Fast Monocular Visual-Inertial Initialization Leveraging Learned Single-View Depth

Nathaniel Merrill* Patrick Geneva* Saimouli Katragadda Chuchu Chen Guoquan Huang

***Abstract*—In monocular visual-inertial navigation systems, it is ideal to initialize as quickly and robustly as possible. State-of-the-art initialization methods typically make linear approximations using the image features and inertial information in order to initialize in closed-form, and then refine the states with a nonlinear optimization. While the standard methods typically wait for a 2sec data window, a recent work has shown that it is possible to initialize faster (0.5sec) by adding constraints from a robust but only up-to-scale monocular depth network in the nonlinear optimization. To further expedite the initialization, in this work, we leverage the scale-less depth measurements instead in the linear initialization step that is performed prior to the nonlinear one, which only requires a single depth image for the first frame. We show that the typical estimation of each feature state independently in the closed-form solution can be replaced by just estimating the scale and offset parameters of the learned depth map. Interestingly, our formulation makes it possible to construct small minimal problems in a RANSAC loop, whereas the typical linear system's minimal problem is quite large and includes every feature state. Experiments show that our method can improve the overall initialization performance on popular public datasets (EuRoC MAV and TUM-VI) over state-of-the-art methods. For the TUM-VI dataset, we show superior initialization performance with only a 0.3sec window of data, which is the smallest ever reported, and show that our method can initialize more often, robustly, and accurately in different challenging scenarios.**

I. Introduction

Visual-inertial odometry (VIO [1]) facilitates real-time 3D motion tracking through the utilization of the camera and an inertial measurement unit (IMU). The small size, low cost, efficiency, and complementary sensing characteristics have made VIO emerge as a foundational technology for AR/VR [2, 3, 4], robotics [5, 6, 7], and autonomous applications [8, 9, 10].

Two typical classes of VIO estimator designs are nonlinear optimization-based approaches [11, 12, 13, 14] and light-weight filter-based ones (e.g. an extended Kalman filter (EKF)) [15, 16, 17, 18, 19]. Both of these approaches rely on good initial conditions (e.g. velocity and gravity) in order to run successfully, and it is highly desirable to calculate the initial conditions as quickly as possible in order to decrease the time the user or end application has to wait to start. The initial conditions can be recovered by making assumptions about the motion (e.g. static), but under dynamic scenarios it is better to solve a visual-inertial structure from motion (VI-SfM) problem in order to initialize without making risky assumptions [20, 21]. However, even VI-SfM can fail, especially under low-excitation scenarios.

To tackle this initialization problem, a recent method [22] proposed to leverage learned monocular depth to provide additional constraints to the VI-SfM and help in the low excitation case, where the monocular priors are applied to each keyframe in the final bundle adjustment (BA) step. To initialize the visual-inertial bundle adjustment (VI-BA), this method utilizes a closed-form solution similar to [23], which compared to the nonlinear VI-BA is far more unstable due to the larger number of linear approximations required. In this work, we instead propose a simple yet effective method to utilize learned monocular depth priors in the *closed-form* linear initialization instead of the VI-BA refinement step, leveraging the single-image depth learned over millions of diverse examples as known prior information to reduce the number of parameters that need to be estimated in the fragile linear system.

The primary contributions of our work include:

- We propose a new formulation for closed-form visual-inertial linear initialization which leverages scale-less single-image depth to reduce the number of feature parameters to just a scale and offset.
- We show that our novel formulation allows for seamless integration of the minimal linear system into a robust RANSAC outlier rejection algorithm, which can be used to reject both bad depth priors as well as outlier feature tracks that may be present, whereas the typical linear system is less suitable for RANSAC.
- We validate our method on two public datasets, and show that our method can improve the final initialization accuracy under the challenging scenario of 0.5sec of data with 5 keyframes. We additionally show superior initialization performance for the new and *even more* challenging scenario of a 0.3sec initialization window, and extensive ablation studies show that our method has superior performance in the presence of outliers and a reduced number of available feature tracks.

The paper is organized as follows: Sec. II provides a review of related works, Sec. III provides background on the typical visual-inertial initialization problem, the proposed method is detailed in Sec. IV, and tested extensively in Sec. V against the state-of-the-art baselines. Finally, we offer some discussion of the limitations of our method in Sec. VI before concluding the paper in Sec. VII.

This work was partially supported by the University of Delaware (UD) College of Engineering, the NSF (IIS-1924897, MRI-2018905, SCH-2014264), and Google ARCore. The authors are with the Robot Perception and Navigation Group, University of Delaware, {nmerrill,pgeneva,saimouli,ccchu,ghuang}@udel.edu

*Denotes equal contribution

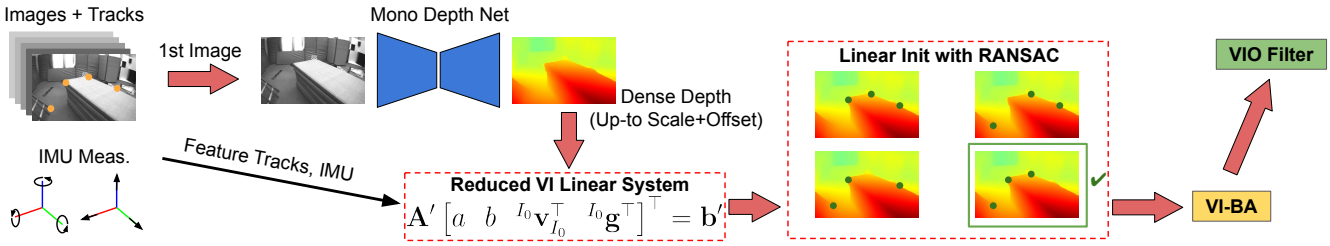

Fig. 1: Overview of the proposed monocular-depth aided visual-inertial initialization method.

II. Related Works

Many works have investigated different methods for performing visual-inertial initialization, and can be generally divided into two different categories: 1) loosely-coupled algorithms, and 2) closed-form solutions. Loosely-coupled algorithms split the problem into first recovering an up-to-scale camera-only SfM trajectory result and then recovering the scale given the inertial measurements, while closed-form solutions directly formulate a linear system involving both visual and inertial measurements.

A. Loosely-coupled Algorithms

The works by Mur-Artal et al. [24] and Qin et al. [12, 25] use a loosely-coupled approach. Mur-Artal et al. [24] leverage ORB-SLAM [14, 26] SfM results and formulate a small linear system involving the up-to-scale poses and inertial preintegration to directly recover scale and gravity – which are then refined along with the accelerometer bias in a secondary step. A later work by Campos et al. [27] additionally use the up-to-scale SfM poses, but instead directly optimizes up-to-scale velocities, gravity direction, biases, and scale. Since an initial guess of scale is required for non-linear optimization, they run the initialization multiple times at different initial scales and select the one which gives the smallest cost.

Qin et al. [12, 25] leverage a simplified SfM pipeline to obtain the up-to-scale trajectory, and then formulate a linear system that recovers scale, gravity, and velocity. A more recent work by Zuñiga-Noël et al. [28] showed that up-to-scale SfM results could be leveraged in a quadratically-constrained least-squares problem, similar to closed-form solutions, which constrains the known magnitude of gravity to improve the accuracy. Another work by Concha et al. [29] proposed a method that quickly initializes the 6 degrees of freedom (DoF) pose without motion parallax by decoupling the problem into the rotation, translation direction (5DoF) and magnitude of the translation (1DoF). While promising due to their robustification with RANSAC to handle outliers, they do not directly leverage inertial information in these low parallax scenarios. A key downside of loosely-coupled algorithms is that they are reliant on good SfM results, which require significant parallax and are typically computationally expensive to obtain.

B. Closed-form Solutions

The earliest works on closed-form solutions are by Dong-Si and Mourikis [20, 30] and Martinelli [21, 31]. In particular, Dong-Si and Mourikis [20, 30] proposed the use of a quadratically-constrained least-squares problem which enforces the gravity magnitude, and showed improvements over methods which did not enforce this constraint. They focused on the recovery of an unknown IMU-to-camera rotation and translation, and directly recovered the 3DoF feature positions in the first reference frame – where Martinelli [21] recovered the depth of each feature for each bearing observation in every frame. A work by Li and Mourikis [23] tried to address the lack of robustness by incorporating measurement noise by using estimated feature depths to simplify the feature reprojection cost into an approximate convex minimization problem. A key drawback is requiring knowledge of the average scene depth.

Another work by Jacques et al. [32] focused on evaluating sensitivities to accelerometer and gyroscope biases, which was further extended by Campos et al. [33] to include an observability and consensus test to remove poor initialization results near pure rotation and with limited acceleration motions. A recent work by Evangelidis and Micusik [34] focused on reducing the computational demands of Martinelli's [21] linear system, and showed that the marginalization (projection) of the depth of each feature bearing and redundant 3DoF feature in a reference frame was possible and efficient.

C. Learning-aided Initialization

Recently a handful of works have emerged which investigate the use of learning-based methods to aid traditional SfM and visual-inertial initialization problems. Liu et al. [35] utilized a large MiDaS [36] depth estimation model to replace the traditional 5-point algorithm [37] with a PnP alignment to the learned depth cloud. Another work by Hruby et al. [38], employed model learning to select a starting problem solution which could numerically be continued without requiring significant samples within a RANSAC formulation. Both of these methods, while outside of the visual-inertial field, utilize learning in the linear initialization stage – similar in spirit to our approach. Linear initialization, whether in visual or visual-inertial systems, has always been a highly-unstable processes, and can gain large benefits from learned prior information.

The work closest to ours is that by Zhou et al. [22]. This work was the first to leverage learned scale-less depth priors to better constrain the VI-BA – which is performed after solving a closed-form solution by Li and Mourikis [23]. This work showed that the inclusion of scale-less depth constraints in their VI-BA improved the problem conditioning, robustness, and accuracy under low-excitation scenarios. As

compared to this work, we look to leverage the scale-less depth directly within the linear initialization stage. As compared to recovering each feature, our linear system is simplified to only recovering the scale and offset of the depth map prediction. This additionally enables the application of RANSAC to further robustify the problem to outliers.

III. Monocular Visual-Inertial Linear Initialization

We consider a sensor platform consisting of a monocular camera and an inertial measurement unit (IMU). During the initialization time period N images at $[t_0, \cdots, t_N]$ are recorded along with IMU readings. The minimal state we wish to recover is [20, 30]:

$$\mathbf{x} = \begin{bmatrix} {}^{I_0}\mathbf{p}_{f_1}^\top & \cdots & {}^{I_0}\mathbf{p}_{f_M}^\top & {}^{I_0}\mathbf{v}_{I_0}^\top & {}^{I_0}\mathbf{g}^\top \end{bmatrix}^\top \tag{1}$$

where $\{I_0\}$ denote the first IMU frame, ${}^{I_0}\mathbf{p}_{f_i}$ is the 3DoF feature position with respect to $\{I_0\}$, and ${}^{I_0}\mathbf{v}_{I_0}$, ${}^{I_0}\mathbf{g}$ are the velocity of the platform and local gravity expressed in the $\{I_0\}$ frame, respectively.

A. Inertial Measurement Model

A canonical three-axis IMU provides linear acceleration, ${}^{I}\mathbf{a}_m$, and angular velocity, ${}^{I}\boldsymbol{\omega}_m$, measurements expressed in the local IMU frame $\{I\}$:

$$\mathbf{a}_m(t) = \mathbf{a}(t) + {}^{I}_{G}\mathbf{R}(t)\,{}^{G}\mathbf{g} + \mathbf{b}_a(t) + \mathbf{n}_a(t) \tag{2}$$

$$\boldsymbol{\omega}_m(t) = \boldsymbol{\omega}(t) + \mathbf{b}_g(t) + \mathbf{n}_g(t) \tag{3}$$

where ${}^{G}\mathbf{g} \simeq [0, 0, 9.81]^\top$ is the gravitational acceleration expressed in the global frame $\{G\}$, and $\mathbf{n}_g$, $\mathbf{n}_a$ are zero-mean white Gaussian noises. ${}^{I}_{G}\mathbf{R}$ denotes the rotation matrix that transforms a position expressed in the global frame to one in the local frame. We assume that the biases $\mathbf{b}_a$ and $\mathbf{b}_g$ are known with reasonable accuracy. The continuous time IMU kinematics which evolve the state from time t_k to t_{k+1} are [39, 40]:

$${}^{I_{k+1}}_{G}\mathbf{R} = {}^{I_{k+1}}_{I_k}\Delta\mathbf{R}\;{}^{I_k}_{G}\mathbf{R} \tag{4}$$

$${}^{G}\mathbf{p}_{I_{k+1}} = {}^{G}\mathbf{p}_{I_k} + {}^{G}\mathbf{v}_{I_k}\Delta T - \frac{1}{2}{}^{G}\mathbf{g}\Delta T^2 + {}^{I_k}_{G}\mathbf{R}^\top {}^{I_k}\boldsymbol{\alpha}_{I_{k+1}} \tag{5}$$

$${}^{G}\mathbf{v}_{I_{k+1}} = {}^{G}\mathbf{v}_{I_k} - {}^{G}\mathbf{g}\Delta T + {}^{I_k}_{G}\mathbf{R}^\top {}^{I_k}\boldsymbol{\beta}_{I_{k+1}} \tag{6}$$

where ${}^{I_k}\boldsymbol{\alpha}_{k+1}$ and ${}^{I_k}\boldsymbol{\beta}_{k+1}$ are the preintegration terms [41, 42, 43]:

$${}^{I_k}\boldsymbol{\alpha}_{I_{k+1}} = \int_{t_k}^{t_{k+1}} \int_{t_k}^{s} {}^{k}_{u}\Delta\mathbf{R}\,(\mathbf{a}_m(u) - \mathbf{b}_a(u) - \mathbf{n}_a(u))\,duds$$

$${}^{I_k}\boldsymbol{\beta}_{I_{k+1}} = \int_{t_k}^{t_{k+1}} {}^{k}_{u}\Delta\mathbf{R}\,(\mathbf{a}_m(u) - \mathbf{b}_a(u) - \mathbf{n}_a(u))\,du$$

We can transform an integration from t_0 to t_k in the global into the first IMU frame $\{I_0\}$:

$${}^{I_k}_{I_0}\mathbf{R} \triangleq {}^{I_k}_{I_0}\Delta\mathbf{R} \tag{7}$$

$${}^{I_0}\mathbf{p}_{I_k} \triangleq {}^{I_0}\mathbf{v}_{I_0}\Delta T_k - \frac{1}{2}{}^{I_0}\mathbf{g}\Delta T_k^2 + {}^{I_0}\boldsymbol{\alpha}_{I_k} \tag{8}$$

$${}^{I_0}\mathbf{v}_{I_k} \triangleq {}^{I_0}\mathbf{v}_{I_0} - {}^{I_0}\mathbf{g}\Delta T_k + {}^{I_0}\boldsymbol{\beta}_{I_k} \tag{9}$$

where $\Delta T_k = (t_k - t_0)$ is the time span for integration. These can be found by rotating the orientation and velocity with ${}^{I_0}_{G}\mathbf{R}$ and computing the relative position change ${}^{I_0}\mathbf{p}_{I_k} = {}^{I_0}_{G}\mathbf{R}({}^{G}\mathbf{p}_{I_k} - {}^{G}\mathbf{p}_{I_0})$, and defines the *relative* IMU integration in the fixed $\{I_0\}$ frame [44].

B. Feature Bearing Observations

Assuming a calibrated perspective camera, the bearing measurement of the i'th feature at timestep t_k can be related to the state by the following:

$$\mathbf{z}_{i,k} := \boldsymbol{\Lambda}({}^{C_k}\mathbf{p}_{f_i}) + \mathbf{n}_k \tag{10}$$

$${}^{C_k}\mathbf{p}_{f_i} = {}^{C}_{I}\mathbf{R}\,{}^{I_k}_{I_0}\mathbf{R}({}^{I_0}\mathbf{p}_{f_i} - {}^{I_0}\mathbf{p}_{I_k}) + {}^{C}\mathbf{p}_I \tag{11}$$

where $\boldsymbol{\Lambda}([x\ y\ z]^\top) = [x/z\ \ y/z]^\top$ is the camera perspective projection model, $\mathbf{z}_{i,k} = [u_{i,k}, v_{i,k}]^\top$ is the normalized feature bearing measurement with white Gaussian noise $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$, and $\{{}^{C}_{I}\mathbf{R}, {}^{C}\mathbf{p}_I\}$ are the known camera-IMU transformation. Eq. (10) can be re-written as the following linear constraint [20]:

$$\begin{bmatrix} 1 & 0 & -u_{i,k} \\ 0 & 1 & -v_{i,k} \end{bmatrix} {}^{C_k}\mathbf{p}_{f_i} \triangleq \boldsymbol{\Gamma}_{i,k}\,{}^{C_k}\mathbf{p}_{f_i} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{12}$$

We can then substitute Eq. (8) and (11) to give:

$$\mathbf{A}_{i,k}\ \mathbf{x} = \mathbf{b}_{i,k} \tag{13}$$

$$\mathbf{A}_{i,k} = \boldsymbol{\Upsilon}_{i,k}\begin{bmatrix} \cdots & \mathbf{I}_3 & \cdots & -\boldsymbol{\Delta}\mathbf{T}_k & \boldsymbol{\Delta}\mathbf{T}_k^2 \end{bmatrix} \tag{14}$$

$$\mathbf{b}_{i,k} = \boldsymbol{\Upsilon}_{i,k}\,{}^{I_0}\boldsymbol{\alpha}_{I_k} - \boldsymbol{\Gamma}_{i,k}\,{}^{C}\mathbf{p}_I \tag{15}$$

where $\boldsymbol{\Delta}\mathbf{T}_k = \Delta T_k\mathbf{I}_3$ and $\boldsymbol{\Upsilon}_{i,k} = \boldsymbol{\Gamma}_{i,k}\,{}^{C}_{I}\mathbf{R}\,{}^{I_k}_{I_0}\mathbf{R}$. This can be "stacked" to recover a complete $\mathbf{A}\mathbf{x} = \mathbf{b}$, and given M features from N images, $\mathbf{A} \in \mathbb{R}^{2MN \times (3M+6)}$ and $\mathbf{b} \in \mathbb{R}^{2MN}$.

C. Constrained Linear Least-Squares

We follow the method by Dong-Si and Mourikis [20, 30, 44], and formulate a constrained linear least-squares problem given the stacked observations (see Eq. (13)):

$$\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 = \|\begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ {}^{I_0}\mathbf{g} \end{bmatrix} - \mathbf{b}\|_2 \tag{16}$$

$$\text{subject to } \|{}^{I_0}\mathbf{g}\|_2 = g \tag{17}$$

The optimal solution can be derived using Lagrange multipliers [30]. The gravity constraint has been shown to have a noticeable impact on shorter trajectory lengths [32].

IV. Single-Image Depth Aided Initialization

We now consider we are given a *single* affine-invariant (up-to scale and offset) depth map, $\mathbf{D}$, in the first frame of reference at time t_0. As compared to recovering the full feature states in Eq. (1), we instead formulate all features as a function of this depth map and the feature bearing in the first camera frame $\{C_0\}$. The minimal state we wish to recover is:

$$\mathbf{x}' = \begin{bmatrix} a & b & {}^{I_0}\mathbf{v}_{I_0}^\top & {}^{I_0}\mathbf{g}^\top \end{bmatrix}^\top \tag{18}$$

where we have assumed that the affine-invariant depth map $\mathbf{D}$ is sufficiently accurate and can provide an estimate of the 3D structure in front of the camera up to a scale a and offset parameter b from just a single frame [36]. An overview of the proposed method can be seen in Fig. 1.

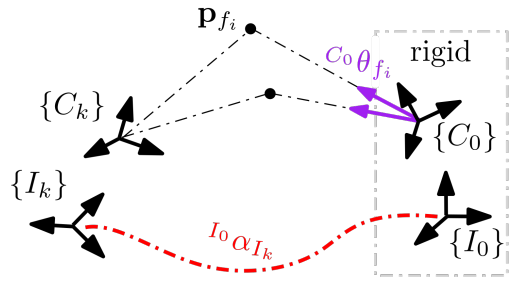

Fig. 2: Frame of references used in the problem. Two features observed from both the $\{C_k\}$ and $\{C_0\}$ frame are shown. The transformation from the $\{I_k\}$ and $\{I_0\}$ is found through IMU integration. The bearing ${}^{C_0}\boldsymbol{\theta}_{f_i}$ is used along with the scale-less depth to recover the scale a and shift b.

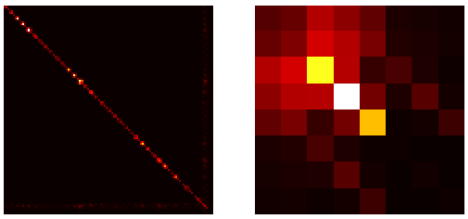

Fig. 3: A comparison of the structure of the Dong-Si [20] $\mathbf{A}^\top\mathbf{A}$ (left) and the proposed $\mathbf{A}'^\top\mathbf{A}'$ (right). The Dong-Si system contains 35 features here (making it 111×111). While sparse, it is much larger than ours, which is 8×8 no matter how many features are included. The log condition number for Dong-Si's is 9.35 while the proposed is 8.15.

A. Depth-Aided Feature Bearing Model

We now modify the feature model in Sec. III-B to be a function of the scale-less depth map. We assume that for a single image the scale a and shift b are constant for the whole depth map. Specifically, for feature ${}^{I_0}\mathbf{p}_{f_i}$ we can express the metric depth scalar $z_i = \mathrm{Z}(u_{i,0}, v_{i,0})$ as a function of a, b, and $d_i = \mathrm{D}(u_{i,0}, v_{i,0})$:

$$\begin{aligned} {}^{I_0}\mathbf{p}_{f_i} &= {}^{I}_{C}\mathbf{R}\ {}^{C_0}\mathbf{p}_{f_i} + {}^{I}\mathbf{p}_{C} \\ &= z_i\ {}^{I_0}\boldsymbol{\theta}_{C_0\to f_i} + {}^{I}\mathbf{p}_{C} \\ &= (ad_i + b)\ {}^{I_0}\boldsymbol{\theta}_{C_0\to f_i} + {}^{I}\mathbf{p}_{C} \end{aligned} \tag{19}$$

where ${}^{I_0}\boldsymbol{\theta}_{C_0\to f_i} = {}^{I}_{C}\mathbf{R}[u_{i,0}\ v_{i,0}\ 1]^\top$ is the bearing vector of the feature rotated (but not translated) into the IMU frame, see Fig. 2 for example frame of references. This treats the normalized 2D coordinates of the feature in the first camera frame $u_{i,0}$ and $v_{i,0}$ as a known quantity. Substituting Eq. (19) into Eq. (11) we can recover the following linear system:

$$\mathbf{A}'_{i,k}\ \mathbf{x}' = \mathbf{b}'_{i,k} \tag{20}$$

$$\mathbf{A}'_{i,k} = \boldsymbol{\Upsilon}_{i,k}\begin{bmatrix}\mathbf{B}_i & -\boldsymbol{\Delta}\mathbf{T}_k & \frac{1}{2}\boldsymbol{\Delta}\mathbf{T}_k^2\end{bmatrix} \tag{21}$$

$$\mathbf{b}'_{i,k} = \boldsymbol{\Upsilon}_{i,k}\ {}^{I_0}\boldsymbol{\alpha}_{I_k} - \boldsymbol{\Upsilon}_{i,k}\ {}^{I}\mathbf{p}_{C} - \boldsymbol{\Gamma}_{i,k}\ {}^{C}\mathbf{p}_{I} \tag{22}$$

$$\mathbf{B}_i = \begin{bmatrix} d_i\ {}^{I_0}\boldsymbol{\theta}_{C_0\to f_i} & {}^{I_0}\boldsymbol{\theta}_{C_0\to f_i}\end{bmatrix}. \tag{23}$$

Given M features from N images, $\mathbf{A}' \in \mathbb{R}^{2MN\times(2+6)}$ and $\mathbf{b}' \in \mathbb{R}^{2MN}$. One can see that the state size remains constant, no matter how many features are included in the problem. The structure of our system can be seen in Fig. 3

Remarks: As evident, this formulation of the linear initialization problem significantly relaxes the original one – reducing the need to estimate the 3D position of every feature to just estimating the scale and offset of the depth map predicted at t_0 – which is shared between all features. Given a reasonable predicted affine-invariant depth $\mathbf{D}$ and a and b are well-constrained, if the recovered scale parameter a is positive, all of the features will be in front of the camera as desired, and there will be no spurious feature positions (e.g. too close or too far due to high uncertainty).

It should be noted that the monocular depth network MiDaS [36] leveraged in this work actually produces affine-invariant *inverse* depth maps $\mathbf{D}_{\text{inv}}$, where $\mathrm{D}(u_i, v_i) = 1/\mathrm{D}_{\text{inv}}(u_i, v_i)$ (dropping the subscript for clarity), and the metric inverse depth is expressed as $\mathrm{Z}_{\text{inv}}(u_i, v_i) = a_{\text{inv}}\mathrm{D}_{\text{inv}}(u_i, v_i) + b_{\text{inv}}$. The use of affine-invariant *depth* instead of *inverse depth* is also reported in [35], which utilizes the same class of depth networks as us. Due to the division, one may suspect that the scale and offset for depth, a and b, would be a nonlinear function of a_{inv} and b_{inv}, but in fact, it can be expressed linearly with the following relationship:

$$(a\ \mathrm{D}\,(u_i, v_i)) + b)\,(a_{\text{inv}}\ \mathrm{D}_{\text{inv}}(u_i, v_i) + b_{\text{inv}}) = 1. \tag{24}$$

Thus, estimating the scale and offset a and b in Eq. (18) instead of a_{inv} and b_{inv} is valid, and $a_{\text{inv}}, b_{\text{inv}}$ can be recovered from a solution of a, b via stacking and solving

$$\begin{bmatrix}(a + b\mathrm{D}_{\text{inv}}(u_i, v_i)) & (a\mathrm{D}(u_i, v_i) + b)\end{bmatrix}\begin{bmatrix}a_{\text{inv}} \\ b_{\text{inv}}\end{bmatrix} = 1 \tag{25}$$

for all u_i, v_i, which is simply Eq. (24) rearranged. Similarly, Eq. (24) can be rearranged to recover a and b from estimates of a_{inv} and b_{inv} by just grouping different terms.

The fact that a, b and $a_{\text{inv}}, b_{\text{inv}}$ can be related linearly also means that we can scale $\mathbf{D}_{\text{inv}}$ arbitrarily before using it in the linear system. To this end, for ensured numeric stability of $\mathbf{D}$, we scale $\mathbf{D}_{\text{inv}}$, which can have arbitrary value, into the range $[1, 2]$ before computing $\mathbf{D}$ via:

$$\mathrm{D}_{\text{inv}}(u_i, v_i) = \frac{\mathrm{D}^0_{\text{inv}}(u_i, v_i) - \min(\mathbf{D}^0_{\text{inv}})}{\max(\mathbf{D}^0_{\text{inv}}) - \min(\mathbf{D}^0_{\text{inv}})} + 1 \tag{26}$$

where $\mathbf{D}^0_{\text{inv}}$ is the raw affine-invariant inverse depth map from the monocular depth network. Note that the range $[1, 2]$ is chosen arbitrarily to avoid possible division by zero.

B. Outlier Rejection in Linear Initialization

A key advantage of our proposed linear system formulation is its ability to be easily inserted into *small* minimal problems in a RANSAC loop to robustify it to outliers. In theory each measurement in the minimal problem for Eq. (18) can be chosen from a different feature since each feature track constrains the same a and b states. However, in practice, we group the measurements by feature and view in order to 1) reject bad feature tracks, and 2) reject bad depth network predictions. An overview of our RANSAC approach can be seen in Algo. 1. A minimal set of features and poses are first randomly grouped and the constrained linear system, Eq. (16), is solved to recover the scale, shift, velocity, and gravity.

Algorithm 1 Linear Initialization with RANSAC

Require: Blocks $\mathbf{A}'_{ki}$, $\mathbf{b}'_{ki}$ of the complete linear system for $i \in \{1,\dots,M\}, k \in \{1,\dots,N\}$, minimal problem size $M_{\text{min}}, N_{\text{min}}$, maximum number of iterations K, thresholds d_{min}, γ
Ensure: Robustified solution to linear system $\mathbf{x}'_{\text{best}}$

$e_{\text{best}} \leftarrow \infty$
for $i \in \{1,\dots,K\}$ **do**
  $\mathcal{S} \leftarrow$ Rand. sample N_{min} meas. from M_{min} feats.
  $\mathbf{A}'_s,\ \mathbf{b}'_s \leftarrow$ Stack blocks $i,k \in \mathcal{S}$
  $a, b, {}^{I_0}\mathbf{v}_{I_0}, {}^{I_0}\mathbf{g} \leftarrow \text{solve}(\mathbf{A}'_s,\ \mathbf{b}'_s)$
  for i,k not in $\mathcal{S}$ **do**
    $\mathbf{r} \leftarrow \mathbf{A}'_{ik}\begin{bmatrix} a & b & {}^{I_0}\mathbf{v}_{I_0}^\top & {}^{I_0}\mathbf{g}^\top \end{bmatrix}^\top - \mathbf{b}'_{ik}$
    if $||\mathbf{r}|| < \gamma$ **then**
      $\mathcal{S} \leftarrow \mathcal{S} \cup (i,k)$
    end if
  end for
  if $|\mathcal{S}| \geq d_{\text{min}}$ **then**
    $\mathbf{A}'_{\text{inl}},\ \mathbf{b}'_{\text{inl}} \leftarrow$ Stack blocks $i,k \in \mathcal{S}$
    $a, b, {}^{I_0}\mathbf{v}_{I_0}, {}^{I_0}\mathbf{g} \leftarrow \text{solve}(\mathbf{A}'_{\text{inl}},\ \mathbf{b}'_{\text{inl}})$
    $\mathbf{r} \leftarrow \mathbf{A}'_{\text{inl}}\begin{bmatrix} a & b & {}^{I_0}\mathbf{v}_{I_0}^\top & {}^{I_0}\mathbf{g}^\top \end{bmatrix}^\top - \mathbf{b}'_{\text{inl}}$
    if $||\mathbf{r}|| < e_{\text{best}}$ **then**
      $e_{\text{best}} \leftarrow ||\mathbf{r}||$
      $\mathbf{x}'_{\text{best}} \leftarrow \begin{bmatrix} a & b & {}^{I_0}\mathbf{v}_{I_0}^\top & {}^{I_0}\mathbf{g}^\top \end{bmatrix}^\top$
    end if
  end if
end for

These states are then used to compute the reprojection error for each measurement not used in the problem, and construct the inlier measurement set $\mathcal{S}$. The solution from the inlier set which gives the minimal error is selected as the best state estimate. We emphasize that the RANSAC approach becomes feasible due to our relaxation of the original linear system from the inclusion of the affine-invariant depth map. While the hard minimal problem for our RANSAC algorithm is 3 views and 2 features, we use 3 views and 4 features in the minimal problems in all experiments for slightly improved conditioning.

C. Nonlinear Refinement

We recover the the 3D position of all features (inlier or not) via Eq. (19), and recover gravity aligned orientation by transforming the recovered gravity ${}^{I_0}\mathbf{g}$ into a gravity aligned frame ${}^{G}\mathbf{g} = [0,0,9.81]^\top$. The VI-BA problem which refines the state estimates, takes into account measurement uncertainties, and recovers the covariance of the initial states can now be formulated. The state vector of this optimization process can be defined as:

$$\mathbf{x}_{\text{mle}} = \begin{bmatrix} \mathbf{x}_{I_0}^\top & \dots & \mathbf{x}_{I_N}^\top & {}^{G}\mathbf{p}_{f_1}^\top & \dots & {}^{G}\mathbf{p}_{f_M}^\top \end{bmatrix}^\top \tag{27}$$

$$\mathbf{x}_{I_k} = \begin{bmatrix} {}^{I_k}_{G}\bar{q}^\top & {}^{G}\mathbf{p}_{I_k}^\top & {}^{G}\mathbf{v}_{I_k}^\top & \mathbf{b}_{g,k}^\top & \mathbf{b}_{a,k}^\top \end{bmatrix}^\top \tag{28}$$

Note that we do *not* include the depth prior in the nonlinear optimization as in [22], because it would require estimating the depth for *all* keyframe images (which could be computational and energy intensive even if possible in real time), rather than the *single* first one (which is *all* that is required in our solution). We empirically found that only including the depth prior in the first keyframe in the VI-BA optimization leads to the exact same result as optimizing without it, but perhaps could improve it if we had a scale prior as in [22]. Thus, we omit the depth prior from the VI-BA and only use it in linear initialization, although including depth priors for all keyframes in the optimization helps as shown in [22].

We solve the optimization problem with inertial $\mathbb{C}_I$, camera $\mathbb{C}_C$, and prior $\mathbb{C}_P$ cost terms:

$$\underset{\mathbf{x}_{\text{mle}}}{\text{argmin}} \quad \mathbb{C}_I + \mathbb{C}_C + \mathbb{C}_P \tag{29}$$

With the following inertial cost function [41, 42, 43]:

$$\mathbb{C}_I \triangleq \sum_k ||\mathbf{x}_{I_{k+1}} \boxminus \mathbf{f}(\mathbf{x}_{I_k}, \mathbf{a}_{m_k}, \boldsymbol{\omega}_{m_k})||^2_{\mathbf{Q}_k} \tag{30}$$

where $\mathbf{Q}_k$ is the linearized measurement noise covariance. The camera re-projection cost is defined as [19]:

$$\mathbb{C}_C \triangleq \sum_{i,k} \|\mathbf{z}_{i,k} - \mathbf{h}(\mathbf{x}_{\text{mle}})\|^2_{\mathbf{R}_i} \tag{31}$$

where $\mathbf{h}(\cdot)$ includes the camera's intrinsic distortion, projection, and camera-to-IMU extrinsic transformation, and $\mathbf{R}_i$ is the image pixel noise covariance.

In addition to constraining the unobservable initial global position and yaw rotation [45, 46], we found that the gyroscope and especially accelerometer biases can nearly be unobservable and hard to initialize, and thus, we provide reasonable priors to these states to avoid numerical instabilities. The prior cost is defined as:

$$\mathbb{C}_P \triangleq \|\mathbf{x}_{\text{mle}} \boxminus \breve{\mathbf{x}}_{\text{mle}}\|^2_{\mathbf{\Omega}_P^{-1}} \tag{32}$$

where $\breve{\mathbf{x}}_{\text{mle}}$ is the fixed state linearization point and $\mathbf{\Omega}_P$ is the prior information matrix – where large values are picked for unobservable state variables.

V. Experimental Validation

To validate the proposed singe-image depth-aided monocular VIO initialization, we employ the two most popular public VI datasets: EuRoC MAV [47] and TUM-VI [48]. We choose an evaluation method similar to that of [22], where we divide each sequence into 10sec windows, run initialization for each of the entry points, and average the results from each successful run (typically 10-15 initializations per trajectory).

In our experiments, we mainly consider the absolute trajectory error (ATE) [49] metric for position and orientation. We additionally use all recovered poses to perform a SIM(3) alignment to the ground truth in order to report the scale error of the problem. For the ATE, trajectories are aligned

TABLE I: Linear system results on EuRoC

Algorithm	ATE (deg)	ATE (m)	#feats
DS 3D	**4.011**	**0.187**	54.9
DS 1D	4.159	0.193	54.8
Ours w/o RANSAC	4.238	0.199	**60.2**
Ours	4.119	0.197	60.0

TABLE II: Scale error (%) on EuRoC after VI-BA (5 KFs, 0.5sec window)

Algorithm	V101	V102	V103	V201	V202	V203	Avg.
DS 3D	11.9	12.1	9.2	8.1	285.1	19.4	57.6
DS + DP	14.1	11.6	7.8	6.4	294.2	19.0	58.8
Ours w/o RANSAC	9.9	1.8	12.8	1.3	72.1	5.6	17.3
Ours	9.3	5.9	10.9	1.9	4.1	2.8	**5.8**

TABLE III: ATE (deg/m) on EuRoC after VI-BA (5 KFs, 0.5sec window).

Algorithm	V101	V102	V103	V201	V202	V203	Average
DS 3D	1.338 / 0.014	0.659 / 0.020	2.173 / 0.025	0.899 / 0.013	1.999 / 0.053	2.483 / 0.043	1.592 / 0.028
DS + DP	1.218 / 0.019	0.706 / 0.022	2.255 / 0.025	0.891 / 0.013	1.600 / 0.039	2.466 / 0.042	1.523 / 0.027
Zhou [22] *	- / 0.021	- / 0.038	- / 0.025	- / 0.015	- / 0.015	- / 0.033	- / 0.024
Ours w/o RANSAC	1.138 / 0.017	0.733 / 0.021	2.074 / 0.025	0.886 / 0.011	1.844 / 0.034	2.128 / 0.046	1.467 / 0.026
Ours	1.111 / 0.019	1.065 / 0.018	2.234 / 0.030	0.882 / 0.011	1.442 / 0.015	1.778 / 0.040	**1.419 / 0.022**

*Results quoted from Table 1 in [22].

to the ground truth using the first frame by solving for the optimal position and yaw transform between the estimate and ground truth (see [49]). Since we do *not* use a scale-aware alignment such as SIM(3) to compute the ATE, scale accuracy will directly impact the position, and gravity errors will also affect the orientation ATE.

A. Implementation Details

Our initialization is implemented on top of the open-source OpenVINS [19] `ov_init` package [44], which re-implements the baseline initialization method by Dong-Si and Mourikis [20, 30] (see Sec. III). Unless otherwise noted, 75 features on average are used during initialization. For the monocular depth network, we leverage an off-the-shelf pre-trained MiDaS network [36] (the v2.1 small model). This particular model is one of the most efficient available from the MiDaS model zoo, and is suitable to run on mobile devices. During all experiments, the network is run directly on the CPU, and network inference typically takes around 0.2sec. A separate thread is launched for initialization from the main tracking thread, but no extra thread is used to run the depth network asynchronously. While this could be done to improve initialization latency, we choose to simply run the network on-demand since it is *only* required to run once per initialization window (the first frame).

B. Baseline Methods

For evaluation, we mainly consider two methods: 1) **DS 3D**, which implements the work of Dong-Si and Mourikis [20] and is the current default initialization available in OpenVINS [19], and 2) **DS + DP**, which is our re-implementation of [22] using the OpenVINS implementation of Dong-Si [20] and the MiDaS v2.1 small network [36]. Note that since we utilize MiDaS, which is completely scale-less, as opposed to the custom depth network in [22] which is weakly-supervised with metric scale, we are unable to include the $1, 0$ prior on the scale and shift in the VI-BA. Including this prior could potentially improve the results, but it is unfortunately not applicable to MiDaS. Other than this difference, we strictly followed the formulation presented in [22] for this re-implementation.

C. EuRoC MAV Dataset

We first evaluate on the EuRoC MAV dataset [47] to provide a relative comparison to the state-of-the-art work by Zhou et al. [22] (denoted as **Zhou [22]**). This comparison is only partial since the implementation in [22] is not open-sourced, thus we are forced to quote results from the paper where applicable. We measure the full orientation error and scale error over the whole trajectory rather than just the gravity and scale error over well-excited trajectory segments, and thus can not directly compare to their orientation and scale. We have selected the closest equivalent challenging configuration of of 5 keyframes (KFs) evenly spaced over a 0.5sec window.

We first evaluate the impact of the proposed linear system. We directly report the results of the linear system solutions (no VI-BA refinement) in Table I. In additional to the ATE, we also report the average number of features which have been successfully initialized in front of the camera – where more features is better in order to better constrain the VI-BA problem. We also include a 1D version of the linear system, Eq. (13), termed **DS 1D**, which estimates scalar feature depths – considering the first bearing to be true. This 1D linear system method has slightly worse accuracy than the 3D version, which is expected since it is not robust to bad initial bearings. We can see that the proposed method without RANSAC is slightly worse in terms of ATE than the linear system with 1D feature states, which is expected since it blindly utilizes all depth priors in the linear system and also considers the first bearings to be true. On the other hand, it can successfully initialize *more* features in front of the camera than either of the baseline linear systems, and after performing VI-BA refinement it is able to have an overall performance improvement (see Table III). Similarly, the proposed method *with* RANSAC is slightly improved in terms of ATE and is able to initialize more features than the other baselines. Note that the OpenVINS KLT tracker attempts to reject bad tracks with a Fundamental matrix RANSAC check, but nevertheless some outliers can pass the check – especially if they are on the epipolar line.

Looking now to results which perform the VI-BA refinement after closed-form recovery, Tables II and III report the scale error and ATE, respectively. We can see that the proposed system without RANSAC enabled (i.e. using all available measurements outlier or not) hurts the performance, while leveraging RANSAC has improved scale and ATE accuracy.

D. TUM-VI Dataset

The second dataset we consider is the TUM-VI dataset [48], where we only evaluate using the left fisheye image. As shown in Fig. 4, the MiDaS v2.1 small network is still able to produce reasonable scale-less depth predictions even without, to the best of our knowledge, explicitly training on this camera model. The results shown in Table IV confirms that the proposed method is able to achieve higher accuracy in the average case for all metrics. One can also see that including

TABLE IV: Initialization window ATE and scale (deg/m (%)) on TUM-VI after VI-BA (5 KFs, 0.5sec window)

Algorithm	room1	room2	room3	room4	room5	room6	Average
DS 3D	1.003 / 0.009 (0.54)	0.957 / 0.011 (4.47)	0.940 / 0.020 (1.58)	1.899 / 0.050 (2.86)	0.654 / 0.014 (5.16)	0.612 / 0.007 (2.43)	1.011 / 0.019 (2.84)
DS + DP	1.021 / 0.010 (0.53)	0.943 / 0.011 (2.94)	0.940 / 0.019 (1.76)	1.899 / 0.051 (3.24)	0.659 / 0.015 (5.16)	0.464 / 0.007 (1.86)	0.988 / 0.019 (2.58)
Ours	0.830 / 0.011 (0.46)	0.710 / 0.013 (3.57)	0.720 / 0.009 (0.66)	0.811 / 0.011 (4.37)	1.167 / 0.016 (3.96)	0.432 / 0.009 (1.24)	**0.779** / **0.012** (**2.37**)

TABLE V: Visual-inertial odometry tracking ATE (deg/m) on TUM-VI after VI-BA (5 KFs, 0.5sec window)

Algorithm	room1	room2	room3	room4	room5	room6	Average
DS 3D	0.688 / 0.036	0.834 / 0.036	1.236 / 0.051	1.501 / 0.079	0.868 / 0.041	0.857 / 0.055	0.997 / **0.050**
DS + DP	0.754 / 0.039	0.804 / 0.039	1.105 / 0.047	1.547 / 0.086	0.829 / 0.033	0.748 / 0.058	**0.964** / **0.050**
Ours	1.330 / 0.187	0.830 / 0.039	1.339 / 0.049	2.114 / 0.294	1.057 / 0.055	1.808 / 0.362	1.413 / 0.164

TABLE VI: Initialization window ATE and scale (deg/m (%)) on TUM-VI with extreme settings (5 KFs, 0.3sec window)

Algorithm	room1	room2	room3	room4	room5	room6	Average
DS 3D	1.475 / 0.026 (13.69)	1.002 / 0.011 (1.25)	2.021 / 0.019 (19.18)	0.673 / 0.024 (8.07)	1.545 / 0.017 (4.36)	0.738 / 0.014 (4.36)	**1.243** / 0.018 (9.20)
DS + DP	1.523 / 0.026 (13.91)	1.043 / 0.012 (0.31)	2.022 / 0.019 (19.19)	0.680 / 0.024 (8.70)	1.677 / 0.023 (1.00)	0.712 / 0.014 (9.28)	1.276 / 0.020 (8.73)
Ours	1.375 / 0.010 (3.50)	0.851 / 0.007 (1.47)	1.707 / 0.014 (6.95)	1.430 / 0.013 (11.80)	1.572 / 0.010 (6.45)	0.710 / 0.013 (8.68)	1.274 / **0.011** (**6.47**)

TABLE VII: Visual-inertial odometry tracking ATE (deg/m) on TUM-VI with extreme settings (5 KFs, 0.3sec window)

Algorithm	room1	room2	room3	room4	room5	room6	Average
DS 3D	1.255 / 0.210	0.859 / 0.043	1.453 / 0.059	2.318 / 0.368	1.457 / 0.045	0.948 / 0.073	1.381 / 0.133
DS + DP	1.246 / 0.205	0.857 / 0.046	1.547 / 0.060	2.245 / 0.295	1.429 / 0.050	0.978 / 0.076	1.384 / 0.122
Ours	0.882 / 0.036	0.825 / 0.044	1.627 / 0.072	1.637 / 0.085	1.533 / 0.064	0.782 / 0.050	**1.214** / **0.059**

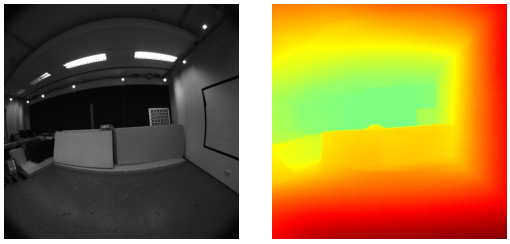

Fig. 4: Qualitative result of the MiDaS [36] v2.1 small on the raw fisheye images of TUM-VI. We found that the network produces reasonable depth maps despite not being explicitly trained for this camera model; however, training the network with fisheye data could potentially improve performance.

the depth priors in the VI-BA, as in [22], improves over the baseline Dong-Si [20] method as expected, but is slightly less-accurate than ours.

In Table V we report the VIO tracking accuracy using the initialization results. While the proposed method yields worse VIO accuracy in this case, this is the only case in all the experiments where we found the proposed method to perform worse – making it overall superior. All methods were successfully initialized for 100% (80/80) of the 10sec windows generated for this experiment.

1) Timing Analysis: Here we investigate the computational cost for the different initialization algorithms on the TUM-VI Room 2 dataset (see Table IX). In particular, we report the network inference time, building and solving the linear system, building and solving the optimization problem, and recovering the covariance. As expected, the proposed method is able to solve the linear system more efficiently due to the simplified linear model and the reduction of state size, but it should be noted that we do not take into account any sparsity when solving the linear system for any method. The depth network inference time is reasonably efficient given it only needs to be performed once for a 0.3 - 0.5 second window and could be done asynchronously. The cost of building and solving the MLE problem is similar across methods, while the covariance recovery takes most of the time – which could be sped up with more engineering effort.

2) Extreme Low-Parallax Scenario: To further showcase the benefit of our method, we investigate a new and *even more* challenging scenario: initialization with 5 keyframes over a *0.3sec window*. To the best of our knowledge, this is the shortest initialization window *ever reported* for monocular VIO with unknown initial conditions.

Table VI reports the ATE and scale error, while Table VII reports the VIO tracking error. The proposed method has overall superior position and scale accuracy in the initialization window, but slightly worse orientation. The VIO tracking accuracy for this extremely challenging scenario shows that the proposed method gains significant accuracy. Additionally, not all methods successfully initialized in every run in this experiment, with both baseline methods being successfully 78 times, while the proposed initialized 80 times out of the 80 10sec windows over the datasets. The initialization is considered successful if all steps are completed and the covariance can be recovered from the final VI-BA result.

3) Robustness to Outliers: We additionally investigate how robust the proposed RANSAC method is to outliers. Given a set of features selected for initialization, a percent of them are selected to be outliers. All observations for these features are perturbed with a normally distributed 10px feature distribution. The mixture of inlier and outlier features are then fed into

TABLE VIII: Outlier ablation study of ATE (deg/m) on TUM-VI dataset with extreme settings (5KFs and 0.3sec window).

Outliers	Algorithm	room1	room2	room3	room4	room5	room6	Average
5%	DS 3D	1.594 / 0.017	0.864 / 0.011	1.441 / 0.014	0.990 / 0.025	1.847 / 0.021	0.805 / 0.015	1.257 / 0.017
	Ours w/o RANSAC	1.103 / 0.012	0.844 / 0.013	1.707 / 0.023	1.573 / 0.017	1.474 / 0.011	0.754 / 0.009	1.242 / **0.014**
	Ours	1.091 / 0.021	1.102 / 0.015	1.219 / 0.015	0.933 / 0.014	1.279 / 0.011	0.661 / 0.009	**1.047** / **0.014**
10%	DS 3D	1.004 / 0.009	0.958 / 0.017	1.586 / 0.014	1.445 / 0.026	1.845 / 0.014	0.840 / 0.016	1.280 / 0.016
	Ours w/o RANSAC	2.148 / 0.014	1.429 / 0.016	0.968 / 0.009	1.978 / 0.017	1.578 / 0.010	0.741 / 0.008	1.474 / 0.012
	Ours	0.983 / 0.010	0.943 / 0.011	0.661 / 0.007	0.935 / 0.016	1.488 / 0.013	0.732 / 0.012	**0.957** / **0.011**
25%	DS 3D	3.683 / 0.027	1.182 / 0.016	2.012 / 0.029	1.819 / 0.017	2.343 / 0.015	0.931 / 0.024	1.995 / 0.021
	Ours w/o RANSAC	3.124 / 0.021	1.506 / 0.017	1.762 / 0.010	2.737 / 0.044	2.814 / 0.022	2.535 / 0.033	2.413 / 0.025
	Ours	2.097 / 0.015	1.111 / 0.009	1.455 / 0.011	1.364 / 0.022	1.503 / 0.014	0.924 / 0.013	**1.409** / **0.014**
45%	DS 3D	6.454 / 0.047	2.280 / 0.046	1.768 / 0.030	2.481 / 0.034	2.700 / 0.016	1.891 / 0.037	2.929 / 0.035
	Ours w/o RANSAC	5.451 / 0.037	2.091 / 0.029	2.049 / 0.023	5.620 / 0.041	5.644 / 0.046	3.355 / 0.056	4.035 / 0.039
	Ours	4.160 / 0.033	1.842 / 0.030	2.064 / 0.013	2.714 / 0.032	2.177 / 0.023	3.023 / 0.051	**2.663** / **0.030**

TABLE IX: Timing analysis of key algorithm components of the baseline and proposed method on all successful initialization in the TUM-VI Room 2 dataset. All values are in seconds.

	DS 3D	Ours w/o RANSAC	Ours
Depth Pred. [36]	-	0.2112 $\pm$ 0.0091	0.2166 $\pm$ 0.0070
Lin. Sys. Build	0.0027 $\pm$ 0.0001	0.0026 $\pm$ 0.0006	0.0027 $\pm$ 0.0003
Lin. Sys. Solve	0.0210 $\pm$ 0.0094	0.0009 $\pm$ 0.0003	0.0024 $\pm$ 0.0008
MLE Build	0.0004 $\pm$ 0.0000	0.0005 $\pm$ 0.0001	0.0005 $\pm$ 0.0001
MLE Solve	0.0155 $\pm$ 0.0078	0.0165 $\pm$ 0.0068	0.0127 $\pm$ 0.0035

TABLE X: Percent of successful initializations on TUM-VI (averaged over all rooms) with 5KFs and 0.3sec window.

Algorithm	60 feats	45 feats	30 feats	15 feats
DS 3D	81.25	17.50	33.75	2.50
DS 1D	**100.00**	81.25	82.50	26.25
DS 3D + DP	78.75	16.25	32.50	2.50
DS 1D + DP	**100.00**	80.00	82.50	25.00
Ours w/o RANSAC	**100.00**	**98.75**	**97.50**	**55.00**
Ours	**100.00**	95.00	96.25	47.50

the rest of the initialization process. Shown in Table VIII, as the outlier percentage increases both the baseline system and non-RANSAC system have increasing errors. The proposed RANSAC method is able to robustly provide reliable initial guesses even in the case of 40% outlier features. We stress that this RANSAC formulation is only enabled by leveraging the scale-less depth map to ensure the state remains independent to the number of features.

4) Robustness to Small Number of Feature Tracks: To showcase the capability of our method to initialize with less information, we experiment with reducing the number of features being tracked during initialization. All experiments up until now have used 75 features, while here we experiment with 60, 45, 30, and 15 features – simulating a reduced number of available measurements due to low texture or other tracking failures. Table X reports the results, and shows that the proposed method can tolerate a severe reduction in the number of features available, while the proposed RANSAC method can still outperform the baselines while remaining also robust to outliers as shown in the previous experiment.

VI. Discussion and Limitations

While we have shown that the proposed method has state-of-the-art initialization performance on short time windows (0.5sec and 0.3sec), we admit that its performance diminishes as the initialization time window increases and more parallax/excitation is available. We believe that this is due to the fact that our method relies on the learned *monocular* depth to aid in the *low excitation* cases, but as a consequence, can not benefit from the classical triangulation that works very well when all the states are observable with sufficient baselines. If extremely fast monocular initialization is desired, then the proposed method reigns supreme, while if a longer initialization window is acceptable or stereo feature tracks are available, we would recommend to simply use a traditional method. We make no claim that the proposed method is able to initialize with *zero* excitation, since some motion and orientation change is required to recover scale. We also do not claim to improve any observability properties of the initialization problem – only that we can reduce the number of states required to be estimated which is shown to improve the robustness.

VII. Conclusion

In this work, we have introduced a new state-of-the-art method to initialize monocular VIO extremely quickly and robustly with the help of a learned single-image depth network. As opposed to utilizing the learned depth in the VI-BA refinement step, we instead proposed to leverage it as known prior information in the fragile linear initialization stage – greatly reducing the number of parameters that need to be estimated. Not only does our method only require the depth to be predicted in one frame instead of all of them, it also conveniently allows for the entire linear initialization to be placed as a small minimal problem in a RANSAC loop – which robustifies the linear system that is already highly unstable outside of ideal conditions. Our results show that we are able to initialize more features in front of the camera than the traditional linear system given the same number of feature tracks, and we display superior initialization accuracy and robustness on two public benchmark datasets (EuRoC and TUM-VI). Additionally, on TUM-VI our method shows an overall superior performance when initializing with only a 0.3sec window of data – which is the shortest ever reported. While our method utilizes monocular depth to aid in initialization, it does not explicitly use it after initialization to benefit the odometry performance as in [50, 51, 52] – which would be an important point to improve upon in the future.

References

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

[2] Google, "ARCore." https://developers.google.com/ar.

[3] Apple, "ARKit." https://developer.apple.com/augmented-reality/.

[4] Meta, "Oculus." https://store.facebook.com/quest/.

[5] C. Chen, Y. Yang, P. Geneva, W. Lee, and G. Huang, "Visual-inertial-aided online mav system identification," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[6] M. Camurri, M. Ramezani, S. Nobili, and M. Fallon, "Pronto: A multi-sensor state estimator for legged robots in real-world scenarios," *Frontiers in Robotics and AI*, vol. 7, p. 68, 2020.

[7] K. J. Wu, C. X. Guo, G. Georgiou, and S. I. Roumeliotis, "VINS on wheels," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5155–5162, IEEE, 2017.

[8] T. Özaslan, G. Loianno, J. Keller, C. J. Taylor, V. Kumar, J. M. Wozencraft, and T. Hood, "Autonomous navigation and mapping for inspection of penstocks and tunnels with mavs," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1740–1747, 2017.

[9] J. Eisele, Z. Song, K. Nelson, and K. Mohseni, "Visual-inertial guidance with a plenoptic camera for autonomous underwater vehicles," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2777–2784, 2019.

[10] D. S. Bayard, D. T. Conway, R. Brockers, J. H. Delaune, L. H. Matthies, H. F. Grip, G. B. Merewether, T. L. Brown, and A. M. San Martin, "Vision-based navigation for the nasa mars helicopter," in *AIAA Scitech 2019 Forum*, p. 1411, 2019.

[11] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[12] 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.

[13] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-inertial mapping with non-linear factor recovery," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 422–429, 2019.

[14] 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*, vol. 37, no. 6, pp. 1874–1890, 2021.

[15] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Rome, Italy), pp. 3565–3572, Apr. 10–14, 2007.

[16] 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.

[17] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 182–201, 2014.

[18] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

[19] 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. https://github.com/rpng/open_vins.

[20] T.-C. Dong-Si and A. I. Mourikis, "Estimator initialization in vision-aided inertial navigation with unknown camera-imu calibration," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1064–1071, IEEE, 2012.

[21] A. Martinelli, "Closed-form solution of visual-inertial structure from motion," *International journal of computer vision*, vol. 106, no. 2, pp. 138–152, 2014.

[22] Y. Zhou, A. Kar, E. Turner, A. Kowdle, C. X. Guo, R. C. DuToit, and K. Tsotsos, "Learned Monocular Depth Priors in Visual-Inertial Initialization," in *European conference on computer vision*, 2022.

[23] M. Li and A. I. Mourikis, "A convex formulation for motion estimation using visual and inertial sensors," in *Proceedings of the Workshop on Multi-View Geometry, held in conjunction with RSS. Berkeley, CA*, 2014.

[24] 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.

[25] T. Qin and S. Shen, "Robust initialization of monocular visual-inertial estimation on aerial robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4225–4232, IEEE, 2017.

[26] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[27] C. Campos, J. M. Montiel, and J. D. Tardós, "Inertial-only optimization for visual-inertial initialization," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 51–57, IEEE, 2020.

[28] D. Zuñiga-Noël, F.-A. Moreno, and J. Gonzalez-Jimenez, "An analytical solution to the imu initialization problem for visual-inertial systems," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6116–6122, 2021.

[29] A. Concha, M. Burri, J. Briales, C. Forster, and L. Oth, "Instant visual odometry initialization for mobile

ar," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 11, pp. 4226–4235, 2021.

[30] T.-C. Dong-Si and A. I. Mourikis, "Closed-form solutions for vision-aided inertial navigation," tech. rep., Dept. of Electrical Engineering, University of California, Riverside, 2011.

[31] A. Martinelli, "Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 44–60, 2011.

[32] J. Kaiser, A. Martinelli, F. Fontana, and D. Scaramuzza, "Simultaneous state initialization and gyroscope bias calibration in visual inertial aided navigation," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 18–25, 2016.

[33] C. Campos, J. M. Montiel, and J. D. Tardós, "Fast and robust initialization for visual-inertial slam," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1288–1294, IEEE, 2019.

[34] G. Evangelidis and B. Micusik, "Revisiting visual-inertial structure-from-motion for odometry and slam initialization," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1415–1422, 2021.

[35] S. Liu, X. Nie, and R. Hamid, "Depth-guided sparse structure-from-motion for movies and tv shows," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15980–15989, 2022.

[36] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1623–1637, 2022.

[37] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 756–770, 2004.

[38] P. Hruby, T. Duff, A. Leykin, and T. Pajdla, "Learning to solve hard minimal problems," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5532–5542, 2022.

[39] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*. AIAA, 1997.

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

[41] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61–76, 2012.

[42] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems XI*, 2015.

[43] K. Eckenhoff, P. Geneva, and G. Huang, "Closed-form preintegration methods for graph-based visual-inertial navigation," *International Journal of Robotics Research*, vol. 38, no. 5, pp. 563–586, 2019.

[44] P. Geneva and G. Huang, "Openvins state initialization: Details and derivations," Tech. Rep. RPNG-2022-INIT, University of Delaware, 2022. Available: https://pgeneva.com/downloads/reports/tr_init.pdf.

[45] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, 2013.

[46] Z. Zhang, G. Gallego, and D. Scaramuzza, "On the comparison of gauge freedom handling in optimization-based visual-inertial state estimation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2710–2717, 2018.

[47] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, 2016.

[48] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers, "The tum vi benchmark for evaluating visual-inertial odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1680–1687, IEEE, 2018.

[49] 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)*, pp. 7244–7251, IEEE, 2018.

[50] X. Zuo, N. Merrill, W. Li, Y. Liu, M. Pollefeys, and G. Huang, "Codevio: Visual-inertial odometry with learned optimizable dense depth," in *Proc. of the IEEE International Conference on Robotics and Automation*, (Xi'an, China), 2021.

[51] L. Koestler, N. Yang, N. Zeller, and D. Cremers, "TANDEM: Tracking and dense mapping in real-time using deep multi-view stereo," in *Conference on Robot Learning (CoRL)*, 2021.

[52] M. Zhao, D. Zhou, X. Song, X. Chen, and L. Zhang, "Dit-slam: real-time dense visual-inertial slam with implicit depth representation and tightly-coupled graph optimization," *Sensors*, vol. 22, no. 9, p. 3389, 2022.