 1000 simulations for all averaging ensembles, the east, north, and up means taken across all averaging ensembles are equivalent to those shown in Figure 5. The more interesting trend is the decreasing standard deviation with increasing size of the averaging ensemble, as shown in Figure 6 for the case of NIS-based multipath rejection and for the east and north error components. As expected, the standard deviation of errors was higher at locations where the vehicle moved at low speed and multipath decorrelated slowly. Additionally, the standard deviation was larger at the beginning of the trajectory where the street was lined with tall buildings on both sides.

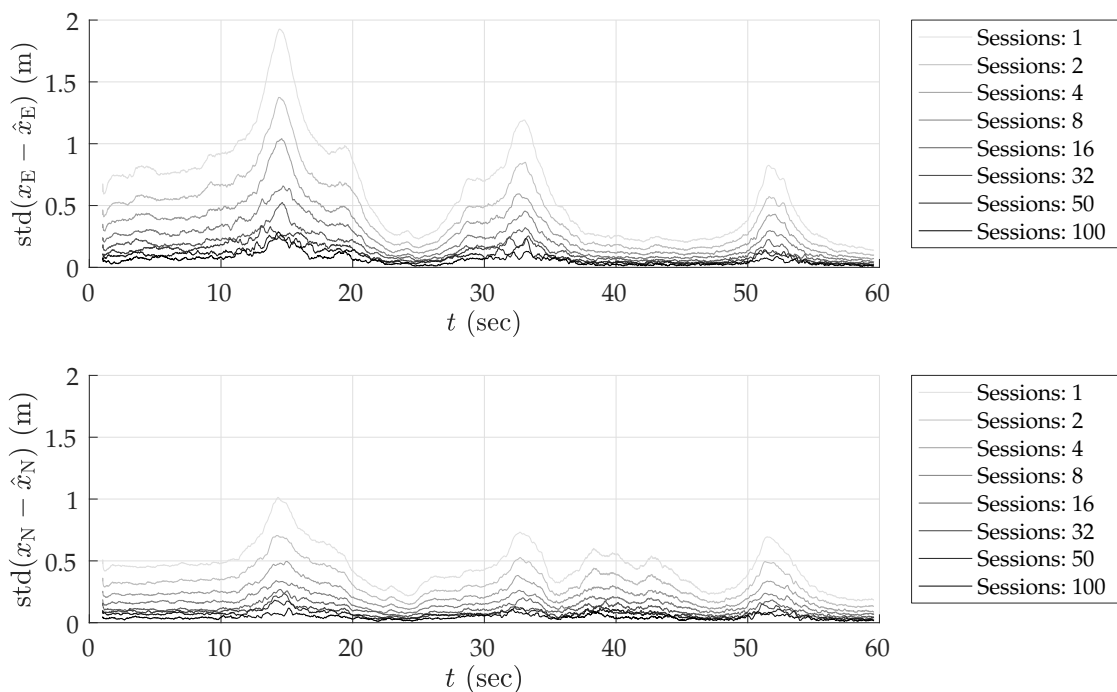


Figure 6. Standard deviation of average position error in east and north directions for NIS-based multipath exclusion as a function of the number of sessions over which the errors are averaged. **Top panel:** standard deviation in the east direction. **Bottom panel:** standard deviation in the north direction.

The standard deviation of the average east and north position error over 100 sessions was bounded below 15–20 cm. Thus, it is highly likely that the ~40-cm error in the north direction between 15–20 s in Figure 5b is in fact a persistent non-zero bias.

Table 3 summarizes the results of the multipath simulation study. It shows the 95-percentile horizontal error magnitude for increasing averaging ensemble sizes and for both ideal and NIS-based multipath exclusion. The 0–60 s average case lists the 95-percentile error over the entire trajectory, whereas the 13–19 s average case lists the 95-percentile error in the worst-case segment of the trajectory in terms of horizontal position bias and standard deviation. This challenging segment is illustrative of persistent problem spots that will arise in urban areas, within which multipath-induced biases will be larger than average. As expected, the 95-percentile error in Table 3 shrank as the averaging ensemble size became larger. For the urban corridor and vehicle dynamics considered in this simulation, NIS-based exclusion achieved 35 cm 95-percentile horizontal error with averaging over 100 sessions. Even in the worst-case region of the trajectory, the 95-percentile horizontal error remained below 50 cm. As multipath exclusion approaches the ideal case, with aid from other sensors or a 3D model of the surroundings, for example, the 95-percentile horizontal error could be reduced to as low as 25 cm for the simulated corridor.

Table 3. 95-percentile horizontal errors.

Averaging Ensemble Size:		1	2	4	8	16	32	50	100
Ideal	0–60 s average (m)	1.5910	1.1262	0.7902	0.5488	0.4078	0.3090	0.2696	0.2147
	13–19 s average (m)	2.5925	1.7809	1.2136	0.8927	0.6416	0.4145	0.3544	0.2609
NIS	0–60 s average (m)	1.7851	1.2795	0.9245	0.6588	0.5169	0.4175	0.3920	0.3526
	13–19 s average (m)	3.1217	2.1953	1.5467	1.1720	0.8456	0.6470	0.5950	0.4702

From the Section 3.3.3's analysis of asymptotic ionospheric errors, and from this section's multipath simulation study, one can draw the following conclusion: so long as the asymptotic horizontal position errors of the ionosphere corrections are below 5 cm, as is true for the Fast-PPP model, and assuming statistical independence of ionospheric and multipath errors, it appears feasible to achieve 50-cm horizontal positioning accuracy at approximately 95% by averaging over 100 mapping sessions.

4. Globally-Referenced Electro-Optical SLAM (GEOSLAM)

This section describes a simultaneous localization and mapping (SLAM) pipeline capable of globally-referenced collaborative multi-session digital mapping. The pipeline combines visual measurements from a stereo visible-light camera system with position measurements from GNSS signals. The objective of this pipeline is to demonstrate the development of an accurate digital map based on multiple mapping sessions with standard GNSS position estimates. The following sections detail the visual SLAM pipeline, the integration of GNSS measurements, and the techniques for multi-session mapping developed in GEOSLAM.

4.1. Visual SLAM

The visual SLAM component of GEOSLAM is similar to existing high-performance SLAM pipelines developed in the robotics community [54–56]. Visual SLAM algorithms may be categorized as either sparse or dense. Sparse visual SLAM algorithms [54,55] create a map of distinctive features such as corners or edges in the scene, while dense SLAM algorithms [56] map the depth for each pixel in the captured frames. The point cloud generated by sparse SLAM algorithms is sufficient for the purpose of localization. Dense reconstruction is appealing to the human eye, but does not provide any tangible

benefit to localization, while consuming much more computational resources. As a result, GEOSLAM implements sparse feature-point-based SLAM.

In [57] it was shown that for the visual SLAM problem, structure-from-motion BA (batch non-linear optimization) outperforms filtering techniques such as the extended Kalman filter, yielding higher accuracy per unit of computing time. It was also noted that having a high number of features points per image frame provides better accuracy than having a large number of frames with fewer feature points per frame. Thus, in typical practice, only a select subset of frames among those captured is retained for processing; frames in this subset are called keyframes. Most recent state-of-the-art visual SLAM algorithms use a keyframe-based BA approach instead of sequential filtering. Likewise, GEOSLAM performs BA-based non-linear optimization to refine both structure and motion.

Figure 7 shows a block diagram representation of the system architecture proposed in this paper. The yellow-colored blocks in this figure are components of the GEOSLAM pipeline, detailed next.

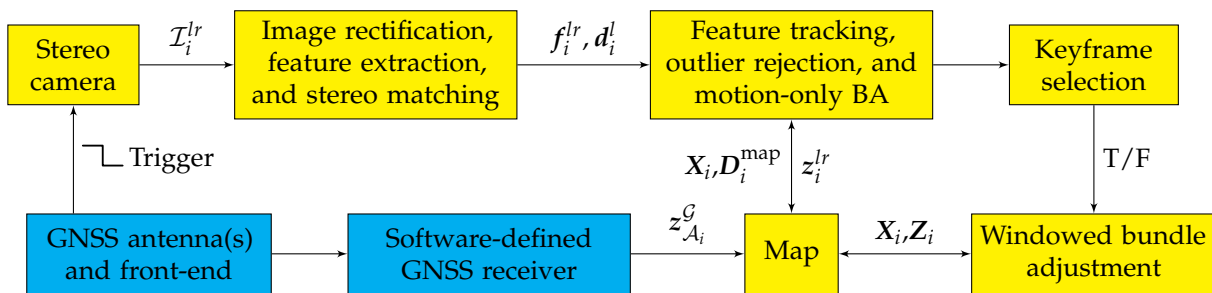


Figure 7. Globally-referenced electro-optical simultaneous localization and mapping (GEOSLAM) block diagram. BA: bundle adjustment.

By way of notation, let

$$z_{nm}^{lr} \triangleq (u_{nm}^l, v_{nm}^l, u_{nm}^r)$$

denote the image plane location of the stereo-matched feature matched to the m th map point, p_m , in the n th stereo keyframe, K_n . The horizontal and vertical coordinates are denoted u and v , respectively, while the superscripts l and r denote the left and right camera frames, respectively. Note that the feature location is specified by only three coordinates. The vertical feature coordinate in the right camera frame, v_{nm}^r , is omitted because for an undistorted and rectified camera model it must hold that $v_{nm}^l = v_{nm}^r$, making one of the coordinates redundant. If p_m is not matched to any feature in K_n , then let $z_{nm}^{lr} = \emptyset$. Furthermore, let

$$\mathcal{M}_n = \{m : z_{nm}^{lr} \neq \emptyset\}$$

denote the set of indices of all map points matched to some feature in K_n . In the visual SLAM literature, the covisibility window of keyframe K_i is defined as the set of keyframes that share at least T map points with K_i . Mathematically, the covisibility window of keyframe K_i is the set of keyframes with indices

$$\text{cov}(i) \triangleq \{n : |\mathcal{M}_n \cap \mathcal{M}_i| > T\},$$

where $|\mathcal{A}|$ denotes the cardinality of the set \mathcal{A} . The covisibility window determines the keyframes to be optimized in a windowed BA. The visibility of common points is regarded as a proxy for correlation between the structure-from-motion states. However, in a sensor fusion architecture, the states for other sensors (e.g., GNSS) may be spatially correlated beyond the covisibility window. Furthermore, other sensors may experience outages that extend beyond the covisibility window. In such a scenario, it would

be desirable to optimize over a batch of keyframes that span the availability gap. Accordingly, GEOSLAM extends the concept of covisibility to N levels as

$$\text{cov}(i, N) \triangleq \begin{cases} \{n : |\mathcal{M}_n \cap (\cup_{k \in \text{cov}(i, N-1)} \mathcal{M}_k)| > T\} & N > 1, \\ \{n : |\mathcal{M}_n \cap \mathcal{M}_i| > T\} & N = 1. \end{cases}$$

When processing K_i , GEOSLAM's objective is to estimate the map point 3D locations $\mathbf{x}_{p_m}^S \in \mathbb{R}^3$ and keyframe poses $(\mathbf{x}_{\mathcal{C}_n}^S, \boldsymbol{\theta}_{\mathcal{C}_n}^S) \in (\mathbb{R}^3, \mathbb{R}^3)$ in the N -level covisibility window for the i th keyframe, where \mathcal{S} stands for the local SLAM frame, \mathcal{C}_n is the left camera coordinate frame associated with K_n , and $\boldsymbol{\theta}_{\mathcal{C}_n}^S$ is the angle-axis representation of the keyframe orientation. The state vector to be estimated is represented as

$$\mathbf{X}_i \triangleq \left[\left\{ (\mathbf{x}_{\mathcal{C}_n}^S, \boldsymbol{\theta}_{\mathcal{C}_n}^S) : n \in \text{cov}(i, N) \right\}, \left\{ \mathbf{x}_{p_m}^S : m \in \cup_{k \in \text{cov}(i, N)} \mathcal{M}_k \right\} \right]^T, \quad (4)$$

where the two sets on the right-hand side are arranged as a concatenation of row vectors so that \mathbf{X}_i becomes a column vector.

When triggered by the GNSS front end, the camera setup captures a pair, denoted \mathcal{I}_i^{lr} , of concurrent images from the left and right cameras, where the subscript denotes that the current pair is a candidate to be the i th stereo keyframe K_i . The intrinsic and extrinsic parameters of the stereo camera setup are assumed to have been calibrated a priori. The stereo image pair is then undistorted and rectified according to the given calibration, and SIFT features are detected and computed separately for each image [58]. SIFT feature matching is performed between the left and right image with the additional constraint that matching features must have approximately the same vertical coordinate to within a few pixels. The set of stereo feature measurements for \mathcal{I}_i^{lr} , f_i^{lr} , and the set of feature descriptors as computed in the left image, d_i^l , are passed on to the tracking module.

The tracking module has access to the 3D map point positions within \mathbf{X}_i and to the set of SIFT descriptors, $\mathbf{D}_i^{\text{map}}$, corresponding to the map points expected to be seen in the candidate keyframe \mathcal{I}_i^{lr} . The tracker performs directed matching of the features between the stereo image and the map. First, a quick feature matching is performed using the Fast Approximate Nearest Neighbor Search Library (FLANN) [59]. With sufficient matches, an initial approximation of the current camera pose is obtained using the five-point algorithm wrapped in random sample consensus (RANSAC) iterations. With this approximate pose, an iteration of exhaustive nearest neighbor search is performed for each map point potentially in view of the camera, but only within a small window of its projected position on the image plane. Subsequently, RANSAC iterations are performed on the full set of feature matches to remove any remaining outliers, and a motion-only BA is performed wherein the current camera pose is optimized based on the feature matches to a fixed set of 3D map points.

After tracking the stereo image pair as described above, GEOSLAM decides whether or not the candidate keyframe \mathcal{I}_i^{lr} must be selected as a keyframe. This decision is made based on the number of map points that were matched to the image features, and the distance traveled by the platform since the last keyframe was chosen. New keyframes are not spawned if the platform is nearly stationary. If the platform is in motion, and the number of feature matches to the map drops below a threshold, then the candidate keyframe \mathcal{I}_i^{lr} is chosen as keyframe K_i , windowed BA is performed over N levels of covisibility, and the unmatched stereo features in K_i are spawned as new map points. Additionally, if \mathcal{I}_i^{lr} is selected to be K_i , then the set of measurements from the features matched to the map, denoted $\mathbf{z}_i^{lr} \triangleq \{z_{im}^{lr} : m \in \mathcal{M}_i\}$, along with their SIFT descriptors, are passed on to the map module for storage, future feature matching, and processing in the windowed BA routine.

In a visual-only SLAM system, the state vector \mathbf{X}_i is optimized with respect to the measurement vector \mathbf{Z}_i , defined as

$$\mathbf{Z}_i \triangleq \left[\left\{ \mathbf{z}_n^{lr} : n \in \text{cov}(i, N) \right\} \right]^T.$$

The windowed BA routine in GEOSLAM minimizes the 3D-to-2D reprojection error. The error term \mathbf{e}_{nm} for observation of map point p_m in the stereo keyframe K_n is given as

$$\mathbf{e}_{nm} = \mathbf{z}_{nm}^{lr} - \Pi(\mathbf{x}_{p_m}^S, \mathbf{x}_{C_n}^S, \boldsymbol{\theta}_{C_n}^S),$$

where Π is the projection function for an undistorted and rectified stereo camera

$$\Pi(\mathbf{x}_{p_m}^S, \mathbf{x}_{C_n}^S, \boldsymbol{\theta}_{C_n}^S) = \begin{bmatrix} f \frac{x_{nm}^S}{z_{nm}^S} + c_u \\ f \frac{y_{nm}^S}{z_{nm}^S} + c_v \\ f \frac{x_{nm}^S}{z_{nm}^S} + c_u - bf \end{bmatrix},$$

$$[x_{nm}, y_{nm}, z_{nm}]^T = R(\boldsymbol{\theta}_{C_n}^S)^T (\mathbf{x}_{p_m}^S - \mathbf{x}_{C_n}^S),$$

in which $R(\cdot)$ denotes the rotation matrix corresponding to the argument angle-axis vector, and f , (c_u, c_v) , and b are the focal length, the principal point, and the baseline distance between the left and right cameras of the rectified stereo camera model, respectively. The cost function to be minimized for visual SLAM is given as

$$C_i = \sum_{n \in \text{cov}(i, N)} \sum_{m \in \mathcal{M}_n} \rho(\mathbf{e}_{nm}^T \boldsymbol{\Omega}_{nm}^{-1} \mathbf{e}_{nm}),$$

where ρ may be the standard least squares cost function $\rho(\cdot) = (\cdot)$ or a more robust cost function such as the Huber or Tukey cost functions, and where $\boldsymbol{\Omega}_{nm} = \sigma_{nm}^2 \mathbf{I}_{3 \times 3}$ is the covariance of the feature measurements.

GEOSLAM performs BA minimization via Google's *ceres-solver*. The automatic differentiation feature of *ceres-solver* is used to compute the Jacobian for the measurement model.

An important feature of the GEOSLAM visual pipeline is the ability to merge maps from multiple mapping sessions. This is embodied in an algorithm similar to the loop closure technique from the visual SLAM literature [60]. Map merging is described in detail in Section 4.3.2.

4.2. GNSS Aiding

Conventional visual SLAM algorithms are known to drift from the true platform trajectory as a function of the distance traveled by the platform. Furthermore, the map of the structure is created in the arbitrary \mathcal{S} frame. Such a map cannot be intelligibly shared with another mapping agent having a different \mathcal{S} frame. Meanwhile, GNSS position estimates are obtained in the global \mathcal{G} frame and do not exhibit any distance-dependent drift. Accordingly, GEOSLAM ingests standard GNSS position estimates from a software-defined GNSS receiver, called GRID/pprx [6,61], in a tightly-coupled architecture to create a globally-referenced map, enable cooperative multi-session mapping, and constrain the drift of visual SLAM. Since the stereo camera setup is triggered by the same clock that drives digitization of the GNSS samples (see Figure 7), it is possible to produce GNSS measurements synchronized with the camera image epochs. This section details the various coordinate frames in GNSS-aided visual SLAM, the updated BA cost function, and an initialization routine required to enable GNSS aiding in SLAM.

4.2.1. Coordinate Frames

The GNSS-aided visual SLAM system has three coordinate frames of interest: the K_i camera frame C_i , the local SLAM frame \mathcal{S} , and the global frame \mathcal{G} . The SLAM frame \mathcal{S} adopts the position and orientation of the first keyframe prior to optimization as its origin and orientation. Thus, \mathcal{S} is fixed relative to \mathcal{G} , but each C_i changes relative to \mathcal{G} as the platform moves.

Note that the structure-from-motion states in Equation (4) are represented in the \mathcal{S} frame, whereas the K_n th keyframe's corresponding GNSS measurement, denoted $z_{\mathcal{A}_n}^{\mathcal{G}} \in \mathbb{R}^3$, is natively represented in the \mathcal{G} frame. The latter is transformed to the \mathcal{S} frame through an unknown but fixed rotation, $R_{\mathcal{G}}^{\mathcal{S}} \in SO(3)$, and translation, $t_{\mathcal{G}}^{\mathcal{S}} \in \mathbb{R}^3$. This transformation is estimated at initialization as explained in Section 4.2.2. After initialization, $z_{\mathcal{A}_n}^{\mathcal{G}}$ is rendered in \mathcal{S} as

$$z_{\mathcal{A}_n}^{\mathcal{S}} = R_{\mathcal{G}}^{\mathcal{S}} z_{\mathcal{A}_n}^{\mathcal{G}} + t_{\mathcal{G}}^{\mathcal{S}}.$$

GEOSLAM estimates the 6-DoF pose of the left camera, but the GNSS antenna phase center is not co-located with the camera center; rather, it is offset from the camera center by a fixed vector (same for all n) in C_n denoted, $t_{\mathcal{A}_n}^{\mathcal{C}_n} \in \mathbb{R}^3$. Thus, the error term associated with the GNSS position estimate for K_n is given as

$$e_{\mathcal{A}_n} = z_{\mathcal{A}_n}^{\mathcal{S}} - \left(x_{\mathcal{C}_n}^{\mathcal{S}} + R(\theta_{\mathcal{C}_n}^{\mathcal{S}}) t_{\mathcal{A}_n}^{\mathcal{C}_n} \right).$$

Under the assumption of temporally-uncorrelated GNSS errors, the updated BA cost function to be minimized is

$$C_i = \sum_{n \in \text{cov}(i, N)} \left[\sum_{m \in \mathcal{M}_n} \rho \left(e_{nm}^T \Omega_{nm}^{-1} e_{nm} \right) + e_{\mathcal{A}_n}^T \Gamma_n^{-1} e_{\mathcal{A}_n} \right],$$

where $\Gamma_n = R_{\mathcal{G}}^{\mathcal{S}} \Gamma'_n (R_{\mathcal{G}}^{\mathcal{S}})^T$ and Γ'_n is the covariance matrix of the GNSS position estimate associated with K_n , expressed in \mathcal{G} .

4.2.2. Initialization in GNSS-Aided SLAM

When initializing, GEOSLAM performs visual-only SLAM for the first N_i keyframes in the \mathcal{S} frame, and stores the GNSS position measurements of the antenna provided in the \mathcal{G} frame. Subsequently, GEOSLAM finds the least-squares Euclidean transformation to obtain the optimal rotation matrix $R_{\mathcal{G}}^{\mathcal{S}}$ and translation vector $t_{\mathcal{G}}^{\mathcal{S}}$ between the two coordinate systems from the set of vector observations. Note that a full similarity transformation is not required since the known stereo baseline renders the \mathcal{S} frame with correct scaling. The estimated Euclidean transformation minimizes the squared difference between the transformed GNSS measurements in \mathcal{S} and the visual SLAM predicted trajectory of the GNSS antenna, also in \mathcal{S} . The specific method used to estimate the transformation is based on SVD decomposition as discussed in [62].

$$\left(R_{\mathcal{G}}^{\mathcal{S}}, t_{\mathcal{G}}^{\mathcal{S}} \right) = \arg \min_{R \in SO(3), t \in \mathbb{R}^3} \sum_{n=1}^{N_i} \left\| \left(R z_{\mathcal{A}_n}^{\mathcal{G}} + t \right) - \left(x_{\mathcal{C}_n}^{\mathcal{S}} + R(\theta_{\mathcal{C}_n}^{\mathcal{S}}) t_{\mathcal{A}_n}^{\mathcal{C}_n} \right) \right\|^2 \quad (5)$$

It must be noted that this transformation need only be approximately correct such that the GNSS estimates, used as measurements, will not diverge with respect to the visually-derived trajectory. Because the jointly estimated trajectory in \mathcal{S} gets transformed back to \mathcal{G} using the same approximate transformation, any errors in the transformation are cancelled.

4.3. Multi-Session Mapping

Refinement of the visual feature map over multiple sessions with time-separated GNSS measurements is central to the idea of approaching the accuracy limit of mapping with standard GNSS. Consider a vehicle revisiting an area mapped previously in one or more sessions. When GEOSLAM matches greater than T features in the current keyframe to the features already present in the prior map, the keyframes from the previous sessions in that section of the map are included in the covisibility window of the current keyframe. After such a merge is detected and verified, a BA may be performed on the covisible keyframes from multiple sessions to average time-separated standard GNSS errors. It is important to note that multi-session mapping can only be realized when sufficient feature matches are found between multiple sessions. This is not a straightforward task, as evidenced by recent efforts on lifelong feature mapping efforts [63]. This issue is further discussed in Section 5. In the current section, multi-session map database management and map merging are discussed.

4.3.1. Map Database

Storage and reuse of maps is a pre-requisite for multi-session mapping. For a given session, the SLAM map is created in the \mathcal{S} frame. However, such a map is not readily usable in successive mapping sessions since the \mathcal{S} frame is distinct for each session. Fortunately, the integration of visual SLAM with GNSS enables transformation of the SLAM map in to the \mathcal{G} frame.

At the end of the p th mapping session in the local frame \mathcal{S}_p , GEOSLAM stores the data to a map database after applying the $(R_{\mathcal{G}}^{\mathcal{S}_p}, \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p})$ transformation, as estimated during initialization for the p th session, to all map point positions, all keyframe poses, and all GNSS measurements associated with each keyframe:

$$\begin{aligned}\mathbf{x}_{\mathcal{C}_n}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{x}_{\mathcal{C}_n}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}); & R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{G}}) &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{S}_p}), \\ \mathbf{x}_{p_m}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{x}_{p_m}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}), \\ \mathbf{z}_{\mathcal{C}_n}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{z}_{\mathcal{C}_n}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}).\end{aligned}$$

At the beginning of the $(p + 1)$ th session, GEOSLAM again estimates the $(R_{\mathcal{G}}^{\mathcal{S}_{p+1}}, \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}})$ transformation during initialization. The map database from previous session(s) is then loaded after applying the transformation for the $p + 1$ session, such that the prior map points, keyframes, and measurements are rendered in the \mathcal{S}_{p+1} frame:

$$\begin{aligned}\mathbf{x}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{x}_{\mathcal{C}_n}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}; & R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}}) &= (R_{\mathcal{G}}^{\mathcal{S}_{p+1}}) R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{G}}), \\ \mathbf{x}_{p_m}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{x}_{p_m}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}, \\ \mathbf{z}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{z}_{\mathcal{C}_n}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}.\end{aligned}$$

After loading the prior map, the standard GEOSLAM pipeline is executed for each stereo image pair in the $(p + 1)$ th session. In addition, GEOSLAM attempts to detect if the vehicle is currently passing through a previously-mapped region. If so, a map merge is declared and the current and prior keyframes are jointly optimized, as detailed in Section 4.3.2. Finally, at the end of the mapping session, the combined map is stored back in the database as described before.

4.3.2. Map Merging

As mentioned before, the matching of feature points across multiple sessions is central to the idea of averaging standard GNSS errors. Once sufficiently many features are matched between the current stereo keyframes and prior map points, GEOSLAM declares a map merging event. This is akin to the well-known problem of detecting loop closure in the visual SLAM literature [60]. This section details GEOSLAM's map merging and loop closing routine. Hereafter, the terms map merging and loop closure are used interchangeably since GEOSLAM treats them identically.

First, note that when detecting a map merge event, feature matching must be attempted against map points that have not been matched in the most recent keyframes. Thus, after processing the i th keyframe, a possible merge is checked for against the set of map points $\{m : m \notin \cup_{n \in \text{cov}(i, N)} \mathcal{M}_n\}$. If this bag-of-words-style feature matching succeeds, then RANSAC iterations are performed to determine whether the matches are geometrically consistent, as well as to robustly estimate the camera pose $(\check{x}_{C_i}^S, \check{\theta}_{C_i}^S)$ implied by the merge event. If enough inliers are found, the map merge routine is executed.

The map merging process is depicted visually in Figures 8–10. A typical merge situation is shown in Figure 8, where the platform pose at the i th keyframe is inconsistent with the prior map at the merge location. To avoid such discontinuity in the ensuing joint BA, as an initial guess GEOSLAM enforces that the i th keyframe pose be consistent with the pose implied by the visual merge matches $(\check{x}_{C_i}^S, \check{\theta}_{C_i}^S)$, and that the keyframes and map points from the prior session(s) be unchanged. To this end, a pose-graph optimization [64] is performed over a large N_m -level covisibility window for K_i , where the relative translations and rotations between covisible keyframes, as estimated in the current session, are provided as delta-pose measurements, while the 6-DoF poses for the terminal nodes in the covisibility window, as well as for K_i , are held constant. In particular, let \mathcal{K}_0 denote the set of terminal keyframes in the covisibility graph $\text{cov}(i, N_m)$. Furthermore, define the delta-pose pseudo-measurements

$$\begin{aligned} \delta x_{nk}^S &\triangleq \hat{x}_{C_n}^S - \hat{x}_{C_k}^S, \\ \delta \theta_{nk}^S &\triangleq \theta \left(R \left(\hat{\theta}_{C_n}^S \right) R \left(\hat{\theta}_{C_k}^S \right)^T \right), \end{aligned}$$

where the superscript (\cdot) denotes GEOSLAM's estimate of the state before the merge event, and $\theta(\cdot)$ denotes the angle-axis representation of the input rotation matrix. The pose-graph optimization minimizes the following cost function with respect to $(x_{C_n}^S, \theta_{C_n}^S) \forall n \in \text{cov}(i, N_m)$:

$$C = \sum_{n \in \text{cov}(i, N_m)} \sum_{k \in \text{cov}(n, 1)} \left[\left\| (x_{C_n}^S - x_{C_k}^S) - \delta x_{nk}^S \right\|_{P_{\delta x}^{-1}}^2 + \left\| \theta \left(R \left(\theta_{C_n}^S \right) R \left(\theta_{C_k}^S \right)^T \right) - \delta \theta_{nk}^S \right\|_{P_{\delta \theta}^{-1}}^2 \right],$$

where $\|q\|_P^2 = q^T P q$, with the following constraints:

$$\begin{aligned} (x_{C_i}^S, \theta_{C_i}^S) &= (\check{x}_{C_i}^S, \check{\theta}_{C_i}^S), \\ (x_{C_n}^S, \theta_{C_n}^S) &= (\hat{x}_{C_n}^S, \hat{\theta}_{C_n}^S) \forall n \in \mathcal{K}_0. \end{aligned}$$

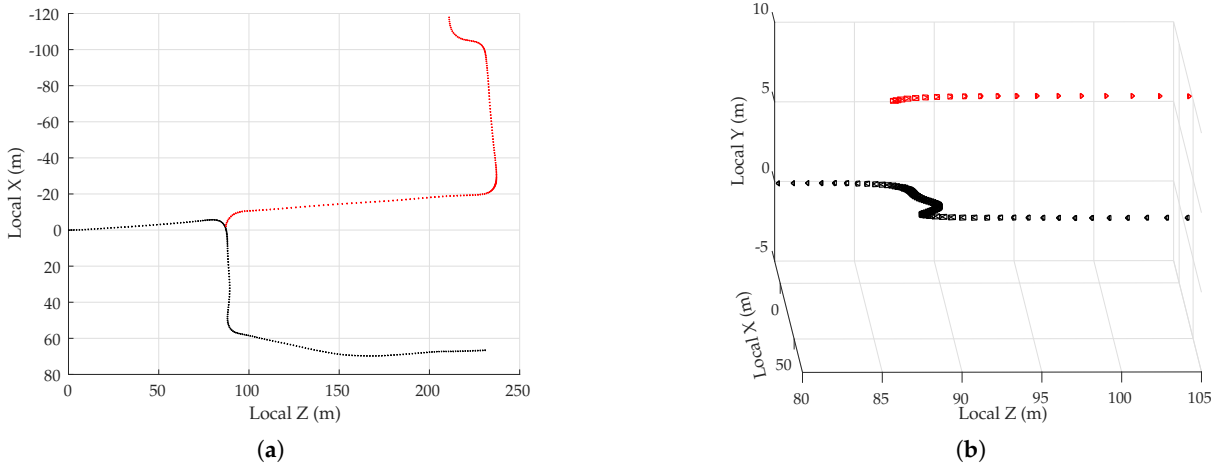


Figure 8. GEOSLAM trajectories at the instant when a merge has been detected and verified. The cameras colored black are keyframes from a prior map, and those colored red are from the current session. (a) Top view of the trajectories. (b) View from 5° elevation showing a discontinuity in the vertical component.

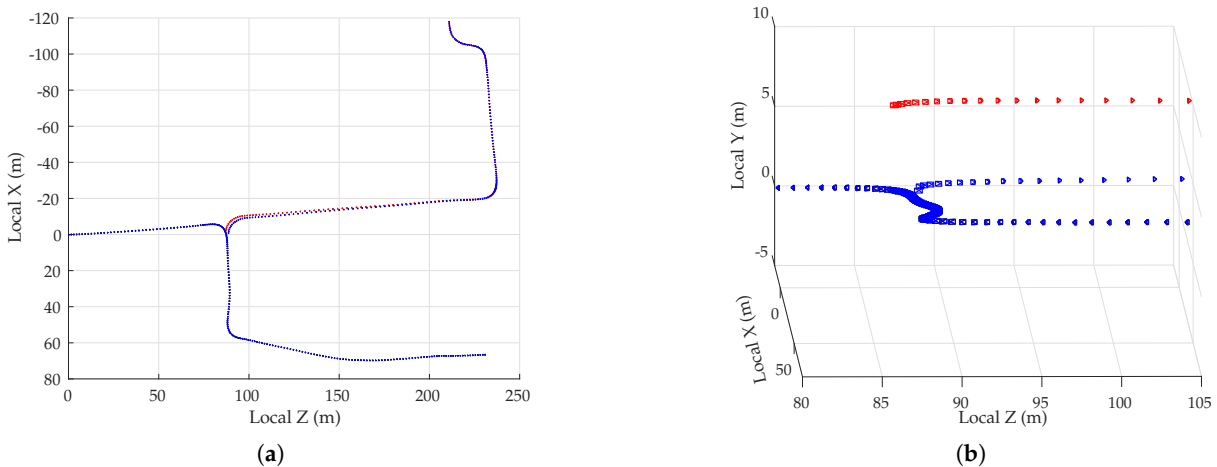


Figure 9. GEOSLAM trajectories post-pose-graph optimization (in blue), overlaid on the corresponding (black and red) trajectories from Figure 8. All keyframes are colored blue at this stage since prior and current keyframes are now connected. (a) Top view of the trajectories. Note that the discontinuity at the merge location is smoothly distributed across N_m levels of covisibility in the current session, and that the keyframe poses from the prior map are unchanged at this stage. (b) View from 5° elevation. Keyframes from the current trajectory have been adjusted to remove the discontinuity, blue and black keyframes exactly overlap. Not shown: the corresponding map points in the current session are also adjusted to match the updated keyframe poses.

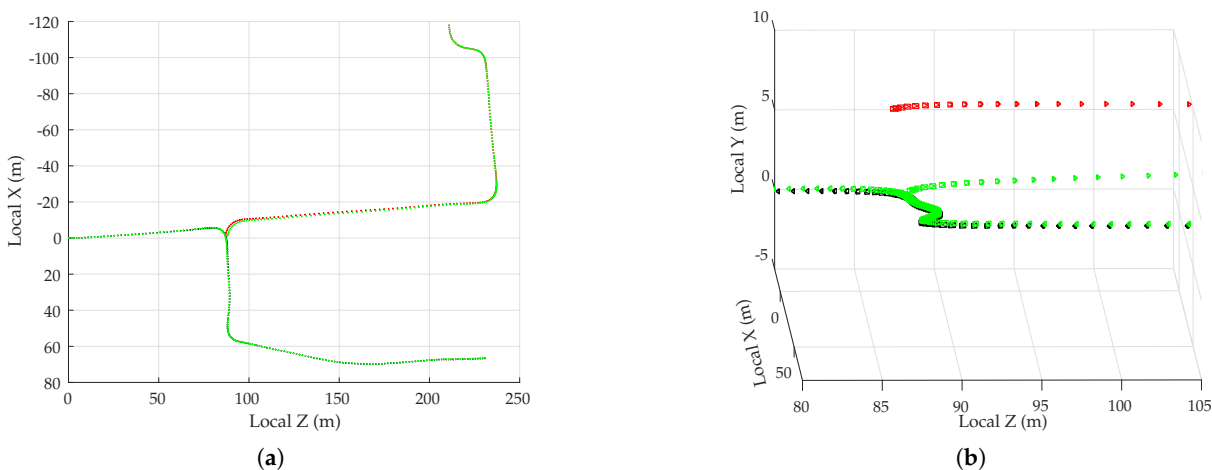


Figure 10. GEOSLAM trajectories after joint BA of current and prior keyframes (in green), overlaid on the corresponding (black and red) trajectories from Figure 8. (a) Top view of the trajectories. Note that both the current and prior keyframes (and map points, not shown) have been adjusted to optimally minimize the BA cost function over N_m levels of covisibility. (b) View from 5° elevation.

As a result of this pose-graph optimization, any discontinuity at the merge location between the prior and current keyframes is smoothed out, as shown in Figure 9. Subsequently, the map points as seen in the current keyframes are also adjusted in accordance to the pose-graph optimization. Finally, the duplicated map points near the merge location are fused together. As a result of shared feature matches between the current and prior keyframes, the updated covisibility window for K_i includes keyframes from prior session(s).

The merged map is then optimized in a windowed BA, this time with feature point coordinates and GNSS positions as measurements. Note that this is a joint windowed BA with both current and prior keyframes and map points. As a result, both the current and prior states are appropriately adjusted based on the number and covariance of the feature point and GNSS measurements. The result of the map merging routine is shown in Figure 10.

5. Empirical Results

To validate the results obtained in the above analyses, GNSS and visual data were collected in a moderate urban area north of the University of Texas at Austin campus in Austin, TX. This section presents the data collection setup, error statistics of various flavors of code-phase GNSS positioning, and results from GEOSLAM’s multi-session GNSS-aided-visual mapping.

5.1. Rover and Reference Platforms

The rover GNSS receiver is one among several sensors housed in an integrated perception platform called the University of Texas Sensorium [6]. Designed for connected and automated vehicle research, the Sensorium is a self-contained sensor housing that can be mounted atop any standard passenger vehicle. Two Antcom G8Ant-3A4TNB1 triple-frequency patch antennas are flush-mounted in the cross-track direction on the Sensorium’s upper plate, separated by just over one meter. The antennas’ signals are routed to a unified radio frequency (RF) front end whose output intermediate frequency (IF) samples are processed in real-time (to within less than 10 ms latency) by the Sensorium’s onboard computer.

The samples are also stored to disk for post-processing. The experimental setup also includes a surveyed GNSS reference station that aids in the generation of the ground truth trajectory.

The GNSS data were processed by a software-defined GNSS receiver tracking signals from GPS L1 C/A, GPS L2CLM, Galileo E1, and SBAS. Data from both the primary (passenger's side) and secondary (driver's side) antennas were used to reconstruct a sub-dm-accurate CDGNSS-based ground truth trajectory, as described in [6]. Enhanced code-phase positioning was performed on the data from the primary antenna, incorporating precise orbit and clock products from IGS, ionospheric corrections from WAAS satellites, and the Saastamoinen model for tropospheric corrections, in addition to NIS-based exclusion of multipath signals. Double-differenced pseudorange-based positioning was also performed with the data from the primary antenna, as discussed later in this section. The code-phase-based position estimates were (i) compared against the ground truth from the primary antenna to study the code-phase positioning error statistics, and (ii) fed to GEOSLAM for vision-GNSS sensor fusion. The primary antenna feed was also input to a ublox M8T receiver for comparison against the enhanced code-phase software receiver.

The Sensorium features a front-facing stereo camera rig composed of two Basler acA2040-35gm cameras that capture synchronous stereo image pairs when triggered by a signal tied to the GNSS front-end's sampling clock. The images are captured in grayscale at 10 frames per second and timestamped by the Sensorium's computer. The cameras are configured to automatically adjust the exposure time based on lighting, while the focal length, focus, and aperture are held fixed, having been adjusted physically prior to capture.

5.2. Test Route

The test route was a 1-km loop north of the University of Texas at Austin campus in Austin, TX. The route included a variety of light-to-moderate urban conditions, from open-sky to overhanging trees to built-up areas. The Dean Keeton corridor, toward the left in Figure 11, was the most challenging stretch along the test route for GNSS positioning. It passes below a pedestrian bridge and is flanked on both sides by buildings ranging from 30 to 65 meters tall set back 28 meters from the center of the roadway.



Figure 11. An overview of the 1-km test route. The Dean Keeton corridor, toward the left, is spanned by a pedestrian bridge and flanked by buildings on both sides. A total of 75 laps of the test route were driven over six separate campaigns.

To study the code-phase-based positioning error characteristics over time-separated sessions in the same area, and to perform multi-session mapping with GEOSLAM, multiple laps of the test route were driven over six separate campaigns. The first two campaigns were conducted on 21 December 2017 and

15 January 2018, while the other four campaigns were conducted in pairs of two on 3 June 2018 and 4 June 2018. The GNSS error charts are presented for a total of 75 laps of the test route, while multi-session mapping with GEOSLAM was performed over eight laps/sessions of data from the four latest campaigns.

Imagery collected over the four June 2018 campaigns exhibits appreciable visual diversity, offering a real-world challenge to multi-session GEOSLAM operation. Figure 12a,b show the variation in lighting and visual features between the data collected on 3 June 2018 and 4 June 2018.



Figure 12. Different visual conditions on two days of data collection. (a) An image captured on the first day of data collection. Note the sharp shadows and absence of parked cars. (b) An image captured on the second day of data collection. Note the absence of sharp shadows and complete blockage of curb due to parked cars.

5.3. Empirical GNSS Error Analysis

Figure 13 shows the error in the enhanced code-phase GNSS position solutions with respect to the ground truth. The error is plotted versus the distance along the 1-km loop. The beginning of this loop was taken to be immediately after the overhead pedestrian bridge along the Dean Keeton corridor. It is observed that the enhanced code-phase GNSS errors are clustered separately for each of the campaigns, and that each cluster is offset from zero by as much as 1 m in the horizontal plane. Such error characteristics are representative of ionospheric modeling errors, which have a long decorrelation time. It is also evident that the error variance was larger as the receiver exits the challenging portion of the loop at which point the tracking loops were recovering from signal loss under the bridge. The effect was especially pronounced in the vertical direction. Figure 14 shows similar errors for the commercial ublox M8T receiver. The error traces from the ublox receiver show a wider spread than the enhanced code-phase receiver, likely due to lack of precise orbit and clock corrections.

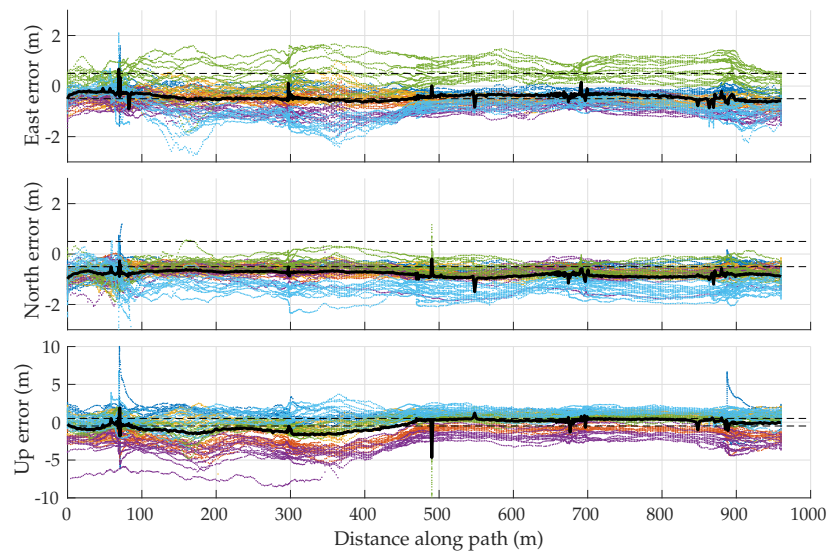


Figure 13. Errors in enhanced code-phase position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. The dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation is nearly constant along the path in the horizontal plane at ~ 0.6 m in the east and ≈ 0.4 m in the north direction. In the up direction, the standard deviation is ~ 2.1 m for the first 400 m along the path, and ≈ 1.3 m for the rest.

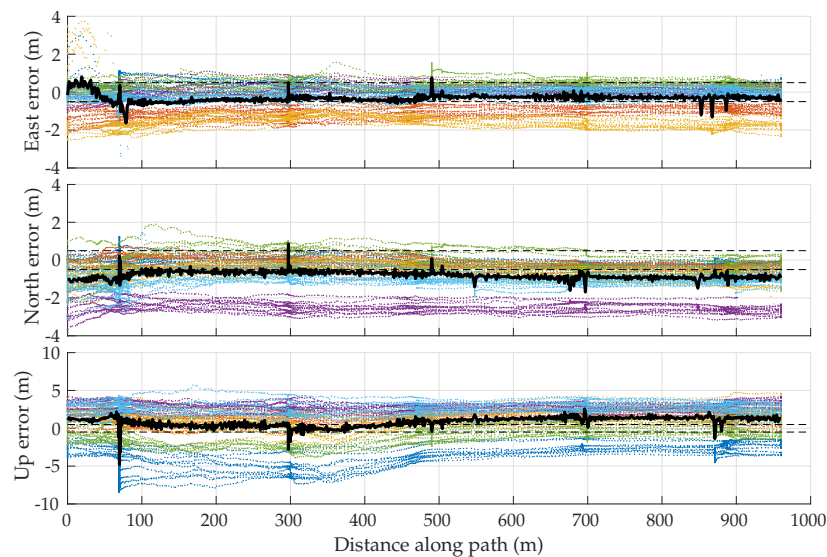


Figure 14. Errors in ublox M8T position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. Dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation in the east is ~ 1.5 m over the first 100 m along the path and ~ 0.7 m over the rest; ~ 0.9 m in the north; and ~ 2.7 m over the first 400 m and ~ 2 m over the rest in the up direction.

On the basis of Figures 13 and 14, one might be tempted to conclude that errors in enhanced code-phase and stand-alone GNSS navigation solutions are substantially non-zero-mean, especially in the

north and up directions, despite the overhead GNSS constellation changing substantially between sessions. It certainly appears that the permanent structures (buildings, bridge) along the test loop left a bias in the vertical direction during the first 400 m along the loop. However, the bias in the north direction, and to a lesser extent in the east, may only be an artifact of the small sample size: ionospheric modeling errors were not yet averaged down to nearly zero in the east and ~ 30 cm in the north, as one would expect from the WAAS ionospheric model (see Table 1).

Given that the asymptotic properties of ionospheric modeling errors are better understood than those of multipath errors, it is instructive to eliminate, insofar as possible, all ionospheric modeling errors from the along-track error histories. To this end, a differential code phase GNSS technique was applied whereby the navigation solution was based on double-difference pseudorange measurements using data from a nearby reference station at a precisely known location. Such double differencing over a short 1-km baseline eliminates virtually all ionospheric and tropospheric errors, but does nothing to reduce vehicle-side multipath. Thus, one can empirically examine multipath effects in isolation from ionospheric effects.

Figure 15 shows the results of this study based on all six data capture campaigns. Note that biases for all components are much smaller. It appears that for the test route chosen, non-zero-mean horizontal errors in the enhanced code phase positions were almost entirely driven by ionospheric modeling errors, and not by persistent effects of multipath due to the permanent structures along the test route. This is broadly consistent with the analyses presented earlier in this paper on position-domain biases due to ionospheric and multipath errors. However, it does appear that a bias due to multipath remained in the vertical direction over the first 400 m, even when ionospheric errors were removed. Apparently, the arrangement of buildings over this segment caused non-line-of-sight effects that did not average away. Mercifully, horizontal errors, which appear to be close to zero-mean over the six campaigns, matter most for high-accuracy digital mapping, since obstacle avoidance and vehicle coordination are largely 2-D problems, and since multiple vehicles can straightforwardly agree on a particular feature's relative vertical position from an inferred road surface.

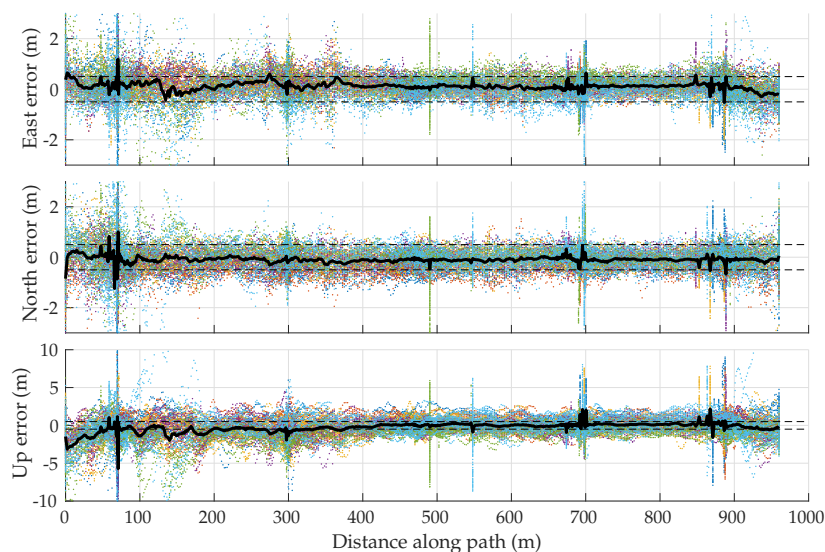


Figure 15. Errors in double-differenced pseudorange-based position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. Dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation in the east and north directions is ~ 0.9 m over the first 200 m along the path and ~ 0.4 m over the rest. In the up direction, the standard deviation is ~ 1.9 m over the first 400 m and ~ 1 m over the rest.

Based on Figure 15, one can conclude that multi-session averaging with a sufficiently accurate ionospheric model, such as the Fast PPP model, yields sub-50-cm global referencing accuracy for digital maps in the horizontal plane with code-phase-based GNSS, even in the presence of persistent multipath.

5.4. Multi-Session Mapping Results

GEOSLAM processed two laps/sessions of data from each of the four campaigns conducted on 4 and 5 June, fusing the visual data from the captured images with the double-differenced pseudorange-based position estimates of the primary antenna. Figure 16 summarizes the result from GEOSLAM's multi-session GNSS-aided-visual SLAM. The black data points denote the difference between the ground truth trajectory of the primary GNSS antenna and GEOSLAM's estimate of the same in local east, north, and up directions for all eight sessions. The gray data points denote the difference between the ground truth trajectory of the primary GNSS antenna and the coincident double-differenced pseudorange-based estimate of the same for all eight sessions.

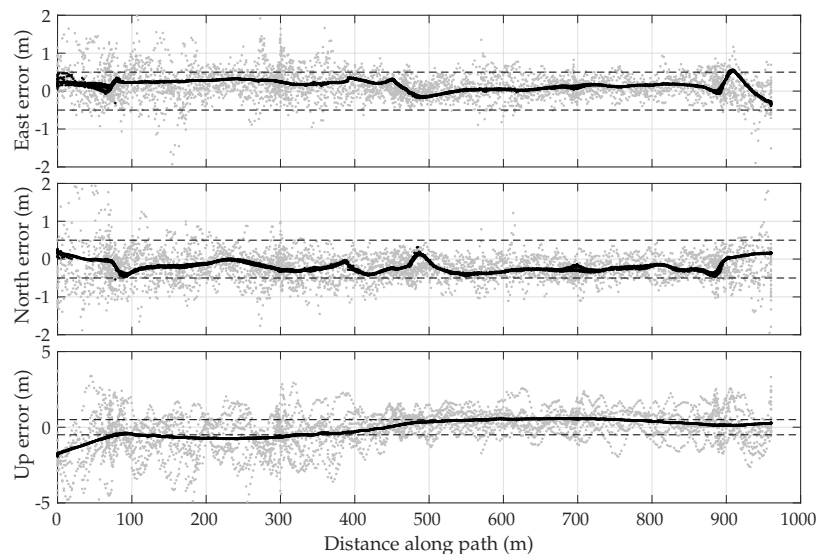


Figure 16. Errors in GEOSLAM's estimate of the primary antenna position (in black) with respect to ground truth in the east, north, and up directions for eight mapping sessions from four different data collection campaigns. The errors in double-differenced pseudorange-based primary antenna position estimates for each of the eight sessions, fed as measurements to GEOSLAM, are plotted in gray for reference. Dashed reference lines are drawn at ± 50 cm.

As one might expect, the error in GEOSLAM's estimate of the antenna position was approximately the same as the average double-differenced pseudorange-based error over eight sessions. Furthermore, due to the approximately zero-mean nature of the double-differenced pseudorange-based estimates, the GEOSLAM estimate of the trajectory was within 50 cm of the truth trajectory in the horizontal plane. Note that the error in GEOSLAM's position estimate was highly repeatable over eight different sessions, so much so that it appears there is a single black trace in Figure 16, while in truth eight independent traces were plotted. This indicates that (i) the localization of the vehicle within the visual map was highly precise: GEOSLAM made the same errors with respect to ground truth over eight different sessions; and (ii) the visual map was merged across eight sessions from four different campaigns: if the maps from any two campaigns were not merged through visual matching of features, then the GNSS position estimate for a

keyframe from one campaign would not affect another keyframe from a different campaign since they would not be covisible, and thus the eight black traces in Figure 16 would not overlap.

6. Conclusions

The accuracy limits of collaborative global referencing of digital maps with standard GNSS were explored through simulation and real data. The asymptotic average of position errors due to thermal noise, satellite orbit and clock errors, and tropospheric modeling errors were assumed to be negligible. It has been shown that the position error due to inaccurate ionospheric modeling may lead to persistent dm-level biases in the horizontal position if the corrections are sourced from the IGS GIM, but other recent models such as the Fast PPP IONEX GIM perform better in this regard. Multipath errors persist with multiple mapping sessions through the same urban corridor and may not be zero mean. With adequate multipath exclusion, persistent multipath biases may be reduced below 50 cm on average. In conclusion, sub-50-cm accurate digital mapping has been shown to be feasible in the horizontal plane after multiple mapping sessions with code-phase-based GNSS, but larger biases persist in the vertical direction. A globally-referenced electro-optical SLAM pipeline, termed GEOSLAM, has been detailed and demonstrated to achieve sub-50-cm horizontal localization accuracy in a moderate urban environment by incorporating code-phase-based GNSS position estimates in the visual SLAM framework and jointly optimizing maps merged across time-separated sessions.

Author Contributions: Conceptualization, L.N. and T.E.H.; Data curation, L.N., J.M.W., M.J.M. and D.M.L.; Funding acquisition, T.E.H.; Methodology, L.N.; Project administration, T.E.H.; Software, L.N., J.M.W. and M.J.M.; Supervision, T.E.H.; Writing—original draft, L.N.; Writing—review & editing, L.N. and T.E.H.

Funding: This work has been supported by the Data-supported Transportation Operations and Planning Center (DSTOP), a Tier 1 USDOT University Transportation Center, and by funding from Toyota, Honda, and Qualcomm through the University of Texas Situation-Aware Vehicular Engineering Systems (SAVES) Center (<http://utsaves.org/>), an initiative of the Wireless Networking and Communications Group.

Conflicts of Interest: The authors declare no conflict of interest. The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Wymeersch, H.; Seco-Granados, G.; Destino, G.; Dardari, D.; Tufvesson, F. 5G mmWave Positioning for Vehicular Networks. *IEEE Wirel. Commun.* **2017**, *24*, 80–86. [[CrossRef](#)]
2. Kenney, J.B. Dedicated short-range communications (DSRC) standards in the United States. *Proc. IEEE* **2011**, *99*, 1162–1182. [[CrossRef](#)]
3. Choi, J.; Va, V.; Gonzalez-Prelcic, N.; Daniels, R.; Bhat, C.R.; Heath, R.W. Millimeter-wave vehicular communication to support massive automotive sensing. *IEEE Commun. Mag.* **2016**, *54*, 160–167. [[CrossRef](#)]
4. Levinson, J.; Montemerlo, M.; Thrun, S. *Map-Based Precision Vehicle Localization in Urban Environments*; Robotics: Science and Systems; MIT Press: Cambridge, MA, USA, 2007; Volume 4, p. 1.
5. Levinson, J.; Thrun, S. Robust vehicle localization in urban environments using probabilistic maps. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA), Anchorage, Alaska, 3–8 May 2010; pp. 4372–4378.
6. Humphreys, T.E.; Murrian, M.; Narula, L. Low-cost Precise Vehicular Positioning in Urban Environments. In Proceedings of the IEEE/ION PLANS Meeting, Monterey, CA, USA, 23–26 April 2018.
7. Hutton, J.J.; Gopaul, N.; Zhang, X.; Wang, J.; Menon, V.; Rieck, D.; Kipka, A.; Pastor, F. Centimeter-Level Robust GnsS-Aided Inertial Post-Processing for Mobile Mapping without Local Reference Stations. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *XLI-B3*, 819–826. [[CrossRef](#)]

8. Rogers, S. Creating and evaluating highly accurate maps with probe vehicles. In Proceedings of the 2000 IEEE Intelligent Transportation Systems, Dearborn, MI, USA, 1–3 October 2000; pp. 125–130.
9. Guo, T.; Iwamura, K.; Koga, M. Towards high accuracy road maps generation from massive GPS traces data. In Proceedings of the 2007 IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007; pp. 667–670.
10. Knoop, V.L.; de Bakker, P.F.; Tiberius, C.C.; van Arem, B. Lane determination with GPS precise point positioning. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2503–2513. [[CrossRef](#)]
11. Circiu, M.S.; Meurer, M.; Felux, M.; Gerbeth, D.; Thöler, S.; Vergara, M.; Enneking, C.; Sgammini, M.; Pullen, S.; Antreich, F. Evaluation of GPS L5 and Galileo E1 and E5a Performance for Future Multifrequency and Multiconstellation GBAS. *Navigation* **2017**, *64*, 149–163. [[CrossRef](#)]
12. Kos, T.; Markezic, I.; Pokrajcic, J. Effects of multipath reception on GPS positioning performance. In Proceedings of the ELMAR-2010, Zadar, Croatia, 15–17 September 2010; pp. 399–402.
13. Le Marchand, O.; Bonnifait, P.; Ibañez-Guzmán, J.; Betaille, D.; Peyret, F. Characterization of GPS multipath for passenger vehicles across urban environments. *ATTI dell'Istituto Italiano di Navigazione* **2009**, *189*, 77–88.
14. Xie, P.; Petovello, M.G. Measuring GNSS multipath distributions in urban canyon environments. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 366–377.
15. Rovira-Garcia, A.; Juan, J.; Sanz, J.; González-Casado, G.; Ibañez, D. Accuracy of ionospheric models used in GNSS and SBAS: Methodology and analysis. *J. Geod.* **2016**, *90*, 229–240. [[CrossRef](#)]
16. Böhm, J.; Möller, G.; Schindelegger, M.; Pain, G.; Weber, R. Development of an improved empirical model for slant delays in the troposphere (GPT2w). *GPS Solut.* **2015**, *19*, 433–441. [[CrossRef](#)]
17. Shepard, D.P.; Humphreys, T.E. Scalable Sub-Decimeter-Accurate 3D Reconstruction. 2017. Available online: <https://ctr.utexas.edu/wp-content/uploads/138.pdf> (accessed on 28 July 2018).
18. Shepard, D.P.; Humphreys, T.E. High-Precision Globally-Referenced Position and Attitude via a Fusion of Visual SLAM, Carrier-Phase-Based GPS, and Inertial Measurements. In Proceedings of the IEEE/ION PLANS Meeting, Monterey, CA, USA, 5–8 May 2014.
19. Pesyna, K.M., Jr. Advanced Techniques for Centimeter-Accurate GNSS Positioning on Low-Cost Mobile Platforms. Ph.D. Thesis, The University of Texas at Austin, Austin, TX, USA, 2015.
20. Bryson, M.; Sukkarieh, S. A Comparison of Feature and Pose-Based Mapping using Vision, Inertial and GPS on a UAV. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011.
21. Ellum, C. Integration of raw GPS measurements into a bundle adjustment. *IAPRS Ser.* **2006**, *35*, 3025.
22. Kume, H.; Taketomi, T.; Sato, T.; Yokoya, N. Extrinsic Camera Parameter Estimation Using Video Images and GPS Considering GPS Positioning Accuracy. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 3923–3926. [[CrossRef](#)]
23. Lhuillier, M. Incremental Fusion of Structure-from-Motion and GPS Using Constrained Bundle Adjustments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2489–2495. [[CrossRef](#)] [[PubMed](#)]
24. Lhuillier, M. Fusion of GPS and structure-from-motion using constrained bundle adjustments. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 3025–3032. [[CrossRef](#)]
25. Soloviev, A.; Venable, D. Integration of GPS and Vision Measurements for Navigation in GPS Challenged Environments. In Proceedings of the IEEE/ION PLANS Meeting, IEEE/Institute of Navigation, Indian Wells, CA, USA, 4–6 May 2010; pp. 826–833.
26. Aumayer, B.M. Ultra-tightly Coupled Vision/GNSS for Automotive Applications. Ph.D. Thesis, University of Calgary, Calgary, AB, Canada, 2016.
27. Chu, T.; Guo, N.; Backén, S.; Akos, D. Monocular camera/IMU/GNSS integration for ground vehicle navigation in challenging GNSS environments. *Sensors* **2012**, *12*, 3162–3185. [[CrossRef](#)] [[PubMed](#)]
28. Wang, J.J.; Kodagoda, S.; Dissanayake, G. Vision Aided GPS/INS System for Robust Land Vehicle Navigation. In Proceedings of the ION International Technical Meeting, Anaheim, CA, USA, 26–28 January 2009; pp. 600–609.
29. Sajad, S.; Michael, T.; Mae, S.; Howard, L. Multiple-robot Simultaneous Localization and Mapping: A Review. *J. Field Robot.* **2016**, *33*, 3–46. [[CrossRef](#)]

30. Zou, D.; Tan, P. CoSLAM: Collaborative Visual SLAM in Dynamic Environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 354–366. [[CrossRef](#)] [[PubMed](#)]
31. Forster, C.; Lynen, S.; Kneip, L.; Scaramuzza, D. Collaborative monocular SLAM with multiple Micro Aerial Vehicles. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3962–3970. [[CrossRef](#)]
32. Piasco, N.; Marzat, J.; Sanfourche, M. Collaborative localization and formation flying using distributed stereo-vision. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1202–1207. [[CrossRef](#)]
33. Indelman, V.; Nelson, E.; Michael, N.; Dellaert, F. Multi-robot pose graph localization and data association from unknown initial relative poses via expectation maximization. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 593–600. [[CrossRef](#)]
34. Cunningham, A.; Wurm, K.M.; Burgard, W.; Dellaert, F. Fully distributed scalable smoothing and mapping with robust multi-robot data association. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 14–18 May 2012; pp. 1093–1100. [[CrossRef](#)]
35. Andersson, L.A.A.; Nygard, J. C-SAM: Multi-Robot SLAM using square root information smoothing. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 2798–2805. [[CrossRef](#)]
36. Dabeer, O.; Ding, W.; Gowaiker, R.; Grzechnik, S.K.; Lakshman, M.J.; Lee, S.; Reitmayr, G.; Sharma, A.; Somasundaram, K.; Sukhvasi, R.T.; Wu, X. An end-to-end system for crowdsourced 3D maps for autonomous vehicles: The mapping component. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 634–641. [[CrossRef](#)]
37. u-blox. u-blox Announces U-Blox F9 Robust and Versatile High Precision Positioning Technology for Industrial and Automotive Applications. 2018. Available online: <https://bit.ly/2GoXaOm> (accessed on 27 July 2018).
38. Murrian, M.J.; Gonzalez, C.W.; Humphreys, T.E.; Pesyna, K.M., Jr.; Shepard, D.P.; Kerns, A.J. Low-cost precise positioning for automated vehicles. *GPS World* **2016**, *27*, 32–39.
39. Misra, P.; Enge, P. *Global Positioning System: Signals, Measurements, and Performance*, 2nd rev. ed.; Ganga-Jumana Press: Lincoln, MA, USA, 2012.
40. GPS Satellite Ephemerides/Satellite & Station Clocks. Available online: <http://www.igs.org/products> (accessed on 30 March 2018).
41. Rovira-Garcia, A.; Juan, J.M.; Sanz, J.; González-Casado, G. A Worldwide Ionospheric Model for Fast Precise Point Positioning. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 4596–4604. [[CrossRef](#)]
42. Yunck, T.P. Chapter 21: Orbit Determination. In *Global Positioning System: Theory and Applications*; American Institute of Aeronautics and Astronautics: Washington, DC, USA, 1996; Volume 2, pp. 559–592.
43. van Bree, R.J.P.; Tiberius, C.C.J.M. Real-time single-frequency precise point positioning: Accuracy assessment. *GPS Solut.* **2012**, *16*, 259–266. [[CrossRef](#)]
44. Le, A.Q.; Tiberius, C. Single-frequency precise point positioning with optimal filtering. *GPS Solut.* **2007**, *11*, 61–69. [[CrossRef](#)]
45. Odijk, D. *Fast Precise GPS Positioning in the Presence of Ionospheric Delays*; No. 52 in Fast Precise GPS Positioning in the Presence of Ionospheric Delays, NCG, Nederlandse Commissie voor Geodesie; Delft University of Technology: Delft, The Netherlands, 2002.
46. Shi, C.; Gu, S.; Lou, Y.; Ge, M. An improved approach to model ionospheric delays for single-frequency Precise Point Positioning. *Adv. Space Res.* **2012**, *49*, 1698–1708. [[CrossRef](#)]
47. Boehm, J.; Werl, B.; Schuh, H. Troposphere mapping functions for GPS and very long baseline interferometry from European Centre for Medium-Range Weather Forecasts operational analysis data. *J. Geophys. Res. Solid Earth* **2006**, *111*. [[CrossRef](#)]
48. Böhm, J.; Niell, A.; Tregoning, P.; Schuh, H. Global Mapping Function (GMF): A new empirical mapping function based on numerical weather model data. *Geophys. Res. Lett.* **2006**, *33*. [[CrossRef](#)]
49. Lehner, A.; Steingass, A. A novel channel model for land mobile satellite navigation. In Proceedings of the ION GNSS Meeting, Long Beach, CA, USA, 13–16 September 2005; pp. 13–16.

50. Lehner, A.; Steingass, A. *Technical Note on the Land Mobile Satellite Channel Model—Interface Control Document*; Association of Radio Industries and Businesses: Tokyo, Japan, 2008.
51. Psiaki, M.; Mohiuddin, S. Modeling, analysis, and simulation of GPS carrier phase for spacecraft relative navigation. *J. Guid. Control Dyn.* **2007**, *30*, 1628. [[CrossRef](#)]
52. Braasch, M.S. *Springer Handbook of Global Navigation Satellite Systems*; Chapter Multipath; Springer: Cham, Switzerland, 2017; pp. 443–468.
53. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; John Wiley and Sons: New York, NY, USA, 2001.
54. Mur-Artal, R.; Tardos, J.D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. *arXiv* **2016**, arXiv:1610.06475.
55. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int. J. Robot. Res.* **2015**, *34*, 314–334. [[CrossRef](#)]
56. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 834–849.
57. Strasdat, H.; Montiel, J.; Davison, A.J. Visual SLAM: Why filter? *Image Vis. Comput.* **2012**, *30*, 65–77. [[CrossRef](#)]
58. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
59. Muja, M.; Lowe, D.G. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proceedings of the VISAPP (1)*, Lisboa, Portugal, 5–8 February 2009; Volume 2, p. 2.
60. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [[CrossRef](#)]
61. Lightsey, E.G.; Humphreys, T.E.; Bhatti, J.A.; Joplin, A.J.; O’Hanlon, B.W.; Powell, S.P. Demonstration of a Space Capable Miniature Dual Frequency GNSS Receiver. *Navig. J. Inst. Navig.* **2014**, *61*, 53–64. [[CrossRef](#)]
62. Sorkine-Hornung, O.; Rabinovich, M. Least-squares rigid motion using SVD. *Computing* **2017**, *1*, 1.
63. Mühlheller, P.; Bürki, M.; Bosse, M.; Derendarz, W.; Philippsen, R.; Furgale, P. Summary maps for lifelong visual localization. *J. Field Robot.* **2016**, *33*, 561–590. [[CrossRef](#)]
64. Strasdat, H.; Davison, A.J.; Montiel, J.M.; Konolige, K. Double window optimisation for constant time visual SLAM. In *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 6–13 November 2011; pp. 2352–2359.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).