LIC-Fusion: LiDAR-Inertial-Camera Odometry

Xingxing Zuo*, Patrick Geneva††, Woosik Lee†, Yong Liu*, and Guoquan Huang†

Abstract— **This paper presents a tightly-coupled multi-sensor fusion algorithm termed LiDAR-inertial-camera fusion (LIC-Fusion), which efficiently fuses IMU measurements, sparse visual features, and extracted LiDAR points. In particular, the proposed LIC-Fusion performs online spatial and temporal sensor calibration between all three asynchronous sensors, in order to compensate for possible calibration variations. The key contribution is the optimal (up to linearization errors) multi-modal sensor fusion of detected and tracked sparse edge/surf feature points from LiDAR scans within an efficient MSCKF-based framework, alongside sparse visual feature observations and IMU readings. We perform extensive experiments in both indoor and outdoor environments, showing that the proposed LIC-Fusion outperforms the state-of-the-art visual-inertial odometry (VIO) and LiDAR odometry methods in terms of estimation accuracy and robustness to aggressive motions.**

I. Introduction and Related Work

It is essential to be able to accurately track the 3D motion of autonomous vehicles and mobile perception systems. One popular solution is inertial navigation systems (INS) aided with a monocular camera, which has recently attracted significant attention [1], [2], [3], [4], [5], [6], in part because of their complimentary sensing modalities, low cost, and small size. However, cameras are limited by lighting conditions and cannot provide high-quality information in low-light or nighttime conditions. In contrast, 3D LiDAR sensors can provide more robust and accurate range measurements regardless of lighting condition, and are therefore popular for robot localization and mapping [7], [8], [9], [10]. 3D LiDARs suffer from point cloud sparsity, high cost, and lower collection rates as compared to cameras. LiDARs are still expensive as of today, limiting their widespread adoptions, but are expected to have dramatic cost reduction in coming years due to emerging new technology [11]. Inertial measurement units (IMUs) measure local angular velocity and linear acceleration and can provide large amount of information in dynamic trajectories but exhibit large drift due to noises if not fused with other information. In this work, we focus on LiDAR-inertial-camera odometry (LIC) which offers the “best” of each sensor modality to provide a fast and robust 3D motion tracking solution in all scenarios.

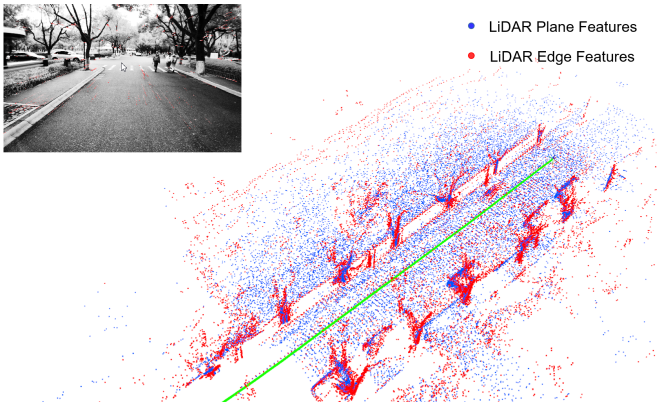

Fig. 1: The proposed LIC-Fusion fuses both sparse visual features tracked in images and LiDAR features extracted in point clouds. The LiDAR points in red and blue are the edge and plane features, respectively. Estimated trajectory is marked in green.

Fusing these multi-modal measurements, in particular, from camera and LiDAR, is often addressed within a SLAM framework [12]. For example, Zhang, Kaess, and Singh [13] associated depth information from LiDAR to visual camera features, resulting in what can be considered as a RGBD system with augmented LiDAR depth. Later, Zhang and Singh [14] developed a general framework for combining visual odometry (VO) and LiDAR odometry which uses high-frequency visual odometry to estimate the overall ego-motion while lower-rate LiDAR odometry, which matches scans to the map and refines the VO estimates. Shin, Park, and Kim [15] have used the depth from LiDAR in a direct visual SLAM method, where photometric errors were minimized in an iterative way. Similarly, in [12] LiDAR was leveraged for augmenting depth to visual features by fitting local planes, which was shown to perform well in autonomous driving scenarios.

Recently, Zhang and Singh [16] developed a laser visual-inertial odometry and mapping system which employed a sequential multi-layer processing pipeline and consists of three main components: IMU prediction, visual-inertial odometry, and scan matching refinement. Specifically, IMU measurements are used for prediction, and the visual-inertial subsystem performs iterative minimization of a joint cost function of the IMU preintegration and visual feature re-projection error. Then, LiDAR scan matching is performed via iterative closet point (ICP) to further refine the prior

This work is supported in part by the National Natural Science Foundation of China under Grant U1509210, 61836015. P. Geneva was also partially supported by the Delaware Space Grant College and Fellowship Program (NASA Grant NNX15AI19H).

*The authors are with the Institute of Cyber-System and Control, Zhejiang University, Hangzhou, China. (Y. Liu is the corresponding author). Email: xingxingzuo@zju.edu.cn, yongliu@iipc.zju.edu.cn

†The authors are with the Department of Mechanical Engineering, University of Delaware, Newark, DE 19716, USA. Email: {ghuang,woosik}@udel.edu

††The author is with the Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA. Email: pgeneva@udel.edu

pose estimates from the VIO module. Note that both the iterative optimization and ICP require sophisticated pipelines and parallel processing to allow for realtime performance. Note also that this essentially is a loosely-coupled fusion approach because only the pose estimation results from the VIO is fed into the LiDAR scan matching subsystem and the scan matching cannot directly process the raw visual-inertial measurements, losing correlation information between LiDAR and VIO.

In this paper, we develop a fast, tightly-coupled, single-thread, LiDAR-inertial-camera (LIC) odometry algorithm with online spatial and temporal multi-sensor calibration within the computationally-efficient multi-state constraint kalman filter (MSCKF) framework [1]. The main contributions of this work are the following:

- We develop a tightly-coupled LIC odometry (termed LIC-Fusion), which enables efficient 6DOF pose estmation with online spatial and temporal calibration. The proposed LIC-Fusion efficiently combines IMU measurements, sparse visual features, and two different sparse LiDAR features (see Figure 1) within the MSCKF framework. The dependence of the calibrated extrinsic parameters and estimated poses on measurements is explicitly modeled and analytically derived.
- We perform extensive experimental validations of the proposed approach on real-world experiments including indoor and outdoor environments, showing that the proposed LIC-Fusion is more accurate and more robust than state-of-the-art methods.

II. The Proposed LIC-Fusion

In this section, we present in detail the proposed LIC-Fusion odometry that tightly fuses LiDAR, inertial, and camera measurements within the MSCKF [1] framework.

A. State Vector

The state vector of the proposed method includes the IMU state $\mathbf{x}_I$ at time k, the extrinsics between IMU and camera $\mathbf{x}_{calib\_C}$, the extrinsics between IMU and LiDAR $\mathbf{x}_{calib\_L}$, a sliding window of clones, including local IMU clones at the past m image times $\mathbf{x}_C$ and at the past n LiDAR scan times $\mathbf{x}_L$. The total state vector is:

$$\mathbf{x} = \begin{bmatrix}\mathbf{x}_I^\top & \mathbf{x}_{calib\_C}^\top & \mathbf{x}_{calib\_L}^\top & \mathbf{x}_C^\top & \mathbf{x}_L^\top\end{bmatrix}^\top \tag{1}$$

where

$$\mathbf{x}_I = \begin{bmatrix}{}^{I_k}_{G}\bar{q}^\top & \mathbf{b}_g^\top & {}^G\mathbf{v}_{I_k}^\top & \mathbf{b}_a^\top & {}^G\mathbf{p}_{I_k}^\top\end{bmatrix}^\top \tag{2}$$

$$\mathbf{x}_{calib\_C} = \begin{bmatrix}{}^{C}_{I}\bar{q}^\top & {}^C\mathbf{p}_I^\top & t_{dC}\end{bmatrix}^\top \tag{3}$$

$$\mathbf{x}_{calib\_L} = \begin{bmatrix}{}^{L}_{I}\bar{q}^\top & {}^L\mathbf{p}_I^\top & t_{dL}\end{bmatrix}^\top \tag{4}$$

$$\mathbf{x}_C = \begin{bmatrix}{}^{I_{a_1}}_{G}\bar{q}^\top & {}^G\mathbf{p}_{I_{a_1}}^\top & \cdots & {}^{I_{a_m}}_{G}\bar{q}^\top & {}^G\mathbf{p}_{I_{a_m}}^\top\end{bmatrix}^\top \tag{5}$$

$$\mathbf{x}_L = \begin{bmatrix}{}^{I_{b_1}}_{G}\bar{q}^\top & {}^G\mathbf{p}_{I_{b_1}}^\top & \cdots & {}^{I_{b_n}}_{G}\bar{q}^\top & {}^G\mathbf{p}_{I_{b_n}}^\top\end{bmatrix}^\top \tag{6}$$

${}^{I_k}_{G}\bar{q}$ is the JPL quaternion [17] corresponding to the 3D rotation matrix ${}^{I_k}_{G}\mathbf{R}$, which denotes the rotation from the global frame of reference $\{G\}$ to the local frame $\{I_k\}$ of IMU at time instant t_k. ${}^G\mathbf{v}_{I_k}$ and ${}^G\mathbf{p}_{I_k}$ represent the IMU velocity and position at time instant t_k in the global frame, respectively. $\mathbf{b}_g$ and $\mathbf{b}_a$ are the biases of gyroscope and accelerometer. ${}^{C}_{I}\bar{q}$ and ${}^C\mathbf{p}_I$ represent the rigid-body transformation between the camera sensor frame $\{C\}$ and the IMU frame $\{I\}$. Analogously, ${}^{L}_{I}\bar{q}$ and ${}^L\mathbf{p}_I$ is the 3D rigid transformation between the LiDAR and IMU frames.

We also co-estimate the time offsets between the exteroceptive sensors and the IMU, which commonly exist in low-cost devices due to sensor latency, clock skew, or data transmission delays. Taking the IMU time to be the "true" base clock, we model that both the camera and LiDAR as having an offset t_{dC} and t_{dL} which can correct the measurement time as follows:

$$t_I = t_C + t_{dC} \tag{7}$$

$$t_I = t_L + t_{dL} \tag{8}$$

where t_C and t_L are the reported time in the camera and LiDAR clock respectively. We refer the reader to [18] for further details.

In the paper, we define that the true value of the state as $\mathbf{x}$, estimated value as $\hat{\mathbf{x}}$, and corresponding error state $\delta\mathbf{x}$, is related by the following generalized update operation:

$$\mathbf{x} = \hat{\mathbf{x}} \boxplus \delta\mathbf{x} \tag{9}$$

The operation $\boxplus$ for a state $\mathbf{v}$ in the vector space is simply the Euclidean addition, i.e., $\mathbf{v} = \hat{\mathbf{v}} + \delta\mathbf{v}$, while for quaternion, it is given by:

$$\bar{q} \approx \begin{bmatrix}\frac{1}{2}\delta\boldsymbol{\theta} \\ 1\end{bmatrix} \otimes \hat{\bar{q}} \tag{10}$$

where $\otimes$ denotes the JPL quaternion multiplication [17].

B. IMU Propagation

The IMU provides angular rate and linear accelerations measurements which we model with the following continuous-time kinematics [17]:

$${}^{I_k}_{G}\dot{\bar{q}}(t) = \frac{1}{2}\boldsymbol{\Omega}\left({}^{I_k}\boldsymbol{\omega}(t)\right){}^{I_k}_{G}\bar{q}(t) \tag{11}$$

$${}^G\dot{\mathbf{p}}_{I_k}(t) = {}^G\mathbf{v}_{I_k}(t) \tag{12}$$

$${}^G\dot{\mathbf{v}}_{I_k}(t) = {}^{I_k}_{G}\mathbf{R}(t)^\top\,{}^{I_k}\mathbf{a}(t) + {}^G\mathbf{g} \tag{13}$$

$$\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg} \tag{14}$$

$$\dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa} \tag{15}$$

where $\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix}-\lfloor\boldsymbol{\omega}\rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^\top & \mathbf{0}\end{bmatrix}$, $\lfloor\cdot\rfloor$ is the skew symmetric matrix, ${}^{I_k}\boldsymbol{\omega}$ and ${}^{I_k}\mathbf{a}$ represent the true angular velocity and linear acceleration in the local IMU frame, and ${}^G\mathbf{g}$ denotes the gravitational acceleration in the global frame. The gyroscope and accelerometer biases $\mathbf{b}_g$ and $\mathbf{b}_a$ are modeled as random walks, which are driven by the white Gaussian noises $\mathbf{n}_{wg}$ and $\mathbf{n}_{wa}$, respectively. This continuous-time system can then be integrated and linearized to propagate the state covariance matrix forward in time [1].

C. State Augmentation

When the system receives a new image or LiDAR scan, the IMU state will propagate forward to that time instant, and the propagated inertial state is cloned into either the $\mathbf{x}_C$ or $\mathbf{x}_L$ state vectors. In order to calibrate the time offsets between different sensors, we will propagate up to IMU time $\hat{t}_{I_k}$, which is the current best estimate of the measurement collection time in the IMU clock. For example, if a new LiDAR scan is received with timestamp t_{L_k}, we will propagate up to $\hat{t}_{I_k} = t_{L_k} + \hat{t}_{dL}$, and augment the state vector $\mathbf{x}_L$ to include this new cloned state estimate:

$$\hat{\mathbf{x}}_{L_k}(\hat{t}_{I_k}) = \begin{bmatrix} {}^{I_k}_{G}\hat{\bar{q}}(\hat{t}_{I_k})^\top & {}^{G}\hat{\mathbf{p}}_{I_k}(\hat{t}_{I_k})^\top \end{bmatrix}^\top \tag{16}$$

We also augment the covariance matrix as:

$$\mathbf{P}(\hat{t}_{I_k}) \leftarrow \begin{bmatrix} \mathbf{P}(\hat{t}_{I_k}) & \mathbf{P}(\hat{t}_{I_k})\mathbf{J}_{I_k}(\hat{t}_{I_k})^\top \\ \mathbf{J}_{I_k}(\hat{t}_{I_k})\mathbf{P}(\hat{t}_{I_k}) & \mathbf{J}_{I_k}(\hat{t}_{I_k})\mathbf{P}(\hat{t}_{I_k})\mathbf{J}_{I_k}(\hat{t}_{I_k})^\top \end{bmatrix} \tag{17}$$

where $\mathbf{J}_{I_k}(\hat{t}_{I_k})$ is the Jacobian of the new cloned $\hat{\mathbf{x}}_{L_k}(\hat{t}_{I_k})$ with respect to the current state (1):

$$\mathbf{J}_{I_k}(\hat{t}_{I_k}) = \frac{\partial \delta \mathbf{x}_{L_k}(\hat{t}_{I_k})}{\partial \delta \mathbf{x}} = \begin{bmatrix} \mathbf{J}_I & \mathbf{J}_{calib\_C} & \mathbf{J}_{calib\_L} & \mathbf{J}_C & \mathbf{J}_L \end{bmatrix}$$

In the above expression, $\mathbf{J}_I$ is the Jacobian with respect to the IMU state $\mathbf{x}_I$, given by:

$$\mathbf{J}_I = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times9} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times9} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{18}$$

$\mathbf{J}_{calib\_L}$ is the Jacobian with respect to the extrinsics (including time offset) between IMU and LiDAR:

$$\mathbf{J}_{calib\_L} = \begin{bmatrix} \mathbf{0}_{6\times6} & \mathbf{J}_{t_{dL}} \end{bmatrix}, \ \mathbf{J}_{t_{dL}} = \begin{bmatrix} {}^{I_k}\hat{\boldsymbol{\omega}}^\top & {}^{G}\hat{\mathbf{v}}_{I_k}^\top \end{bmatrix}^\top \tag{19}$$

and ${}^{I_k}\hat{\boldsymbol{\omega}}$ denotes the local angular velocity of IMU at time $\hat{t}_{I_k}$, and ${}^{G}\hat{\mathbf{v}}_{I_k}$ is the global linear velocity of IMU at time $\hat{t}_{I_k}$. Similarly, $\mathbf{J}_{calib\_C}$, $\mathbf{J}_C$, $\mathbf{J}_L$ are the Jacobian with respect to extrinsics between IMU and camera, clones at camera time, clones at LiDAR time, respectively, which should be zero in this case. It is important to note that the dependence of the new cloned IMU state corresponding to the LiDAR measurement on t_{dL} is modeled via the IMU kinematics and thus allows our measurement models (see Section II-D.1) to be directly a function of the clones which are at the "true" measurement time in the IMU clock frame. This LiDAR cloning procedure is analogous to the procedure used for when a new camera measurement occurs.

D. Measurement Models

1) LiDAR Feature Measurement: To limit the required computational cost, we wish to select a sparse set of high quality features from the raw LiDAR scan for state estimation. In analogy to [7], we extract high and low curvature sections of LiDAR scan rings which correspond to edge and planar surf features respectively (see Figure 1). We track the extracted edge and surf features in the current LiDAR scan back to the previous scan by projecting and finding the closest corresponding features using KD-tree for fast indexing [19]. For example, we project one feature point ${}^{L_{l+1}}\mathbf{p}_{fi}$ in the LiDAR scan $\{L_{l+1}\}$ to $\{L_l\}$, the projected point is denoted as ${}^{L_l}\mathbf{p}_{fi}$:

$${}^{L_l}\mathbf{p}_{fi} = {}^{L_l}_{L_{l+1}}\mathbf{R}\,{}^{L_{l+1}}\mathbf{p}_{fi} + {}^{L_l}\mathbf{p}_{L_{l+1}} \tag{20}$$

where ${}^{L_l}_{L_{l+1}}\mathbf{R}$ and ${}^{L_l}\mathbf{p}_{L_{l+1}}$ are the relative rotation and translation between two LiDAR frames, which can be computed from the states in the state vector:

$${}^{L_l}_{L_{l+1}}\mathbf{R} = {}^{L}_{I}\mathbf{R}\,{}^{I_l}_{G}\mathbf{R}\left({}^{L}_{I}\mathbf{R}\,{}^{I_{l+1}}_{G}\mathbf{R}\right)^\top \tag{21}$$

$${}^{L_l}\mathbf{p}_{L_{l+1}} = {}^{L}_{I}\mathbf{R}\,{}^{I_l}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_{I_{l+1}} - {}^{G}\mathbf{p}_{I_l} + {}^{I_{l+1}}_{G}\mathbf{R}^\top\,{}^{I}\mathbf{p}_L\right) + {}^{L}\mathbf{p}_I \tag{22}$$

$${}^{I}\mathbf{p}_L = -{}^{L}_{I}\mathbf{R}^\top\,{}^{L}\mathbf{p}_I \tag{23}$$

After this tracking, we would find two edge features in the old scan, ${}^{L_l}\mathbf{p}_{fj}, {}^{L_l}\mathbf{p}_{fk}$, corresponding to the projected edge feature ${}^{L_l}\mathbf{p}_{fi}$. We assume they are sampled from the same physical edge as ${}^{L_l}\mathbf{p}_{fi}$. If the closest edge feature ${}^{L_l}\mathbf{p}_{fj}$ is on the r-th scan ring, then the second nearest edge feature ${}^{L_l}\mathbf{p}_{fk}$ should be on the immediate neighboring ring $r-1$ or $r+1$. As a result, the measurement residual of the edge feature ${}^{L_{l+1}}\mathbf{p}_{fi}$ is the distance between its projected feature point ${}^{L_l}\mathbf{p}_{fi}$ and the straight line formed by ${}^{L_l}\mathbf{p}_{fj}$ and ${}^{L_l}\mathbf{p}_{fk}$:

$$r({}^{L_{l+1}}\mathbf{p}_{fi}) = \frac{\left\| \left({}^{L_l}\mathbf{p}_{fi} - {}^{L_l}\mathbf{p}_{fj}\right) \times \left({}^{L_l}\mathbf{p}_{fi} - {}^{L_l}\mathbf{p}_{fk}\right) \right\|_2}{\left\| {}^{L_l}\mathbf{p}_{fj} - {}^{L_l}\mathbf{p}_{fk} \right\|_2} \tag{24}$$

where $\|\cdot\|_2$ is the Euclidean norm and $\times$ denotes the cross product of two vector.

We linearize the above distance measurement of edge features at the current state estimate:

$$\begin{aligned} r({}^{L_{l+1}}\mathbf{p}_{fi}) &= h(\mathbf{x}) + n_r \\ &= h(\hat{\mathbf{x}}) + \mathbf{H}_{\mathbf{x}}\delta\mathbf{x} + n_r \end{aligned} \tag{25}$$

where $\mathbf{H}_{\mathbf{x}}$ is the Jacobian of the distance with respect to the states in the state vector and n_r is modeled as white Gaussian with variance C_r. The non-zero elements in $\mathbf{H}_{\mathbf{x}}$ are only related to the cloned poses ${}^{I_l}_{G}\bar{q}, {}^{G}\mathbf{p}_{I_l}$ and ${}^{I_{l+1}}_{G}\bar{q}, {}^{G}\mathbf{p}_{I_{l+1}}$ along with the rigid calibration between the IMU and LiDAR ${}^{L}_{I}\bar{q}, {}^{L}\mathbf{p}_I$. Thus we have:

$$\mathbf{H}_{\mathbf{x}} = \frac{\partial \delta r({}^{L_{l+1}}\mathbf{p}_{fi})}{\partial^{L_l}\delta\mathbf{p}_{fi}} \frac{\partial^{L_l}\delta\mathbf{p}_{fi}}{\partial\delta\mathbf{x}} \tag{26}$$

the non-zero elements in $\frac{\partial^{L_l}\delta\mathbf{p}_{fi}}{\partial\delta\mathbf{x}}$ are computed as:

$$\begin{aligned} \frac{\partial^{L_l}\delta\mathbf{p}_{fi}}{\partial^{I_l}_{G}\delta\boldsymbol{\theta}} &= {}^{L}_{I}\hat{\mathbf{R}}\lfloor {}^{I_l}_{G}\hat{\mathbf{R}}\,{}^{I_{l+1}}_{G}\hat{\mathbf{R}}^\top\,{}^{L}_{I}\hat{\mathbf{R}}\,{}^{L_{l+1}}\mathbf{p}_{fi} \rfloor \\ &\quad + {}^{L}_{I}\hat{\mathbf{R}}\lfloor {}^{I_l}_{G}\hat{\mathbf{R}}({}^{G}\hat{\mathbf{p}}_{I_{l+1}} - {}^{G}\hat{\mathbf{p}}_{I_l} + {}^{I_{l+1}}_{G}\hat{\mathbf{R}}^\top\,{}^{I}\hat{\mathbf{p}}_L) \rfloor \\ \frac{\partial^{L_l}\delta\mathbf{p}_{fi}}{\partial^{G}\delta\hat{\mathbf{p}}_{I_l}} &= -{}^{L}_{I}\hat{\mathbf{R}}\,{}^{I_l}_{G}\hat{\mathbf{R}} \\ \frac{\partial^{L_l}\delta\mathbf{p}_{fi}}{\partial^{I_{l+1}}_{G}\delta\boldsymbol{\theta}} &= -{}^{L}_{I}\hat{\mathbf{R}}\,{}^{I_l}_{G}\hat{\mathbf{R}}\,{}^{I_{l+1}}_{G}\hat{\mathbf{R}}^\top \lfloor {}^{L}_{I}\hat{\mathbf{R}}^\top\,{}^{L_{l+1}}\mathbf{p}_{fi} + {}^{I}\hat{\mathbf{p}}_L \rfloor \end{aligned}$$

$$\begin{aligned}\frac{\partial {}^{L_l}\delta \mathbf{p}_{fi}}{\partial {}^G\delta\hat{\mathbf{p}}_{I_{l+1}}} &= {}^L_I\hat{\mathbf{R}}{}^{I_l}_G\hat{\mathbf{R}} \\ \frac{\partial {}^{L_l}\delta \mathbf{p}_{fi}}{\partial {}^L_I\delta\boldsymbol{\theta}} &= \lfloor {}^{L_l}_{L_{l+1}}\hat{\mathbf{R}}({}^{L_{l+1}}\mathbf{p}_{fi} - {}^L\hat{\mathbf{p}}_I)\rfloor \\ &\quad - {}^{L_l}_{L_{l+1}}\hat{\mathbf{R}}\lfloor {}^{L_{l+1}}\mathbf{p}_{fi} - {}^L\hat{\mathbf{p}}_I\rfloor \\ \frac{\partial {}^{L_l}\delta \mathbf{p}_{fi}}{\partial {}^L\delta\hat{\mathbf{p}}_I} &= -{}^{L_l}_{L_{l+1}}\hat{\mathbf{R}} + \mathbf{I}_{3\times 3}\end{aligned}$$

In order to perform EKF update, we need to know the explicit covariance C_r of the distance measurement. As this measurement is not directly obtained from the LiDAR sensor, we propagate the covariance of raw measurements (point) in LiDAR scan C_r. Assuming the covariance of point ${}^{L_{l+1}}\mathbf{p}_{fi}, {}^{L_l}\mathbf{p}_{fj}, {}^{L_l}\mathbf{p}_{fk}$ are $\mathbf{C}_i, \mathbf{C}_j, \mathbf{C}_k$ respectively, C_r can be computed as:

$$\begin{aligned} C_r = \sum_{x=i,j,k} \mathbf{J}_x\mathbf{C}_x\mathbf{J}_x^\top, \quad \mathbf{J}_i = \frac{\partial \delta r({}^{L_{l+1}}\mathbf{p}_{fi})}{\partial^{L_{l+1}}\delta\mathbf{p}_{fi}} \\ \mathbf{J}_j = \frac{\partial \delta r({}^{L_{l+1}}\mathbf{p}_{fi})}{\partial^{L_l}\delta\mathbf{p}_{fj}}, \quad \mathbf{J}_k = \frac{\partial \delta r({}^{L_{l+1}}\mathbf{p}_{fi})}{\partial^{L_l}\delta\mathbf{p}_{fk}} \end{aligned} \tag{27}$$

We perform simple probabilistic outlier rejection based on the Mahalanobis distance:

$$r_m = r({}^{L_{l+1}}\mathbf{p}_{fi})^\top \left(\mathbf{H_x P_x H_x^\top} + C_r\right)^{-1} r({}^{L_{l+1}}\mathbf{p}_{fi})$$

where $\mathbf{P_x}$ denotes the covariance matrix of the related states. r_m should subject to a $\boldsymbol{\chi}^2$ "chi-squared" distribution and thus $r({}^{L_{l+1}}\mathbf{p}_{fi})$ will used in our EKF update if it passes this test.

Similarly, for the projected planar surf features ${}^{L_l}\mathbf{p}_{fi}$, we will find three corresponding surf features, ${}^{L_l}\mathbf{p}_{fj}, {}^{L_l}\mathbf{p}_{fk}, {}^{L_l}\mathbf{p}_{fl}$, which are assumed to be sampled on the same physical plane as ${}^{L_l}\mathbf{p}_{fi}$. The measurement residual of surf feature ${}^{L_{l+1}}\mathbf{p}_{fi}$ is the distance between its projected feature point ${}^{L_l}\mathbf{p}_{fi}$ and the plane formed by ${}^{L_l}\mathbf{p}_{fj}, {}^{L_l}\mathbf{p}_{fk}, {}^{L_l}\mathbf{p}_{fl}$. The covariance propagation of the distance measurement of surf features, linearization and Mahalanobis distance test are similar to the edge feature.

2) Visual Feature Measurement: Given a new image, we similarly propagate and augment the state. FAST features are extracted from the image and tracked into future frames using KLT optical flow. Once a visual feature is lost or has been tracked over the entire sliding window, we triangulate the feature in 3D space using the current estimate of the camera clones [1]. The standard visual feature reprojection error is used in the update. For a given set of feature bearing measurements $\mathbf{z}_i$ of a 3D visual feature ${}^G\mathbf{p}_{fi}$ the general linearized residual is:

$$\begin{aligned}\mathbf{r}(\mathbf{z}_i) &= \mathbf{h}(\mathbf{x}, {}^G\mathbf{p}_{fi}) + \mathbf{n}_r && (28)\\ &= \mathbf{h}(\hat{\mathbf{x}}, {}^G\hat{\mathbf{p}}_{fi}) + \mathbf{H_x}\delta\mathbf{x} + \mathbf{H_f}{}^G\delta\mathbf{p}_{fi} + \mathbf{n}_r && (29)\end{aligned}$$

where $\mathbf{H}_f$ is the Jacobian of visual feature measurement with respect to the 3D feature ${}^G\mathbf{p}_{fi}$ and both Jacobians are evaluated at the current best estimates. Since our measurements are a function of ${}^G\hat{\mathbf{p}}_{fi}$ (see (29)), we leverage the MSCKF nullspace projection to remove this dependency [1]. After the nullspace projection we have:

$$\mathbf{r}_o(\mathbf{z}_i) = \mathbf{H_{xo}}\delta\mathbf{x} + \mathbf{n}_{ro} \tag{30}$$

It should be noted that the Jacobian with respect to the rigid transformation between IMU and camera $\{{}^C_I\bar{q}, {}^C\mathbf{p}_I\}$ is non-zero, which means the transformation between IMU and camera can be calibrated online.

E. Measurement Compression

After linearizing the LiDAR feature and visual feature measurements at current state estimate, we could naively perform an EKF update, but this comes with a large computational cost due to the large number of visual and LiDAR feature measurements. Consider the stack of all measurement residuals and Jacobians (which are from LiDAR or visual features):

$$\mathbf{r} = \mathbf{H_x}\delta\mathbf{x} + \mathbf{n} \tag{31}$$

where $\mathbf{r}$ and $\mathbf{n}$ are vectors with block elements of residual and noise in (25) or (30). By commonly assuming all measurements statistically independent, the noise vector $\mathbf{n}$ would be uncorrelated. To reduce the computational complexity, we employ Givens rotation [20] to perform thin QR to compress the measurements [1], i.e.,

$$\mathbf{H_x} = \begin{bmatrix}\mathbf{Q}_{H1} & \mathbf{Q}_{H2}\end{bmatrix}\begin{bmatrix}\mathbf{T}_H \\ \mathbf{0}\end{bmatrix} \tag{32}$$

where $\mathbf{Q}_{H1}$ and $\mathbf{Q}_{H2}$ are unitary matrices. After the measurement compression, we obtain:

$$\mathbf{r}_c = \mathbf{T}_H\delta\mathbf{x} + \mathbf{n}_c \tag{33}$$

where the compressed Jacobian matrix $\mathbf{T}_H$ should be square with the dimension of the state vector $\mathbf{x}$, and the compressed noise is given by $\mathbf{n}_c = \mathbf{Q}_{H1}^\top\mathbf{n}$. This compressed linear measurement residual is then used to efficiently update the state estimate and coviance with the standard EKF.

III. Experimental Results

To validate the performance of the proposed algorithm, several experiments were performed both in outdoor and indoor environments. The sensor rig, shown in Figure 2, consists of an Xsens MTi-300 AHRS IMU, Velodyne VLP-16 LiDAR, and monochrome global-shutter Blackfly BFLY-PGE-23S6M camera. The extrinsics between sensors are calibrated offline and refined during online estimation. For evaluation, we compare the proposed LIC-Fusion against the state-of-the-art visual-inertial and LiDAR odometry methods. Specifically, we compare the proposed to our implementation of the standard MSCKF-based VIO [1] and the open sourced implementation of LOAM LiDAR odometry [7]. It is also important to note that we directly compare to the output of LOAM which leverages ICP matching to its constructed *global map* and thus has leveraged implicit loop-closure information, while our LIC-Fusion is purely an odometry based method which estimates states in a sliding window, neither maintain a global map, nor leverage loop closures.

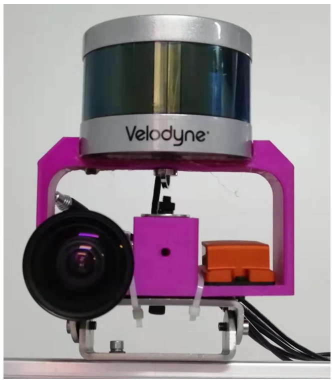

Fig. 2: The self-assembled LiDAR-inertial-camera rig with Velodyne LiDAR, Xsens IMU, and monochrome camera.

TABLE I: Outdoor Experimental Results: Average of average absolute trajectory errors (ATE) and their standard deviation/variability.

	MSCKF [1]	**LIC-Fusion**	**LOAM [7]**
Average ATEs (m)	10.75	4.06	23.08
1 Sigma (m)	3.56	3.42	2.63

A. *Outdoor Tests*

Firstly, the proposed system is tested on an outdoor sequence collected by mounting the self-assembled sensor rig (see Figure 2) on a custom Ackermann robot platform. This outdoor sequence is around 800 meters in length and is recorded over a duration of 4 minutes. RTK GPS with centimeter-level accuracy is also mounted on and the GPS measurements are used as the groundtruth for evaluation.

Each algorithm was run six different times to account for their inherent randomness due to the use of RANSAC and to provide a representative evaluation of typical performance. Figure 3 shows the resulting mean trajectories estimated by the proposed LIC-Fusion, MSCKF, and LOAM. The average mean squared errors (MSE) of each method is presented in Fig. 4, in which the trajectories are aligned to the RTK groundtruth using the “best fit” transform that minimized the overall trajectory error. The proposed LIC-Fusion showed a 2.5 meter decrease in the average error as compared to the standard MSCKF, and 5 meter decreased when compared to LOAM. We can find that the drift of LIC-Fusion grows much slower over time as compared to the other two methods and maintains the smallest error for most of the trajectory. The average absolute trajectory errors (ATE) [21] and their one sigma deviation/variability are also reported in Table I. These results show that the proposed system is able to localize with high accuracy by fusing different sensing modalities (that being camera, inertial, and LiDAR).

B. *Indoor Tests*

We further evaluate the system on a series of indoor datasets which were collected in various normal to low-light lighting conditions with slow to aggressive motion profiles. The indoor sequences are collected by holding the sensor rig (see Figure 2) in hand at chest height. Since groundtruth was not available indoors, we returned the sensor platform to the initial location and evaluate the start-end error. Table II, summarizes the average start-end error results with the trajectories being shown in Figure 5. The results show that the proposed LIC-Fusion is able to localize with high accuracy and is able to handle even extreme cases of high motion and low light due to the fusion of three different sensing modalities. Shown in Figure 6, the Indoor-C sequence recorded while we shook the sensor rig as strongly as we could, hence it has both high angular velocities and high linear accelerations with aggressive motion. The proposed LIC-Fusion is able to localize in this sequence, while the compared two methods fail with large amounts of errors.

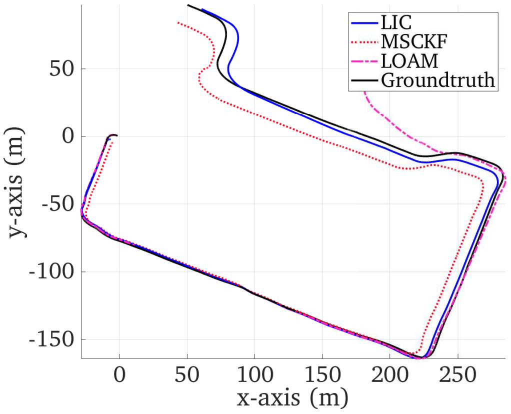

Fig. 3: Top view of outdoor sequence trajectories, showing the trajectories resulted from proposed LIC-Fusion (blue), MSCKF (red), LOAM (pink), and RTK GPS groundtruth (black)

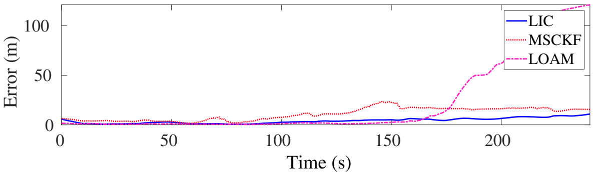

Fig. 4: Average mean squared errors (MSE) of the proposed LIC-Fusion (blue), MSCKF (red), and LOAM (pink) on the outdoor sequence, over the duration of the trajectory.

IV. Conclusions and Future Work

In this paper, we have developed a tightly-coupled efficient multi-modal sensor fusion algorithm for LiDAR-inertial-camera odometry (i.e., LIC-Fusion) within the MSCKF framework. Online spatial and temporal calibration between all three sensors is performed to compensate for calibration sensitivities as well as to ease sensor deployment. The proposed approach detects and tracks sparse edge and pla-

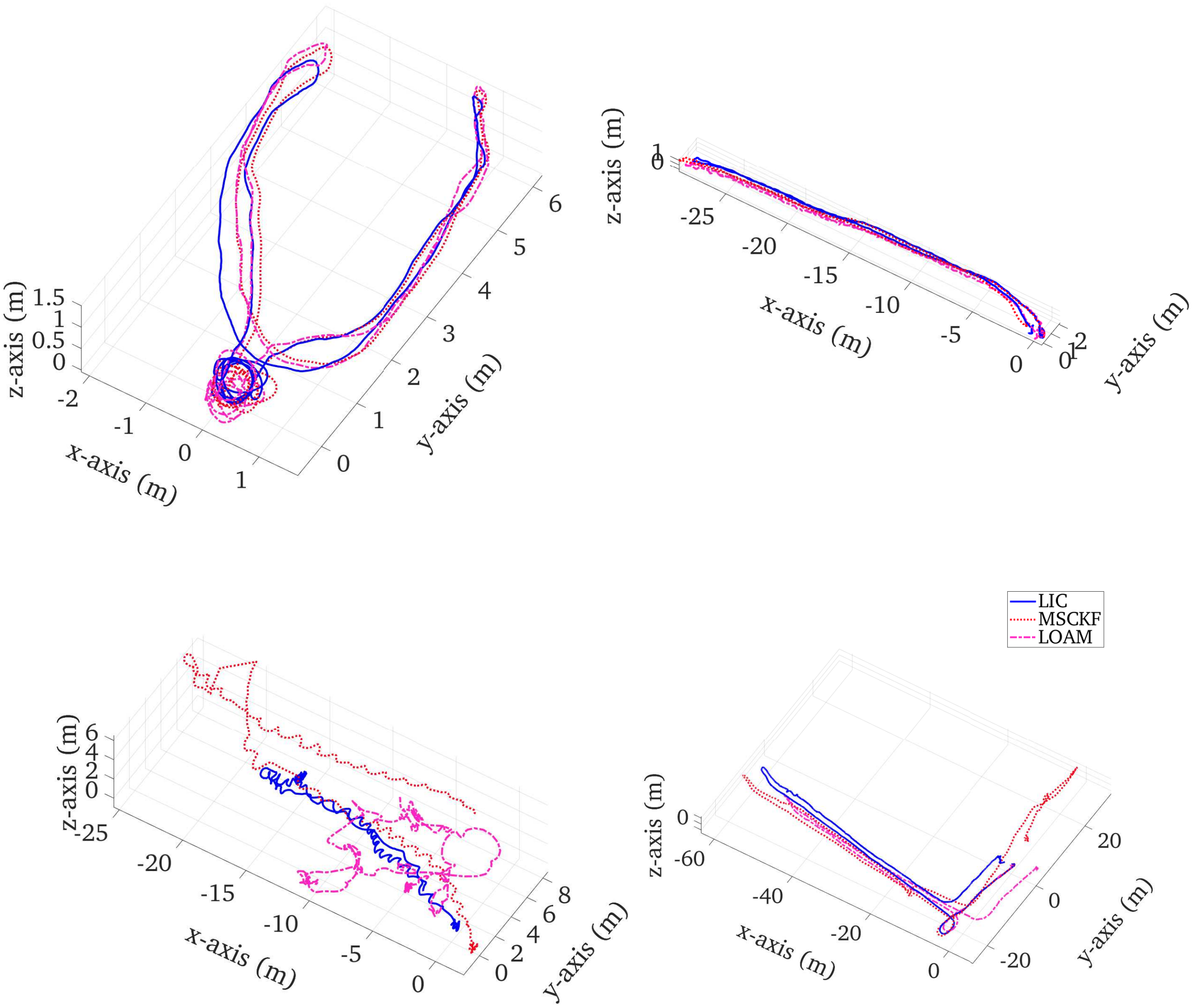

Fig. 5: Isometric views of the estimated trajectories on indoor sequences A, B, D and C (clockwise from top left).

TABLE II: Indoor Experimental Results: Average trajectory start-end errors

Sequence	MSCKF [1]	LIC-Fusion	LOAM [7]
Indoor-A (39m)	0.99	0.98	0.66
Indoor-B (86m)	1.55	1.04	0.46
Indoor-C (55m)	49.94	1.55	2.44
Indoor-D (189m)	46.03	3.68	5.99

nar surf feature points over LiDAR scans and fuses these measurements along with the visual features extracted from monocular images. As a result, by taking advantages of different sensing modalities, the proposed LIC-Fusion is able to provide accurate and robust 6DOF motion tracking in 3D in different environments and under aggressive motions. In the future, we will investigate how to efficiently integrate loop closure constraints obtained from both the LiDAR and camera in order to bound navigation errors.

References

[1] 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.

[2] 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.

[3] 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.

[4] G. Huang, K. Eckenhoff, and J. Leonard, "Optimal-state-constraint EKF for visual-inertial navigation," in *Proc. of the International Symposium on Robotics Research*, (Sestri Levante, Italy), Sept. 12–15, 2015. (to appear).

[5] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial slam using nonlinear optimization," *Proceedings of Robotis Science and Systems (RSS) 2013*, 2013.

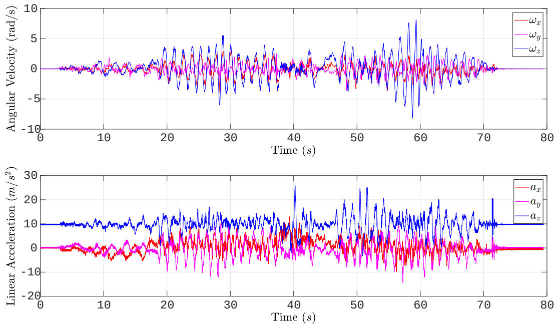

Fig. 6: Raw IMU measurements over the high-dynamic Indoor-C sequence.

[6] Z. Huai and G. Huang, "Robocentric visual-inertial odometry," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Madrid, Spain), Oct. 1-5, 2018.

[7] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time.," in *Robotics: Science and Systems*, vol. 2, p. 9, 2014.

[8] C. Park, S. Kim, P. Moghadam, C. Fookes, and S. Sridharan, "Probabilistic surfel fusion for dense lidar mapping," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2418–2426, 2017.

[9] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4758–4765, IEEE, 2018.

[10] J. Behley and C. Stachniss, "Efficient surfel-based slam using 3d laser range data in urban environments," in *Robotics: Science and Systems (RSS)*, 2018.

[11] Grand View Research, "Lidar technology: Recent developments and market overview." Available: https://www.grandviewresearch.com/blog/lidar-technology-recent-developments-and-market-overview.

[12] J. Graeter, A. Wilczynski, and M. Lauer, "Limo: Lidar-monocular visual odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 7872–7879, IEEE, 2018.

[13] J. Zhang, M. Kaess, and S. Singh, "Real-time depth enhanced monocular odometry," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4973–4980, IEEE, 2014.

[14] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2174–2181, IEEE, 2015.

[15] Y.-S. Shin, Y. S. Park, and A. Kim, "Direct visual slam using sparse depth for camera-lidar system," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, IEEE, 2018.

[16] J. Zhang and S. Singh, "Laser–visual–inertial odometry and mapping with high robustness and low drift," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1242–1264, 2018.

[17] 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.

[18] M. Li and A. I. Mourikis, "Online temporal calibration for camera–imu systems: Theory and algorithms," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 947–964, 2014.

[19] M. De Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf, "Computational geometry," in *Computational geometry*, pp. 1–17, Springer, 1997.

[20] G. H. Golub and C. F. Van Loan, *Matrix computations*, vol. 3. JHU press, 2012.

[21] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 573–580, IEEE, 2012.