LedMapper: Towards efficient and accurate LED mapping for visible light positioning at scale

Qing Liang, Yuxiang Sun, Chengju Liu, Ming Liu, Senior Member, IEEE, and Lujia Wang

Abstract—Indoor localization of high accuracy has been widely interested. Among competitive solutions, visible light positioning (VLP) is promising due to its ability to deliver high-accuracy 3D position and orientation with low-cost sensors by sharing the LED lighting infrastructure widespread in buildings. Most VLP systems require a prior LED location map for which manual surveys are costly in practical deployment at scale. In this paper, to address this difficulty, we propose a novel system for efficient and accurate offline mapping of LEDs for VLP. With input from visual-inertial sensors and existing or surveyed priors, it builds the map by posing a full-SLAM (simultaneous localization and mapping) problem within a factor graph formulation. Compared to manual surveys, it greatly saves human labor and time while yielding an accurate and workspace-aligned LED map. With real-world experiments in a room-scale testbed and a 15x larger lab office, we extensively evaluate the LED mapping system to verify its efficacy and performance gains.

Index Terms—Factor graph optimization, indoor localization, LED mapping, visible light communication (VLC), visible light positioning (VLP), visual-inertial odometry (VIO).

I. INTRODUCTION

INDOOR localization is needed by many moving platforms indoors, e.g., for robot navigation and a wide variety of location-based services on mobile devices like people way-finding in GPS-denied venues. With the growing adoption of LEDs for energy-efficient lighting in buildings and the advance of visible light communication (VLC), LEDs hold great potential to be a kind of GPS-like ubiquitous infrastructure that allows accurate and efficient indoor localization [1]–[5]. This approach is known as visible light positioning (VLP). Compared to other infrastructure-based approaches of similar high accuracy (cm to dm) that use ultra-wideband (UWB) radio [6] and ultrasound [7], VLP has the advantage of reusing LED lights as infrastructure. This avoids the extra burden and cost of installing specialized positioning hardware.

*This work was supported in part by the National Natural Science Foundation of China under Grant U1713211, in part by Zhongshan Municipal Science and Technology Bureau Fund under Project ZST21EG06, in part by the Collaborative Research Fund by Research Grants Council Hong Kong under Project C4063-18G, in part by the Department of Science and Technology of Guangdong Province Fund under Project GDST20EG54, and in part by the Foshan-HKUST Projects University-Enterprise Collaborative R&D Program under Project FSUST20-SHCHR06C, awarded to Prof. Ming Liu. (Corresponding author: Ming Liu.)

Qing Liang, Ming Liu, and Lujia Wang are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (email: qliangah@ust.hk; eelium@ust.hk; eewanglj@ust.hk).

Yuxiang Sun is with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong (e-mail: yx.sun@polyu.edu.hk; sun.yuxiang@outlook.com).

Chengju Liu is with the School of Electronics and Information Engineering, Tongji University, Shanghai, China (e-mail: luchengju@tongji.edu.cn).

In VLP, information is transmitted from modulated LEDs that change intensities quickly beyond human perception and is received by a photodiode (PD) or camera sensor. It carries a unique identifier for each LED, e.g., an identity code (ID) [8]–[12] and a frequency [13]–[16]. In VLP, LEDs act as artificial beacons in the environment, and each allows known data association using its identifier. The sensor takes measurements of LEDs in the VLC range, e.g., bearing, ranging, and received signal strength (RSS). These can be used to infer the sensor pose (or position) standalone, such as by geometry-based [15] (e.g., trilateration) and fingerprinting [16] methods, or be combined with other sensors for a fused pose estimate [8]–[12]. To achieve this, most VLP systems require a prior map composed of global LED locations in the environment alongside their identifiers for data association.

If LEDs reside on a precisely assembled frame, the map can be known from the frame geometry and be otherwise by manual surveys using measuring devices (e.g., laser rangefinder, total station, and motion capture system). This can be straightforward for small-scale experiments and is the de facto way in many studies. However, for real applications at scale, manual surveys are difficult and are prone to human errors due to the increase of LEDs and coverage. It entails intensive human labor and time, thereby posing non-negligible deployment costs of VLP systems. To overcome such practical challenges, we seek novel solutions to efficient, accurate LED mapping with less human effort. Yet, this is rarely studied in the VLP field [14]. Our main focus is hence in this direction.

Today, a mobile device has rich sensors onboard, e.g., a MEMS (micro-electro-mechanical system) inertial measurement unit (IMU) and multiple rolling-shutter (RS) cameras. In
VLP, the RS camera is a ready-to-use receiver, and the IMU is widely used for aided VLP, e.g., in loosely or tightly coupled manners [8]–[11]. Moreover, with camera-IMU sensors, many visual-inertial odometry (VIO) algorithms are well-developed [17]–[19] in robotics, capable of low-drift, accurate 6-degrees-of-freedom (DoF) pose estimates in a local frame. Indeed, VIO has been integrated into recent mobile devices as mature software, e.g., ARKit\(^1\) and ARCore\(^2\). In applications, it acts as a virtual 6-DoF odometer of low drift (e.g., a few percent of traveled distance or less). To the best of our knowledge, it has not yet been well explored in VLP usages.

In this paper, to tackle the difficulty in LED mapping for VLP, we propose LedMapper, a novel system developed for efficient and accurate offline mapping of modulated LEDs in a 3D workspace, leading to much-reduced human effort than manual surveys. We utilize a rig of low-cost visual-inertial (VI) sensors already existing on mobile devices, i.e., an IMU and two RS cameras (one for VLC and the other for VIO), as the mapping device. The mapping process entails data acquisition in the workspace, for which a surveyor wanders around with the handheld device to form looped paths and points the VLC camera to LEDs when passing by. For VLP, the LED map should align with the global reference frame of the workspace. To allow this, the mapper needs a few known global LED locations (so-called anchors or control points from surveys beforehand) as prior input. Nevertheless, the required human effort is much less than a complete manual survey since the control points account for only a minor portion of all LEDs. The mapping task is solved by posing a full-SLAM (simultaneous localization and mapping) problem within a factor graph formulation [20]. In this work, we assume point-source LEDs but explore LED geometry priors for mapping with the benefit of absolute metric scale. We consider that our contributions are mainly in the VLP field and credit the novelty to the proposed LED mapping system itself. We highlight the novel contributions below:

- A novel LED mapping system for efficient and accurate mapping of LEDs offline. With input from visual-inertial sensors and existing or surveyed priors, it builds the map by solving a full-SLAM problem within a factor graph. Compared to manual surveys, it effectively saves human labor and time while yielding an accurate and workspace-aligned LED map for a wide range of VLP systems.
- Extensive evaluations with real-world experiments in a room-scale testbed and a 15x larger lab office. The results show the efficacy and performance gains of our system.

The remainder of this paper is structured as follows. Section II lists the related work. Section III overviews the proposed system. Section IV briefly reviews the VLC front-end. Section V details the mapping approach. Section VI and Section VII shows experimental results and our discussions of limitations, respectively. Section VIII concludes this paper.

2https://developers.google.com/ar/

II. RELATED WORK

There is a rich body of literature on VLP, among which [1]–[5] provide fundamentals and comprehensive surveys. To our knowledge, however, only a handful of works aim to map LED locations efficiently for VLP. In this section, we review these closely related to our proposed LedMapper in detail.

In [13] and its follow-up [14], a VLP calibration (i.e., LED mapping) method is proposed using a mobile robot equipped with a 2D Lidar and an upward-facing RS camera. The Lidar data is processed by a SLAM algorithm [21] to give drift-less robot poses. The camera takes images of overhead LEDs and decodes identifiers. The robot must approach each LED until it appears in the image center. One can thereby obtain the 2D LED position from the robot pose. Assuming a known height, this yields an LED map with 3D locations and unique identifiers. Evaluated in a small testbed with four LEDs, [14] shows a good map accuracy of centimeters. Yet, the map is expressed in a local SLAM frame, not necessarily aligned with the global workspace [14]. In addition, data collection may be problematic in a non-traversable area by a wheeled robot.

A handheld device has better mobility in complex scenes. Using a PD receiver and a Tango tablet (running VIO), [22] proposes a light registration method to map 2D light locations onto a floor plan (say height is known). This entails a surveyor who holds the device, starts from a known pose (set by a few anchored lights of known locations), and walks across ceiling lights. Upon crossing a lamp, its identifier is decoded, and its 2D location is recorded using the tracked VIO pose. To bound VIO drifts, [22] divides the mapping area into smaller sections and repeats the process. The VIO paths and light locations are manually aligned to the floor plan. It attains successful results in large-scale scenarios. However, the final map accuracy is not clarified due to the lack of evaluation. Like [14], it may face data collection problems when lights are located above an area not traversable by a person. The required human intervention (e.g., manual alignment) can be prone to errors.

In [22], unmodified lights are in place of modulated LEDs for VLP. Please refer to [1] for a review of VLP based on modified and unmodified lights. As in [14], we focus on mapping modulated LEDs and use an RS camera as the VLC receiver, but for flexible data collection in complex venues, we use a handheld device like in [22]. Likewise, we consider offline mapping since LED positions are fixed and only require a one-time registration. The methods of [13], [14], [22] are heuristic and ignore the uncertainties of sensor measurements and prior knowledge. By contrast, our LedMapper follows a principled design based on probabilistic state estimation [23]. The mapping task is formulated as a full-SLAM problem within a factor graph that has been well studied in robotics [20]. The input information can be utilized in a sounder way. Moreover, we expect our system to be way self-contained, not relying on an accurate floor plan or the presence of other SLAM/localization systems in the environment.

Rather than using an LED location map, some VLP methods (e.g., fingerprinting [16]) require a detailed map of location-labeled light fingerprints. A robot or mobile device can assist in the mapping process, as in [24], [25]. Yet, a detailed review...
of such methods is beyond our scope. Overall, unlike in WiFi-based positioning systems [26], fingerprinting methods attract less attention in VLP studies [3], partly due to the high cost of light fingerprinting [4]. In contrast, due to the line-of-sight (LOS) light propagation, geometry methods allow for high accuracy using LED location maps that are easier to obtain.

III. SYSTEM OVERVIEW

In this section, we describe the hardware setup for our LED mapping task in Section III-A, give an overview of the system workflow in Section III-B, and clarify the notations and reference frames used in this paper in Section III-C.

A. Hardware Setup

![Photo of self-assembled hardware](a)

![Visual-inertial sensors](b)

Fig. 2: Photos of our self-assembled hardware. (a) Modulated LEDs. (b) Visual-inertial sensors.

The VLP hardware often consists of modulated LEDs at the infrastructure side and VLC receivers at the user side. Here, an RS camera is chosen as the receiver. We build a few LED prototypes with rechargeable batteries (see Fig. 2a). Each has a round radiation surface of 15.5 cm in diameter and rating power of 3 W. The portable design allows for a flexible LED deployment for experiments. The LEDs are modified from off-the-shelf products by adding a microcontroller unit (MCU) and reusing most inbuilt components, such as MOSFET, LED beads, batteries, and housings. The MCU runs VLC logic and modulates LEDs via the MOSFET to broadcast LED IDs. Due to the small amount of data required for LED IDs in VLP, we trade the reduction of data packet length for an acceptable maximum decoding distance (e.g., a few meters). This is achieved by sacrificing the payload size and omitting error checking in the protocol [11]. Finally, our VLC implementation gives $d_{\text{m}} \approx 2.3 m$ while allowing for a data payload of one byte. We find it sufficient for this study.

B. System Workflow

Fig. 3 shows the workflow of the proposed LED mapping system. It entails three blocks: VLC front-end, VIO estimator, and LED mapper. The VLC front-end takes the RS images from Cam1 as input and produces feature tracks of LED blobs with IDs by LED detection, tracking, and decoding. The gyroscope measurements are utilized to assist LED tracking. The VIO estimator fuses IMU measurements and the natural visual features from Cam0 and provides 6-DoF VIO poses of the IMU frame. We treat it as a black box and assume a favorable lighting condition for VIO operation. It can be implemented by well-established VIO algorithms such as VINS-Mono [18] and OpenVINS [30]. In our case, we choose VINS-Mono due to its support for RS cameras. Using all historical LED tracks and VIO poses as input, the LED mapper aims to build a globally consistent and accurate LED map by offline batch optimization. Besides sensor inputs, we consider prior map information essential for global workspace alignment or relatively easy to obtain, such as control points, ceiling height, and LED geometry. These priors, if available, can be seamlessly incorporated into the mapper as additional constraints for improved quality. After mapping, the built LED map is assumed accurate (with uncertainties) and can later be used by VLP systems for online localization.

---


2https://lp-research.com/lpms-me1-dk/
C. Notations and Reference Frames

A transformation matrix $T_{A}^{B} \in SE(3)$ takes a vector $p_{B} \in \mathbb{R}^{3}$ in the frame $F_{B}$ to the frame $F_{A}$. It can be divided into a rotation matrix $R_{A}^{B} \in SO(3)$ and a translation vector $p_{A}^{B} \in \mathbb{R}^{3}$, i.e., $T_{A}^{B} = [R_{A}^{B}, p_{A}^{B}]$. The transformed vector $p_{A}^{B} \in \mathbb{R}^{3}$ in $F_{A}$ is given by $p_{A} = R_{A}^{B} p_{B} + p_{A}^{B}$. With slight abuse of notation for brevity, we also write $p_{A} = T_{A}^{B} p_{B}$. Often, we use the unit quaternion under Hamilton convention [31], $q_{A}^{B}$, to represent the rotation. $R(\cdot)$ converts $q_{A}^{B}$ to the rotation matrix $R_{A}^{B}$, $\odot$ denotes the quaternion multiplication. For a variable $(\cdot)$, we write its measurement as $\hat{(\cdot)}$. As depicted by Fig. 4, we work with four coordinate frames. The global base frame $F_{G}$ is gravity-aligned and fixed in the workspace. It sets the origin for all global measurements. The map to be built is aligned to this frame. The local base frame $F_{L}$ is gravity-aligned and sets the origin for VIO pose estimation. $F_{I}$ is attached to the IMU body frame, and $F_{C}$ is attached to the optical frame of the VLC camera. The IMU-camera extrinsic transformation $T_{C}^{I}$ is a known constant from prior calibration.

IV. VLC FRONT-END

The blobs from modulated LEDs are detected on incoming images, tracked over consecutive frames, and decoded to have correct LED IDs for long-term data association. The front-end has three modules: blob detection, blob tracking, and VLC decoding. It is mostly inherited from our previous works [10–12]. We briefly review it for completeness.

Due to the fast exposure of the VLC camera, bright LEDs have high contrast against the background. After binarization and dilation on grayscale image input, LED blobs are detected with standard techniques. For each blob, we take its centroid with pixel coordinate, $[u, v]^T$. With the known camera intrinsics, the normalized pixel location, $\bar{z} \in \mathbb{R}^{2}$, is also computed.

To achieve tracking, we detect new blobs on every frame and find their best matches [32] in the previous frame. Each blob has a unique tracking ID for short-term data association. We assume the mutual blob distances in an image is greater than the inter-frame pixel displacements. This works in our case due to the sparsity of lights and the non-rapid camera motion. Since camera rotation is more likely to yield large pixel displacements, we compensate for it using the short-term integration of gyroscope measurements before matching.

V. LED MAPPING FROM BATCH OPTIMIZATION

We present the proposed LED mapper based on factor graph optimization in detail, including problem statement in Section V-A, graph construction in Section V-B, factor description in Section V-C, and batch optimization in Section V-D.

A. Problem Statement

The mapping task entails solving a full-SLAM problem that seeks to estimate the entire IMU poses, $\{x_{1}, \ldots, x_{N}\}$, and the locations of LED landmarks, $\{l_{1}, \ldots, l_{M}\}$, given all historical sensor measurements and prior knowledge about LEDs. $N$ and $M$ are the total number of poses and landmarks, respectively. It is formulated as a factor graph optimization problem [20]. The structure of the built factor graph is shown in Fig. 5.

![Fig. 4: Illustration of the involved reference frames and their relations. The global LED priors (e.g., control points) are expressed in $F_{G}$.](image)

VLC decoding applies to blobs with barcode-like patterns. The row-parallel strips of varying widths and pixel intensities carry VLC information, e.g., LED ID. Given a blob, we pick up the grayscale pixel values on the centering column of its image region. As the camera’s sampling rate is known, these ordered pixel values form a time-varying 1D signal. After binarization, OOK demodulation, and Manchester decoding, the LED ID can be obtained. The blobs that are part of a track have the same LED ID, and we can identify them all if any single is decodable. The blob tracking allows more instances of LED detections for later mapping and VLP.
where \( \{T^t_{ij}, \ldots, T^t_{ji}\} \) are the \( M \) IMU poses; \( \{p^t_{ij}, \ldots, p^t_{ji}\} \) are the \( M \) LED positions; \( T^t_{ij} \) is the base frame transformation between \( F_G \) and \( F_L \); and \( s \) is the VIO scale factor.

### B. Graph Construction

We explain how to construct this graph. A new pose node, \( x_i \), is added as the latest VIO pose measurement, \( T^t_{ij} \), arrives (at 10 Hz in our case). This VIO pose gives the initial estimate of \( x_i \). Also, we can apply a simple keyframe selection strategy based on translation and rotation changes to further sparsify the graph. The LED detections are available at maximally 10 Hz from the VLC front-end. Yet, in general, we do not assume the same update rate for the VIO poses and LED detections. When a new LED blob first arrives (as per the unique tracking ID), we create a temporary landmark node for it. As each LED has been tracked over frames, its subsequent detections are associated with this landmark node. After the LED track is complete, we check all the detections to find whether one or more blobs have a valid LED ID (say successfully decoded). If not, the landmark node is immediately dropped due to decoding failure. Note this ID is critical for long-term data association when loop-closure occurs later on. Otherwise, we proceed to check if a landmark of the same LED ID exists in the graph. If so, we re-associate all related LED detections to the existing node and drop the temporal one. Otherwise, we will triangulate the LED position using the involved LED detections and IMU poses alongside the known \( T^t_{ij} \). After this, the landmark node is marked matured and added to the graph as \( l_j \), where \( j \) counts the number of unique LED IDs obtained so far. The associated LED ID is denoted as \( ID_j \).

In general, the timestamps of LED detections are not aligned with those of VIO poses, due to the lack of hardware synchronization between the VLC and VIO sensors. We assume the VIO time is based on the IMU clock as in VINS-Mono. Like in [35], we match the LED detections to a past VIO pose with the closest timestamp and point them virtually to the related pose node in the graph. The true pose from which the LED is detected is in between the two bounding IMU poses, which can be obtained by linear interpolation. To facilitate this, we expect the sensors to move smoothly, as in normal walking.

### C. Factor Description

As illustrated in Fig. 5, the graph incorporates three sources of factors posed by sensor measurements and prior knowledge. We will describe them in detail below.

1) VIO factors: Given input VIO poses, \( \{\hat{T}^t_{ij}\} \), the relative transformation measurement between consecutive poses, \( x_i \) and \( x_{i+1} \), is given by \( \hat{T}^t_{i+1} = \hat{T}^t_{i}\hat{T}^{-1} \hat{T}^t_{i+1} \). It is written as \( \hat{p}_{i+1}, \hat{q}_{i+1} \). The predicted motion is \( \hat{T}^t_{i+1} = T^t_{i+1}T^t_{i} = \hat{p}_{i+1}, \hat{q}_{i+1} \). This yields the odometry residue:

\[
\mathbf{r}^O(x_i, x_{i+1}, s) = \begin{bmatrix}
sp\hat{p}_{i+1} - \hat{p}_{i+1} \\
2 \cdot \text{vec3}(\hat{q}_{i+1} \hat{q}_{i+1}^{-1})
\end{bmatrix},
\]

where \( s \) is the VIO scale factor, and \( \text{vec3}(\cdot) \) returns the vector part \( (q_x, q_y, q_z)^T \) of a quaternion \( q \). We write the covariance of this relative measurement as \( \Sigma^O_{i+1} = \text{diag}(\sigma^2_{pos}, \sigma^2_{rot}) \), where \( \sigma^2_{pos} \) and \( \sigma^2_{rot} \) describe the uncertainties in translation and rotation, respectively, and are set empirically in this work.

The roll and pitch in VIO are accurate without drift in \( F_L \). To ensure this property, we exploit them as absolute measurements. For each pose \( x_i \), we apply a rotational constraint due to gravity with the residue:

\[
\mathbf{r}^V(x_i) = 2 \cdot \text{vec2}(\hat{q}_{i}^{L-1} \hat{q}_{i}^{L-1}),
\]

where \( \text{vec2}(\cdot) \) returns the vector part \( (q_x, q_y)^T \) of \( q \). The covariance is written as \( \Sigma^V = \sigma^2_{rot} \mathbf{I}_2 \), where \( \sigma^2_{rot} \) describes the small uncertainty in the absolute roll and pitch measurements.

2) Vision factor: We assume an undistorted, pinhole RS camera model for the VLC camera. An RS camera captures image rows sequentially at varying times. In the case of general motions, this leads to different camera poses for each row. Following the convention in [34], we assume the image timestamp corresponds to the middle image row. For an image of \( K \) rows in total with timestamp \( t \), the sampling time of the \( k \)th row away from the middle is \( t^k = t + \frac{k}{2r} \), \( k \in \{-K, \ldots, K\} \). \( t_r \) is the RS frame readout time which is assumed known from pre-calibration or the sensor’s datasheet if available.

Consider the landmark \( l_j \) associated with the pose \( x_i \) in the graph, for which the VLC front-end gives the normalized image measurement \( \hat{z}_j \) with timestamp \( t_{ij} \). The corresponding pixel coordinate \( [\hat{u}, \hat{v}]^T \) lies in row \( \hat{v} = \hat{v} - \frac{K}{2}, \hat{v} \in [1, K] \). Due to the time misalignment between VIO poses and LED detections, \( \hat{z}_{ij} \) is indeed taken from an intermediate pose, \( x_{ij} \), between \( x_i \) and \( x_{i+1} \) with timestamps \( t_i \) and \( t_{i+1} \), respectively. With RS imaging, the exact timestamp is given by \( t^k_{ij} = t_{ij} + \frac{k}{r} t_r \). This is depicted by the diagram in Fig. 6.

![Fig. 6: Illustration of the time misalignment alongside the RS imaging time used for pose interpolation. Note the time interval of VIO poses and that of LED detections are not necessarily the same.](image)

We can obtain \( x_{ij} \) by interpolating \( x_i \) and \( x_{i+1} \), i.e., using spherical linear interpolation (Slerp) for the rotation and linear interpolation for the translation [36]. Having \( x_i = [p^L_{ij}, q^L_{ij}] \) and \( x_{i+1} = [p^L_{ij+1}, q^L_{ij+1}] \), we write \( x_{ij} = [p^L_{ij}, q^L_{ij}] \) with

\[
q^L_{ij} = \text{Slerp}(q^L_{ij}, q^L_{ij+1}, \tau),
\]

\[
p^L_{ij} = (1 - \tau)p^L_{ij} + \tau p^L_{ij+1},
\]

\[
\tau = \frac{t^k_{ij} - t_i}{t_{i+1} - t_i} = \frac{t_{ij} + \frac{1}{2} \pi - \frac{1}{2} t_r}{t_{i+1} - t_i} - \frac{1}{2} t_r - t_i.
\]

where Slerp(\( q_0, q_1, \tau \)) is the Slerp function [31] that linearly interpolates from \( q_0 \) to \( q_1 \) as \( \tau \) evolves from 0 to 1 (\( \tau \in [0, 1] \)).

The re-projection residue relating to the landmark \( l_j = p^L_j \) and the interpolated pose \( x_{ij} = T^t_{ij} \) is given by

\[
\mathbf{r}^V(x_i, x_{i+1}, l_j) = \hat{z}_j - \pi \left( T^T_{C} T^t_{ij} p^L_j \right)
\]

where \( \pi(\cdot) \) projects a 3D point \( p^C \) onto the normalized image plane as a 2D point \( z \) according to \( z = [p^C_x/p^C_z, p^C_y/p^C_z]^T \).
The covariance matrix is given by $\Sigma^V_{ij} = \sigma^2_n I_2$, where $\sigma^2_n$ describes the normalized pixel noise for LED observations.

3) Prior factors: We consider prior map information that is deemed essential or readily available to our mapping task. It can be absolute or relative depending on if it is given in $\mathcal{F}_G$. The absolute prior on $I_1$ is specified with a known LED position $\hat{p}_i^G$. It provides absolute geometric constraints for the system states. The introduced metric scale information can help correct VIO scale errors. With the transformation $\hat{T}_L^G$, this leads to the absolute prior residue below:

$$r_{i}^{P_a}(1, \hat{T}_L^G) = \hat{p}_i^G - \hat{T}_L^G p_i^L$$

with covariance matrix $\Sigma^{P_a} = \text{diag}(\sigma^2_x, \sigma^2_y, \sigma^2_z)$ that describes the measurement uncertainty on each axis.

For absolute priors, we distinguish between control points and weak priors. The former is precisely known 3D positions in $\mathcal{F}_G$ for a few selected LEDs. These absolute locations can be obtained by manual surveys using measuring equipment (e.g., laser rangefinder, total station) and are necessary if we need to align the built map with the workspace. The latter is partial knowledge about the LED position (only certain on one axis), e.g., the common height of ceiling LEDs. This can be known from one control point on the ceiling or the 3D architectural plan when available. For control points, we set $\sigma^2_x, \sigma^2_y, \sigma^2_z$ to small values. For weak priors with known height, we set small values for $\sigma^2_x$ and large values for others.

An initial guess of $\hat{T}_L^G$ is needed for its optimization. Given a known pose $\hat{T}_L^G$ in $\mathcal{F}_G$ and the related VIO pose $\hat{T}_L^f$, it is given by $\hat{T}_L^G = \hat{T}_L^f \hat{T}_L^{-1}$. To achieve this, we currently need at least two control points that are closely located. Using such two LEDs co-visible in a single frame, we can compute $\hat{T}_L^f$ by the 2-point pose initialization [11] based on a closed-form P2P solution [37]. With initialization success, we estimate $\hat{T}_L^G$ while zeroing the roll and pitch. Otherwise, we fix it to an identity matrix and ignore absolute priors in mapping.

The relative priors come from the known shape and size of LED geometry without effort. Note in this work, we evaluate the system using small-sized circular LEDs and assume point landmarks. In reality, squared panels or linear tubes are often used, and each can be, e.g., represented by a set of corner (end) points. Like in square fiducial markers [38], the side length of LED panels (tubes) provides extra distance measurements of metric scale. Consider a relative prior between two corners on an LED landmark, $I_1$ and $I_2$, with the known distance $\hat{d}_{ij}$. The residue is simply written as

$$r_{i}^{P_a}(1, 1) = \hat{d}_{ij} - \|p_i^L - p_j^L\|$$

with covariance $\Sigma^{P_a}_{ij} = [\sigma^2_d]$, where $\sigma^2_d$ is the noise uncertainty.

D. Batch Optimization

To obtain the maximum-a-posteriori estimate for the entire state $x_m$, we minimize a cost function $f(x_m)$ that sums over the Mahalanobis norm of all measurement residues as follows:

$$f(x_m) = \sum_{i \in \mathcal{O}} \|r_i^G(x_m)\|^2_{\Sigma_i^G} + \sum_{i \in \mathcal{G}} \|r_i^G(x_m)\|^2_{\Sigma_i^G}$$

$$+ \sum_{(i,j) \in \mathcal{V}} \rho\left(\|r_{ij}^G(x_m)\|^2_{\Sigma_{ij}^G}\right)$$

$$+ \sum_{i \in \mathcal{P}_a} \|r_{i}^{P_a}(x_m)\|^2_{\Sigma_i^{P_a}} + \sum_{(i,j) \in \mathcal{P}_r} \|r_{ij}^{P_a}(x_m)\|^2_{\Sigma_{ij}^{P_r}},$$

where $\rho(\cdot)$ is a robust loss function [23] to reduce the effect of LED outliers. $\mathcal{O}$ and $\mathcal{G}$ are sets of the relative odometry measurements and the absolute rotation measurements around the gravity, respectively, derived from VIO input. $\mathcal{V}$ is the set of visual measurements of LEDs that are successfully decoded. $\mathcal{P}_a$ is the set of absolute map priors, including control points and weak priors, while $\mathcal{P}_r$ is the set of relative map priors. The residuals and their covariances are defined in Section V-C.

This nonlinear problem is solved using Ceres solver [39]. After optimization, we obtain the global position of each LED, $l_j$, as per $p_j^G = \hat{T}_L^G p_j^L$. The set of $\{(l_j, p_j^L)\}, j \in [1, M]$ constitutes the final LED map anchored to the workspace. For later VLP use, we empirically set a fixed uncertainty, based on the experimental evaluation, for the mapped locations.

Remark: Without map priors, the VIO scale factor cannot be determined and is hence fixed in optimization (i.e., $s = 1$). As such, the optimized LED positions and the IMU poses can be subject to an inaccurate metric scale estimate from the VIO input. If control points are not available or when $\hat{T}_L^G$ fails in initialization, as explained previously, the built LED map can not align to $\mathcal{F}_G$ in the workspace. Yet, it is acceptable to use if we allow localization solutions within a local frame $\mathcal{F}_L$.

VI. EVALUATION

In this section, we evaluate the proposed LED mapping system by real-world experiments. We first introduce the experiment setup in Section VI-A. We assess the LED mapping accuracy in a controlled testbed with ground truth LED locations in Section VI-B. We show the influence of VIO scale errors on results and describe how they are compensated for by using various priors if available. In Section VI-C, we study the possible impact of LED sparsity on map accuracy. In Section VI-D, we show the influence of the inhomogeneity of LED layout. Finally, in Section VI-E, we evaluate the mapping system at a 15x larger office area of more realistic settings.

A. Experiment Setup

To verify the efficacy of our LedMapper, we compare its four variants: M1–M4. The difference lies in the constraints needed for optimizing the cost function $f(x_m)$, including sensor measurements $\{\mathcal{O}, \mathcal{G}, \mathcal{V}\}$ and prior knowledge $\{\mathcal{P}_a, \mathcal{P}_r\}$, as explained in Section V-D.

- M1: The baseline method uses sensor measurements $\{\mathcal{O}, \mathcal{G}, \mathcal{V}\}$ only without prior. The mapping results can suffer from a non-unit metric scale due to VIO.
- M2: Besides $\{\mathcal{O}, \mathcal{G}, \mathcal{V}\}$ as in M1, it uses absolute priors $\mathcal{P}_a$ from control points. We aim to study their effects
on scale estimates and mapping accuracy, alongside their usage for global map alignment.

- **M3**: Like M2, it also exploits \( \{O, G, V, P_a\} \). However, \( P_a \) now includes both control points and weak priors (e.g., ceiling height). Comparing M3 to M2 allows us to see the impact of weak priors.

- **M4**: Besides \( \{O, G, V\} \) as in M1, it uses relative priors \( P_r \) from LED geometry. We aim to test their efficacy in improving VIO scale estimates and mapping accuracy.

For comparison, we align the results to the ground truth by SE(3) or Sim3 transformation using [40]. To assess the map accuracy, we compute the root-mean-square error (RMSE) of estimated LED positions. The metric scale error in LED positions is based on the scale computed during Sim3 alignment. The trajectory accuracy is evaluated by the absolute trajectory error (ATE) [41].

All experimental data are collected using the self-assembled sensor rig (cf. Section III-A) and are processed on a desktop computer (Intel i7-7700K CPU@4.2 GHz, 16 GB RAM).

![Fig. 7: The photo (left) and illustration (right) of the Mocap testbed for experiments, showing the locations and IDs of mounted LEDs. The black dots show the ground truth locations. The red pentagons are for control points. The blue squares are for LED geometry prior.](image)

### B. Mapping Accuracy

To study the mapping accuracy, we conduct experiments in a room-sized testbed \((5 \text{ m} \times 4 \text{ m} \times 2.35 \text{ m})\) instrumented with a precise OptiTrack\(^7\) motion capture system (Mocap). There are 25 modulated LEDs evenly distributed on the ceiling, as shown in Fig. 7. We set \( F_G \) the same as the world frame of Mocap. High-accuracy 3D LED positions are available as ground truth by a tedious manual calibration procedure using Mocap and a laser rangefinder for height compensation.\(^8\) The ground truth IMU poses (trajectory) are computed from offline batch optimization, using all available sensor inputs (VIO poses and LED detections) and taking all ground truth LED positions as control points.\(^9\) To collect data in the testbed, we carry the handheld sensor, point the VLC camera to ceiling LEDs and walk around normally to close loops. While walking, we do not require the camera to face upright to the ceiling (say it can tilt forward). The sensor height is kept relatively constant (1 m above the floor) in the experiment. Five datasets with different motion profiles are collected, and each lasts about one minute.\(^10\) At the start of each run, the sensor is put on the ground still for a few seconds and then moved with sufficient motion excitation to aid VIO initialization. The start point sits beneath LED-110 and LED-114, as shown in Fig. 7.

We now detail the test settings for the four mapper variants (i.e., M1–M4). As indicated in Fig. 7, three LEDs of known positions in \( F_G \) (red pentagons) are chosen as control points for M2/M3. With LED-110 and LED-114, an initial estimate for \( T_G^V \) can be obtained by 2-point pose initialization [11]. We use the rough ceiling height of 2.35 m as weak priors for M3. The affected standard deviation is set as \( \sigma_z = 0.2 \text{ m} \). For M4, we select three pairs of LEDs (blue squares) with known pairwise distances (dashed blue lines) and treat this knowledge as a simulated source of relative priors from LED geometry.

![Fig. 8: Mapping results on dataset #1 using four methods (M1–M4). (a) visualizes the mapped LED positions and the optimized trajectory. Cyan shows results after SE(3) alignment, blue shows results after Sim3 alignment, and red shows the ground truth. (b) plots the scale ratio of optimized trajectories and VIO input over time.](image)

Fig.8 shows the results on dataset #1 of M1–M4 after SE(3) alignment. 22 LEDs are mapped successfully. The Sim3 results of M1 are also included (denoted as M1-sim3). We align the VIO trajectory using its first 50 poses. Fig.8a compares M1’s results with the ground truth and VIO input. The results of

\(^7\)https://optitrack.com/

\(^8\)The ceiling LED locations are beyond the operation scope of our Mocap system. We do calibration in two steps using Mocap and a leveler-mounted laser rangefinder. We first measure the 3D orthogonal projection position of the LED on the floor using Mocap and then obtain the truth LED location by compensating for the height difference using the laser distance measurement.

\(^9\)The ground truth trajectory data provided by Mocap were not recorded at the time of data collection due to some reason. Yet, this does not hamper the goal of assessing the LED mapping accuracy in this work.

\(^10\)Running on these datasets, batch optimization for mapping 25 LEDs can finish within a fraction of seconds.
M2–M4 are very close to M1-sim3 and are omitted here for clarity. The Sim3 results (blue) well fit the ground truth, while the SE(3) results (cyan) show larger mismatches (see outer rings). This suggests an inaccurate scale estimate in M1. To examine the scale errors in optimized trajectories, we compare the scale ratio [33] among different methods, as shown in Fig.8b. The scale ratio is computed as the traveled distance of the trajectory estimate divided by the ground truth and subtracted by one. A ratio closer to zero means a better metric of the trajectory estimate divided by the ground truth and one degree. M3 performs even better, with smaller errors in translation (≤2.2 cm) and rotation (≤0.53 deg). This accuracy gain is due to the weak priors about LED height.

C. Impact of LED Sparsity on Mapping Accuracy

In what follows, we study the impact of the sparsity of LED observations on mapping accuracy. The previous setup of 25 LEDs in a 5 m × 4 m area means a dense LED placement and provides rich LED observations. In reality, due to the variances of LED deployment density and ceiling height in complex indoor settings, LED observations valid for mapping can be much sparser. It will be helpful to assess the mapping performance under sparse LED setups. Explicitly, we consider four LED setups of different sparsity levels with 3/6/12/25 LEDs, as shown in Fig. 9. Rather than altering the physical setup, for evaluation convenience, we selectively use measurements from the chosen LEDs (see blue squares in Fig. 9a-c). These are evenly scattered over the test area.

We test M1 on the five testbed datasets as used previously. To minimize the influence of VIO scale errors, we pre-calibrate the scale factor $s$ by batch optimization using all sensor measurements and the ground truth LED map and keep it fixed in the experiments. That is, we here assume a scale-correct input. As the VIO scale can change with motion profiles across datasets, we calibrate the scale factor individually for each but apply it to all four LED setups. For comparison, the mapping results are SE(3)-aligned to the ground truth.

![Fig. 9: Mapping results (blue) on dataset #1 by M1 under LED setups of varying sparsities, compared to the ground truth (red). From (a) to (d) are with 3, 6, 12, and 25 LEDs. 3 LEDs are not mapped in (d).](image-url)

In Fig. 9, we show the qualitative results (blue) obtained on dataset #1 for four LED setups, compared to the ground truth (red). We report the quantitative results on five datasets in the boxplots of Fig. 10, including LED position errors and trajectory RMSE. With the four LED setups, the mapper can recover LED positions and the IMU trajectory on all datasets. As shown in Fig. 9, these LED positions match the ground truth.
truth well, even when LEDs are sparse. The map accuracy is consistent among the four LED setups, as seen from the boxplot of LED position errors (5 cm@max) in Fig. 10. As a result, the LED mapper can run under very sparse LED distribution and build an accurate LED map. The trajectory errors, as shown by the boxplot in Fig. 10, grow evidently as LEDs become sparser. This degradation is due to insufficient loop-closure constraints from LED observations on IMU motions.

To examine the availability of LED observations, we report in Fig. 10 the LED outage rate, the maximum outage, and the average number of observations per frame. The outage rate is computed as the accumulated time of LED outages divided by the total time (the higher this rate, the severer the outage). The maximum outage measures the longest period without LED observations. The results clearly show the lack of LED constraints when LEDs are sparsely placed.

### D. Impact of the Inhomogeneity of LED Layout

So far, we consider only those LED setups of homogeneous layout (LEDs are evenly scattered over the area for mapping). Sometimes, LEDs may not be evenly deployed but are clustered on one side of the mapping area. It is helpful to study the impact of inhomogeneous LED layouts on mapping accuracy.

In the experiment, we consider eight setups of inhomogeneous LED layouts with 6 LEDs on the testbed, as shown in Fig. 11. In each setup, all the 6 LEDs for mapping are clustered on one corner of the testbed. For comparison, we take the homogeneous LED setup with 6 LEDs (see Fig. 9b) as a baseline. These LED layouts have different homogeneity but the same sparsity (i.e., the same number of LEDs in a given area). Like in Section VI-C, we test the M1 variant of the LedMapper on the five testbed datasets. Also, we follow the same experimental settings, e.g., fixing the VIO scale by pre-calibration and aligning the results by SE(3).

In Fig. 12, for each dataset, we report the mapping results by M1 under inhomogeneous LED setup (see boxplots) alongside the baseline result under homogeneous LED setup (see triangles). Fig. 13 shows the time-evolving number of LED observations per frame on dataset #1, corresponding to one inhomogeneous and one homogeneous LED setup.

As seen from Fig. 12a, the LED position RMSE on the five datasets is smaller than the baseline result. In the experiment, the system achieves improved LED mapping accuracy in the inhomogeneous case. This is likely due to more LED observations in the small clustered area. Under our inhomogeneous LED setups, the LEDs are all within a local region, in which the camera can observe multiple LEDs per frame (see Fig. 13a). This can help yield more consistent and accurate LED position estimates. By contrast, in the homogeneous case, the camera can often observe one LED per frame (see Fig. 13b) due to the scattered distribution.

Meanwhile, the accuracy of trajectory estimates tends to degrade under inhomogeneous LED setups, as shown in Fig. 12b. We suspect this is because of the less frequent correction to the drifting VIO poses based on intermittent LED detections (see Fig. 13a). The batch optimized trajectory can serve as the
best achievable positioning result by a real-time VLP system, which takes as input the VIO estimates and LED detections as used by LED mapping. In this sense, to allow for better VLP accuracy, we prefer homogeneous LED layouts in a given area.

E. Mapping Tests at a Lab Office

Now, we aim to test LedMapper in more realistic settings and at a larger scale. We carry out experiments in our lab at HKUST, a typical office scene that covers about 20 m \( \times \) 15 m. We place 12 LEDs randomly on the ground along pathways and leave them facing the ceiling. In principle, our system can work from LEDs placed at will since there are no more than 3D landmarks. The choice of putting them on the ground is for evaluation convenience.

![Fig. 14: Results in the lab-scale test on five datasets using 12 LEDs. (a) shows the positions and IDs of mapped LEDs. For each LED, the blue dots are estimates, and the red plus is the mean. (b) shows the statistics of the LED position RMSE among five datasets.](image)

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In Fig. 1 of Section I, we show the results on a typical lab dataset with the locations and IDs of mapped LEDs and the optimized trajectory, as well as the VIO input. All 12 LEDs are mapped successfully. As expected, the optimized trajectory has less drift (see z-axis) and better consistency. We take this map as a reference and align the results from the other datasets to it by Sim3. This is because M1 is subject to a non-unit VIO scale as a reference and align the results from the other datasets to it. We take this map as a reference and align the results from the other datasets to it by Sim3. This is because M1 is subject to a non-unit VIO scale as a reference and align the results from the other datasets to it. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

![In Fig. 1 of Section I, we show the results on a typical lab dataset with the locations and IDs of mapped LEDs and the optimized trajectory, as well as the VIO input. All 12 LEDs are mapped successfully. As expected, the optimized trajectory has less drift (see z-axis) and better consistency. We take this map as a reference and align the results from the other datasets to it by Sim3. This is because M1 is subject to a non-unit VIO scale as a reference and align the results from the other datasets to it. We take this map as a reference and align the results from the other datasets to it by Sim3. This is because M1 is subject to a non-unit VIO scale as a reference and align the results from the other datasets to it.](image)

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

In this experiment, we do not have ground truth for the IMU trajectory and LED positions. To evaluate the mapping performance, we examine the consistency of mapped locations among multiple runs. Five datasets are collected in the lab using the mobile mapping device. During collection, we point the cameras forward and face them to ground LEDs when walking by. We revisit these LEDs to form closed loops before returning to the start point. We test the mapper by the baseline method M1 due to the lack of ground truth measurements of control points or pairwise distances (needed by M2–M4).

VII. Discussion of Limitations

Currently, we use homemade circular LEDs for evaluation. The blob detector and tracker of the VLC front-end are designed for circular LEDs. Still, the system can run with other shaped LEDs (e.g., linear tubes, square panels) by adapting the front-end. Moreover, pose estimation using a single such LED is achievable, while the appearance (often symmetrical) may need extra modification (e.g., a colored marker on corners) for distinctiveness [42]. For a modified square LED of known size, the camera observation model for square fiducials [38] can be readily applied to our system. Also, due to limited LEDs for the study, we have assessed the system in a room-sized testbed and a 300 m\(^2\) office area. There is a practical difficulty in preparing enough homemade LEDs for experiments at a larger scale. With more LEDs available in future work, evaluation in wider environments will be desired.

The data payload and maximum decoding distance \( d_m \) by our VLC implementation could be insufficient in reality. A larger-sized data payload is essential to large-scale deployment with thousands of LEDs. Also, to enable operation in scenarios with high ceilings (e.g., shopping malls rather than our office buildings), a greater \( d_m \) is desired. Otherwise, the LED-camera distance can easily exceed \( d_m \), causing decoding failure. As mentioned previously, these limitations are mainly due to the small LED surface (i.e., 15.5 cm) in use. In real applications, one can effectively increase the data payload and (or) the maximum decoding distance by simply using LEDs of a larger surface. As for standard LEDs for daily lighting, a square panel can be 50 cm wide, while a linear tube can be 120 cm long. In future work, advanced VLC modulation/coding schemes can be explored for improved performance.

To ease hardware setup, we resort to a low-quality VI sensor without hardware triggering. Under the same environments and motion profiles, it yields inferior VIO performance with more drifts and scale errors. This can affect the best achievable accuracy of the LedMapper. Also, VI measurements are now loosely fused by our system since these are preprocessed by the VI front-end. Moreover, pose estimation using a single such LED is achievable, while the appearance (often symmetrical) may need extra modification (e.g., a colored marker on corners) for distinctiveness [42]. For a modified square LED of known size, the camera observation model for square fiducials [38] can be readily applied to our system. Also, due to limited LEDs for the study, we have assessed the system in a room-sized testbed and a 300 m\(^2\) office area. There is a practical difficulty in preparing enough homemade LEDs for experiments at a larger scale. With more LEDs available in future work, evaluation in wider environments will be desired.

LedMapper is not fully automated as it still requires manual input, e.g., surveying a few LEDs as control points. Yet, these are often necessary to align the LED map with the workspace for VLP. Even so, the human effort has been much reduced than manual surveys, as control points take only a minor portion of LEDs (e.g., 3/25 in our case). To align the map, the mapper now needs at least two close control points. We safely expect improved map accuracy if using more. However, this will cause increased human effort and hence less efficiency. In practice, a trade-off should be sought between accuracy and efficiency. When the architectural floor plan has updated LED locations, it can assist as informative priors. Yet, one cannot accept the map without any prior knowledge.
directly turn it into a usable map for VLP due to the lack of LED identifiers on a standard floor plan.

Compared to photodiode-based VLP systems, camera-based systems are less affected by multipath effects [2], [3] from diffuse reflections off rough surfaces (e.g., walls, floors). In daily scenarios, specular reflective materials (e.g., glass, mirrors) are relatively few [43] but challenging to camera-based systems, alongside our LedMapper. When LEDs are close to a mirror surface, the LED and its mirroring can be observed and decoded by the camera at the same time. While it is not easy to disambiguate between them, we can circumvent this issue by discarding the affected LED detections. In a worse situation, only the LED mirroring is detected. Currently, our LedMapper can not handle this corner case. In practice, specular reflections can be in part reduced by adding a polarizer on the camera lens. To solve this issue, however, much research effort is required in future work.

VIII. CONCLUSION

This paper introduced a novel system designed for efficient and accurate offline mapping of modulated LEDs for VLP, named LedMapper. Compared to manual surveys, it required much less human effort in building a usable LED map, thereby reducing the deployment costs of VLP systems in reality. A handheld mapping device with low-cost visual-inertial sensors was utilized. The mapping process entailed a surveyor wandering around the workspace with the device for data collection. Given collected sensor data and some existing or surveyed priors, it can build an accurate and workspace-aligned LED map by formulating a full-SLAM problem within a factor graph. Compared to its heuristic counterparts, LedMapper exploited input information in a sounder way, credited to the principled design following probabilistic state estimation. Finally, the system was extensively evaluated with real-world experiments in a room-scale controlled testbed and a 15x larger lab office, showing its efficacy and performance gains.

In future work, we will adapt the system to different-sized LEDs and evaluate it in larger-scale settings. It is rewarding to do tightly coupled integration for higher mapping accuracy or to explore advanced VLC modulation/coding methods for better performance. Lastly, improving the system robustness to specular reflections is challenging and yet to be solved.

REFERENCES

Qing Liang (Member, IEEE) received the B.A. degree in automation from Xi’an Jiaotong University, Xi’an, China, in 2013, and the master’s degree in instrument science and technology from Beihang University, Beijing, China, in 2016. He is currently pursuing the Ph.D. degree with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong.

His current research interests include sensor fusion, low-cost localization, and mobile robots.

Chengju Liu (Member, IEEE) received the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 2011.

From October 2011 to July 2012, she was a Research Associate with the BEACON Center, Michigan State University, East Lansing, USA. From March 2011 to June 2013, she was a Postdoctoral Researcher with Tongji University where she is currently a Professor with the College of Electrical and Information Engineering. She is also the Team Leader of the TJ Ark Robot Team, Tongji University.

His research interests include intelligent control, motion control of legged robots, and evolutionary computation.

Ming Liu (Senior Member, IEEE) received the B.A. degree in automation from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree from the Department of Mechanical and Process Engineering, ETH Zürich, Zürich, Switzerland, in 2013.

During his master’s study at Tongji University, he stayed one year in Erlangen-Nunberg University and Fraunhofer Institute IISB, Germany, as a master visiting scholar. He is currently with the Department of Electronic and Computer Engineering, the Department of Computer Science and Engineering, and the Robotics Institute, The Hong Kong University of Science and Technology, Hong Kong. His research interests include dynamic environment modeling, deep-learning for robotics, 3-D mapping, machine learning, and visual control.

Lujia Wang (Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong, in 2015.

From 2015 to 2016, she was a Research Fellow with the School of Electrical Engineering, Nanyang Technological University, Singapore. From 2016 to 2021, she was an Associate Professor with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, China. She is currently a Research Assistant Professor at the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. Her current research interests include cloud robotics, lifelong federated robotic learning, resource/task allocation for robotic systems, and applications on autonomous driving.