Map-based Visual-Inertial Localization: A Numerical Study

Patrick Geneva and Guoquan Huang

***Abstract*— We revisit the problem of efficiently leveraging prior map information within a visual-inertial estimation framework. The use of traditional landmark-based maps with 2D-to-3D measurements along with the recently introduced keyframe-based maps with 2D-to-2D measurements are investigated. The full joint estimation of the prior map is compared within a visual-inertial simulator to the Schmidt-Kalman filter (SKF) and measurement inflation methods in terms of their computational complexity, consistency, accuracy, and memory usage. This study shows that the SKF can enable efficient and consistent estimation for small workspace scenarios and the use of 2D-to-3D landmark maps have the highest levels of accuracy. Keyframe-based 2D-to-2D maps can reduce the required state size while still enabling accuracy gains. Finally, we show that measurement inflation methods, after tuning, can be accurate and efficient for large-scale environments if the guarantee of consistency is relaxed.**

I. Introduction

Camera and inertial sensors have become increasingly prominent in robotic and autonomous applications due to their small form factor, complimentary sensing nature, weight, and low cost [1]. Visual-inertial navigation systems (VINS) look to fuse this visual and inertial dynamic information into an estimate of the platform's pose and corresponding uncertainty. One of the barriers which prevents the wider deployment of VINS is that visual-inertial sensors can only provide the relative change to an arbitrary frame and cannot recover global yaw and position [2], [3].

To overcome this, VINS is typically a sub-system of the complete localization solution and provides high frequency *relative* pose information that is later fused with global information (e.g., a prior map with loop-closures). Incorporating global information increases the computational complexity and memory resources, thus many works have relegated this problem to secondary non-realtime background threads [4]–[9]. This has typically been achieved by splitting the underlying *joint* optimization problem into one which recovers the global pose and introduces loop-closure constraints, and another odometry method which provides high frequency relative poses. The two optimization problems do *not* model the correlations between each other and thus are, at best, an imperfect inconsistent approximation of the original joint problem. The popular design which uses a visual-inertial odometry (VIO) front-end, and a backend optimization which incorporates loop-closure information typically has the additional downside that global correction information cannot be leveraged in the "front-end" VIO [4]–[6], [10]. Works which have allowed the front-end to use an optimized map which includes global information, treat the map as perfectly know or do not track the correlations [7], [11]–[16].

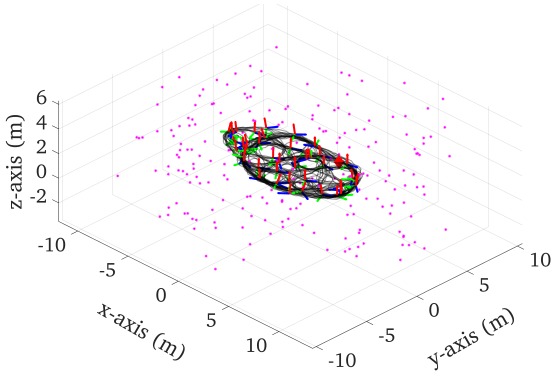

Fig. 1: Simulated 1.2km hand-held Room trajectory, axes are in units of meters. Every other keyframe is shown to increase clarity. Feature depths (purple) are between 5 and 7 meters.

A few works have looked to efficiently incorporate global information *directly* into the estimator in a consistent manor. Most leverage the Schmidt-Kalman filter (SKF) [17] formulation which allows for estimation of the active states and "schmidt'ed" states which are treated as nuisance parameters and not updated. The SKF's computational cost is linear in the number of schmidt'ed states which is ideal for large prior map sizes. Dutoit et al. [18] introduced the Cholesky-Schmidt-Kalman filter which explicitly considered the uncertainty of a given fixed prior map. They showed that, as compared to doing simple measurement inflation and treating the prior map as true, their method was consistent and computationally efficient. Ke et al. [19] presents an approximate information variant of the Schmidt-Kalman filter, termed RISE-SLAM, which enabled efficient information filtering, but required state re-ordering between "exploration" and "relocalization" phases. Huai and Huang [20] later expanded in this direction and showed that tracking of two states in opposite state ordering and incorporating loop-closure constraints asynchronously was feasible. Additionally, we have previously proposed two works which have investigated the use of the SKF for efficient online map-based localization [21], [22]. Both leveraged the SKF and built either a point-based or keyframe pose-based map and directly updated the state with loop-closure information in real-time.

There are a few different ways to relate global map information to the state. The most common method is to leverage sparse visual landmark maps and constrain the front-end state with 2D-to-3D constraints [7], [12], [15], [16], [18], [21]. Typically, due to the large map size, these methods

This work was partially supported by the University of Delaware (UD) College of Engineering, the NSF (IIS-1924897, SCH-2014264), the ARL (W911NF-19-2-0226, W911NF-20-2-0098), and Google ARCore. P. Geneva is also supported by the UD University Doctoral Fellowship.

The authors are with the Robot Perception and Navigation Group (RPNG), University of Delaware, Newark, DE 19716, USA. Email: {pgeneva,ghuang}@udel.edu

either treat the sparse landmark map as true or perform measurement inflation. We presented an alternative method in [22], which leveraged prior keyframes with historical observations. After matching to features tracked in the current frame, these historical observations could be used to related the active feature to the historical keyframe pose, and thus indirectly constrain the current pose to the global prior map.

In this paper we investigate how these different methods can be incorporated within a VINS, and how each impacts the computational complexity, accuracy, and consistency. The key contributions of our work are as follows:

- We investigate in depth the use of landmark 2D-to-3D and keyframe-based 2D-to-2D prior maps within a filter-based real-time visual-inertial estimator.
- We study different techniques for incorporating loop-closure information: (i) full estimation of the prior map in a joint manor, (ii) Schmidt'ing of the prior map, and (iii) a variety of different measurement inflation methods.
- All variations are evaluated numerically within a realistic visual-inertial simulator leveraging real-world trajectories and discussed in terms of their accuracy, consistency, computational complexity, and memory.

II. Problem Statement

A. Map-based EKF-SLAM

We build off MSCKF-based VIO [23]–[25] and incorporate prior landmark or keyframe maps. Specifically, we can define the following state of the system:

$$\mathbf{x}_k = \begin{bmatrix}\mathbf{x}_A^\top & \mathbf{x}_M^\top & \mathbf{x}_K^\top\end{bmatrix}^\top \tag{1}$$

$$\mathbf{x}_A = \begin{bmatrix}\mathbf{x}_{I_k}^\top & \mathbf{x}_C^\top & \mathbf{x}_L^\top\end{bmatrix}^\top \tag{2}$$

$$\mathbf{x}_M = \begin{bmatrix}{}^G\mathbf{p}_{f_1}^\top \cdots {}^G\mathbf{p}_{f_m}^\top\end{bmatrix}^\top \tag{3}$$

$$\mathbf{x}_K = \begin{bmatrix}\mathbf{x}_{T_1}^\top \cdots \mathbf{x}_{T_n}^\top\end{bmatrix}^\top \tag{4}$$

where:

$$\mathbf{x}_{I_k} = \begin{bmatrix}{}^{I_k}_G\bar{q}^\top & \mathbf{b}_{\omega_k}^\top & {}^G\mathbf{v}_{I_k}^\top & \mathbf{b}_{a_k}^\top & {}^G\mathbf{p}_{I_k}^\top\end{bmatrix}^\top \tag{5}$$

$$\mathbf{x}_C = \begin{bmatrix}\mathbf{x}_{T_{k-1}}^\top \cdots \mathbf{x}_{T_{k-c}}^\top\end{bmatrix}^\top \tag{6}$$

$$\mathbf{x}_L = \begin{bmatrix}{}^G\mathbf{p}_{f_1}^\top \cdots {}^G\mathbf{p}_{f_d}^\top\end{bmatrix}^\top,\ \mathbf{x}_{T_i} = \begin{bmatrix}{}^{I_i}_G\bar{q}^\top & {}^G\mathbf{p}_{I_i}^\top\end{bmatrix}^\top \tag{7}$$

where we define the "active" state $\mathbf{x}_A$, map of m features $\mathbf{x}_M$, and map of n keyframe poses $\mathbf{x}_K$. The clone state $\mathbf{x}_C$ contains c historical IMU poses and a local temporal SLAM feature map $\mathbf{x}_L$ stores features which are tracked and updated until lost. ${}^I_G\bar{q}$ is the unit quaternion parameterizing the rotation ${}^I_G\mathbf{R}$ from the global frame of reference $\{G\}$ to the IMU local frame $\{I\}$ [26], $\mathbf{b}_\omega$ and $\mathbf{b}_a$ are the gyroscope and accelerometer biases, and ${}^G\mathbf{v}_I$ and ${}^G\mathbf{p}_I$ are the velocity and position of the IMU expressed in the global frame, respectively.

B. Propagation

The state evolves according to the inertial nonlinear IMU kinematics [27]. All states besides the inertial state, $\mathbf{x}_I$, have zero dynamics. Integrating incoming IMU linear accelerations, $\mathbf{a}_m$, and angular velocities, $\boldsymbol{\omega}_m$, we get:

$$\mathbf{x}_{I_{k+1}} = \mathbf{f}(\mathbf{x}_{I_k}, \mathbf{a}_{m_k} - \mathbf{n}_{a_k}, \boldsymbol{\omega}_{m_k} - \mathbf{n}_{\omega_k}) \tag{8}$$

where $\mathbf{n}_a$ and $\mathbf{n}_\omega$ are the zero-mean white Gaussian noise of the IMU measurements. We can then linearize the nonlinear model and propagate the state and covariance forward [23].

C. Feature Measurement Function

As the sensor platform moves, bearings to environmental features are tracked on the image plane using KLT optical flow [28]. A bearing measurement $\mathbf{z}$ seen at timestep k can be related to the state by the following (simplified for presentation, model in [25] is used):

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_{T_k}, {}^G\mathbf{p}_f) + \mathbf{n}_k =: \boldsymbol{\Lambda}({}^{C_k}\mathbf{p}_f) + \mathbf{n}_k \tag{9}$$

$$\boldsymbol{\Lambda}([x\ y\ z]^\top) = \begin{bmatrix}x/z & y/z\end{bmatrix}^\top \tag{10}$$

$${}^{C_k}\mathbf{p}_f = {}^C_I\mathbf{R}\,{}^{I_k}_G\mathbf{R}({}^G\mathbf{p}_f - {}^G\mathbf{p}_{I_k}) + {}^C\mathbf{p}_I \tag{11}$$

where $\mathbf{n}_k$ is the white Gaussian noise with covariance $\mathbf{R}_k = \sigma_{pix}^2\mathbf{I}$. We can now linearize this measurement model and obtain the following residual:

$$\mathbf{r}_k = \mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_{T_k}, {}^G\hat{\mathbf{p}}_f) \tag{12}$$

$$\simeq \mathbf{H}_{T_k}\tilde{\mathbf{x}}_{T_k} + \mathbf{H}_{f_k}{}^G\tilde{\mathbf{p}}_f + \mathbf{n}_k \tag{13}$$

where $\mathbf{H}_{T_k}$ and $\mathbf{H}_{f_k}$ are the measurement Jacobians, and $\tilde{\mathbf{x}}_{T_k}$ and ${}^G\tilde{\mathbf{p}}_f$ are the error states for the observation pose and feature, respectively.[1] After sufficient observations of the feature, we can "stack" them to get:

$$\mathbf{r} = \mathbf{H}_T\tilde{\mathbf{x}}_{T_{1..c}} + \mathbf{H}_f{}^G\tilde{\mathbf{p}}_f + \mathbf{n} \tag{14}$$

where the measurement is a function of c clone poses, $\tilde{\mathbf{x}}_{T_{1..c}} = [\tilde{\mathbf{x}}_{T_1}^\top \cdots \tilde{\mathbf{x}}_{T_c}^\top]^\top$, corresponding to each observation time the feature was seen, and the stacked measurement noise is $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ where $\mathbf{R} = \sigma_{pix}^2\mathbf{I}$.

D. Prior Map Update - 2D-to-3D

Consider an actively tracked feature that has matched to the i'th prior map feature in $\mathbf{x}_M$. This is the traditional 2D-to-3D measurement model that has been typically seen throughout the literature [12], [15], [18], [21]. We have:

$$\mathbf{r} = \mathbf{H}_T\tilde{\mathbf{x}}_{T_{1..c}} + \mathbf{H}_{f_i}{}^G\tilde{\mathbf{p}}_{f_i} + \mathbf{n} \tag{15}$$

where ${}^G\mathbf{p}_{f_i} \in \mathbf{x}_M$. This measurement function can directly update the state.

E. Prior Map Update - 2D-to-2D

Presented in [22], consider the case that we observe a feature from the k'th keyframe in our prior map of keyframes and has also been seen by c active clones:

$$\mathbf{r} = \mathbf{H}_T\tilde{\mathbf{x}}_{T_{1..c}} + \mathbf{H}_{T_k}\tilde{\mathbf{x}}_{T_k} + \mathbf{H}_f{}^G\tilde{\mathbf{p}}_f + \mathbf{n} \tag{16}$$

[1] Throughout this paper $\hat{\mathbf{x}}$ is used to denote the estimate of a random variable $\mathbf{x}$, while $\tilde{\mathbf{x}} = \mathbf{x} \boxminus \hat{\mathbf{x}}$ is the error in this estimate. The updated estimate from a correction $\boldsymbol{\delta}\mathbf{x}$ is $\hat{\mathbf{x}}^\oplus = \hat{\mathbf{x}} \boxplus \boldsymbol{\delta}\mathbf{x}$.

where $\mathbf{x}_{T_k} \in \mathbf{x}_K$. We then remove the dependency on the feature ${}^G\mathbf{p}_f$, ${}^G\mathbf{p}_f \notin \mathbf{x}_k$, by projecting $\mathbf{r}$ onto the left nullspace of $\mathbf{H}_f$ (i.e., $\mathbf{N}^\top\mathbf{H}_f = \mathbf{0}$) [23]:

$$\begin{aligned}\mathbf{N}^\top\mathbf{r} &= \mathbf{N}^\top\mathbf{H}_T\tilde{\mathbf{x}}_{T_{1..c}} + \mathbf{N}^\top\mathbf{H}_{T_k}\tilde{\mathbf{x}}_{T_k} \\ &\quad + \mathbf{N}^\top\mathbf{H}_f{}^G\tilde{\mathbf{p}}_f + \mathbf{N}^\top\mathbf{n} \qquad (17)\\ \Rightarrow\ \mathbf{r}' &= \mathbf{H}'_T\tilde{\mathbf{x}}_{T_{1..c}} + \mathbf{H}'_{T_k}\tilde{\mathbf{x}}_{T_k} + \mathbf{n}' \qquad (18)\end{aligned}$$

where $\mathbf{n}' = \mathbf{N}^\top\mathbf{n}$ with covariance $\mathbf{R}' = \mathbf{N}^\top\mathbf{R}\mathbf{N}$. This measurement function can directly update the state.

F. Non-Map Feature Updates

Features which have not been matched to the prior maps can be processed as local "SLAM features" and inserted into the state vector, $\mathbf{x}_L$, as part of a local temporal map or as "MSCKF features" which directly update the state by removing the feature position through the MSCKF nullspace projection procedure [23]. The updates are similar to Eq. (15) and Eq. (18) (see [25] for details).

III. Methods for Prior Map Updates

We now detail the different methods and techniques which enable efficient incorporation of global information. In general, for a filter or non-linear least squares the complexity is on order $O(x^2)$ or $O(x^3)$ (where x is the total map state size) in the worst case if the measurement residual size is assumed to be much smaller in order. We now present different approximations which aim to reduce this complexity.

A. Extended Kalman Filter

We first begin with the standard EKF which jointly estimates all variables. We split the state into an active and map portion at timestep k.

$$\mathbf{x}_k = \begin{bmatrix}\mathbf{x}_{A_k}\\ \mathbf{x}_{S_k}\end{bmatrix}\ ,\quad \mathbf{P}_k = \begin{bmatrix}\mathbf{P}_{AA_k} & \mathbf{P}_{AS_k}\\ \mathbf{P}_{SA_k} & \mathbf{P}_{SS_k}\end{bmatrix} \qquad (19)$$

where $\mathbf{x}_{S_k} = [\mathbf{x}_M^\top\ \mathbf{x}_K^\top]^\top$. We can define the following linearized measurement function:

$$\mathbf{r} \simeq \mathbf{H}_{A_k}\tilde{\mathbf{x}}_{A_k} + \mathbf{H}_{S_k}\tilde{\mathbf{x}}_{S_k} + \mathbf{n} \qquad (20)$$

where $\tilde{\mathbf{x}}_{S_k} = {}^G\tilde{\mathbf{p}}_{f_i}$ if we matched to a historical feature [see Sec. II-D], or $\tilde{\mathbf{x}}_{S_k} = \tilde{\mathbf{x}}_{T_k}$ in the case that we matched to a historical keyframe [see Sec. II-E]. We can then define the Kalman gain as follows:

$$\begin{aligned}\mathbf{K}_k = \begin{bmatrix}\mathbf{K}_{A_k}\\ \mathbf{K}_{S_k}\end{bmatrix} &= \begin{bmatrix}\mathbf{P}_{AA_k}\mathbf{H}_{A_k}^\top + \mathbf{P}_{AS_k}\mathbf{H}_{S_k}^\top\\ \mathbf{P}_{SA_k}\mathbf{H}_{A_k}^\top + \mathbf{P}_{SS_k}\mathbf{H}_{S_k}^\top\end{bmatrix}\mathbf{S}_k^{-1}\\ &=: \begin{bmatrix}\mathbf{L}_{A_k}\\ \mathbf{L}_{S_k}\end{bmatrix}\mathbf{S}_k^{-1} \qquad (21)\end{aligned}$$

where $\mathbf{H}_k = [\mathbf{H}_{A_k}\ \mathbf{H}_{S_k}]$ and $\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^\top + \mathbf{R}$ is the measurement residual innovation. This then leads to the following mean and covariance update equations:

$$\hat{\mathbf{x}}_{A_k}^\oplus = \hat{\mathbf{x}}_{A_k} + \mathbf{K}_{A_k}\mathbf{r}\ ,\ \hat{\mathbf{x}}_{S_k}^\oplus = \hat{\mathbf{x}}_{S_k} + \mathbf{K}_{S_k}\mathbf{r} \qquad (22)$$

$$\mathbf{P}_k^\oplus = \mathbf{P}_k - \begin{bmatrix}\mathbf{K}_{A_k}\mathbf{S}_k\mathbf{K}_{A_k}^\top & \mathbf{K}_{A_k}\mathbf{H}_k\begin{bmatrix}\mathbf{P}_{AS_k}\\ \mathbf{P}_{SS_k}\end{bmatrix}\\ \begin{bmatrix}\mathbf{P}_{AS_k}\\ \mathbf{P}_{SS_k}\end{bmatrix}^\top{\mathbf{H}_k}^\top\mathbf{K}_{A_k}^\top & \mathbf{K}_{S_k}\mathbf{S}_k\mathbf{K}_{S_k}^\top\end{bmatrix} \qquad (23)$$

We note that this process is $O(x^2)$ complexity if the number of measurements is far smaller than the state size (i.e., ${\mathbf{S}_k}^{-1}$ is cheap) due to the covariance update multiplication $\mathbf{K}\mathbf{S}_k\mathbf{K}^\top$. The memory requirement for landmark-based and keyframe based maps are $O((3m)^2)$ and $O((6n)^2)$, respectively.

B. Linear Schmidt-Kalman Filter

A consistent alternative to the standard EKF is the Schmidt-Kalman filter (SKF) [17]. This has been successfully used (along with different variations) to reduce the update complexity for map-based localization [18]–[22]. Using the same state definition as in Eq. (19), we set $\mathbf{K}_{S_k} = \mathbf{0}$ and get the following update equations [see Eq. (22) and (23)]:

$$\hat{\mathbf{x}}_{A_k}^\oplus = \hat{\mathbf{x}}_{A_k} + \mathbf{K}_{A_k}\mathbf{r}\ ,\ \hat{\mathbf{x}}_{S_k}^\oplus = \hat{\mathbf{x}}_{S_k} \qquad (24)$$

$$\mathbf{P}_k^\oplus = \mathbf{P}_k - \begin{bmatrix}\mathbf{K}_{A_k}\mathbf{S}_k\mathbf{K}_{A_k}^\top & \mathbf{K}_{A_k}\mathbf{H}_k\begin{bmatrix}\mathbf{P}_{AS_k}\\ \mathbf{P}_{SS_k}\end{bmatrix}\\ \begin{bmatrix}\mathbf{P}_{AS_k}\\ \mathbf{P}_{SS_k}\end{bmatrix}^\top{\mathbf{H}_k}^\top\mathbf{K}_{A_k}^\top & \mathbf{0}\end{bmatrix} \qquad (25)$$

This process is $O(x)$ and its memory requirement for landmark-based and keyframe based maps are $O((3m)^2)$ and $O((6n)^2)$, respectively. This is due to only updating the cross-covariance terms. We note that this is a *consistent*[2] approximation which ensures that the filter is never more confident then the original EKF [see [18]].

C. Noise Inflation - Measurement

Another method for incorporating global information is to not explicitly estimate the map states (landmarks or keyframes). The downside is that this prevents the modeling of the correlation between the state and the map and thus is inconsistent. Specifically, in Eq. (12) and (18) we treat the feature position and keyframe pose as known, and thus their Jacobians, $\mathbf{H}_{f_i}$ and $\mathbf{H}'_{T_k}$, become zero.

Naively the simplest way to solve this inconsistency due to an over confident measurement is to inflate the measurement noise. For landmark-based or keyframe-based maps we can simply inflate the measurement observation noise as:

$$\mathbf{R} = (\gamma\sigma_{pix})^2\mathbf{I} \qquad (26)$$

The key advantage of this method is that the computational cost is now constant $O(1)$ since only the inertial state, sliding window, and temporal SLAM map is tracked. The memory requirement for both landmark and keyframe-based maps is $O(0)$. This can have profound impacts on large maps and thus giving up the guarantee of consistency for this computational advantage is very alluring.

[2] An estimator is consistent when its errors are zero-mean (unbiased) and covariance matrix is equal to that reported by the estimator [29, Section 5.4].

TABLE I: Simulation parameters and priors that perturbations of measurements and initial states were drawn from.

Parameter	Value	Parameter	Value
Pixel Proj. (px)	1	Num. Camera	1
IMU Freq. (hz)	400	Cam Freq. (hz)	10
Avg. Feats	15	Num. SLAM	10
Num. Clones	11	Feat. Rep.	GLOBAL
Gyro. White Noise	1.6968e-04	Gyro. Rand. Walk	1.9393e-05
Accel. White Noise	2.0000e-03	Accel. Rand. Walk	3.0000e-03
Prior Key. Ori. (deg)	1.0	Prior Key. Pos. (cm)	6
Prior Feat. Pos. (cm)	12	% Feat. Lost Btw Key.	75
Max Dist. Btw Key. (m)	1	Max Deg. Btw Key. (deg)	15
Map PTS	210	Map KF	86

D. Noise Inflation - Marginal Covariance Inflation

Many works have leveraged the marginal covariance of the prior map to both reduce the complexity and memory requirements of the system (e.g., [15]). The main advantage is that this allows for each landmark or keyframe to have different levels of uncertainty and the use of its Jacobian to map the additional error to the observed measurement. More concretely we have the following modified measurement noise for landmark-based and keyframe-based prior maps respectively [see Eq. (12) and (18)]:

$$\mathbf{R} = \mu \mathbf{H}_{f_i} \mathbf{P}_{f f_i} \mathbf{H}_{f_i}^\top + \sigma_{pix}^2 \mathbf{I} \tag{27}$$

$$\mathbf{R} = \mu \mathbf{H}'_{T_k} \mathbf{P}_{T T_k} \mathbf{H}_{T_k}'^\top + \sigma_{pix}^2 \mathbf{I} \tag{28}$$

where $\mathbf{P}_{f f_i}$ and $\mathbf{P}_{T T_k}$ are the 3x3 and 6x6 prior landmark and keyframe covariances, respectively. This process also ensures the computational cost is also now constant $O(1)$, with memory requirements of $O(9m)$ and $O(36n)$, respectively.

E. Noise Inflation - Alpha Beta Inflation

The final noise inflation variation investigated is the one presented in [30], which incorporates not only inflation due to the marginal prior map covariance but also the current state covariance (originally adopted by NASA's Apollo program [31] and used to "intentionally slow adaptation in linearized estimation problems"). Specifically we have the following:

$$\mathbf{R} = \alpha \mathbf{H}_{f_i} \mathbf{P}_{f f_i} \mathbf{H}_{f_i}^\top + \beta \mathbf{H}_T \mathbf{P} \mathbf{H}_T^\top + \sigma_{pix}^2 \mathbf{I} \tag{29}$$

$$\mathbf{R} = \alpha \mathbf{H}'_{T_k} \mathbf{P}_{T T_k} \mathbf{H}_{T_k}'^\top + \beta \mathbf{H}'_T \mathbf{P} \mathbf{H}_T'^\top + \sigma_{pix}^2 \mathbf{I} \tag{30}$$

This process is constant $O(1)$ in terms of computational cost, with memory requirements of $O(9m)$ and $O(36n)$ for landmark and keyframe-based maps. We normally "whitten" the linearized measurement function with the now dense noise to regain an identity noise covariance form.

IV. Numerical Study

To investigate and compare the different methods for global measurement inclusion we simulated a realistic indoor single room dataset which is approximately 15 minutes long and 1.2km in length (see Fig. 1). We employ the OpenVINS simulator [22] to generate realistic visual-bearing and inertial measurements from the trajectory generated by an existing VINS. Simulation parameters used are documented in Tab. I, while details on how the prior map is generated are specified in the following section. First-estimates Jacobians (FEJ) [32], [33] were used to improve the estimator consistency as the use of environmental landmarks is known to introduce inconsistent information gains. For metrics we report the Absolute Trajectory Error (ATE), Normalized Estimation Error Squared (NEES), and Relative Pose Error (RPE) throughout the different experiments (see [34] and [29]). NEES's magnitude should match the 3 degree-of-freedom orientation and position state sizes.

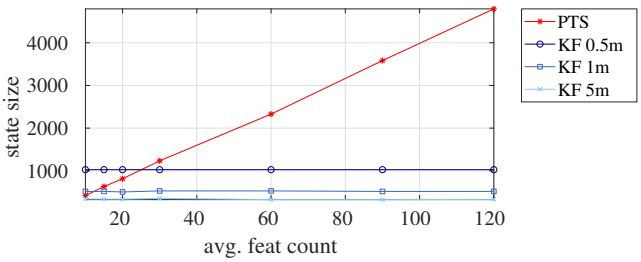

Fig. 2: Relation between state state size (number of variables) and the average number of features observed for both landmark-based (PTS) and keyframe-based (KF) maps in the Room dataset. Different maximum keyframe distance thresholds are also plotted.

Feature matching to historical keyframes to gain additional feature observations was simulated by selecting the closest keyframe and using groundtruth labels, while for map features the groundtruth labels were directly used (thus perfect matching). In real-world experiments, where incorrect feature associations are prevalent, chi-squared thresholding can be leveraged before update to reject outliers. Additional simulation results for different trajectories and noise perturbations can be found in the companion technical report [35]

A. Prior Map Generation

We now describe the procedure on how we generate a prior map of environmental landmarks and keyframes (e.g., Fig. 1). Starting at the beginning of the trajectory we move the camera forward in time at a rate of 4 Hz. At each timestep we project the current landmark map into the camera frame and if the number of seen features falls below our average feature tracking amount we generate new features. This is repeated until the end of the trajectory is reached and our prior landmark map is complete after applying perturbations.

To generate the keyframe map, we repeat this procedure. Specifically at each timestep the current camera must be near an existing keyframe and share a sufficient percentage of common overlapping features; otherwise a new keyframe is created. After generating our keyframes, we project the landmark map into each to generate bearing observations, and both the keyframe poses and observations are perturbed.

Shown in Fig. 2, we perform a small study on how the prior map state size changes with the average number of feature tracks. Landmark-based maps have a state of $3m$, where m is the number of landmarks, and keyframe-based maps have $6n$, where n is the number of keyframes. The landmark map has a very linear relationship with the average number of tracked features and grows to a very large size, which is expected. We additionally show three keyframe

TABLE II: Average ATE and NEES over 5 Room dataset runs for different map priors and algorithms.

	Prior	Algo.	ATE (deg / m)	NEES (3)
VIO	-	-	2.603 / 0.271	3.524 / 1.591
2D-to-2D	0.5°, 3cm	EKF	0.324 / 0.090	2.933 / 3.327
		SKF	0.374 / 0.099	2.758 / 3.248
	1.0°, 6cm	EKF	0.442 / 0.105	3.236 / 3.698
		SKF	0.518 / 0.130	2.806 / 3.466
	3.0°, 12cm	EKF	0.629 / 0.127	4.353 / 5.335
		SKF	0.941 / 0.167	3.009 / 3.585
2D-to-3D	3cm	EKF	0.051 / 0.010	5.975 / 6.586
		SKF	0.064 / 0.021	2.898 / 3.188
	6cm	EKF	0.068 / 0.014	8.224 / 9.292
		SKF	0.087 / 0.036	2.863 / 3.210
	12cm	EKF	0.079 / 0.015	9.321 / 9.472
		SKF	0.122 / 0.065	2.761 / 3.175

prior maps with different maximum distances between generated keyframes. For the keyframe-based maps there is a clear advantage in state size, as the average number of tracked features increase, since more features just increases the number of observations in all keyframes. For the rest of the experiments we select a keyframe distance of 1 meter since the state size is close to the size of a point-based map when using 15 average features, and thus this advantage won't be shown.

B. Map Prior Noise Sensitivity

A natural question is how good will the "best" estimator perform with different prior map noises. We first investigate this using the standard EKF and SKF to see how the accuracy is affected by the quality and uncertainty levels of the prior map. Shown in Tab. II, we report the VIO, which doesn't leverage any prior map, the landmark-based 2D-to-3D map, and the keyframe-based 2D-to-2D map. The simulator parameters used are reported in Tab. I.

First, we can see that all the prior map methods are able to outperform the odometry VIO method. Additionally, even at large noise levels of 12cm, both the landmark and keyframe methods are still able to gain in both the orientation and position accuracy. Additionally, we can see that the 2D-to-3D methods greatly outperform the 2D-to-2D method. This makes sense since the 2D-to-2D indirectly constrain the current pose of the system through additional feature observations, while the 2D-to-3D directly constrain *all* observations for a feature. It is also interesting to note that while the EKF 2D-to-3D has very good levels of accuracy the NEES increases with noise. We conjecture this is due to FEJ, which can introduce linearization errors at high noise levels (the SKF hides this due to its naturally conservative covariance, see [36] for a discussion). Given these results we pick our priors used during the rest of the simulations, Tab. I, as 12cm for the landmark-based map and 1 degree and 6 centimeters for the keyframe-based map with 1 pixel observation noise.

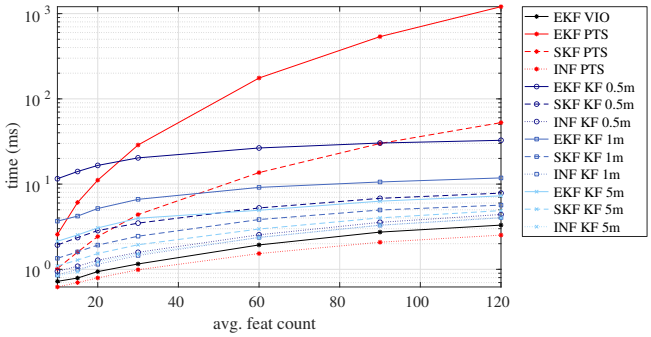

Fig. 3: Runtime in milliseconds for both propagation and update without (VIO) and with both landmark-based (PTS) and keyframe-based (KF) maps for the Room dataset. Keyframe-based map are reported for different max keyframe distances.

C. Inflation Tuning Sensitivity

A downside of the inflation methods is that their inflation multipliers need to be tuned. The results reported in Tab. III look to answer if they are sensitive to their value and determine what the optimal is. We can first see that the measurement-based inflation, γ, requires the largest amount of inflation levels to reach consistent estimation, and with an inflation value of 1 the estimator quickly diverged since it is equivalent to treating the feature position as true. The amount of inflation when using the marginal covariance μ, alpha α, and beta β inflation does not have that large of an affect on accuracy which is ideal. Additionally, there seems to be little difference between the two. We can therefore recommend inflating using the marginal or alpha beta covariance with a conservative (one order) multiplier (this does not *guarantee* consistency). It is also important to note that while these two methods do have some invariance to different prior map noises, the measurement inflation parameter γ highly depends on the prior map quality. We select an inflation of $\gamma = 20$, $\mu = 10$, $\alpha = 10$, and $\beta = 5$ for the rest of the experiments.

D. Map and Algorithm Comparison

We now look to compare the different prior map types and methods that incorporate global information. We report the results in Tab. IV. In general we see that the 2D-to-3D landmark-based methods are able to achieve an order of magnitude better accuracy across all variants, with near constant error as the RPE segments grow in length. The 2D-to-2D method is able to halve the orientation error, but the position error has marginal improvements when compared to that of the 2D-to-3D method. The majority of improvements are at longer trajectory lengths of 200-240m as compared to the shorter segments. This is likely due to the fact that it takes *many* historical 2D-to-2D observations to improve the state as compared to the "strong" constraint a 3D position of the feature in the 2D-to-3D method provides.

We additionally report on the right of Tab. IV and in Fig. 3 each method's average timing. The EKF takes the most time, the SKF second, and the inflation methods all around

TABLE III: Average ATE and NEES over 5 Room runs for different inflation values.

	γ	ATE (deg / m)	NEES (3)	μ	ATE (deg / m)	NEES (3)	α,β	ATE (deg / m)	NEES (3)
VIO	-	2.381 / 0.267	3.522 / 1.590	-	2.381 / 0.267	3.522 / 1.590	-	2.381 / 0.267	3.522 / 1.590
2D-to-2D	1	* / *	* / *	1	0.853 / 0.187	4.219 / 6.928	1,1	0.883 / 0.187	3.796 / 6.070
	5	0.737 / 0.219	5.197 / 17.377	5	0.846 / 0.182	3.124 / 3.198	5,2	0.810 / 0.182	2.826 / 2.916
	10	0.931 / 0.181	3.960 / 6.099	10	0.787 / 0.180	2.699 / 2.385	10,5	0.899 / 0.192	2.688 / 2.275
	20	0.886 / 0.184	2.949 / 3.557	20	0.822 / 0.185	2.574 / 1.893	20,5	0.928 / 0.193	2.650 / 1.867
2D-to-3D	1	* / *	* / *	1	0.132 / 0.045	12.438 / 18.407	1,1	0.131 / 0.045	12.184 / 17.957
	5	0.178 / 0.055	17.185 / 27.854	5	0.110 / 0.040	4.387 / 4.537	5,2	0.110 / 0.040	4.323 / 4.442
	10	0.163 / 0.054	7.584 / 10.841	10	0.109 / 0.041	3.308 / 2.731	10,5	0.109 / 0.041	3.233 / 2.611
	20	0.156 / 0.057	3.861 / 3.795	20	0.111 / 0.043	2.761 / 1.743	20,5	0.112 / 0.043	2.726 / 1.688

TABLE IV: Average RPE over the Room dataset for different prior map types and algorithms. Units are in degrees and meters. Additionally the NEES and total time to process each image is reported.

	Algo.	40m	80m	120m	160m	200m	240m	NEES (ori / pos)	Time (ms)
VIO	-	0.373 / 0.088	0.536 / 0.119	0.636 / 0.141	0.717 / 0.163	0.811 / 0.175	0.888 / 0.187	3.228 / 3.796	0.8 $\pm$ 0.3
2D-to-2D	EKF	0.225 / 0.091	0.323 / 0.111	0.372 / 0.120	0.402 / 0.121	0.424 / 0.122	0.394 / 0.125	3.298 / 4.311	3.6 $\pm$ 1.8
	SKF	0.260 / 0.097	0.339 / 0.129	0.415 / 0.146	0.448 / 0.155	0.492 / 0.167	0.542 / 0.171	3.074 / 3.596	1.4 $\pm$ 0.7
	Inf. Meas.	0.276 / 0.099	0.353 / 0.134	0.449 / 0.152	0.518 / 0.163	0.531 / 0.173	0.562 / 0.180	3.016 / 3.647	0.9 $\pm$ 0.3
	Inf. Marg.	0.265 / 0.091	0.350 / 0.122	0.447 / 0.142	0.520 / 0.156	0.560 / 0.169	0.613 / 0.175	2.795 / 2.784	0.9 $\pm$ 0.3
	Inf. $\alpha\beta$	0.269 / 0.091	0.353 / 0.122	0.456 / 0.142	0.546 / 0.156	0.599 / 0.168	0.656 / 0.173	2.781 / 2.689	0.9 $\pm$ 0.3
2D-to-3D	EKF	0.041 / 0.009	0.041 / 0.009	0.041 / 0.009	0.041 / 0.009	0.041 / 0.009	0.041 / 0.009	9.612 / 7.792	5.8 $\pm$ 1.1
	SKF	0.090 / 0.040	0.092 / 0.038	0.091 / 0.040	0.090 / 0.038	0.092 / 0.039	0.091 / 0.039	3.051 / 2.963	1.4 $\pm$ 0.2
	Inf. Meas.	0.125 / 0.068	0.139 / 0.065	0.141 / 0.067	0.141 / 0.064	0.142 / 0.066	0.136 / 0.065	3.663 / 3.528	0.6 $\pm$ 0.1
	Inf. Marg.	0.102 / 0.046	0.103 / 0.045	0.102 / 0.046	0.098 / 0.044	0.103 / 0.046	0.100 / 0.045	3.201 / 2.546	0.6 $\pm$ 0.1
	Inf. $\alpha\beta$	0.102 / 0.047	0.103 / 0.046	0.102 / 0.047	0.098 / 0.045	0.103 / 0.046	0.100 / 0.046	3.126 / 2.437	0.6 $\pm$ 0.1

the same.[3] In Fig. 3, we additionally show the computational cost as we increase the average number of features and for different keyframe distance thresholds. The 2D-to-2D (KF) methods have near constant offset from the VIO time as the number of average features only marginally increases the computational cost due to more measurements. This is a clear advantage when the number of tracked features is large. The 2D-to-3D (PTS) method quickly increases an order of magnitude slower than VIO, which is expected as the state size dramatically grows (see Fig. 2). The inflation methods (INF) for both landmark and keyframe prior maps perform as efficiently as VIO due to their near constant run-time and constant state vector size.

E. Findings and Discussions

In summary, we have investigated through simulation the: relation between state size and the average number of features, achievable accuracy given different map priors, sensitivity of inflation methods to their tuning parameters, and how all methods compare in terms of accuracy, consistency, and computational cost for both 2D-to-3D landmark and 2D-to-2D keyframe maps.

We showed that even at extremely high noise levels, in general, the 2D-to-3D maps outperform the 2D-to-2D methods in accuracy. Keyframe maps have an computational advantage due to their state size when using a large number of features. The marginal and alpha beta covariance inflation methods are relatively invariant to their inflation parameters making them ideal for large environmental maps where EKF and SKF estimators become prohibitively expensive or the loss of consistency guarantees is acceptable.

Finally, we evaluated all methods against each other and make the following general recommendations: (1) the SKF should be used for small workspaces to ensure consistency and achieve high accuracy levels with low computational cost, (2) keyframe-based maps can be leveraged to reduce the computational cost while still reducing drift, (3) for large environments and map sizes, inflation methods can practically be leveraged with conservative inflation values.

[3]All timings were run on an Intel(R) Xeon(R) CPU E3-1505M v6 @ 3.00GHz processor in single threaded execution.

V. Conclusion

In this work we have revisited the map-based visual-inertial estimation problem in detail. A thorough investigation of 2D-to-3D landmark-based and 2D-to-2D keyframe-based prior maps was conducted. Different methods which incorporate this global information were presented and discussed. Simulation experiments were performed to show the achievable accuracy of estimators given different map priors, the sensitivities of inflation-based methods to their parameters, and how all variants compare in terms of their accuracy, consistency, and computational cost. We finally gave a series of general recommendations to leverage the SKF for small workspace consistent estimation, the use of keyframe-based maps to reduce state size and still limit navigation drift, and that inflation methods, after tuning, can be accuracy and efficient for large-scale environments. In the future we plan to investigate these methods in real-world scenarios.

REFERENCES

[1] G. Huang, "Visual-inertial navigation: A concise review," in *Proc. International Conference on Robotics and Automation*, Montreal, Canada, May 2019.

[2] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011.

[3] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Observability-constrained vision-aided inertial navigation," *University of Minnesota, Dept. of Comp. Sci. & Eng., MARS Lab, Tech. Rep*, vol. 1, p. 6, 2012.

[4] A. Kasyanov, F. Engelmann, J. Stückler, and B. Leibe, "Keyframe-based visual-inertial online slam with relocalization," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 6662–6669.

[5] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[6] ——, "Relocalization, global optimization and map merging for monocular visual-inertial slam," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1197–1204.

[7] R. Mur-Artal and J. D. Tardós, "Visual-inertial monocular slam with map reuse," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 796–803, 2017.

[8] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, 2021.

[9] G. Bresson, R. Aufrère, and R. Chapuis, "Making visual slam consistent with geo-referenced landmarks," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 553–558.

[10] H. Liu, M. Chen, G. Zhang, H. Bao, and Y. Bao, "Ice-ba: Incremental, consistent and efficient bundle adjustment for visual-inertial slam," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1974–1982.

[11] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[12] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 264–280, 2009.

[13] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg, "Global localization from monocular slam on a mobile phone," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 4, pp. 531–539, 2014.

[14] S. Middelberg, T. Sattler, O. Untzelmann, and L. Kobbelt, "Scalable 6-dof localization on mobile devices," in *European conference on computer vision*. Springer, 2014, pp. 268–283.

[15] S. Lynen, T. Sattler, M. Bosse, J. A. Hesch, M. Pollefeys, and R. Siegwart, "Get out of my lab: Large-scale, real-time visual-inertial localization." in *Robotics: Science and Systems*, vol. 1, 2015, p. 1.

[16] S. Lynen, B. Zeisl, D. Aiger, M. Bosse, J. Hesch, M. Pollefeys, R. Siegwart, and T. Sattler, "Large-scale, real-time visual–inertial localization revisited," *The International Journal of Robotics Research*, vol. 39, no. 9, pp. 1061–1084, 2020.

[17] S. F. Schmidt, "Application of state-space methods to navigation problems," in *Advances in control systems*. Elsevier, 1966, vol. 3, pp. 293–340.

[18] R. C. DuToit, J. A. Hesch, E. D. Nerurkar, and S. I. Roumeliotis, "Consistent map-based 3d localization on mobile devices," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 6253–6260.

[19] T. Ke, K. J. Wu, and S. I. Roumeliotis, "Rise-slam: A resource-aware inverse schmidt estimator for slam," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 354–361.

[20] Z. Huai and G. Huang, "Markov parallel tracking and mapping for probabilistic slam," in *Proc. of the IEEE International Conference on Robotics and Automation*, Xi'an, China, 2021.

[21] P. Geneva, J. Maley, and G. Huang, "An efficient schmidt-ekf for 3D visual-inertial SLAM," in *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, June 2019.

[22] P. Geneva, K. Eckenhoff, and G. Huang, "A linear-complexity EKF for visual-inertial navigation with loop closures," in *Proc. International Conference on Robotics and Automation*, Montreal, Canada, May 2019.

[23] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 3565–3572.

[24] M. Li and A. I. Mourikis, "High-precision, consistent ekf-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

[25] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: a research platform for visual-inertial estimation," in *Proc. of the IEEE International Conference on Robotics and Automation*, Paris, France, 2020. [Online]. Available: https://github.com/rpng/open_vins

[26] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep., Mar. 2005.

[27] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 1997.

[28] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *International Joint Conference on Artificial Intelligence*, Vancouver, BC, Aug. 1981, pp. 674–679.

[29] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[30] K. Sartipi, R. C. DuToit, C. B. Cobar, and S. I. Roumeliotis, "Decentralized visual-inertial localization and mapping on mobile devices for augmented reality," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2145–2152.

[31] D. S. Bayard and P. B. Brugarolas, "An estimation algorithm for vision-based exploration of small bodies in space," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005, pp. 4589–4595.

[32] G. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A first-estimates Jacobian EKF for improving SLAM consistency," in *Proc. of the 11th International Symposium on Experimental Robotics*, Athens, Greece, July 2008.

[33] ——, "Observability-based rules for designing consistent EKF SLAM estimators," *International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, Apr. 2010.

[34] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7244–7251.

[35] P. Geneva and G. Huang, "Map-based visual-inertial localization: A numerical study," University of Delaware, Tech. Rep. RPNG-2022-MAPPING, 2022. [Online]. Available: http://udel.edu/~ghuang/papers/tr_mapping.pdf

[36] C. Chen, Y. Yang, P. Geneva, and G. Huang, "FEJ2: a consistent visual-inertial state estimator design," in *Proc. International Conference on Robotics and Automation*, Philadelphia, USA, May 2022.