IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

A Tightly Coupled VLC-Inertial Localization System by EKF

Qing Liang and Ming Liu

***Abstract*— Lightweight global localization is favorable by many resource-constrained platforms working in GPS-denied indoor environments, such as service robots and mobile devices. In recent years, visible light communication (VLC) has emerged as a promising technology that can support global positioning in buildings by reusing the widespread LED luminaries as artificial visual landmarks. In this paper, we propose a novel VLC/IMU integrated system with a tightly coupled formulation by an extended-Kalman filter (EKF) for robust VLC-inertial localization. By tightly fusing the inertial measurements with the visual measurements of LED fiducials, our EKF localizer can provide lightweight real-time accurate global pose estimates, even in LED-shortage situations. We further complete it with a 2-point global pose initialization method that loosely couples the two sensor measurements. We can hence bootstrap our system with two or more LED features observed in one camera frame. The proposed system and method are verified by extensive field experiments using dozens of self-made LED prototypes.**

I. Introduction

Localization is essential for many robot tasks like planning and navigation, as well as for a wide range of location-based services. We are interested in global solutions in GPS-denied indoor environments. State-of-the-art Lidar odometry and mapping systems [1], [2] can provide consistent low-drift pose estimates using multi-scan Lidar sensors. However, they are computationally intensive for resource-constrained platforms, such as service robots and mobile devices. We aim to reach a lightweight solution that is accurate, consistent, reliable and more easily affordable with inexpensive sensors.

In recent years, localization based on visible light communication (VLC) [3] has emerged as a competitive lightweight solution to be deployed at scale in modern buildings. Besides illumination, LED lights can be reused as artificial landmarks for positioning. The modulated LED broadcasts its unique identity by VLC, which can later be recognized by a rolling-shutter camera. The lights can be mapped once for all, as they are normally fixed and not easily vulnerable to environmental changes. Hence, we are solving a localization problem with known data associations via VLC and a priori map. We can obtain camera poses by solving a perspective-n-point (PnP) problem, given more than three LED features observed in one camera frame. Yet we find that such a requirement is usually demanding, if not impossible, to meet in real situations.

*This work was supported by the National Natural Science Foundation of China, under grant No. U1713211, and the Shenzhen Science, Technology and Innovation Commission (SZSTI) under grant JCYJ20160428154842603, the Research Grant Council of Hong Kong SAR Government, China, under Project No. 11210017, No. 21202816, awarded to Prof. Ming Liu (*Corresponding author*).

The authors are with Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong. {qliangah,eelium}@ust.hk

Note that each square-shaped fiducial (e.g., AprilTag [4]) provides four distinctive corner features, which are sufficient to determine a camera pose. By contrast, normal LED lights offer less usable point features in each due to the lack of distinguishable appearance, e.g., one feature for a circular LED. The number of LEDs decodable in a camera view is limited by a couple of practical factors, such as the density of lights, ceiling height, camera field-of-view (FoV) and the effective VLC range supported by chosen hardware. It would deteriorate further with line-of-sight obstruction by surroundings. Therefore, vision-only methods like PnP may suffer the shortage of decodable LEDs in reality. We can relax this problem by a more careful LED arrangement, e.g., using special LEDs of distinguishable appearance or increasing the density of lights. Yet, it may also raise the associated cost during system deployment.

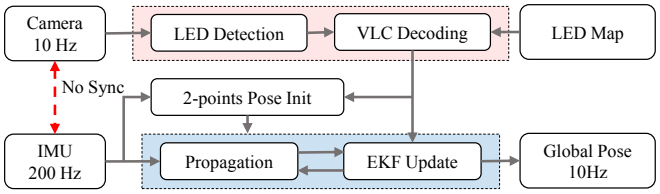


Fig. 1: Overview of the proposed VLC-inertial localization system.

In this work, we aim to overcome the challenge of LED shortage more economically. We are motivated to relax the requirement on the number of LED observations for VLC localization by fusing inertial measurements for improved robustness. Hence, we propose a novel VLC/IMU integrated system with a tightly-coupled formulation by an extended Kalman filter (EKF). We follow the standard EKF-based framework for localization. Especially, our approach exploits visual measurements of VLC-enabled LED luminaries for visual-inertial fusion in a tightly-coupled manner. Moreover, we expect our system to work properly with low-end visual-inertial sensors (e.g., a rolling-shutter camera and an inexpensive IMU) that can be found on low-cost service robots or smartphones. Yet for these sensors, hardware synchronization is not readily available. As such, there may exist a time offset between two sensor streams, e.g., due to different triggering and transmission delays [5], [6]. It may vary slowly further due to clock drift in long-term operation. To solve this problem, we turn to online temporal calibration by following the standard approach proposed by [5], which also adopts an EKF-based formulation for visual-inertial pose estimation.

As shown in Fig. 1, a VLC frontend first extracts LED features with known data associations from a built map by LED detection and VLC decoding. EKF then corrects the

propagated IMU states using such absolute measurements and ensures globally consistent pose estimates free of drift. To initialize the filter's global pose, we introduce a 2-point method based on an IMU-aided P2P solution in [7]. By doing so, our EKF localizer can safely bootstrap from at least two LED features in one camera view, and enable failure recovery in extreme cases of long-term LED outage. The main contributions are summarized as follows:

- A novel VLC/IMU integrated system with a tightly-coupled EKF formulation is proposed for robust VLC-inertial localization in LED-shortage situations.
- A 2-point global pose initialization method is integrated to aid system bootstrapping and failure recovery.
- The system and method are verified by extensive field experiments using a prototype VLC network.

The remainder of this paper is organized as follows. Section II introduces the related works. Section III and Section IV explain our VLC frontend and EKF-based localizer, respectively. Section V presents the experimental results. Section VI concludes this paper.

II. Related Works

A. VLC-based Localization

VLC-based systems [8]–[12] employ modulated LEDs of known locations as landmarks, measure bearings or ranges of visible LEDs with cameras or photodiodes, associate each measurement as per the decoded LED identity by VLC, and solve the location using the measured constraints. Geometry-based methods (e.g., triangulation) need at least three LED features to fix a 3D pose. This is a major cause of their fragile performance in real situations with insufficient visible LEDs. Several methods were proposed to relax this issue by fusing IMU measurements. Epsilon [11] employed an IMU and a digital compass to measure the photodiode's 3D orientation w.r.t. the geomagnetic north. By tedious user intervention, it managed to fix locations in meter-level accuracy using one LED, yet not in real-time. Epsilon may suffer large compass errors due to magnetic anomalies. Lookup [10] solved the camera's 2D position using two LEDs by measuring its roll and pitch angles with an accelerometer, as well as assuming knowledge of the camera's altitude. Further, by knowing the camera's yaw angle from a digital compass, it was able to handle the case of one visible LED for 2D localization. Some larger errors were reported, e.g., due to incorrect orientation measurements of the compass. In this work, we are interested in solving the real-time 3D pose of a free-moving camera in LED shortage or outage scenarios with the aid of an IMU (i.e., not using a compass) by tightly-coupled fusion.

B. Pose Estimation with Fiducial Markers

The paper printable squared fiducials [4], [13] are popular artificial visual landmarks for lightweight pose estimation in robot applications. Similar to modulated LEDs, the fiducial maker can be uniquely identified by its encoded code patterns from a camera image. Yet, each marker can provide four distinctive corner features. By integrating inertial measurements, in either a loosely- or tightly-coupled manner, some methods [14]–[16] can provide very accurate and robust pose estimates with fiducials. They have a trivial solution to the pose initialization problem, as it is sufficient to determine the camera pose from a single observation of known fiducials. By contrast, it is more technically difficult to obtain an initial pose guess for our system, especially under the LED-shortage condition. We note that fiducial-based systems are suitable for specialized robot workspaces where the environmental appearance is of no concern, such as warehouses, factories, and laboratories. However, fiducials may look unappealing or even weird in daily environments, such as shopping malls and museums. As an alternative, our LED-based system can be naturally compatible with most daily scenarios, as well as some specialized workspaces.

III. VLC with a Rolling-shutter Camera

The time-varying light signals from LEDs are perceived by the rolling-shutter camera as spatially-varying strips. We intend to retrieve VLC messages from such barcode-like strip patterns. To do so, we first extract candidate regions from the image that may possibly contain LEDs. For each of them, we try to decode its unique identity (ID) and find its normalized centroid pixel as feature measurements. We can then obtain its absolute 3D position from a prebuilt LED feature map.

A. VLC Preliminaries

We consider a rolling-shutter camera with row exposure time τ_e and row read-out time τ_r. The effective sampling rate, also known as the rolling-shutter frequency [17], is $f_s = 1/\tau_r$. We assume that the LED transmitter can switch on and off under the control of binary signals. We use an on-off-keying (OOK) modulation scheme with Manchester coding for data packaging. The OOK modulation frequency is $f_m = 1/\tau_m$ with τ_m as the sampling interval. That is, τ_m is the minimum pulse duration in the modulated binary signals. The upper bound of the square wave fundamental frequency is $f_h = f_m/2$. To recover the signals, the Nyquist sampling theorem must apply[1], i.e., $f_h < f_s/2$ and hence $f_m < f_s$. The modulated pulses are captured by the camera as bright or dark strips with varying widths proportional to the pulse durations. The minimum strip width, measured in pixels, is computed as $\mathrm{w}_0 = \tau_m/\tau_r$. An L-bit long data packet yields a strip pattern extending $\mathrm{w}_0 L$ pixels in height. That is, to recover the complete information carried by the data packet, we need a strip pattern with at least $\mathrm{w}_0 L$ rows of pixels. The pattern size is bounded by the image height H, i.e., $\mathrm{w}_0 L \leq H$. It follows $\tau_r < \tau_m \leq \tau_r H/L$.

We further consider a circular-shaped LED of diameter A. The maximum image size S of the LED radiation surface at a given distance d is described by $S = Af/d$, where f is the camera focal length in pixels. The data packet is decodable only if the condition $S \geq \mathrm{w}_0 L$ holds:

$$d \leq d_{\mathrm{m}} = \frac{Af}{\mathrm{w}_0 L} = \frac{\tau_r A f}{\tau_m L} \tag{1}$$

[1]We consider the fundamental frequency components for analysis convenience. In a more strict sense, we should consider high-order harmonics of the square wave signals. For example to recover its third-order harmonics, we should have $3f_h < f_s/2$ and $f_m < f_s/3$.

where d_{m} is the maximum range for VLC decoding. The determining factors include the focal length f and row read-out time τ_r of the rolling-shutter camera, the radiation surface size A and the OOK modulation interval τ_m of the LED transmitter, and also the data packet size L in use.

B. Protocol Definition

The designed data packet begins with a 4-bit preamble $\mathrm{PS} = \mathrm{b0001}$, precedes with a 16-bit Manchester-coded data payload DATA, and ends with another 4-bit symbol $\mathrm{ES} = \mathrm{b0111}$. This format yields a 24-bit long data packet with balanced DC-components to circumvent the LED flicker issue. The payload carries one byte of IDs, e.g., labeling up to 256 LEDs. The channel capacity can be extended by a larger payload. Yet, we are motivated to improve the maximum VLC decoding distance d_{m} by using a smaller packet size L instead, as suggested by Eq. 1, due to hardware limitations in our implementation. To do so, we further omit any special packet section for error checking or data recovery.

C. LED Detection

Rolling-shutter cameras can capture strip patterns from a flashing LED during underexposure. Natural features are not observable, while bright objects (e.g., LEDs) can be easily distinguished. Normally, the strips are parallel to image rows and interleaving in the column direction. We are interested in those regions as they carry VLC information. To locate the bright blobs in the image and extract such regions of interest (ROI), we first binarize the grayscale image by thresholding. We then dilate the binary image in the column direction to fill strip gaps. After that, the bright strips from a given LED can join together as a connected blob. We detect blobs and retain large ones as ROIs for subsequent VLC decoding, as they are more likely to carry a complete data packet. We crop the grayscale image using the ROI masks and send the cropped images to the VLC decoder. In addition, the centroid pixel for each ROI is undistorted and normalized with the calibrated camera intrinsics, as image measurements of the LED feature. Readers may refer to our technical report [18] for more details. Note that the perspective projection of the LED (e.g., a circle) centroid, in general, does not squarely coincide with the centroid of the LED image (e.g., an oval). Yet in practice, such an approximation error is acceptable for small objects and can be accommodated by the image noise.

D. VLC Decoding

VLC information is encoded by strips of varying widths. In each ROI, we pick up the grayscale pixels in the centering column. We consider the column pixels as 1D time-varying intensity signals, as the camera's sampling frequency is fixed and known. The binary versions are used for OOK demodulation and Manchester decoding. We adopt adaptive thresholding to counter the nonuniform illumination artifacts of LEDs. Now we can obtain the LED's ID from the decoding result. The data packet may start at a random location in an ROI due to the asynchronous communication mechanism. It happens that only shifted packet versions are available in some ROIs. To address this problem, we adopt a bidirectional decoding scheme [17] to improve decoding success rates. Note that decoding mistakes may happen due to the lack of a special data integrity checking mechanism in our protocol. Therefore, the pose estimator should be resilient to possible data association errors.

E. Implementation Details

We customize dozens of battery-powered LEDs as VLC transmitters. The LED has a circular radiation surface of diameter 15.5cm. The rating power is around 3 watts. We employ a cheap microcontroller to run the VLC protocol on its firmware and use a MOSFET transistor for driving the LED current. The modulation frequency f_m is set to 16kHz. We use a Raspberry Pi rolling-shutter camera (Sony IMX219 with a vertical FoV of 48.8 deg) as the VLC receiver. It has a focal length of 1284 pixels under the image resolution of 1640 by 1232. We manually adjust the camera exposure time to capture sharp patterns. We experimentally determine the maximum decoding distance of our hardware setup be around 2.5m, which coincides with the theoretical upper bound of 2.76m computed from Eq. 1. Readers interested in details can refer to our technical report [18].

IV. Global Localization by EKF

We consider an indoor environment with modulated LED lights at known locations (e.g., on the ceiling). The EKF uses the camera observations to known LED features extracted by the VLC frontend to correct its state estimates, after bootstrapping from 2-point global pose initialization.

A. Notations

We define a gravity-aligned global reference frame $\{G\}$ with its z-axis pointing upwards to the ceiling. The gravity vector expressed in $\{G\}$ is ${}^{G}\mathbf{g} = [0,\, 0,\, -g]$. The IMU frame $\{I\}$ and camera frame $\{C\}$ are rigidly connected. The two sensors run freely without any hardware or software synchronization. The IMU-camera spatial transformation can be obtained from offline calibration or manual measurements. To account for calibration inaccuracy, we further include these extrinsic parameters in the filter state for refinement by joint estimation. Besides this, the time offset t_d between the two sensors is assumed as an unknown constant. We use the IMU time as the time reference, i.e., $t_{imu} = t_{cam} + t_d$, following the convention in [5]. For a camera image timestamped at t, its actual sampling time instance is $t + t_d$. We use the unit quaternion ${}^{A}_{B}\bar{\mathbf{q}}$ under JPL convention [19] to represent the rotation ${}^{A}_{B}\mathbf{R}$ from frame $\{B\}$ to $\{A\}$, i.e., ${}^{A}_{B}\mathbf{R} = \mathbf{R}\left({}^{A}_{B}\bar{\mathbf{q}}\right)$. $\otimes$ denotes the quaternion multiplication. $\lfloor \cdot_{\times} \rfloor$ denotes the skew-symmetric matrix. For a quantity $\mathbf{a}$, we use $\hat{\mathbf{a}}$ for its estimate and $\tilde{\mathbf{a}}$ for the residue.

B. Filter State Definition

The IMU state $\mathbf{x}_I \in \mathbb{R}^{24}$ is defined as follows [5]:

$$\mathbf{x}_I = \left[{}^{I}_{G}\bar{\mathbf{q}}^{\top}\ {}^{G}\mathbf{p}_I^{\top}\ {}^{G}\mathbf{v}_I^{\top}\ \mathbf{b}_g^{\top}\ \mathbf{b}_a^{\top}\ {}^{C}_{I}\bar{\mathbf{q}}^{\top}\ {}^{C}\mathbf{p}_I^{\top}\ t_d\right]^{\top} \tag{2}$$

where ${}^{I}_{G}\bar{\mathbf{q}}$ is the unit quaternion that describes the rotation ${}^{I}_{G}\mathbf{R}$ from $\{G\}$ to $\{I\}$, i.e., ${}^{I}_{G}\mathbf{R} = \mathbf{R}\left({}^{I}_{G}\bar{\mathbf{q}}\right)$; ${}^{G}\mathbf{p}_I$ and ${}^{G}\mathbf{v}_I$

are the global IMU position and velocity, respectively; $\mathbf{b}_g$ and $\mathbf{b}_a$ are the gyroscope and accelerometer biases; ${}^C_I\bar{\mathbf{q}}$ is the unit quaternion that represents the rotation ${}^C_I\mathbf{R}$ from the IMU frame $\{I\}$ to the camera frame $\{C\}$; ${}^C\mathbf{p}_I$ denotes the IMU position in $\{C\}$; and t_d is the time offset.

The error state $\tilde{\mathbf{x}}_I \in \mathbb{R}^{22}$ is then given by:

$$\tilde{\mathbf{x}}_I = \begin{bmatrix} {}^I\tilde{\boldsymbol{\theta}}^\top & {}^G\tilde{\mathbf{p}}_I^\top & {}^G\tilde{\mathbf{v}}_I^\top & \tilde{\mathbf{b}}_g^\top & \tilde{\mathbf{b}}_a^\top & {}^I\tilde{\boldsymbol{\phi}}^\top & {}^C\tilde{\mathbf{p}}_I^\top & \tilde{t}_d \end{bmatrix}^\top \quad (3)$$

where for quaternions, we employ the multiplicative error definition with local perturbations in the IMU frame. That is, we have ${}^I_G\bar{\mathbf{q}} \simeq \begin{bmatrix} \frac{1}{2}{}^I\tilde{\boldsymbol{\theta}} \\ 1 \end{bmatrix} \otimes {}^I_G\hat{\bar{\mathbf{q}}}$ and ${}^C_I\bar{\mathbf{q}} \simeq {}^C_I\hat{\bar{\mathbf{q}}} \otimes \begin{bmatrix} \frac{1}{2}{}^I\tilde{\boldsymbol{\phi}} \\ 1 \end{bmatrix}$ where ${}^I\tilde{\boldsymbol{\theta}}$ and ${}^I\tilde{\boldsymbol{\phi}}$ are the 3×1 small angle rotation error vectors expressed in $\{I\}$. The standard additive errors apply to other quantities, e.g., ${}^G\mathbf{p}_I = {}^G\hat{\mathbf{p}}_I + {}^G\tilde{\mathbf{p}}_I$.

C. IMU Propagation

The IMU measures the true angular velocity ${}^I\boldsymbol{\omega}$ and linear acceleration ${}^I\mathbf{a}$ in its local frame $\{I\}$ as [5]: $\boldsymbol{\omega}_m = {}^I\boldsymbol{\omega} + \mathbf{b}_g + \mathbf{n}_g$ and $\mathbf{a}_m = {}^I\mathbf{a} - {}^I_G\mathbf{R}{}^G\mathbf{g} + \mathbf{b}_a + \mathbf{n}_a$. The measurement errors are modeled as zero-mean white Gaussian noises, i.e., $\mathbf{n}_g \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_g^2)$ and $\mathbf{n}_a \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_a^2)$. The continuous-time dynamics of the state $\mathbf{x}_I$ is given by:

$$\begin{aligned} &{}^I_G\dot{\bar{\mathbf{q}}} = \frac{1}{2}\boldsymbol{\Omega}\left({}^I\boldsymbol{\omega}\right){}^I_G\bar{\mathbf{q}},\ {}^G\dot{\mathbf{p}}_I = {}^G\mathbf{v}_I,\ {}^G\dot{\mathbf{v}}_I = {}^I_G\mathbf{R}^\top{}^I\mathbf{a}, \\ &\dot{\mathbf{b}}_g = \mathbf{n}_{wg},\ \dot{\mathbf{b}}_a = \mathbf{n}_{wa},\ {}^C_I\dot{\bar{\mathbf{q}}} = \mathbf{0},\ {}^C\dot{\mathbf{p}}_I = \mathbf{0},\ \dot{t}_d = 0 \end{aligned} \quad (4)$$

where $\boldsymbol{\Omega}({}^I\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor {}^I\boldsymbol{\omega}_\times \rfloor & {}^I\boldsymbol{\omega} \\ -{}^I\boldsymbol{\omega}^\top & 0 \end{bmatrix}$; and $\mathbf{n}_{wg}$ and $\mathbf{n}_{wa}$ are the underlying noise processes that drive the IMU biases, with $\mathbf{n}_{wg} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{wg}^2)$ and $\mathbf{n}_{wa} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{wa}^2)$. Note that, for long-term operation, the time offset may vary slowly due to clock drift between unsynchronized sensors. We can then model it as a random walk process, i.e., $\dot{t}_d = \mathbf{n}_{td}$ with $\mathbf{n}_{td}$ as the underlying zero-mean Gaussian noise. The propagation of nominal state $\hat{\mathbf{x}}_I$ derives from the expectation of Eq. 4:

$$\begin{aligned} {}^I_G\dot{\hat{\bar{\mathbf{q}}}} = \frac{1}{2}\boldsymbol{\Omega}\left(\hat{\boldsymbol{\omega}}\right){}^I_G\hat{\bar{\mathbf{q}}},\ {}^G\dot{\hat{\mathbf{p}}}_I = {}^G\hat{\mathbf{v}}_I,\ {}^G\dot{\hat{\mathbf{v}}}_I = {}^I_G\hat{\mathbf{R}}^\top\hat{\mathbf{a}} + {}^G\mathbf{g} \end{aligned} \quad (5)$$

where $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g$, $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$, and ${}^I_G\hat{\mathbf{R}} = \mathbf{R}\left({}^I_G\hat{\bar{\mathbf{q}}}\right)$. The other quantities such as $\hat{\mathbf{b}}_g$ remain constant. We can now predict $\hat{\mathbf{x}}_I$ in discrete-time by numerical integration.

The linearized continuous-time error state equation is written as $\dot{\tilde{\mathbf{x}}}_I = \mathbf{F}\,\tilde{\mathbf{x}}_I + \mathbf{G}\,\mathbf{n}_I$, where $\mathbf{n}_I = \begin{bmatrix} \mathbf{n}_g^\top & \mathbf{n}_{wg}^\top & \mathbf{n}_a^\top & \mathbf{n}_{wa}^\top \end{bmatrix}^\top$ denotes the continuous-time IMU noise with its covariance matrix $\mathbf{Q}_c$ as $\mathbf{diag}\{\boldsymbol{\sigma}_g^2, \boldsymbol{\sigma}_{wg}^2, \boldsymbol{\sigma}_a^2, \boldsymbol{\sigma}_{wa}^2\}$. The detailed expressions of the system matrix $\mathbf{F}$ and the noise input matrix $\mathbf{G}$ are omitted here due to space limitations and can be found in our technical report [18]. We can now propagate the EKF state covariance using the discrete-time implementation of the above-mentioned error state equation.

D. Camera Measurement Update

We assume a calibrated pinhole camera model[2]. For an image timestamped at t, we consider the ith feature f_i of the decoded LEDs from the VLC frontend. Its measurement $\{\mathbf{z}_i, {}^G\mathbf{p}_{f_i}\}$ is known, where $\mathbf{z}_i$ is the normalized pixel of the LED centroid and ${}^G\mathbf{p}_{f_i}$ is the global LED position. The feature observation $\mathbf{z}_i$ taken at camera time t is given by:

$$\begin{aligned} \mathbf{z}_i(t) &= \mathbf{h}\left({}^C\mathbf{p}_{f_i}(t+t_d)\right) + \mathbf{n}_{im}(t+t_d) \\ {}^C\mathbf{p}_{f_i}(t+t_d) &= {}^C_I\mathbf{R}\,{}^I_G\mathbf{R}(t+t_d)\left({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_I(t+t_d)\right) + {}^C\mathbf{p}_I \end{aligned} \quad (6)$$

where $\mathbf{n}_{im} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{im}^2)$ is the image noise; $\mathbf{h}(\cdot)$ is the perspective projection function, i.e., $\mathbf{h}\left([x,y,z]^\top\right) = [x/z, y/z]^\top$; and ${}^C\mathbf{p}_{f_i}$ is the feature position with respect to the current camera frame at IMU time $t + t_d$.

With the latest state estimate $\hat{\mathbf{x}}_I(t + \hat{t}_d)$ from the IMU propagation, we can now derive the expected measurement as $\hat{\mathbf{z}}_i(t) = \mathbf{h}\left({}^C\hat{\mathbf{p}}_{f_i}(t+\hat{t}_d)\right)$, and computed its residue term $\tilde{\mathbf{z}}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i$ by first-order approximation: $\tilde{\mathbf{z}}_i \simeq \mathbf{H}_{\mathbf{x},i}\tilde{\mathbf{x}}_I + \mathbf{H}_{f_i}{}^G\tilde{\mathbf{p}}_{f_i} + \mathbf{n}_{im}$. The LED position error ${}^G\tilde{\mathbf{p}}_{f_i}$ is modeled as zero-mean white Gaussian noise with covariance $\boldsymbol{\sigma}_f^2$. The measurement Jacobian w.r.t. the IMU state $\mathbf{H}_{\mathbf{x},i}$ and the Jacobian w.r.t. the LED feature position $\mathbf{H}_{f_i}$ are given by:

$$\begin{aligned} &\mathbf{H}_{\mathbf{x},i} = \begin{bmatrix} \mathbf{H}_{\boldsymbol{\theta},i} & \mathbf{H}_{\mathbf{p},i} & \mathbf{0}_{2\times 9} & \mathbf{H}_{\boldsymbol{\phi},i} & \mathbf{H}_{\mathbf{p}_c,i} & \mathbf{H}_{t_d,i} \end{bmatrix} \\ &\mathbf{H}_{\boldsymbol{\theta},i} = \mathbf{J}_i\,{}^C_I\hat{\mathbf{R}}\left\lfloor {}^I_G\hat{\mathbf{R}}\left({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_I\right)_\times \right\rfloor \\ &\mathbf{H}_{\mathbf{p},i} = -\mathbf{J}_i\,{}^C_I\hat{\mathbf{R}}\,{}^I_G\hat{\mathbf{R}},\ \mathbf{H}_{\boldsymbol{\phi},i} = \mathbf{H}_{\boldsymbol{\theta},i},\ \mathbf{H}_{\mathbf{p}_c,i} = \mathbf{J}_i \\ &\mathbf{H}_{t_d,i} = \mathbf{H}_{\boldsymbol{\theta},i}\,\hat{\boldsymbol{\omega}} + \mathbf{H}_{\mathbf{p},i}\,{}^G\hat{\mathbf{v}}_I,\ \mathbf{H}_{f_i} = -\mathbf{H}_{\mathbf{p},i} \end{aligned} \quad (7)$$

where $\mathbf{J}_i = \partial\mathbf{h}(\mathbf{f})/\partial\mathbf{f}$ is the Jacobian of $\mathbf{h}(\cdot)$ evaluated at the expected feature position in the camera frame ${}^C\hat{\mathbf{p}}_{f_i} = [\hat{x}, \hat{y}, \hat{z}]^\top$, i.e., $\mathbf{J}_i = \frac{1}{\hat{z}}\begin{bmatrix} 1 & 0 & -\hat{x}/\hat{z} \\ 0 & 1 & -\hat{y}/\hat{z} \end{bmatrix}$.

The filter state and covariance estimates can be updated by following the general EKF equations [19]. To account for false data associations from VLC decoding errors, we perform the Mahalanobis gating test for each observation before the measurement update. The EKF can naturally process multiple LED observations in a single image if more LEDs are successfully decoded.

E. 2-point Pose Initialization

For global localization, we need to initialize the filter with a 6-degrees-of-freedom (DoF) pose w.r.t. the global frame, as well as its velocity. Since vision-only methods like PnP easily suffer from large errors or failure in LED-shortage scenarios, we steer to an IMU-aided P2P solution that can work more reliably with two point-feature measurements [7]. IMU measures roll and pitch angles accurately by monitoring gravity, leaving four unknowns in the camera pose. It has been proved that there are two closed-form solutions to this problem [7]. In our applications, moreover, we can obtain a unique solution by checking its z-direction as the sensor suite is alway beneath the ceiling. We further refine the pose by minimizing camera re-projection errors once more than two LEDs are decoded in the image. Specially, we use the P2P solution as an initial guess and optimize the pose in 4-DoF by fixing its roll and pitch. The IMU-centric pose can be resolved given the sensor extrinsics. The velocity computed by pose difference is noisy and unreliable to use, especially

[2]In this section, we assume a simplified global-shutter camera measurement model without considering the rolling-shutter effect on feature measurements. We leave this issue for our future work.

when the sensor moves slowly, e.g., for handheld cases and low-speed robots. Alternatively, we provide the filter with zero velocity and a large variance. So far, our system can bootstrap with two or more LEDs decoded in a single image.

V. Experiments

We evaluate our system through real-world experiments. We use the absolute trajectory error (ATE) for global position accuracy and use the axis-angle error for orientation accuracy assessment. We set up a room-sized (5*4*2.3 m^3) test field with 25 LEDs evenly mounted on the ceiling (see Fig. 2). The spacing is around 1-1.5m. We use a customized sensor suite for data collection, as shown by the right side of Fig. 2. It comprises a Raspberry Pi camera (IMX219, 1640*1232 @10Hz) and a MicroStrain IMU (3DM-GX3-25 @200Hz) without any synchronization. The motion capture system (OptiTrack Mocap @120Hz) provides ground truth poses for our experiments. We set the Mocap world frame to coincide with the global frame $\{G\}$. The extrinsic parameters between the camera and the Mocap rigid body (i.e., reflective markers on the sensor suite) are known from hand-eye calibration. We measured the global 3D location of LEDs using Mocap, as well as a commodity laser ranger for height compensation. The algorithm runs on a desktop computer (Intel i7-7700K CPU @4.20GHz, 16G RAM).

A. Localization Performance

To assess the localization performance[3], we have collected a few datasets in eight trials [see Table I]. Specifically, we move the handheld sensor suite smoothly by walking in the test field. We orient the camera upwards facing the ceiling lights. For the ease of filter initialization, we put the sensor on the ground and keep it still for a few seconds at the start of each run. Unless otherwise specified, the global pose in EKF is initialized by the 2-point initialization method, which will later be evaluated in section V-C. The extrinsic parameters $\{{}^C_I\bar{\mathbf{q}}, {}^C\mathbf{p}_I\}$ are initialized with coarse manual measurements. The remaining parameters in the filter (e.g., the IMU biases and time offset) are simply set to zeros.

TABLE I: Description of the eight datasets in use.

Trials	1	2	3	4	5	6	7	8
Time [s]	39.5	33.4	40.7	34.6	66.8	43.4	67.7	133.5
Dist. [m]	30.2	37.1	35.0	27.8	67.6	42.0	69.0	158.6
MaxVel [m/s]	1.40	1.99	1.49	1.36	1.48	1.55	1.51	1.67
Shape	circle		square		eight		random	

Fig. 3 shows the results for trial 7 as we walk randomly in the field for 68s. We use Mocap to denote the ground truth and use EKF for the estimates. As shown in Fig. 3a, the estimated trajectory well matches the ground truth. The global position, orientation and velocity estimates for this trial, as well as the respective errors compared against the ground truth, are shown in Fig. 4. We illustrate the number of decodable LEDs in each camera frame in Fig. 3b. On the one hand, we have a very low chance to decode three or more LEDs in one image despite the dense LED deployment. As such, vision-only methods can rarely be used in our setting. On the other hand, we can concurrently decode two LEDs at a much higher possibility, and thus, bootstrap the proposed system more easily by 2-point initialization. Furthermore, we show the time offset estimate $\hat{t}_d$ in Fig. 3c, as well as its standard deviation from the filter covariance matrix. It converges soon after the sensor starts moving.

The absolute pose errors for the eight trials are shown in Fig. 5, where the position error is evaluated by ATE and the orientation error is based on the axis-angle representation. Fig. 5c shows the time offset estimation results for the last 20s in each run. We can observe that most of these estimates are consistent, e.g., lying between -24ms and -32ms. There is

[3]Online demonstrations can be found in our supplementary video.

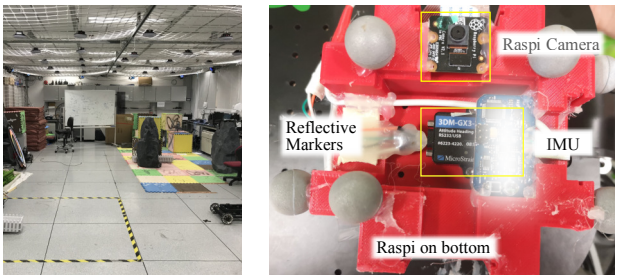


Fig. 2: Test field (left) and self-assembled sensor suite (right).

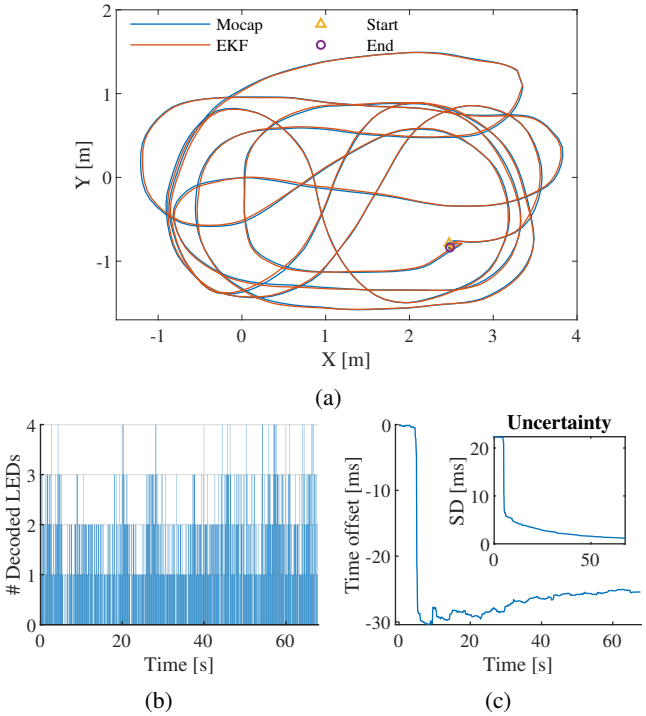
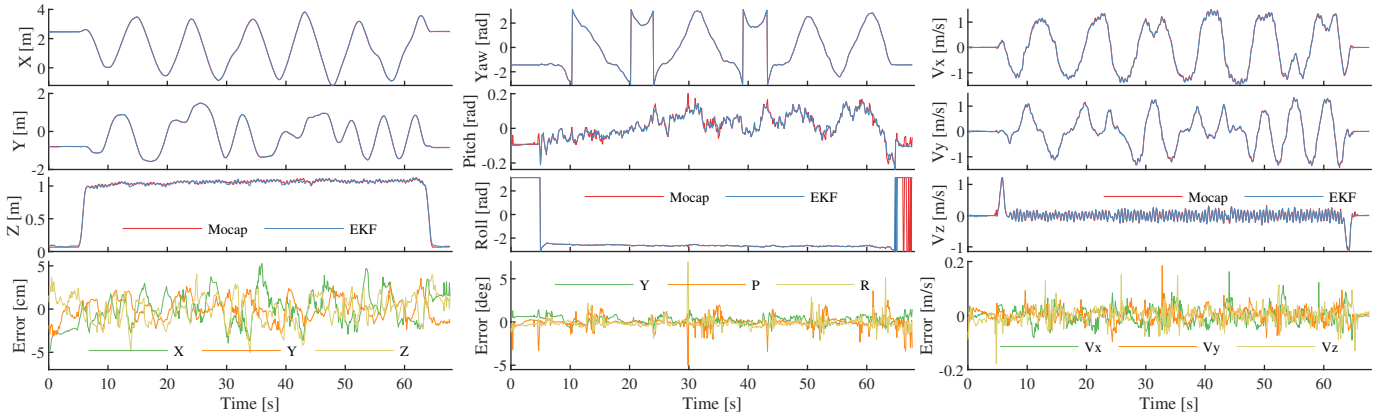
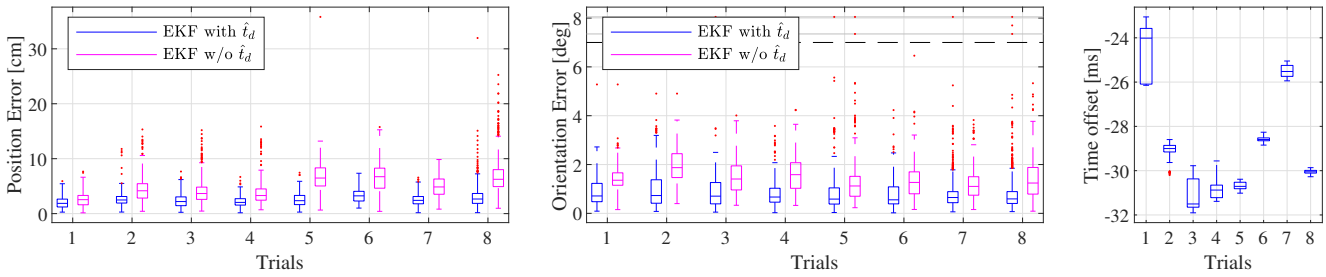


Fig. 3: The random trajectory (a) travels approximately 69m in 68s. The EKF estimates are very close to the ground truth by visual comparison. (b) shows the number of LEDs that are successfully decoded from each camera frame by the VLC frontend. (c) shows the time offset estimates as well as their uncertainty described by the standard deviation (SD), as shown in the inner subplot.



(a) Position (b) Orientation (c) Velocity

Fig. 4: Global position (a), orientation (b), and velocity (c), as well as their respective errors in trial 7 compared with the ground truth.



(a) Position error (b) Orientation error (c) Time offset

Fig. 5: Absolute position (a) and orientation (b) errors on eight trials. We show the consistency of time offset estimates in (c) by using results over the last 20s, and compare the performance of the proposed method both with and without online temporal calibration.

TABLE II: Statistics on localization errors and counts of decoded LEDs in eight trials using both dense and sparse feature maps.

Trial		1	2	3	4	5	6	7	8
Position Error [cm]	RMSE	2.20 / 2.91	3.02 / 3.91	2.67 / 3.22	2.41 / 3.29	2.80 / 2.99	3.59 / 3.97	2.75 / 3.00	3.47 / **4.00**
	std	0.96 / 1.43	1.40 / 1.88	1.23 / 1.47	0.97 / 1.43	1.20 / 1.39	1.30 / 1.63	1.05 / 1.46	1.79 / 2.11
Rotation Error [deg]	RMSE	1.07 / 1.09	**1.27** / 1.25	1.12 / 1.15	1.00 / 0.97	1.22 / 1.21	0.99 / 1.00	1.10 / 1.12	1.04 / 1.06
	std	0.59 / 0.61	0.80 / 0.74	0.71 / 0.73	0.57 / 0.58	0.91 / 0.90	0.59 / 0.57	0.73 / 0.73	0.68 / 0.74
#LED	mean	1.59 / 0.86	1.38 / 0.69	1.21 / 0.67	1.21 / 0.69	1.20 / 0.65	1.60 / 0.91	1.79 / 0.95	1.16 / 0.59
Pct. of #LED	≥ 1	0.86 / 0.69	0.83 / 0.58	0.77 / 0.59	0.78 / 0.57	0.83 / 0.61	0.90 / 0.73	0.92 / 0.76	0.77 / 0.54
	≥ 2	0.52 / 0.15	0.42 / 0.11	0.35 / 0.08	0.36 / 0.12	0.32 / 0.05	0.52 / 0.17	0.61 / 0.18	0.33 / 0.05
	≥ 3	0.16 / 0.01	0.11 / 0.00	0.08 / 0.00	0.07 / 0.00	0.05 / 0.00	0.15 / 0.01	0.22 / 0.01	0.05 / 0.00

no ground truth time offset for our sensor suite. It may even vary slightly in different runs due to the lack of hardware synchronization. Further, we study the impact of temporal calibration on our localization performance. With online time offset estimation activated, the proposed method significantly outperforms its counterpart without such a calibration, say on the eight trials in terms of both the global position accuracy and orientation accuracy. We note that the extreme outliers in orientation, as shown in Fig. 5b, are most probably caused by the occasional Mocap tracking errors (especially the rotation) at certain places, e.g., due to the blockage of reflective markers by the experimenter. By revisiting the orientation plots in Fig. 4b, we observe that the yaw direction is consistently smooth while the roll and pitch directions have a few spikes (e.g., at the 30s). Since EKF estimates are normally smooth after converging, those spikes are most probably caused by the Mocap system.

B. Robustness Test under LED Shortage/Outage

We aim to explore the robustness of our system in more challenging scenarios, e.g., with less decodable LEDs in a single view (say LED shortage) or with the complete absence of LEDs in a certain period (say LED outage). These problems may arise from many practical factors, such as the lights deployment density and the maximum VLC decoding range supported by the hardware setup. We here look into the LED shortage problem by altering the deployment density. To do so, we uniformly remove half of the 25 LEDs from

the original dense map. This results in a sparse map with 12 LEDs. We simply discard measurements from those removed ones. Unlike commercial lights for illumination, our prototype LEDs have a much smaller radiation surface (e.g., 15cm in diameter), as well as a reduced VLC decoding range. The 12 circular LEDs are reasonably sparse for localization in the test field. As a comparison, 10 pairs of standard fluorescent tubes are deployed in the same area.

Table. II summarizes the statistics on absolute position and rotation errors, along with the counts of decodable LEDs in the camera view. The results from the dense map are shown before that from the sparse map side by side. We show the root-mean-squared error (RMSE) and the respective standard deviation for the estimated poses. We notice that the position errors increase as the map density decreases, e.g., with larger RMSE errors and standard deviations. Yet, we do not find any substantial variation in rotation errors. The maximum RMSE position error (e.g., 4 cm in trial 8) arises from the sparse map, while the maximum RMSE rotation error (e.g., 1.27 deg in trial 2) comes from the dense map. The average number of decodable LEDs in the sparse map is almost half of that in the dense map, indicating a substantial loss of usable LED features. In the meantime, the performance degradation in positioning accuracy is relatively marginal.

Also, we show the percentage of frames that can decode a certain number of LEDs in Table. II. The percentage of decoding three or more LEDs is extremely low, especially in the sparse map. Meanwhile, we have a much higher possibility to decode one or more LEDs. As we know, EKF can still keep correcting its estimates with one observation only. The chance remains to observe two decodable LEDs simultaneously in the sparse map. As such, our system can still bootstrap from 2-point initialization. Therefore, our method has better usability than those vision-only counterparts.

Further, we explore the system performance in situations with an intermediate LED outage. Specifically, we study the short-term outage problem by dropping different quantities of camera frames in a given period. For example, we can simulate an effective camera rate of 1Hz by dropping 9 out of 10 frames for every 1s. We choose five different camera rates from 10Hz to 0.5Hz. The respective pose errors are shown in Fig. 6. The system can bootstrap on its own at the camera rate of 1Hz. Aided by its normal initialization at 10Hz, the system can finally sustain at 0.5Hz without diverging. In other words, the system is tolerant to a certain short-term LED outage, e.g., less than 2s, during normal walking.

C. 2-point Initialization and Failure Recovery

The 2-point initialization plays important roles in filter bootstrapping and recovery from failure, e.g., due to the long-term LED outage. We want to evaluate the accuracy of our IMU-aided P2P solution with 4-DoF pose refinement. Moreover, we are interested in investigating the impact of the initial pose estimate on the overall localization performance. To this end, we initialize the filter using both the P2P-based solution and the ground truth. Fig. 7 shows the pose errors on trial 1, 3, 5, and 7. We use P2P to denote the results

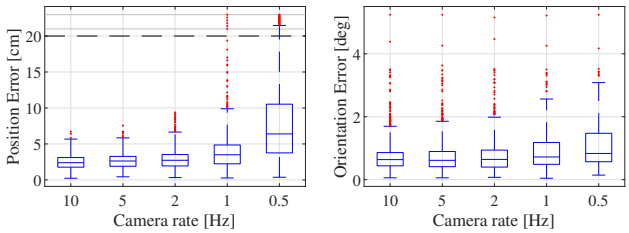


Fig. 6: Pose errors in trial 7 at different camera rates. The maximum position errors are 27cm at 1Hz and 37cm at 0.5Hz. Note that we manually remove an extreme rotation outlier at the 30s (around 10 deg) caused by Mocap tracking errors, as illustrated by Fig. 4b.

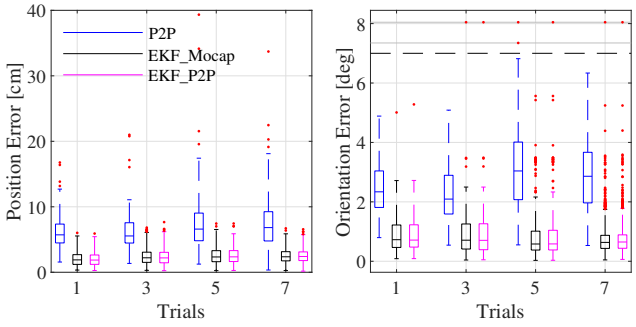


Fig. 7: Pose errors evaluated on trial 1, 3, 5, and 7. We compared the results from P2P, the EKF initialized by the Mocap ground truth, and the EKF initialized by P2P. There is no statistically significant difference in performance between the latter two cases.

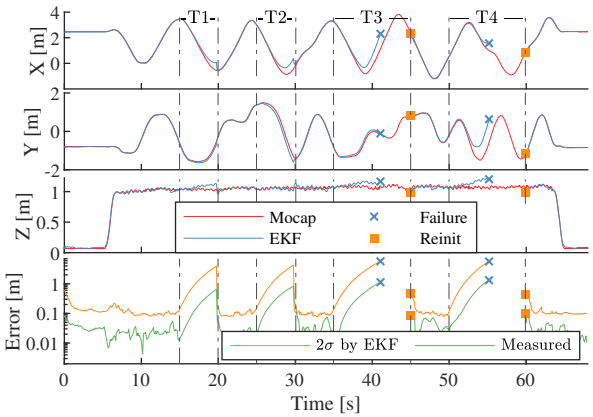


Fig. 8: Position estimates in trial 7 under longer periods of outage: T1=[15s 20s], T2=[25s 30s], T3=[35s 45s], T4=[50s 60s]. Besides this, we show the two times error standard deviation 2σ estimated by EKF, as well as the measured position error in the error plot. It is shown in log-scale for better visualization.

from our 2-point pose initialization. We use EKF-Mocap for indicating the results from the Mocap-initialized EKF while using EKF-P2P for the P2P-initialized EKF. The P2P acts as a baseline for comparison. We notice that P2P suffers from larger pose errors. Even though, we can achieve a median position error around 5 cm and a median orientation error around 3 degrees. The pose estimation results from both EKF-Mocap and EKF-P2P are almost the same. We can not find any statistically significant difference. So far, we may safely prove the efficacy of the proposed 2-point initialization method for filter bootstrapping.

Fig. 8 illustrates the case of failure recovery under the long-term outage, where we take trial 7 as an example. We manually introduce four outage periods: T1 and T2 last for 5s, while T3 and T4 last for 10s. In the first two periods, the filter begins to diverge after losing LED observations but can converge again once new features are available. The increasing position error is well estimated by EKF, as indicated by the error plot of Fig. 8. In the latter two cases, the filter uncertainty grows too high that the failure recovery mechanism is triggered, preventing the output of erroneous estimates. The filter can be reinitialized soon after the camera observes two or more decodable LEDs.

D. Runtime Analysis

To evaluate the runtime efficiency, we also run the proposed algorithm on a Raspberry Pi 3B single-board computer (Cortex-A53 @1.2GHz, 1G RAM). We implement two threads: one for VLC decoding and the other for EKF estimation. We summarize the average runtime to process an image taken by each thread in Table. III. The runtime is dominated by the VLC thread. The algorithm efficiency can be improved by optimizing the image processing pipeline for VLC decoding. Nevertheless, we can achieve real-time performance on Raspberry Pi 3B without any code optimization for ARM processors, considering a camera rate of 10Hz. The proposed VLC-inertial localization system is hence lightweight to use on resource-constrained computational platforms.

TABLE III: Runtime statistics.

Module	VLC (Thread 1)	EKF (Thread 2)
Desktop PC	2.3 ms	0.7 ms
Raspberry Pi 3B	40.5 ms	9.7 ms

E. Discussions

The proposed system suffers a few limitations. We use only circular LEDs of the same form factor for evaluation due to the difficulty in hardware preparation. The number of encodable LEDs is subject to the usable VLC channel capacity supported by our hardware. Besides, we resort to a simplified global-shutter camera model for the EKF update. In future work, we plan to employ a rolling-shutter model instead. We exploit ceiling-mounted LEDs in our system and thus assume an upward-facing camera for normal operation. The change in camera orientation (say roll and pitch) is often limited during motions. The system can accommodate some temporally larger orientation changes by inertial tracking at the risk of losing LED observations though. We leave this issue for our future work. It would be interesting to improve the system by using natural visual features as well for better tracking performance in LED absent periods.

VI. Conclusion

This paper presented an EKF-based tightly coupled VLC-inertial localization system by using modulated LED lights in modern buildings as artificial visual landmarks, especially for lightweight global localization on resource-constrained platforms. Our system employed a rolling-shutter camera and an unsynchronized IMU. The EKF localizer tightly fused inertial measurements with visual measurements of VLC-enabled LEDs. We further completed our system by 2-point global pose initialization for filter bootstrapping and failure recovery. Our system managed to be bootstrapped from two and more LED features in a single image and then sustained by EKF. The system and method were verified by extensive field experiments in a Mocap-room mounted with dozens of LED prototypes. It has been shown that our system can reliably provide lightweight real-time accurate global pose estimates in LED-shortage situations. The robustness under short-term LED outage, as well as the failure recovery behavior under long-term outage, was also demonstrated.

References

[1] J. Zhang and S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time," in *Robotics: Science and Systems*, vol. 2, 2014, p. 9.

[2] H. Ye, Y. Chen, and M. Liu, "Tightly Coupled 3D Lidar Inertial Odometry and Mapping," in *Proc. ICRA*. IEEE, 2019.

[3] Y. Zhuang, L. Hua, L. Qi, J. Yang, P. Cao, Y. Cao, Y. Wu, J. Thompson, and H. Haas, "A survey of positioning systems using visible LED lights," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1963–1988, 2018.

[4] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. ICRA*. IEEE, 2011, pp. 3400–3407.

[5] M. Li and A. I. Mourikis, "Online temporal calibration for camera–IMU systems: Theory and algorithms," *Int. J. Rob. Res.*, vol. 33, no. 7, pp. 947–964, 2014.

[6] T. Qin and S. Shen, "Online temporal calibration for monocular visual-inertial systems," in *Proc. IROS*. IEEE, 2018, pp. 3662–3669.

[7] Z. Kukelova, M. Bujnak, and T. Pajdla, "Closed-form solutions to minimal absolute pose problems with known vertical direction," in *Proc. ACCV*. Springer, 2010, pp. 216–229.

[8] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," in *Proc. MobiCom'14*. ACM, 2014, pp. 447–458.

[9] A. Jovicic, "Qualcomm Lumicast: A high accuracy indoor positioning system based on visible light communication," 2016.

[10] G. Simon, G. Zachár, and G. Vakulya, "Lookup: Robust and accurate indoor localization using visible light communication," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 9, pp. 2337–2348, 2017.

[11] L. Li, P. Hu, C. Peng, G. Shen, and F. Zhao, "Epsilon: A visible light based positioning system," in *Proc. NSDI'14*, 2014, pp. 331–343.

[12] Q. Liang, L. Wang, Y. Li, and M. Liu, "Plugo: a Scalable Visible Light Communication System towards Low-cost Indoor Localization," in *Proc. IROS*. IEEE, 2018, pp. 3709–3714.

[13] R. Munoz-Salinas, M. J. Marin-Jimenez, and R. Medina-Carnicer, "SPM-SLAM: Simultaneous localization and mapping with squared planar markers," *Pattern Recognition*, vol. 86, pp. 156–171, 2019.

[14] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision," *Autonomous Robots*, vol. 33, no. 1-2, pp. 21–39, 2012.

[15] M. Neunert, M. Bloesch, and J. Buchli, "An open source, fiducial based, visual-inertial motion capture system," in *Proc. FUSION*. IEEE, 2016, pp. 1523–1530.

[16] G. He, S. Zhong, and J. Guo, "A lightweight and scalable visual-inertial motion capture system using fiducial markers," *Autonomous Robots*, pp. 1–21, 2019.

[17] Y. Yang, J. Hao, and J. Luo, "CeilingTalk: Lightweight indoor broadcast through LED-camera communication," *IEEE Trans. Mobile Comput.*, vol. 16, no. 12, pp. 3308–3319, 2017.

[18] Q. Liang and M. Liu, "Technical Report: A Tightly Coupled VLC-Inertial Localization System by EKF," 2019. [Online]. Available: http://ram-lab.com/papers/2019/tr_vlcinertial.pdf

[19] N. Trawny and S. I. Roumeliotis, "Indirect kalman filter for 3d attitude estimation," *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep*, vol. 2, p. 2005, 2005.