SchurVINS: Schur Complement-Based Lightweight Visual Inertial Navigation System

Yunfei Fan, Tianyu Zhao, Guidong Wang
PICO, ByteDance
{frank.01, zhaotianyu.1998, guidong.wang}@bytedance.com

Abstract

*Accuracy and computational efficiency are the most important metrics to Visual Inertial Navigation System (VINS). The existing VINS algorithms with either high accuracy or low computational complexity, are difficult to provide the high precision localization in resource-constrained devices. To this end, we propose a novel filter-based VINS framework named **S**chur**V**INS (SV), which could guarantee both high accuracy by building a complete residual model and low computational complexity with Schur complement. Technically, we first formulate the full residual model where Gradient, Hessian and observation covariance are explicitly modeled. Then Schur complement is employed to decompose the full model into ego-motion residual model and landmark residual model. Finally, Extended Kalman Filter (EKF) update is implemented in these two models with high efficiency. Experiments on EuRoC and TUM-VI datasets show that our method notably outperforms state-of-the-art (SOTA) methods in both accuracy and computational complexity. The experimental code of SchurVINS is available at* https://github.com/bytedance/SchurVINS.

1. Introduction

High-precision localization technologies have become a cornerstone in various industrial fields, playing an indispensable role particularly in robotics, augmented reality (AR), and virtual reality (VR). In recent decades, visual inertial navigation system (VINS) has attracted significant attentions due to its advantages of low-cost and ubiquitousness. Composed of only cameras and inertial measurement units (IMU), the VINS module can provide six-degree-of-freedom (6-DOF) positioning as accurate as expensive sensors such as Lidar, and is more competent in being installed in portable devices like smartphone and micro aerial vehicles (MAV).

It has been reported that kinds of excellent open-source VINS algorithms could achieve high-precision pose estimation, which mainly includes two methodologies: optimization-based and filter-based methods. Typical optimization-based methods [4, 17, 21, 24, 33, 34] model poses and the corresponding observed landmarks jointly. Benefitting from Schur complement technique [1], this high-dimensional model with special sparsity could be solved efficiently by bundle adjustment (BA [32]). In theory [11], although notable in high-precision of localization, optimization-based methods may suffer from high computational complexity. In contrast, mainstream filter-based methods [2, 7, 10, 30] derived from MSCKF [22] utilize the left nullspace method to simplify the residual model. EKF [29] update is then executed on the simplified residual model to estimate corresponding poses. Finally, they achieve high efficiency but compromise accuracy, since landmarks are not optimized with camera poses jointly and all observations are utilized only once. To sum up, optimization-based methods are advantageous in accuracy while filter-based methods are more efficient.

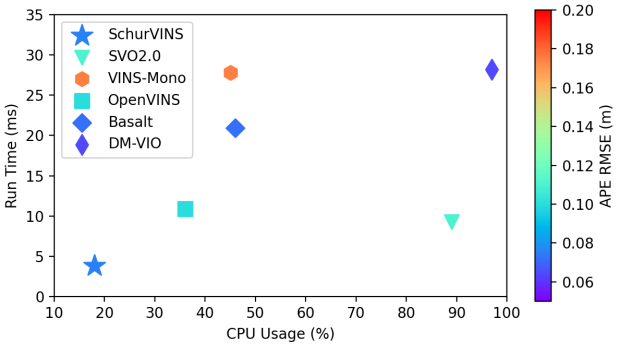

Figure 1. Comparison of run time, CPU usage and RMSE evaluated on EuRoC dataset. Different shapes and colors indicate different methods and precision, respectively.

Therefore, it is urgent to develop a framework combines their high precision and efficiency. As discussed

above, traditional residual model without simplification can achieve high accuracy. In spite of this, when both landmarks and poses are incorporated into the state vector for joint estimation, the efficiency of EKF-SLAM significantly decreases [22]. Inspired by the Schur complement in optimization-based methods, we make full use of the sparse structure inherent in the high-dimensional residual model constructed with poses and landmarks to achieve high efficiency in EKF. Thus, an EKF-based VINS framework that achieves both high efficiency and accuracy is presented. In the framework, the equivalent residual model, consisting of gradient, Hessian and the corresponding observation covariance, is derived based on the traditional residual model. Taking the special sparse structure of Hessian into account, Schur complement is carried out to break the equivalent residual equation into two smaller equations: equivalent pose residual model and equivalent landmark residual model. The equivalent landmark residual model is able to be further split into a collection of small equivalent residual models due to its own sparse structure. Finally, EKF update is implemented with the derived equivalent residual model to estimate the poses and corresponding landmarks jointly. As shown in Fig. 1, the resulting framework outperforms SOTA methods in latency, computational complexity and accuracy. Our main contributions are summarized as follows:

- An equivalent residual model is proposed to deal with hyper high-dimension observations, which consists of gradient, Hessian and the corresponding observation covariance. This method is of great generality in EKF systems.
- A lightweight EKF-based landmark solver is proposed to estimate position of landmarks with high efficiency.
- A novel EKF-based VINS framework is developed to achieve ego-motion and landmark estimation simultaneously with high accuracy and efficiency. The experimental code is published to benefit community.

2. Related Work

Improving the efficiency and accuracy is an ongoing effort for VINS algorithms. To date, significant research has been carried out to reduce the computational complexity and improve the precision.

Many VINS algorithms focus on efficiency improvement. Some studies reuse the intermediate results of previous optimization to decrease the amount of repetitive computation [14–16, 21]. While these approaches may yield a slight loss in accuracy, the computational process can be notably accelerated. Some other studies try to achieve high efficiency through engineering technologies. In [23, 36], efficient Hessian construction and Schur complement calculation is employed to improve cache efficiency and avoid redundant matrix representation. In [6, 35], variables are declared by single precision instead of traditional double precision to speed up the algorithm.

Besides efficiency, some studies concentrate on improving the accuracy. In [12, 13, 20], high accuracy is guaranteed through improving the consistency in EKF-based VINS. Some improved MSCKF namely Hybrid MSCKF [10, 18] (combined MSCKF and EKF-SLAM), proposed in recent to balance efficiency and accuracy, model informative landmarks selectively as part of their state variables to estimate jointly [19]. Some researchers construct the local bundle adjustment (LBA) running on other threads to reduce drift [4, 9]. However, LBA requires massive computational resources which might not be practical for implementation on small devices.

3. SchurVINS Framework

In this paper, the proposed SchurVINS is developed based on open-source SVO2.0 [8, 9] with stereo configuration, in which sliding window based EKF back-end is employed to replace the original back-end in SVO2.0, and EKF-based landmark solver is utilized to replace the original landmark optimizer. The framework of SchurVINS algorithm and the relationship between SVO and SchurVINS are shown in Fig. 2.

3.1. State Definition

Normally, for a traditional EKF-based VINS system [7, 10, 20], the basic IMU state is defined as:

$$\mathbf{x}_I = \begin{bmatrix} {}^G_I\mathbf{q}^\mathsf{T} & {}^G\mathbf{p}_I{}^\mathsf{T} & {}^G\mathbf{v}_I{}^\mathsf{T} & \mathbf{b}_a^\mathsf{T} & \mathbf{b}_g^\mathsf{T} \end{bmatrix}^\mathsf{T} \tag{1}$$

where $\{\mathrm{G}\}$, $\{\mathrm{I}\}$ and $\{\mathrm{C}\}$ are the global frame, local frame and camera frame, respectively. ${}^G\mathbf{p}_I$ and ${}^G\mathbf{v}_I$ are the position and velocity of IMU expressed in $\{\mathrm{G}\}$, respectively. ${}^G_I\mathbf{q}$ represents the rotation quaternion from $\{\mathrm{I}\}$ to $\{\mathrm{G}\}$ (in this paper, quaternion obeys Hamilton rules [29]). The vectors $\mathbf{b}_a$ and $\mathbf{b}_g$ individually represent the biases of the angular velocity and linear acceleration measured by the IMU device. And thus the corresponding EKF error-state of $\mathbf{x}_I$ is defined as Eq. (2)

$$\tilde{\mathbf{x}}_I = \begin{bmatrix} {}^G_I\tilde{\boldsymbol{\theta}}^\mathsf{T} & {}^G\tilde{\mathbf{p}}_I{}^\mathsf{T} & {}^G\tilde{\mathbf{v}}_I{}^\mathsf{T} & \tilde{\mathbf{b}_a}^\mathsf{T} & \tilde{\mathbf{b}_g}^\mathsf{T} \end{bmatrix}^\mathsf{T} \tag{2}$$

where, ${}^G_I\tilde{\boldsymbol{\theta}}$ represents the error-state of ${}^G_I\mathbf{q}$. Except for quaternion, other states can be used with standard additive error (e.g. $\mathbf{x} = \hat{\mathbf{x}} + \tilde{\mathbf{x}}$). Similar to [29], the extended additive error of quaternion is defined as Eq. (3) (in this paper, quaternion error is defined in frame $\{G\}$)

$$\mathbf{q}^G_I = \delta^G_I\mathbf{q} \otimes {}^G_I\hat{\mathbf{q}}, \quad \delta^G_I\mathbf{q} = \begin{bmatrix} 1 & \frac{1}{2}\delta^G_I\tilde{\boldsymbol{\theta}} \end{bmatrix}^\mathsf{T} \tag{3}$$

Similarly, the extended additive error of rotation matrix is defined as Eq. (4)

$$\mathbf{R}({}^G_I\mathbf{q}) = {}^G_I\mathbf{R}, \quad {}^G_I\mathbf{R} = \left(\mathbf{I} + \left[{}^G_I\tilde{\boldsymbol{\theta}}\right]_\times\right) {}^G_I\hat{\mathbf{R}} \tag{4}$$

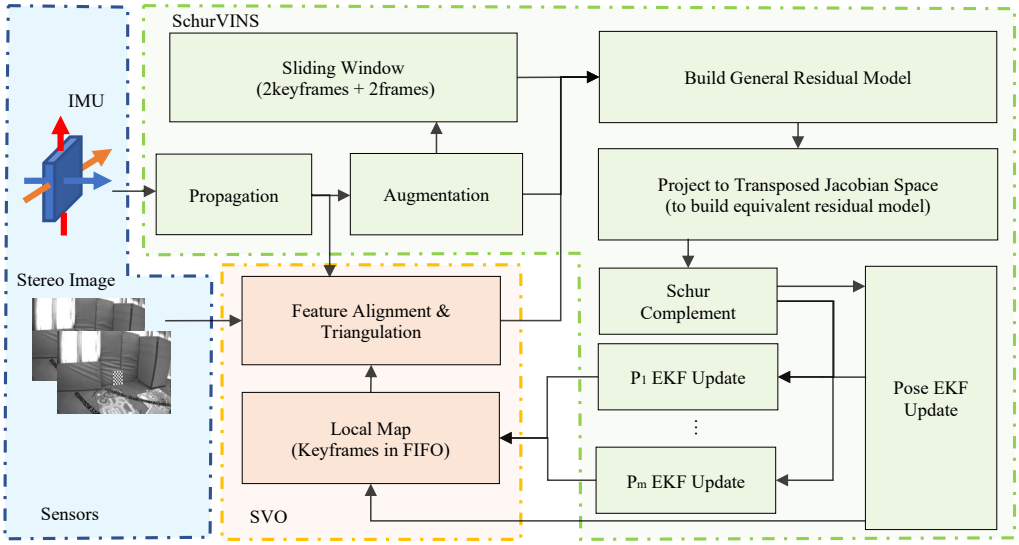

Figure 2. Framework of SchurVINS, which shows the relationship between SVO and SchurVINS. P_1 to P_m represent the valid landmarks of the surrounding environment which are employed to construct residual model.

3.2. Propagation and Augmentation

SchurVINS follows the policy introduced in [29] on state propagation. The time evolution of IMU states are described as

$$ {}^{G}_{I}\dot{\hat{\mathbf{q}}} = \tfrac{1}{2}{}^{G}_{I}\hat{\mathbf{q}} \otimes \Omega(\hat{\boldsymbol{\omega}}), \quad \Omega(\hat{\boldsymbol{\omega}}) = \begin{pmatrix} 0 & -\hat{\boldsymbol{\omega}}^{\mathsf{T}} \\ \hat{\boldsymbol{\omega}} & -[\hat{\boldsymbol{\omega}}]_{\times} \end{pmatrix} \tag{5} $$

$$ \dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3\times 1}, \quad \dot{\hat{\mathbf{b}}}_a = \mathbf{0}_{3\times 1} \tag{6} $$

$$ {}^{G}\dot{\hat{\mathbf{p}}}_I = {}^{G}\mathbf{v}_I, \quad {}^{G}\dot{\hat{\mathbf{v}}}_I = {}^{G}_{I}\hat{\mathbf{R}}\hat{\mathbf{a}} + {}^{G}\mathbf{g} \tag{7} $$

where $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g$ and $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$ are IMU measurements with biases discarded. where $[\hat{\boldsymbol{\omega}}]_{\times}$ is skew symmetric matrix of $\hat{\boldsymbol{\omega}}$. Based on Eqs.(5) to (7), the linearized continuous dynamics for the error IMU state is defined as

$$ \dot{\tilde{\mathbf{x}}}_I = \mathbf{F}\tilde{\mathbf{x}}_I + \mathbf{G}\mathbf{n}_I \tag{8} $$

where $\mathbf{n}_I = \begin{bmatrix} \mathbf{n}_a{}^{\mathsf{T}} & \mathbf{n}_{a\omega}{}^{\mathsf{T}} & \mathbf{n}_g{}^{\mathsf{T}} & \mathbf{n}_{g\omega}{}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$. Vectors $\mathbf{n}_a$ and $\mathbf{n}_g$ represent the Gaussian noise of the accelerometer and gyroscope measurement, while $\mathbf{n}_{a\omega}$ and $\mathbf{n}_{g\omega}$ are the random walk rate of the accelerometer and gyroscope measurement biases. $\mathbf{F}$ and $\mathbf{G}$ are defined as

$$ \mathbf{F} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -{}^{G}_{I}\mathbf{R} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -[{}^{G}_{I}\mathbf{R}\hat{\mathbf{a}}]_{\times} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -{}^{G}_{I}\mathbf{R} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} \end{bmatrix} \tag{9} $$

$$ \mathbf{G} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -{}^{G}_{I}\mathbf{R}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -{}^{G}_{I}\mathbf{R}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{10} $$

4^{th} Runge-Kutta numerical integration method is employed in Eqs. (3) to (7) for propagating the estimated IMU state. Based on Eq. (8), the discrete time state transition matrix $\boldsymbol{\Phi}$ and discrete time noise covariance $\mathbf{Q}$ are formulated as follows:

$$ \begin{aligned} \boldsymbol{\Phi} &= \mathbf{I}_{15\times15} + \mathbf{F}dt + \mathbf{F}^2dt^2 + \mathbf{F}^3dt^3 \\ \mathbf{Q} &= \boldsymbol{\Phi}\mathbf{G}\mathbf{Q}_I\mathbf{G}^{\mathsf{T}}\boldsymbol{\Phi}^{\mathsf{T}}dt \end{aligned} \tag{11} $$

where $\mathbf{Q}_I = E[\mathbf{n}_I\mathbf{n}_I{}^{\mathsf{T}}]$ is the continuous time noise covariance matrix of the system. Hence, the formulations of covariance propagation are built as:

$$ \mathbf{P}_{II} \leftarrow \boldsymbol{\Phi}\mathbf{P}_{II}\boldsymbol{\Phi}^{\mathsf{T}} + \mathbf{Q}, \quad \mathbf{P}_{IA} \leftarrow \boldsymbol{\Phi}\mathbf{P}_{IA} \tag{12} $$

The covariance $\mathbf{P}$ is partitioned as Eq. (13). $\mathbf{P}_{II}$ is the covariance of basic state. $\mathbf{P}_{IA}$ and $\mathbf{P}_{AI}$ is the covariance between basic state and augmented state. $\mathbf{P}_{AA}$ is covariance of the augmented state.

$$ \mathbf{P} = \begin{bmatrix} \mathbf{P}_{II} & \mathbf{P}_{IA} \\ \mathbf{P}_{IA}^{\mathsf{T}} & \mathbf{P}_{AA} \end{bmatrix} \tag{13} $$

When a new image arrives, the current IMU pose $\mathbf{x}_{Ai} = \begin{bmatrix} {}^{G}_{I}\mathbf{q}^{\mathsf{T}} & {}^{G}\mathbf{p}_I{}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$ is augmented as well as its covariance. The augmentation formulations are:

$$ \begin{aligned} \mathbf{X} &= \begin{bmatrix} \mathbf{x}_I{}^{\mathsf{T}} & \mathbf{x}_{A0}{}^{\mathsf{T}} & \mathbf{x}_{A1}{}^{\mathsf{T}} & \cdots & \mathbf{x}_{Ai}{}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \\ \mathbf{P} &\leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}_{21}{}^{\mathsf{T}} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \end{aligned} \tag{14} $$

where $\mathbf{P}_{21} = \mathbf{J}_a\mathbf{P}$, $\mathbf{P}_{22} = \mathbf{J}_a\mathbf{P}\mathbf{J}_a{}^{\mathsf{T}}$. And $\mathbf{J}_a$ is the Jacobian of $\tilde{\mathbf{x}}_{Ai}$ with respect to error states, which is defined as

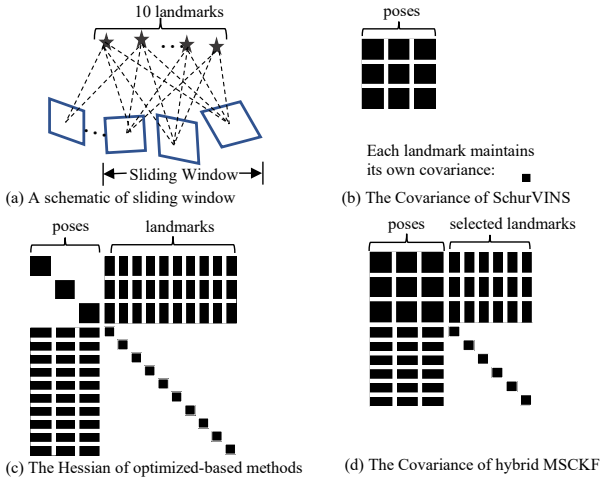

Figure 3. A schematic of our system for ten landmarks and the sliding window of size three shown in (a), and the Hessian or Covariance of different methods shown in (b)-(d). (b) shows our algorithm in which the covariance of every single landmark is independent from the entire covariance of poses in the sliding window. (c) demonstrates the Hessian of both landmarks and poses in the sliding window. (d) demonstrates traditional hybrid MSCKF with the Covariance of both selected landmarks and poses in the sliding window.

follows:

$$\mathbf{J} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times(9+6N)} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times(9+6N)} \end{bmatrix} \tag{15}$$

3.3. Schur Complement-Based State Update

In the SchurVINS scheme, unlike MSCKF methods [10, 30], the EKF update is conducted based on all the successfully triangulated landmarks and their observations in the sliding window, which can eliminate the drift caused by state propagation in every single image timestamp interval as much as possible. For single observation, the reprojection error $\mathbf{r}_{i,j}$ of camera measurement is formulated as:

$$\begin{aligned} \mathbf{r}_{i,j} &= \mathbf{z}_{i,j} - \hat{\mathbf{z}}_{i,j} \\ \mathbf{r}_{i,j} &= \mathbf{J}_{x,i,j}\tilde{\mathbf{X}} + \mathbf{J}_{f_j}{}^{G}\tilde{\mathbf{p}}_{f_j} + \mathbf{n}_{i,j} \\ \hat{\mathbf{z}}_{i,j} &= \frac{1}{{}^{C_i}\hat{Z}_j}\begin{bmatrix} {}^{C_i}\hat{X}_j \\ {}^{C_i}\hat{Y}_j \end{bmatrix} \end{aligned} \tag{16}$$

where $\mathbf{r}_{i,j}$ and $\mathbf{z}_{i,j}$ are the reprojection error and the camera measurement of j^{th} landmark at i^{th} pose in sliding window, respectively, and $\hat{\mathbf{z}}_{i,j}$ is the corresponding theoretical measurement formulated by estimated states. ${}^{C_i}\mathbf{p}_j = \begin{bmatrix} {}^{C_i}\hat{X}_j & {}^{C_i}\hat{Y}_j & {}^{C_i}\hat{Z}_j \end{bmatrix}$ is the landmark coordinate in camera pose of i^{th} sliding window. $\mathbf{n}_{i,j}$ represents the corresponding measurement standard deviation (or measurement noise). $\tilde{\mathbf{X}}$ and ${}^{G}\tilde{\mathbf{p}}_{f_j}$ are respectively the state perturbation and landmark position perturbation. $\mathbf{J}_{x,i,j}$ and $\mathbf{J}_{f_i}$ are the Jacobians of residual with respect to system state and landmark position, respectively. The Jacobians are defined as follows:

$$\begin{aligned} \mathbf{J}_{x,i,j} &= \begin{bmatrix} \mathbf{0}_{2\times(15+6i)} & \mathbf{J}_A & \mathbf{0}_{2\times6(N-i-1)} \end{bmatrix} \\ \mathbf{J}_A &= \mathbf{J}_{i,j}\begin{bmatrix} {}^{I}_{C}\hat{\mathbf{R}}^{\mathsf{T}}[{}^{I_i}\hat{\mathbf{p}}_{f_j}]_\times {}^{G}_{I_i}\hat{\mathbf{R}}^{\mathsf{T}} & -{}^{G}_{C_i}\hat{\mathbf{R}}^{\mathsf{T}} \end{bmatrix} \\ \mathbf{J}_{f_j} &= \begin{bmatrix} \mathbf{J}_{i,j}{}^{G}_{C_i}\hat{\mathbf{R}}^{\mathsf{T}} \end{bmatrix} \end{aligned} \tag{17}$$

where, for convenience, we define the camera model using the pinhole model. Therefore, $\mathbf{J}_{i,j}$ is defined as:

$$\mathbf{J}_{i,j} = \frac{1}{{}^{C_i}\hat{Z}_j{}^2}\begin{bmatrix} {}^{C_i}\hat{Z}_j & 0 & -{}^{C_i}\hat{X}_j \\ 0 & {}^{C_i}\hat{Z}_j & -{}^{C_i}\hat{Y}_j \end{bmatrix} \tag{18}$$

Aiming at all the observations of landmarks in the sliding window, we can acquire the full residual model by stacking all the residual equations:

$$\mathbf{r} = \begin{bmatrix} \mathbf{J}_x & \mathbf{J}_f \end{bmatrix}\begin{bmatrix} \tilde{\mathbf{X}} \\ {}^{G}\tilde{\mathbf{p}}_f \end{bmatrix} + \mathbf{n}, \quad \mathbf{n} = \begin{bmatrix} u, u, u, \cdots, u \end{bmatrix}^{\mathsf{T}} \tag{19}$$

where, $\mathbf{r}$ and $\begin{bmatrix} \mathbf{J}_x & \mathbf{J}_f \end{bmatrix}$ are respectively the stacked residual and stacked Jacobian. $\mathbf{J}_x$ and $\mathbf{J}_f$ are jacobian with respect to states and landmark positions, respectively. $\mathbf{n}$ is the stacked measurement noise, and the measurement covariance of $\mathbf{n}$ is $\mathbf{R} = diag(u^2, u^2, \cdots, u^2)$, where u is the element of standard deviation of $\mathbf{n}$.

Unlike [7, 10, 30], in this paper, the residual model Eq. (19) is projected into the jacobian space $\begin{bmatrix} \mathbf{J}_x & \mathbf{J}_f \end{bmatrix}^{\mathsf{T}}$ for formulating equivalent residual equations, which consist of gradient and hessian and observation covariance shown in Eqs. (20) and (21) below. It is worth highlighting that this strategy is an alternative to QR decomposition strategy [22] for speeding-up in any EKF systems with high-dimensional measurements.

$$\begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}} \\ \mathbf{J}_f{}^{\mathsf{T}} \end{bmatrix}\mathbf{r} = \begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}} \\ \mathbf{J}_f{}^{\mathsf{T}} \end{bmatrix}\begin{bmatrix} \mathbf{J}_x & \mathbf{J}_f \end{bmatrix}\begin{bmatrix} \tilde{\mathbf{X}} \\ {}^{G}\tilde{\mathbf{P}}_f \end{bmatrix} + \mathbf{n}' \tag{20}$$

$$\mathbf{R}' = \begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}} \\ \mathbf{J}_f{}^{\mathsf{T}} \end{bmatrix}\mathbf{R}\begin{bmatrix} \mathbf{J}_x & \mathbf{J}_f \end{bmatrix} \tag{21}$$

where $\mathbf{n}'$ and $\mathbf{R}'$ are the equivalent observation noise and covariance, respectively. Obviously, Eqs. (20) and (21) could be simplified as:

$$\underbrace{\begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}}\mathbf{r} \\ \mathbf{J}_f{}^{\mathsf{T}}\mathbf{r} \end{bmatrix}}_{\begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}} = \underbrace{\begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}}\mathbf{J}_x & \mathbf{J}_x{}^{\mathsf{T}}\mathbf{J}_f \\ \mathbf{J}_f{}^{\mathsf{T}}\mathbf{J}_x & \mathbf{J}_f{}^{\mathsf{T}}\mathbf{J}_f \end{bmatrix}}_{\begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \\ \mathbf{C}_2{}^{\mathsf{T}} & \mathbf{C}_3 \end{bmatrix}}\begin{bmatrix} \tilde{\mathbf{X}} \\ {}^{W}\tilde{\mathbf{P}}_f \end{bmatrix} + \underbrace{\mathbf{n}'}_{\begin{bmatrix} \mathbf{n}'_1 \\ \mathbf{n}'_2 \end{bmatrix}} \tag{22}$$

$$\mathbf{R}' = \begin{bmatrix} \mathbf{J}_x{}^{\mathsf{T}}\mathbf{J}_x & \mathbf{J}_x{}^{\mathsf{T}}\mathbf{J}_f \\ \mathbf{J}_f{}^{\mathsf{T}}\mathbf{J}_x & \mathbf{J}_f{}^{\mathsf{T}}\mathbf{J}_f \end{bmatrix}u^2 \tag{23}$$

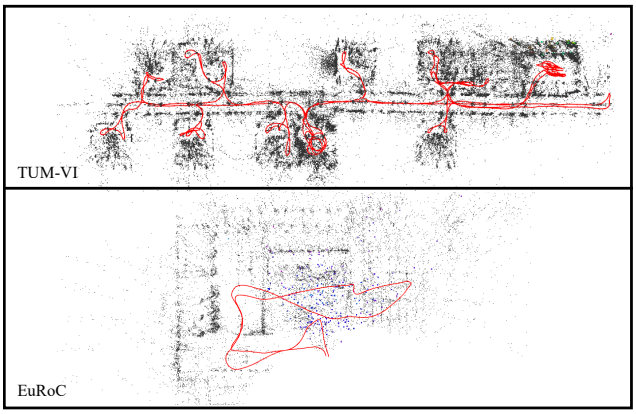

Figure 4. The experimental trajectory and point cloud of SchurVINS on TUM-VI and EuRoC datasets.

Since ${}^{G}\tilde{\mathbf{P}}_f$ is not included in the states in Eq. (14), it is necessary to employ Schur complement [28] on Eqs. (20) and (21) to marginalize the implicit states. To be straightforward, Eqs. (22) and (23) should be projected into $\mathbf{L}$ space as Eqs. (24) and (25).

$$\mathbf{L}\begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{bmatrix} = \mathbf{L}\begin{bmatrix}\mathbf{C}_1 & \mathbf{C}_2\\ \mathbf{C}_2{}^{\mathsf{T}} & \mathbf{C}_3\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{X}}\\ {}^{W}\tilde{\mathbf{P}}_f\end{bmatrix} + \begin{bmatrix}\mathbf{n}_1''\\ \mathbf{n}_2''\end{bmatrix} \tag{24}$$

$$\mathbf{R}'' = \mathbf{L}\begin{bmatrix}\mathbf{C}_1 & \mathbf{C}_2\\ \mathbf{C}_2{}^{\mathsf{T}} & \mathbf{C}_3\end{bmatrix}\mathbf{L}^{\mathsf{T}}u^2 = \begin{bmatrix}\mathbf{R}_1'' & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_2''\end{bmatrix} \tag{25}$$

where $\begin{bmatrix}\mathbf{n}_1''^{\mathsf{T}} & \mathbf{n}_2''^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}}$ and $\mathbf{R}''$ are the derived observation noise and covariance. And $\mathbf{L}$ is defined as:

$$\mathbf{L} = \begin{bmatrix}\mathbf{I} & -\mathbf{C}_2\mathbf{C}_3^{-1}\\ \mathbf{0} & \mathbf{I}\end{bmatrix} \tag{26}$$

Substituting Eq. (26) into Eqs. (24) and (25) yields the simplified formulations:

$$\begin{bmatrix}\mathbf{b}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{b}_2\\ \mathbf{b}_2\end{bmatrix} = \mathbf{C}\begin{bmatrix}\tilde{\mathbf{X}}\\ {}^{W}\tilde{\mathbf{P}}_f\end{bmatrix} + \begin{bmatrix}\mathbf{n}_1''\\ \mathbf{n}_2''\end{bmatrix} \tag{27}$$

$$\mathbf{R}'' = \begin{bmatrix}(\mathbf{C}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{C}_2^{\mathsf{T}}) & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_3\end{bmatrix}u^2 \tag{28}$$

where

$$\mathbf{C} = \begin{bmatrix}(\mathbf{C}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{C}_2^{\mathsf{T}}) & \mathbf{0}\\ \mathbf{C}_2^{\mathsf{T}} & \mathbf{C}_3\end{bmatrix} \tag{29}$$

Eqs. (27) and (28) could be decomposed into Eqs. (30) to (31) and Eqs. (32) to (33) as follows:

$$\begin{aligned}\left[\mathbf{b}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{b}_2\right] &= \left[\mathbf{C}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{C}_2^{\mathsf{T}}\right]\tilde{\mathbf{X}} + \mathbf{n}_1'' && (30)\\ \mathbf{R}_1'' &= \left[\mathbf{C}_1 - \mathbf{C}_2\mathbf{C}_3^{-1}\mathbf{C}_2^{\mathsf{T}}\right]u^2 && (31)\\ \left[\mathbf{b}_2 - \mathbf{C}_2^{\mathsf{T}}\tilde{\mathbf{X}}\right] &= \left[\mathbf{C}_3\right]{}^{W}\tilde{\mathbf{P}}_f + \mathbf{n}_2'' && (32)\\ \mathbf{R}_2'' &= \left[\mathbf{C}_3\right]u^2 && (33)\end{aligned}$$

Obviously, Eqs. (30) and (31) are equivalent residual equation and observation noise covariance. They could be substituted into standard EKF model Eqs. (34) and (37) to conduct state update directly.

$$\begin{aligned}\mathbf{K} &= \mathbf{P}\mathbf{J}^{\mathsf{T}}(\mathbf{J}\mathbf{P}\mathbf{J}^{\mathsf{T}} + \mathbf{R})^{-1} && (34)\\ \Delta\mathbf{x} &= \mathbf{K}\mathbf{r} && (35)\\ \mathbf{P} &= (\mathbf{I} - \mathbf{K}\mathbf{J})\mathbf{P}(\mathbf{I} - \mathbf{K}\mathbf{J})^{\mathsf{T}} + \mathbf{K}\mathbf{R}\mathbf{K}^{\mathsf{T}} && (36)\\ \mathbf{x} &= \mathbf{x} \oplus \Delta\mathbf{x} && (37)\end{aligned}$$

3.4. EKF-based Landmark Solver

$\tilde{\mathbf{X}}$ can be obtained by substituting Eqs. (30) and (31) into Eqs. (34) to (37). Then, the resulting $\tilde{\mathbf{X}}$ could be substituted into Eq. (32) to establish the landmark equivalent residual equation

$$\begin{bmatrix}\mathbf{r}_1\\ \mathbf{r}_2\\ \vdots\\ \mathbf{r}_m\end{bmatrix} = \begin{bmatrix}\mathbf{C}_{3_1} & & & \\ & \mathbf{C}_{3_2} & & \\ & & \ddots & \\ & & & \mathbf{C}_{3_m}\end{bmatrix}\begin{bmatrix}{}^{W}\tilde{\mathbf{P}}_{f_1}\\ {}^{W}\tilde{\mathbf{P}}_{f_2}\\ \vdots\\ {}^{W}\tilde{\mathbf{P}}_{f_m}\end{bmatrix} + \mathbf{n}_2'' \tag{38}$$

where $\mathbf{C}_{3_1}, \cdots, \mathbf{C}_{3_m}$ are diagonal elements of $\mathbf{C}_3$ clarified in Eq. (22). And the corresponding covariance $\mathbf{R}_2''$ is:

$$\mathbf{R}_2'' = \begin{bmatrix}\mathbf{C}_{3_1}u^2 & & & \\ & \mathbf{C}_{3_2}u^2 & & \\ & & \ddots & \\ & & & \mathbf{C}_{3_m}u^2\end{bmatrix} \tag{39}$$

Benefited from the sparsity of the resulting landmark equivalent residual equation, Eqs. (38) and (39) is split as a bunch of small independent residual models, shown as Eq. (40), which allows the EKF update of each landmark to conduct one by one. This significantly reduces the computational complexity.

$$\begin{aligned}\left[\mathbf{r}_i\right] &= \left[\mathbf{C}_{3_i}\right]\left[{}^{W}\tilde{\mathbf{P}}_{f_i}\right] + \mathbf{n}_{2_i}'', i = 1, \cdots, m\\ \mathbf{R} &= \left[\mathbf{C}_{3_i}u^2\right]\end{aligned} \tag{40}$$

3.5. Frontend

Our code implementation makes full use of SVO2.0 as the front-end of SchurVINS. The integrated components of SchurVINS include feature alignment and depth-filter modules from original SVO2.0. Meanwhile, sparse image alignment module is replaced by the proposed EKF propagation scheme to guarantee delivering an accurate pose to feature alignment module. Compared with frame-to-frame feature tracking[10, 24, 30], the strategy of feature alignment, implemented by projecting and matching the co-visible landmarks from local map to frames, achieves excellent long-term landmark tracking performance due to

Sequence	S/M[1]	F/O[2]	MH1	MH2	MH3	MH4	MH5	V11	V12	V13	V21	V22	Avg
OKVIS[4][17]	M	O	0.160	0.220	0.240	0.340	0.470	0.090	0.200	0.240	0.130	0.160	0.225
VINS-mono[24]	M	O	0.150	0.150	0.220	0.320	0.300	0.079	0.110	0.180	0.080	0.160	0.174
Kimera[26]	S	O	0.110	0.100	0.160	0.240	0.350	0.050	0.080	0.070	0.080	0.100	0.134
ICE-BA[21]	S	O	0.090	0.070	0.110	0.160	0.270	0.050	0.050	0.110	0.120	0.090	0.112
SVO2.0[5][9]	S	O	0.080	0.080	0.088	0.211	0.231	0.052	0.082	0.073	0.084	0.116	0.109
BASALT[33]	S	O	0.070	0.060	**0.070**	0.130	0.110	0.040	0.050	0.100	0.040	0.050	0.072
DM-VIO[34]	M	O	0.065	**0.044**	0.097	**0.102**	**0.096**	0.048	**0.045**	0.069	**0.029**	0.050	**0.064**
MSCKF[4][22]	S	F	0.420	0.450	0.230	0.370	0.480	0.340	0.200	0.670	0.100	0.160	0.342
ROVIO[4][2]	M	F	0.210	0.250	0.250	0.490	0.520	0.100	0.100	0.140	0.120	0.140	0.232
OpenVINS-4[5][10][3]	S	F	0.084	0.084	0.127	0.218	0.360	0.038	0.054	**0.050**	0.064	0.061	0.114
OpenVINS[5][10]	S	F	0.072	0.143	0.086	0.173	0.247	0.055	0.060	0.059	0.054	**0.047**	0.096
SV(ours)[5]	S	F	**0.049**	0.077	0.086	0.125	0.125	**0.035**	0.053	0.082	0.046	0.075	0.075

[1] S and M mean stereo and monocular methods, respectively.
[2] F and O mean filter-based and optimization-based methods, respectively.
[3] OpenVINS-4 means that the maximum size of the sliding window in OpenVINS is configured to be 4.
[4] results taken from [5].
[5] evaluated by author manually.
All other results are taken from the respective paper.

Table 1. Accuracy evaluation of various mono and stereo VINS algorithms on EuRoC. In the upper part, we summarize the results for the optimization-based methods that run sliding window optimization to estimate pose. In the lower part, we evaluate the results of filter-based methods. Best result in bold, underline is the best result among filter-based methods. SchurVINS achieves the lowest average APE RMSE in filter-based methods and surpasses the majority of optimization-based methods.

Sequence	S/M	F/O	c1	c2	c3	c4	c5	r1	r2	r3	r4	r5	r6	Avg
VINS-Mono[1]	M	O	0.630	0.950	1.560	0.250	0.770	0.070	**0.070**	0.110	0.040	0.200	0.080	0.430
OKVIS[1]	M	O	0.330	0.470	0.570	0.260	0.390	0.060	0.110	0.070	0.030	0.070	0.040	0.218
BASALT[1]	S	O	0.340	0.420	0.350	0.210	0.370	0.090	**0.070**	0.130	0.050	0.130	**0.020**	0.198
DM-VIO[1]	M	O	**0.190**	0.470	**0.240**	**0.130**	**0.160**	**0.030**	0.130	0.090	0.040	0.060	**0.020**	**0.141**
ROVIO[1]	M	F	0.470	0.750	0.850	**0.130**	2.090	0.160	0.330	0.150	0.090	0.120	0.050	0.471
OpenVINS[2]	S	F	0.413	0.322	1.536	0.186	0.644	0.062	0.093	0.079	**0.027**	0.074	**0.020**	0.314
SV[2]	S	F	0.329	**0.285**	0.555	0.162	0.274	0.048	0.160	**0.066**	0.049	**0.054**	0.021	0.182

[1] results taken from [34].
[2] evaluated by author manually.

Table 2. Accuracy evaluation on TUM-VI datasets. c1 to c5 denote corridor1 to corridor5 in TUM-VI datasets. r1 to r6 denote room1 to room6 in TUM-VI datasets. Best result in bold, underline is the best result among filter-based methods.

the fact that the lost landmarks in short time is capable to be tracked again. Depth-filter is utilized to execute landmark position initialization. Once the landmark is initialized sufficiently, it would be transferred to the proposed EKF-based landmark solver to proceed estimation with sliding window jointly.

Based on First In First Out (FIFO) strategy, local map only maintains the most recent ten keyframes to support landmark tracking. Since high accuracy is already achieved, the traditional LBA is no longer necessary, which is abandoned in the proposed SchurVINS.

3.6. Keyframe Selection

The strategy of keyframe selection is important in VINS system. There are three strategies to select keyframes in SchurVINS. If the average parallax between the candidate frame and the previous keyframe reaches the threshold or the count of tracked landmarks drops below the certain threshold, the corresponding frame is defined as keyframe. Once the keyframe is selected, the FAST corners [31] are extracted to generate new landmarks via depth-filter module. Additionally, when the gap in both orientation and position between the candidate frame and the co-visible keyframes maintained in the local map is out of the certain range, the keyframe would be determined, by which the

	Avg CPU	Std CPU	Speed
DM-VIO	98/172[1]	-/30	1x/1.76x
BASALT	46/203[1]	-/46	1x/4.37x
VINS-Mono	45	13	1x
OpenVINS	37	10	1x
OpenVINS-4	32	8	1x
SMSCKF[30]	25	4	1x
SVO2.0	89	20	1x
SVO2.0-wo[2]	17	6	1x
SV	18	6	1x

[1] The 1x evaluation results of DM-VIO and BASALT are the converted results by author manually.
[2] SVO2.0-wo means SVO2.0 without the enabled LBA.

Table 3. Evaluation of CPU overhead for different wellknown VINS algorithms. GBA, PGO and LC are disabled on all the mentioned algorithms, with the exception of SVO2.0, which has the LBA module enabled. Our method provides a notable improvement in efficiency compared to the SOTA VINS algorithms.

	SV	SV-GN[1]	SVO2.0-wo[2]	SVO2.0	SMSCKF[30]	OpenVINS	OpenVINS-4
SparseImageAlign	-	-	1.35	1.43	-	-	-
FeatureAlign	1.39	1.39	1.79	1.91	-	-	-
KLT	-	-	-	-	2.63	2.69	2.67
Propagation	0.11	0.11	-	-	0.55	0.21	0.18
optimizePose	0.67	0.67	0.48	-	3.16	0.99/4.30[2]	0.34/2.46[2]
optimizeStructure	0.11	0.42	0.07	-	-	0.93	0.44
LBA	-	-	-	26.3[3]	-	-	-
Total time[4]	3.83	4.11	3.77	9.28	8.53	10.91	7.89

[1] denotes SchurVINS with Gauss-Newton optimization-based (GN-based) landmark estimation as originally used in SVO2.0.
[2] Running time of MSCKF update and SLAM update.
[3] It contains some running time of SVO2.0 LBA in asynchronous thread.
[4] The total time also contains other modules.

Table 4. Running time evaluation of the main parts of SchurVINS compared with SVO2.0 and OpenVINS on EuRoC MH01 (mean time in ms). Note that the different overhead of optimizeStructure between SVO-NonBA and SchurVINS-GN is primarily attributed to the variation in the count of feature matches, which is a consequence of the localization accuracy.

tracking module could overcome divergence between the candidate frame and the local map.

4. Experiments

The accuracy and efficiency of SchurVINS algorithms are evaluated by two experiments. And the additional ablation experiment is carried out to demonstrate the effectiveness of the proposed EKF-based landmark solver.

System Configuration: We have developed SchurVINS based on the open source code repository of SVO2.0, specifically, svo_pro_open. The majority of system parameters are not required to be modified. For high efficiency, edgelet features, loop closure (LC), pose graph optimization (PGO), LBA and Global BA (GBA) are discarded or deactivated. For our experiments below, we have configured the threshold on the quantity of keyframes in the local map to a maximum of ten. This local map mainly maintains co-visible keyframes and landmarks to achieve feature alignment. In the backend of SchurVINS, there is a sliding window consists of 2 old keyframes and 2 latest temporal frames. The keyframe strategy is similar to original SVO2.0.

4.1. Accuracy

The overall accuracy of the mentioned algorithms is evaluated using Root Mean Square Error (RMSE) on two wellknown datasets, EuRoC [3] and TUM-VI [27]. The corresponding experimental trajectory and point cloud of SchurVINS on TUM-VI and EuRoC datasets are shown on Fig. 4. To prevent the fluctuation of the algorithm from causing unreasonable evaluation results, our own evaluation method is to run the algorithm for 7 rounds, remove the maximum and minimum values, and then calculate the average of the remaining results as the evaluation result. In Tab. 1, our method obtains the lowest average RMSE in filter-based methods reported on the dataset so far, as well as outperforms the majority of optimization-based methods. Besides, our approach obtains the similar accuracy with wellknown optimization-based method BASALT and slightly lower accuracy than the recent competitor DM-VIO. Besides, the well-known VINS algorithms, VINS-Fusion [24] and SMSCKF [30], are not included in Tab. 1, since VINS-mono and OpenVINS surpass VINS-Fusion and SMSCKF in terms of accuracy, respectively [10, 25]. The re-evaluation experiment in Tab. 2 is within expectation absolutely. It is worth highlighting that, although degrading in accuracy slightly compared with the two optimization-based competitors, our method achieves obviously lower computational complexity than both of them with details in the next subsection.

4.2. Efficiency

The efficiency evaluations are carried out on Intel i7-9700 (3.00GHZ) desktop platform. Global BA (GBA), pose graph optimization and loop closure are disabled on all of the following algorithms. Besides, LBA is only enabled on the original SVO2.0. The efficiency experiment is divided into two parts: profiling processor usage and overhead time, which are reported in Tab. 3 and Tab. 4, respectively.

As demonstrated in Tab. 3, SchurVINS achieves almost the lowest processor usage compared with all the mentioned VINS algorithms. Especially, SVO2.0-wo requires similar cpu usage with SchurVINS, but it suffers from notable inaccuracy since it is almost pure Visual Odometry (VO). To thoroughly investigate the underlying reasons contributing to the efficiency advantages of SchurVINS, the experiment to meticulously analyze the overhead time

Sequence	MH1	MH2	MH3	MH4	MH5	V11	V12	V13	V21	V22	Avg
SV	**0.049**	0.077	**0.086**	**0.125**	0.125	**0.035**	0.053	0.082	0.046	**0.075**	0.075
SV-GN	0.057	**0.055**	0.097	0.135	**0.116**	0.038	**0.051**	**0.068**	**0.037**	0.083	**0.073**
SV-OFF[1]	0.067	0.103	0.107	0.137	0.143	0.038	0.062	-	0.057	0.255	0.107

[1] SV-OFF denotes SchurVINS with disabled EKF-based landmark solver only uses depth-filter to initialize landmark.

Table 5. Ablation Evaluation on EuRoC.

of SchurVINS including the comparison with SVO2.0, the widely-recognized filter-based OpenVINS and SMSCKF is carried out in Tab. 4.

In Tab. 4, the optimizeStructure module in SchurVINS is nearly 3 times faster than that of SchurVINS-GN. Because our method obtains significant computational savings by leveraging the intermediate results of Schur complement. In contrast, SchurVINS-GN reconstructs problems to estimate landmarks. Compared with SVO2.0-wo, SchurVINS is faster due to its replacement from the high-computational SparseImageAlign to propagation module. In contrast, the optimizeStructure of SVO2.0-wo is obviously faster than SchurVINS-GN. The reason is that the latter utilizes almost 4 times measurements than the former to conduct optimization. Compared with SVO2.0, the root cause leads to the obviously increased run time of algorithm is the high computational complexity of LBA. In consideration of OpenVINS, it is noteworthy that neither the default configuration nor the configuration with a maximum size of sliding window of 4 could achieve that OpenVINS outperforms SchurVINS in efficiency. What stands out from this analysis is that the update of SLAM points in OpenVINS requires noticeably more computational resources compared with the EKF-based landmark estimation presented in SchurVINS. Illustrated on Fig. 3, SchurVINS makes full use of the sparsity of problem than both hybrid MSCKF and optimization-based methods.

4.3. Ablation Study

The experiments above strongly support SchurVINS. And thus it is necessary to study the impact of different components of our algorithm. Based on SchurVINS, we replace or discard the EKF-based landmark solver to analyse its effectiveness.

As illustrated in Tab. 5, if without either GN-based or EKF-based landmark solver, SchurVINS cannot sufficiently limit the global drift. Moreover, in some challenge scenarios, lack of estimating landmarks simultaneously in SchurVINS may lead to system divergency. The comparison between SchurVINS and SchurVINS-GN in Tab. 5 indicates that both the proposed EKF-based landmark solver and the GN-based landmark solver belonging to original SVO2.0 are effective and reliable to guarantee high precision. In addition, the comparison between them in Tab. 4 and Tab. 5, illustrates that although the proposed EKF-based landmark solver leads to slight accuracy degradation, it could achieve the obviously low computational complexity. An intuitive explanation for the decreased accuracy is that our method only uses all the observations in sliding window for landmark estimation.

5. Conclusions and Future Work

In this paper, we have developed an EKF-based VINS algorithm, including the novel EKF-based landmark solver, to achieve 6-DoF estimation with both high efficiency and accuracy. In particular, the formulated equivalent residual model consisting of Hessian, Gradient and the corresponding observation covariance is utilized to estimate poses and landmarks jointly to guarantee high-precision positioning. To achieve high efficiency, the equivalent residual model is decomposed as pose residual model and landmark residual model by Schur complement to conduct EKF update respectively. Benefited from the probabilistic independence of surrounding environment elements, the resulting landmark residual model are split as a bunch of small independent residual models for the EKF update of each landmark, which significantly reduces the computational complexity. To best of our knowledge, we are the first to utilize Schur complement factorizing residual model in EKF-based VINS algorithms for acceleration. The experiments based on EuRoC and TUM-VI datasets demonstrate that our approach notably outperforms the overall EKF-based methods [10, 30] and the majority of optimization-based methods in both accuracy and efficiency. Besides, our approach requires almost less than 50% computational resource than the SOTA optimization-based methods [33, 34] with comparable accuracy. In the meanwhile, the ablation studies clearly demonstrate that our proposed EKF-based landmark solver is not only significantly efficient but also capable of ensuring high accuracy.

In future work, we will focus on the local map refinement in SchurVINS to explore more accuracy.

6. Acknowledgment

We would like to thank Taoran Chen, Chen Chen, and Jiatong Li in ByteDance as well as Zihuan Cheng in SCUT for their kind help. Moreover, I (Frank) would like to deeply thank my wife, Linan Guo.

References

[1] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part II 11*, pages 29–42. Springer, 2010. 1

[2] Michael Bloesch, Sammy Omari, Marco Hutter, and Roland Siegwart. Robust visual inertial odometry using a direct ekf-based approach. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 298–304. IEEE, 2015. 1, 6

[3] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart. The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research*, 35(10):1157–1163, 2016. 7

[4] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6): 1874–1890, 2021. 1, 2

[5] Jeffrey Delmerico and Davide Scaramuzza. A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 2502–2509. IEEE, 2018. 6

[6] Nikolaus Demmel, Christiane Sommer, Daniel Cremers, and Vladyslav Usenko. Square root bundle adjustment for large-scale reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11723–11732, 2021. 2

[7] Yunfei Fan, Ruofu Wang, and Yinian Mao. Stereo visual inertial odometry with online baseline calibration. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1084–1090. IEEE, 2020. 1, 2, 4

[8] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 15–22, 2014. 2

[9] Christian Forster, Zichao Zhang, Michael Gassner, Manuel Werlberger, and Davide Scaramuzza. SVO: Semidirect visual odometry for monocular and multicamera systems. *IEEE Trans. Robot.*, 33(2):249–265, 2017. 2, 6

[10] Patrick Geneva, Kevin Eckenhoff, Woosik Lee, Yulin Yang, and Guoquan Huang. Openvins: A research platform for visual-inertial estimation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4666–4672. IEEE, 2020. 1, 2, 4, 5, 6, 7, 8

[11] Guoquan Huang. Visual-inertial navigation: A concise review. In *2019 international conference on robotics and automation (ICRA)*, pages 9572–9582. IEEE, 2019. 1

[12] Guoquan P Huang, Anastasios I Mourikis, and Stergios I Roumeliotis. Analysis and improvement of the consistency of extended kalman filter based slam. In *2008 IEEE International Conference on Robotics and Automation*, pages 473–479. IEEE, 2008. 2

[13] Guoquan P Huang, Anastasios I Mourikis, and Stergios I Roumeliotis. A first-estimates jacobian ekf for improving slam consistency. In *Experimental Robotics: The Eleventh International Symposium*, pages 373–382. Springer, 2009. 2

[14] Viorela Ila, Lukas Polok, Marek Solony, and Pavel Svoboda. Slam++-a highly efficient and temporally scalable incremental slam framework. *The International Journal of Robotics Research*, 36(2):210–230, 2017. 2

[15] Michael Kaess, Ananth Ranganathan, and Frank Dellaert. isam: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, 2008.

[16] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert. isam2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, 31(2):216–235, 2012. 2

[17] Stefan Leutenegger, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34(3):314–334, 2015. 1, 6

[18] Mingyang Li and Anastasios I Mourikis. Vision-aided inertial navigation for resource-constrained systems. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1057–1063. IEEE, 2012. 2

[19] Mingyang Li and Anastasios I Mourikis. Optimization-based estimator design for vision-aided inertial navigation. In *Robotics: Science and Systems*, pages 241–248. Berlin Germany, 2013. 2

[20] Mingyang Li and Anastasios I Mourikis. High-precision, consistent ekf-based visual-inertial odometry. *The International Journal of Robotics Research*, 32(6):690–711, 2013. 2

[21] Haomin Liu, Mingyu Chen, Guofeng Zhang, Hujun Bao, and Yingze 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*, pages 1974–1982, 2018. 1, 2, 6

[22] Anastasios I Mourikis and Stergios I Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *Proceedings 2007 IEEE international conference on robotics and automation*, pages 3565–3572. IEEE, 2007. 1, 2, 4, 6

[23] Lukas Polok, Marek Solony, Viorela Ila, Pavel Smrz, and Pavel Zemcik. Efficient implementation for block matrix operations for nonlinear least squares problems in robotic applications. In *2013 IEEE International Conference on Robotics and Automation*, pages 2263–2269. IEEE, 2013. 2

[24] Tong Qin, Peiliang Li, and Shaojie Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018. 1, 5, 6, 7

[25] Tong Qin, Jie Pan, Shaozu Cao, and Shaojie Shen. A general optimization-based framework for local odometry estimation with multiple sensors. *CoRR*, abs/1901.03638, 2019. 7

[26] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *2020*

IEEE International Conference on Robotics and Automation (ICRA), pages 1689–1696. IEEE, 2020. 6

[27] David Schubert, Thore Goll, Nikolaus Demmel, Vladyslav Usenko, Jörg Stückler, and Daniel Cremers. The tum vi benchmark for evaluating visual-inertial odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1680–1687. IEEE, 2018. 7

[28] Gabe Sibley, Larry Matthies, and Gaurav Sukhatme. Sliding window filter with application to planetary landing. *Journal of Field Robotics*, 27(5):587–608, 2010. 5

[29] Joan Sola. Quaternion kinematics for the error-state kalman filter. *arXiv preprint arXiv:1711.02508*, 2017. 1, 2, 3

[30] Ke Sun, Kartik Mohta, Bernd Pfrommer, Michael Watterson, Sikang Liu, Yash Mulgaonkar, Camillo J Taylor, and Vijay Kumar. Robust stereo visual inertial odometry for fast autonomous flight. *IEEE Robotics and Automation Letters*, 3(2):965–972, 2018. 1, 4, 5, 7, 8

[31] Miroslav Trajković and Mark Hedley. Fast corner detection. *Image and vision computing*, 16(2):75–87, 1998. 6

[32] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings*, pages 298–372. Springer, 2000. 1

[33] Vladyslav Usenko, Nikolaus Demmel, David Schubert, Jörg Stückler, and Daniel Cremers. Visual-inertial mapping with non-linear factor recovery. *IEEE Robotics and Automation Letters*, 5(2):422–429, 2019. 1, 6, 8

[34] Lukas Von Stumberg and Daniel Cremers. Dm-vio: Delayed marginalization visual-inertial odometry. *IEEE Robotics and Automation Letters*, 7(2):1408–1415, 2022. 1, 6, 8

[35] Kejian Wu, Ahmed M Ahmed, Georgios A Georgiou, and Stergios I Roumeliotis. A square root inverse filter for efficient vision-aided inertial navigation on mobile devices. In *Robotics: Science and Systems*, page 2. Rome, Italy, 2015. 2

[36] Zhichao Ye, Guanglin Li, Haomin Liu, Zhaopeng Cui, Hujun Bao, and Guofeng Zhang. Coli-ba: Compact linearization based solver for bundle adjustment. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3727–3736, 2022. 2