This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

Camera Pose Estimation using a VLC-Modulated Single Rectangular LED for Indoor Positioning

Babar Hussain, *Member, IEEE*, Yiru Wang, *Student Member, IEEE*, Runzhou Chen, *Student Member, IEEE*, and C. Patrick Yue, *Fellow, IEEE*

***Abstract*— An indoor positioning algorithm is presented that uses a single rectangular visible light communication (VLC)-modulated LED as a transmitter and the CMOS image sensor of a smartphone camera as a receiver. The real-world location of the LED is transmitted as a VLC signal that is decoded by the smartphone using optical camera communication (OCC). Then the location of the smartphone is calculated by finding the camera pose using the perspective-n-point algorithm on the real-world coordinates and corresponding image coordinates of the LED. The algorithm is experimentally evaluated as a real-time application running on a smartphone. The experimental results show that the algorithm can achieve an average 3D positioning error of less than 5 cm for an arbitrarily tilted smartphone.**

***Index Terms*— Visible Light Positioning, Optical Camera Communication (OCC), perspective-n-point (PnP) algorithm.**

I. Introduction

WITH the rapid deployment of energy-efficient LED lighting, visible light communication (VLC) is seen as a promising technology for enabling futuristic internet-of-things-driven location-based services [1], [2]. Thanks to the line of sight (LOS) property of light and ubiquity of LEDs in indoor environments, VLC-based indoor positioning systems (IPSs) can potentially achieve high accuracy and precision at a relatively low deployment cost as compared to other positioning technologies that require costly infrastructure [3]. For this reason, this area has been a focus of extensive research recently, with several solutions proposed based on localization algorithms measuring received signal strength (RSS) [4]–[6], angle of arrival (AOA) [7], time of arrival (TOA) [8], and time difference of arrival (TDOF) [9], or based on cameras [10]–[17]. Although, most of these positioning algorithms can achieve a high precision with positioning error of a few cm, there are several practical deployment challenges associated with each. Measuring RSS, AOA, TOA, and TDOF requires a photodetector (PD)-based receiver front-end circuit, which adds complexity to the implementation and limits the practical deployment scope of the system. In addition, TOA and TDOF pose additional requirements of synchronization between the transmitter and the receiver. Meanwhile, thanks to the ubiquity of smartphones, camera-based positioning systems have become highly desirable for practical deployment of IPSs for not only consumers but also robotic applications [18]–[19].

Recent progress on camera-based visible light positioning (VLP) systems has been focused on a twofold target of increasing the positioning accuracy and reducing the number of required LEDs. The earlier works demonstrated VLP using five LEDs [11], followed by four LEDs [12], with some using three LEDs [13]. More recent works, through a combination of sensors and geometric techniques, have proposed VLP systems that only require one LED [14], [15] or even a half LED [16], with all of them achieving a high precision of < 10 cm. However, all of the above works have only focused on using a circular LED light, which limits the application scope of VLP. In addition, several of these works rely on inertial measurement unit (IMU) sensors, for instance gyroscope and compass, which suffer from inherent noise and drift and affect the overall accuracy and stability of the system.

As shown in Fig. 1, a majority of the LED luminaries and signage used nowadays come in a rectangular shape. This includes ceiling lighting panels in large venues such as train stations, supermarkets, hospitals etc., which are the types of venues where high-accuracy IPSs are of most utility. In addition, LED-based advertisement signboards in shopping malls and other popular consumer venues are ideal for enabling

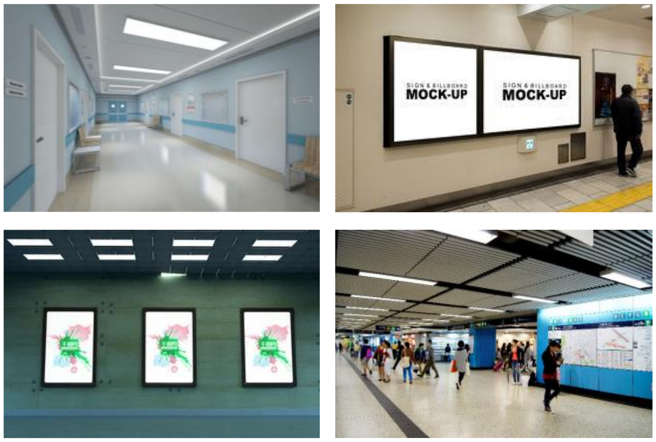


Fig. 1. Rectangular-shaped LED lighting at large scale venues.

This work was supported in part by the Foshan-HKUST Project, Government of Foshan municipality under the project FSUST20-SHCIRI06C, in part by the HKUST-BDR Joint Research Institute under HBJRI Fund Scheme project no. HBJRI-FTP-005, and Hong Kong Innovation and Technology Fund under Grant GHP/004/18SZ.

Babar Hussain, and C. Patrick Yue are with the HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, 518055, Shenzhen, China, and the Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology. (E-mail: bhussain@connect.ust.hk, eepatrick@ust.hk). Yiru Wang, and Runzhou Chen are with the Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology. (E-mail: ywangkf@connect.ust.hk, rchenas@connect.ust.hk).

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

TABLE I
RELATED WORK ON CAMERA-BASED VLP SYSTEMS

Reference	Method	Real-time/Offline	Number of LEDs	Shape of LED	Limitations
Luxapose [11]	Photogrammetry-based AOA	Offline	5	Circular	Requires high density of LEDs
Lookup [10]	Fisheye lens + Sensors	Offline	4	Circular	Requires a camera with fisheye lens
Li et al. [12]	4-LED PnP	Offline	4	Circular	Requires high density of LEDs
Lin et al. [13]	Trilateration	Offline	3	Circular	Requires high density of LEDs
Zhang et al. [14]	Circle Projection	Offline	1	Circular	Requires a marker on LEDs surface
Hao et al. [15]	Projective Geometry + IMU	No mentioned	1	Circular	Only applicable to circular LEDs and must rely on IMU sensors
Cheng et al. [16]	Geometric Features	Offline	1	Circular	
Wang et al. [17]	LED Shape Reconstruction	Offline	< 1	Circular	
Our Work	Rectangular LED-based PnP	Real-time	1	Rectangular	For ceiling mounted LEDs, requires compass for corner index determination

location-aware augmented reality (AR) advertisement and virtual reality (VR) for futuristic personalized consumer experiences. To address these needs, we present an algorithm for calculating the high-precision position of rectangular-shaped lighting and signage using a CMOS camera. Our proposed algorithm calculates the position of the camera by applying the perspective-n-point (PnP) algorithm on the corners of a single rectangular LED. In addition, our method does not require the use of gyroscope sensors to measure the tilt angles, which can reduce the cost and complexity of the system. In certain use cases, our method requires a compass to be used for resolving ambiguity in the lights corner. However, our proposed technique is immune to compass noise. We implement our algorithm as a real-time Android application and conduct experiments to verify the performance. Our contributions are as follows.

1) We propose an algorithm that utilizes a single rectangular LED to estimate the camera pose by applying the PnP problem on the four corners of the LED.
2) We reveal key image processing techniques to accurately determine the corners of the LED by removing the rolling shutter pattern from an image without affecting the geometry of the polygon.
3) We conduct simulations to study the tradeoff between various system parameters that affect positioning accuracy, including accuracy of the corners position, the size of the LED, and the distance between LED and camera.
4) We implement the positioning algorithm on an Android application with real-time performance, and experimentally evaluate the positioning accuracy in various scenarios including different sizes of LEDs, LED with display graphics, and various heights. In addition, we experimentally validate the robustness of our image processing algorithm in the presence of background lights. Our system can achieve a positioning accuracy of < 5 cm in most of the measurement results which is comparatively higher than the state-of-the-art reported results.

The rest of the paper is organized as follows. Section II discusses the related works. The proposed methodology is presented in Section III, followed by the implementation and performance evaluation in Section IV. Finally, the conclusion is provided in Section V.

II. RELATED WORK

A great deal of work has been done in utilizing LEDs for VLP [3], [20]. In terms of the receiver, these works can be categorized into PD-based and camera-based. In this section, we will present a review of camera-based VLP works and compare them with our proposed approach.

One of the pioneering works on VLP using cameras is [11], where the authors demonstrated the use of photogrammetry techniques to achieve a positioning accuracy of 7 cm using five LEDs and a smartphone. Later, in [12], a VLP system was presented that required at least four LEDs to apply the PnP algorithm to calculate the camera's position with an accuracy of < 5 cm. The use of a global shutter camera as the VLP receiver and three under-sampled phase shift-modulated LEDs as location beacons was presented in [13]. The authors recorded a video stream of the LEDs and processed it offline to demonstrate a positioning accuracy of ~6 cm. In another work [10], the authors proposed a geometric and consensus-based positioning technique by combining inclinometer and magnetometer sensors with a fisheye lens-based camera. In a large-scale venue, the system was able to achieve an accuracy of 17 cm using four LEDs.

The use of a single LED for VLP was presented in [14] using a circle projection technique to achieve a positioning error of ~17 cm. However, the proposed algorithm requires a marker to be

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

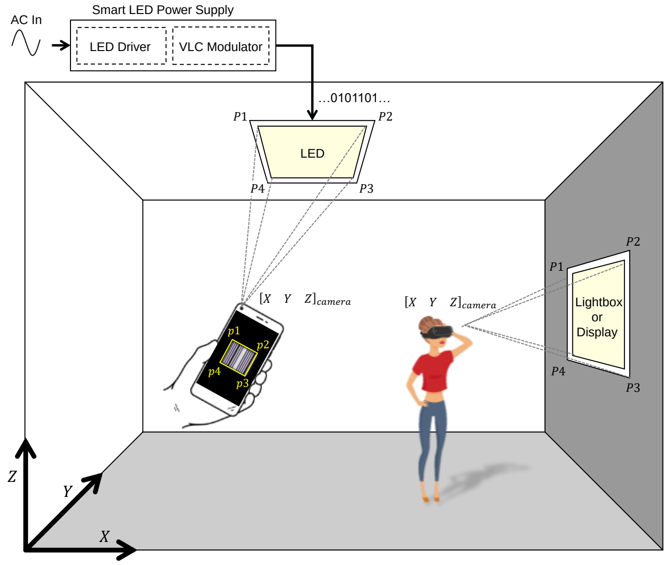


Fig. 2. System overview of a rectangular LED-based VLP system.

placed in front of the LED for identifying the orientation. Later, in [15], a single LED was combined with a smartphone's gyroscope and compass sensors to further improve the average accuracy to ~11 cm. A more recent work on single LED-based VLP [16] exploited the geometric features of a circular LED and proposed a planes intersection-line method to achieve a high accuracy for tilted smartphones. The algorithm was implemented offline and achieved a positioning accuracy of ~5.5 cm. In our most recent work [17], we proposed an LED shape reconstruction algorithm to enhance the robustness of circular LED-based VLP systems during partial blockage and tilting, and demonstrated < 5 cm accuracy through offline experiments.

The above-discussed works are summarized in Table I. It is evident that the majority of the works on VLP have been based on circular LEDs, which only represent one type of lighting form factor. In practice, a large majority of lights and displays come in a rectangular shape. In addition, single LED-based systems that use a circular LED must rely on the use of a gyroscope and compass, which makes them susceptible to noise and instability, introduces more complexity into the algorithm, and limits the scope of the system to devices that contain both a camera and IMU. Whereas, in our proposed method the use of compass is only required for ceiling mounted LEDs for finding corner index based on a relative value and can tolerate a compass error of up to 45 degrees. Lastly, most of the works presented above are based on offline implementations, with no consideration of parameter selection to support real-time practical use cases.

III. Methodology

In this section we will first give the system overview, followed by the camera pose estimation algorithm. Finally, the details of the image processing steps for VLC signal decoding and light corner detection will be provided.

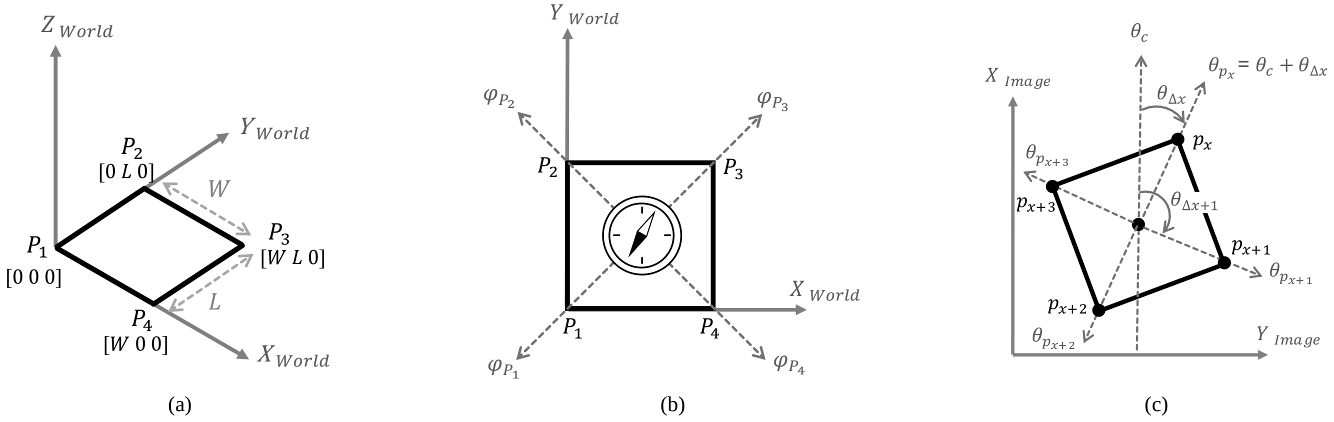
A. System Overview

An overview of the system is illustrated with an exemplary scenario in Fig. 2. A rectangular LED light is installed in a ceiling and is powered by a smart LED power supply containing an LED driver and a VLC modulator [21]. The modulator controls the intensity of the light based on a pre-stored unique binary ID to broadcast an on-off keying (OOK) signal. A smartphone with a frontside camera captures the light image to decode the VLC ID, and calculates the camera pose w.r.t the real-world coordinates of the LED by extracting the corner information from the image. The real-world 3-D coordinates of the LED light are associated with its unique ID and are either pre-stored in the smartphone or accessed through the cloud. The obtained position of the LED can be further translated into global positioning system (GPS) coordinates for displaying on a digital map [22].

In another scenario, using the same principle as described above, a rectangular LED lightbox or display can be used to identify the location of the smartphone or VR headset using the backside camera, as shown in Fig. 2 on the right.

B. Camera Pose Estimation

Camera pose estimation is a well-known problem in computer vision, with several solutions proposed that vary in their computational complexity and robustness [23]. The proposed algorithm uses the information of the four corners of a rectangular LED in an image to solve for the camera pose. The method requires the LED size i.e., W and L to be known beforehand which can be easily obtained from the product specifications. However, in certain scenarios when the size information of the LED lighting is not available, for instance in unknown environments with lighting



(a) (b) (c)

Fig. 3. (a) Real world planar corners of the rectangular LED. (b) Real word corner orientation of the LED measured using a compass. (c) Corner orientation of the LED in image plane measured by calculating angular displacement of the center-to-corner vector w.r.t smartphone compass reading

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

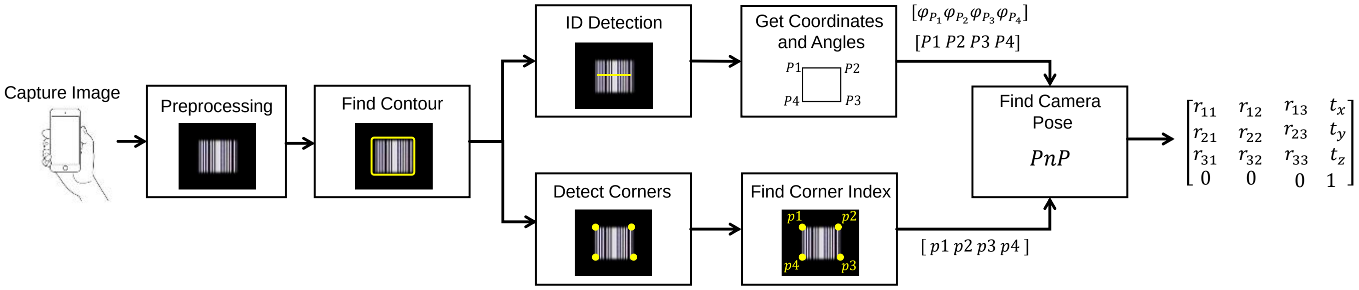

Capture Image
Preprocessing
Find Contour
ID Detection
Get Coordinates and Angles
$P1$
$P2$
$P4$
$P3$
$[\varphi_{P_1}\varphi_{P_2}\varphi_{P_3}\varphi_{P_4}]$
$[P1\ P2\ P3\ P4]$
Detect Corners
Find Corner Index
$p1$
$p2$
$p4$
$p3$
$[\ p1\ p2\ p3\ p4\ ]$
Find Camera Pose
PnP
$\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Fig. 4. Image processing and algorithm implementation steps of the proposed VLP system.

pre-installed, the size can be obtained using the camera of the smartphone itself. For instance, by placing the smartphone camera under the light with its plane parallel to LED lights plane and following the method in [24], or using stereo vision technique, as discussed in [25]. In a typical indoor environment, usually a large group of lights are of the same type, therefore, only one of the lights size needs to be determined.

Since the LED surface can be treated as a 2-D shape, the four corners can be considered coplanar points with Z = 0, as shown in Fig. 3 (a). This makes the transformation of the world coordinates to the camera coordinates a 2-D homography-based transformation. The camera pose can be extracted from the homography matrix as rotation and translation vectors [26]. The description of each step in camera pose estimation is discussed as follows.

1) *Estimation of Homography:* Assuming we already know the world coordinates of the four corners of the LED $P_i, i = 1, 2, 3, 4$, where the ith corner is adjacent to the $(i+1)th$ and $(i-1)th$ corner along the perimeter, we can express their corresponding points in the camera coordinates system, i.e., $p_i, i = 1, 2, 3, 4$, using the camera-projection equation, as follows:

$$s\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K\,[R \mid t]\,[X_i \quad Y_i \quad Z_i]^T, i = 1,2,3,4, \tag{1}$$

where K is the intrinsic camera matrix, R is the 3×3 rotation matrix, t is the translation vector, and s is the scaling factor. Since all the world coordinates points are on the same plane, $Z_i = 0$ can be put in (1) and $[R \mid t]$ can be expanded, for the equation to be written as

$$s\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K \begin{bmatrix} r_{11} & r_{21} & 0 & t_x \\ r_{12} & r_{22} & 0 & t_y \\ r_{13} & r_{23} & 0 & t_x \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ 0 \\ 1 \end{bmatrix}, i = 1,2,3,4. \tag{2}$$

If we represent the first and second column of the rotation matrix with $r_1$and r_2 respectively, then (2) can be written as

$$s\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K\,[r_1\,,\, r_2\,, t] \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}, i = 1,2,3,4. \tag{3}$$

With the removal of Z_i, (3) becomes a 2-D to 2-D mapping, and $K\,[r_1\,,\, r_1\,, t]$ can be replaced by an equivalent homography matrix H as follows:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}, i = 1,2,3,4. \tag{4}$$

If we combine and rearrange (4) for all four point correspondences, we get the following:

$$\overbrace{\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1 \\ & \vdots & & & \vdots & & & \vdots & \\ & \vdots & & & \vdots & & & \vdots & \\ X_4 & Y_4 & 1 & 0 & 0 & 0 & -x_4X_4 & -x_4Y_4 & -x_4 \\ 0 & 0 & 0 & X_4 & Y_4 & 1 & -y_4X_4 & -y_4Y_4 & -y_4 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} h_{11} \\ h_{12} \\ \vdots \\ \vdots \\ \vdots \\ h_{33} \end{bmatrix}}^{H'} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}. \tag{5}$$

The above equation is in the form of $AH^{'} = 0$, where A is an 8×9 matrix and H' is a 9×1 matrix, i.e., column vector representation of H (order 3×3). The solution for H' lies in the null space of A and can be computed using singular value decomposition.

2) *Calculation of Camera Pose:* From (3) and (4), H is related to K, R and t through a scaling factor λ as shown in (6). Since the columns of the rotation matrix have a unit norm, $\|r_1\|$ can be used as the scaling factor λ to re-write (6) as (7), while r_3 can be found by the cross product of r_1 and r_2 based on orthogonality, as shown in (8).

$$H = \lambda\, K\,[r_1\,,\, r_2\,, t]\,. \tag{6}$$

$$\frac{K^{-1}H}{\|r_1\|} = [r_1\,,\, r_2\,, t]\,. \tag{7}$$

$$r_3 = r_1 \times r_2\ . \tag{8}$$

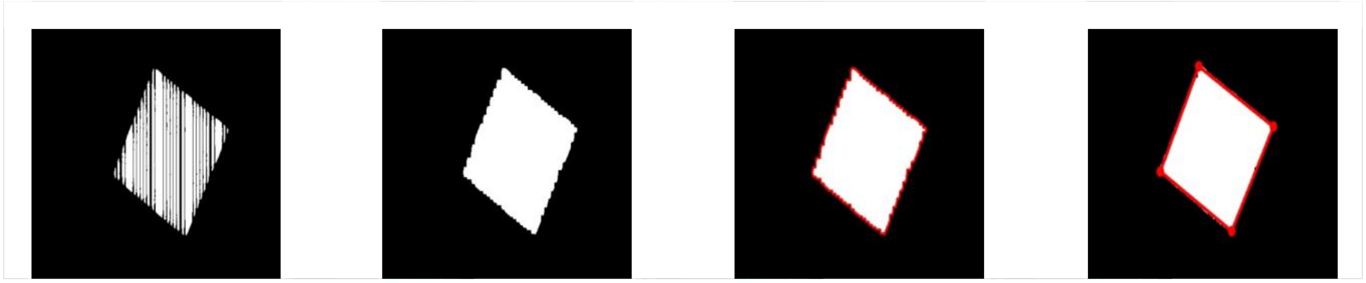
3) *Iterative Optimization:* The camera pose $[R|t]$ calculated via (7) and (8) is quite susceptible to image noise. Therefore, it needs to be further refined through iterative optimization. We choose the Levenberg-Marquardt (LM) algorithm for optimizing the camera pose iteratively by minimizing the reprojection error [27]. The reprojection error, d, is the difference between image point p_i and the reprojection $\hat{p}_i$ of its corresponding world coordinates point P_i for both the x and y axis, and it is calculated based on the camera matrix K and estimated camera pose $[R|t]$:

$$\begin{aligned} d_{i,x} &= p_{i,x} - \hat{p}_{i,x}(K, R, t, P_i), i = 1,2,3,4, \\ d_{i,y} &= p_{i,y} - \hat{p}_{i,y}(K, R, t, P_i), i = 1,2,3,4. \end{aligned} \tag{9}$$

In order to apply the LM optimization, the cost function f, defined as $f = g^T g$, with g given in (10), is optimized over six pose parameters, which include three axes of the translation vector $[t_x, t_y, t_z]^T$ and three axis-angle representations of the rotation $[\omega_x, \omega_y, \omega_z]^T$calculated using the Rodrigues transformation of R [27]. Iterative minimization is carried out based on the step

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980



(a) (b) (c) (d)

Fig. 5. Contour and polygon detection: (a) binary image (b) after morphological operations (c) contour detection (d) polygon fitting

function h and damping factor μ as given in (11).

$$g = [d_{1,x}, d_{1,y}, d_{2,x}, d_{2,y}, d_{3,x}, d_{3,y} d_{4,x}, d_{4,y}]^T. \tag{10}$$

$$(\mathbf{J}^T\mathbf{J} + \mu\ \mathrm{I}\ )h = \mathbf{J}^T g\ , \tag{11}$$

where $\mathbf{J}$ is the Jacobian matrix containing first-order partial derivatives of g w.r.t each pose parameter.

4) *Finding the Corner Indexes:* In order to match the light corners in the image (p_i) to their corresponding world corners (P_i), the general orientation along the z-axis of the camera must be known. In the case of an LED on the ceiling and smartphone being held in the hand, the compass of the smartphone can be utilized to find the correspondence between p_i and P_i based on the real-world orientation of each corner, which can be measured using the compass during system deployment. In addition, modern building plans for large scale buildings also indicate the direction of north on the blueprint eliminating the need for on-site compass measurement [28].

As illustrated in Fig. 3 (b), it is assumed that the orientation of each corner of the LED is measured in the form of the orientation angle φ_{P_i} and prestored in the location database along with the 3-D world coordinates of the LED. Here φ_{P_i} represents the orientation of a vector originating from the center of the LED and passing through the ith corner. During the image processing stage, the smartphone compass reading θ_c, which represents the real-world orientation of the x-axis of the smartphone and image plane, is used to estimate the corner orientation θ_{p_x} of one of the LED corners p_x in the image by adding θ_c to the difference angle $\theta_{\Delta x}$ which is the angle between + x-axis and the point p_x i.e. atan2 of p_x, as shown in Fig. 3 (c). Similarly, the orientation of the subsequent corners i.e., $p_{x+1}, p_{x+2}, p_{x+3}$ is calculated. The unknown index x of the corners can be matched to a real-world corner index i for which the sum of differences between each θ_{p_x} and its corresponding real-world φ_{P_i} is minimized as in (12).

$$\min\left(\sum_{x,i=1}^{4}\left\|\theta_{p_{x+k}} - \varphi_{P_i}\right\|\right)\Bigg|_{k=0,1,2,3}\ , \tag{12}$$

where k represents the index offset to be added in x to match it with its corresponding real-world corner index. Here, it is important to note that, ideally, θ_{p_i} should be equal to φ_{P_i}. However, in practice, the smartphone compass is prone to errors due to magnetic disturbances in the indoor environment. Therefore, the difference $\left\|\theta_{p_i} - \varphi_{P_i}\right\|$ represents the compass error and is tolerable up to 45 degrees for a four-cornered rectangular light, making the system substantially immune to compass error. This compass error immunity can be further validated based on the measurement data of several phones provided in [29], which measured a compass error of below 45 degrees 80% of the time.

On the other hand, in the case of an LED lightbox, or display on a wall, and the camera of a handled smartphone or VR headset, finding the indexes of the corners is relatively straightforward. Since, in this case, the orientation along the z-axis is generally known due to the limited tilting movement of the VR headset or posture at which the smartphone must be held when pointing towards a display on the wall. Alternatively, at the expense of higher computation cost, image recognition techniques could be employed to recognize some features of the graphics or text shown on the display to infer the camera orientation and identify the correct indexes of the corners.

5) *Going from one LED to another:* In practical settings, due to relatively large spacing between LED lights as compared to the camera field of view, the camera may encounter blind spots while going from one LED to another. In such scenarios, the smartphone can potentially utilize other positioning technologies to maintain a consistent positioning estimate, for instance pedestrian dead reckoning, as described in [22]. On the other hand, in some rare scenarios, the LED lights maybe installed in very close proximity such that the camera could see multiple lights at once. In such scenarios, the LED light that is closer to the optical center of the camera maybe chosen as the more reliable source during image processing stage while still using the single LED-based pose estimation method described above.

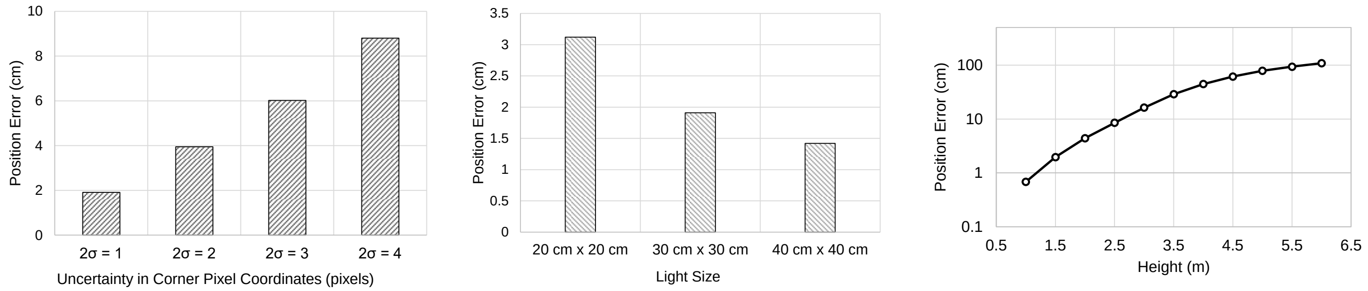
C. Image Processing for ID Decoding and Corner Detection

The image processing flow of the proposed method is shown in Fig. 4. The captured image containing the LED is first preprocessed, followed by contour detection to identify the region-of-interest (ROI) for light ID decoding and corner detection. The light ID helps to select the real-world coordinates and angles which are used in the PnP algorithm along with the corners in the image to calculate the camera pose. The details of the key steps are provided below.

1) *Preprocessing:* The preprocessing involves converting the captured image, which comes in a three-channel color format to a single-channel grayscale image. The grayscale image needs to be converted into a binary image before a contour detection algorithm can be applied for extracting the ROI and LED corners. Image binarization can be done using a number of thresholding algorithms. Here, we use Otsu's adaptive thresholding algorithm

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980



(a) Light size = 30 × 30 cm^2, Height = 1.5 m

(b) Height = 1.5 m, 2 σ = 1

(c) Light size = 30 × 30 cm^2, 2 σ = 1

Fig. 6. Simulation results showing the impact of various parameters on the positioning performance of VLP

[30], as it works well for low-exposure rolling shutter images of LEDs.

2) *Contour Detection:* The contour detection is the most critical step in the image processing as it determines how accurately LEDs corners can be detected in the image, which affects the overall accuracy of the camera pose. One of the main obstacles in the accurate contour determination is the light ID pattern on the image of the LED, which causes partial loss of shape information. Fig. 5 (a) shows a typical LED ID pattern of an 18-bit LED ID code captured using a Huawei P30 Pro front camera with a resolution of 1280 × 960 pixels. In the case of circular LEDs, blurring is generally applied to the image to smooth out the rolling shutter pattern. However, blurring will lead to the loss of shape information in the case of a rectangular LED. Therefore, to address this issue, we use a dual approach. First, we use a high frequency modulation to reduce the width of the rolling shutter pattern to the extent possible without affecting the ID decoding capability. Second, we utilize mathematical morphology to close the chopped sections of the striped LED image to form a closed rectangle shape, as shown in Fig. 5 (b). We apply a rectangular kernel to first dilate the white lines in the rectangle, followed by erosion to erode the extra pixels around the boundary which appear due to the dilation. The size of the kernel is proportional to the width of the rolling shutter pattern. If ω represents the number of pixels/bit of the rolling shutter and there are a maximum of N number of consecutive 0s in the binary sequence of the light ID, the recommended size of the kernel will be $m \times m$, where $m = (N\omega + 1)$ with all elements 1, as given in (13). The details of mathematical morphology can be found in [31].

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}_{m \times m} , m = (N\omega + 1) \ . \tag{13}$$

Once the rectangular shape of the LED has been smoothed, its contours can be detected using the boundary-following algorithm described in [32] (Fig. 5 (c)). The exact shape of the LED with corners can be found by applying the Douglas-Peucker (DP) approximation method [33] on the contours to approximate a four-point polygon by reducing the line segments in the contour, as shown in Fig. 5 (d). The four corners can then be used in the PnP algorithm for pose calculation, as described earlier.

3) *Light ID Decoding:* The light ID decoding process is performed on the ROI found using the contour detection in the previous step. We use the conventional method of rolling shutter decoding by first calculating the binary threshold of the center line of the ROI in the grayscale image to extract the binary data. The binary data are then converted into the light ID by grouping the ones and zeros using the width factor ω and identifying the start and end of the frame based on a predefined preamble. The details of the light ID decoding can be found in [19] and [34]. Each light ID represents a unique location which is stored in the database in the form of a look-up table (LUT) with a set of the 3-D world coordinates of the LED and its real-world corner orientation angles. Once a light ID is detected, its real-world corners and orientation angles are passed to the PnP algorithm for pose estimation as described above.

IV. Implementation and Evaluation

A. System Simulation

There are several parameters that could impact the positioning performance, including the light size, height, and the uncertainty in the lights corner coordinates in the image due to noise. Therefore, simulations are conducted to reveal the tradeoff between these parameters. A set of reference positioning points $(X,\ Y)$ are generated on a grid of 100 cm × 100 cm with a spacing of 20 cm between adjacent points. These points, along with a given height (H), and a set of tilting angles including roll, pitch and yaw ranging from 0 – 30°, 0 – 30°, and 0 – 180° respectively, are used to generate a set of reference camera poses (R, t) which would be used as ground truth. These poses are then used to calculate corresponding lights corner coordinates in the image using the projective geometry relation in (1). These corner points when input in the pose estimation algorithm described in Section II, would generate the pose identical to the reference pose, i.e., with zero positioning error. However, we introduced noise in these corner coordinates based on a normal distribution with zero mean (μ), and a variance (σ^2) chosen according to the maximum amount of desired deviation in lights corners. The deviation in the resulting camera poses after adding the noise is measured as the positioning error. We conducted three sets of simulations with each simulation varying one of the three parameters i.e., height, LED size, and noise in corner pixels coordinates, while keeping the others constant as shown in Fig 6.

The impact of increasing uncertainty in the corner pixel coordinates is shown in Fig. 6 (a). Here $2\sigma = 1, 2,3,4$ denotes the addition of random noise to the lights corner coordinates with 2 standard deviation i.e., 95% probability of up to 1, 2, 3, and 4 pixels respectively. It is revealed that every increase of 1 pixel in the uncertainty results in an average increase of 2 cm in the positioning error. Fig. 6 (b) shows the impact of light size

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

TABLE II
SYSTEM SPECIFICATIONS

Parameter	Specification
LED Transmit Power	5 – 10 Watt
LED Size	19×19 cm^2, 29×29 cm^2
Height	1 m, 1.5 m, 2 m
Smartphone Model	Huawei P30 Pro
Camera Resolution	1280×960
Exposure	1/10000 sec
Frame Rate	~30 fps
Frame Processing Time	16 ms

on positioning performance. As the light size decreases, the positioning error increases due to reduction in Euclidian distance [23]. Similarly, as shown in Fig. 6 (c), the increase in height also results in the increase in positioning error as the increasing height would result in a smaller projection on the image which is equivalent to a smaller light size. From Fig. 6 (c), it can also be observed that for very high ceilings of 6 m and above, the positioning error for the single-LED-based positioning system can reach up to 1 m. However, for such scenarios, multiple LED-based positioning systems could be considered as camera could cover a wider field-of-view and see multiple lights when looking up at high ceilings.

B. System Specifications and Implementation

A summary of the system design parameters is provided in Table II. The proposed algorithm is implemented as an Android application running on a Huawei P30 Pro smartphone. The frontside camera is used with a frame rate of 30 fps and the exposure value set to a minimum of 1/10,000 sec. A resolution of 1280×960 pixels is chosen for several reasons. First, it can capture enough details to successfully decode a 16 kHz modulated OCC signal. Second, it can precisely track the LED's shape for position estimation with a high accuracy that is comparable to the state-of-the-art VLP systems. Third, it is computationally light enough to achieve real-time performance on lower-end smartphones and IoT devices. Image acquisition on the Android is achieved via Camera2 API [35] whereas the compass reading for corners orientation is acquired through geomagnetic rotation vector from Android motion sensors API [36]. Both of these APIs return their data to the main activity in the application where the algorithm is implemented. On the transmitter side, A 5-Watt and a 10-Watt square-shaped LED panels are used with an LED power supply comprising LED driver and a VLC modulator [21] that is programmed to modulate the LEDs supply current to the binary pattern of the VLC ID. The light panels have a surface area of 19×19 cm^2, and 29 × 29 cm^2, respectively. In order to verify that the size of the pre-installed LEDs can be measured in unknown environments, we measured the physical dimensions of these lights using the single-shot calibration method of [24] by placing the smartphone on the ground with front-camera facing up and taking 5 readings at the heights of 1 m and 1.5 m. The size is measured with a sufficiently high average accuracy of 99% and worst-case accuracy of 98%.

C. Evaluation of Computational Performance

To see the impact of each processing step on the computation performance, we quantify the processing time of each stage during execution by tracking the Android System time. A total of 100 frames are used to measure an average value for each stage, with the summary of results shown in Table III. The total computation time of each frame is measured to be 16 ms, which leaves sufficient margin to maintain a 30 fps real-time performance. This also validates that the proposed algorithm can be implemented on computationally limited hardware, for instance low-power IoT devices, etc.

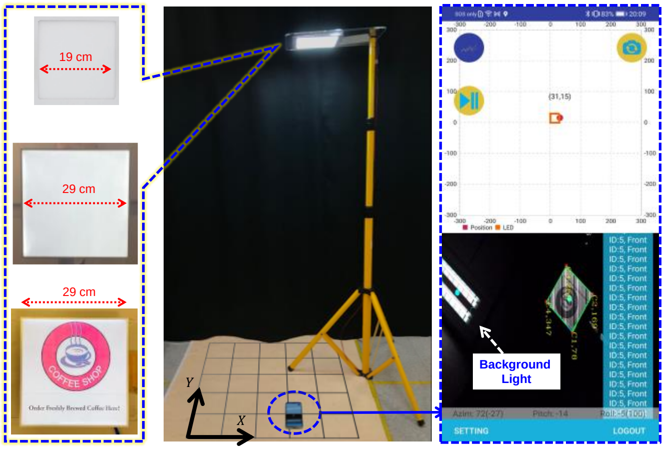


Fig. 7. Measurement setup (left) and VLP application screenshot (right).

TABLE III
SUMMARY OF COMPUTATION TIME

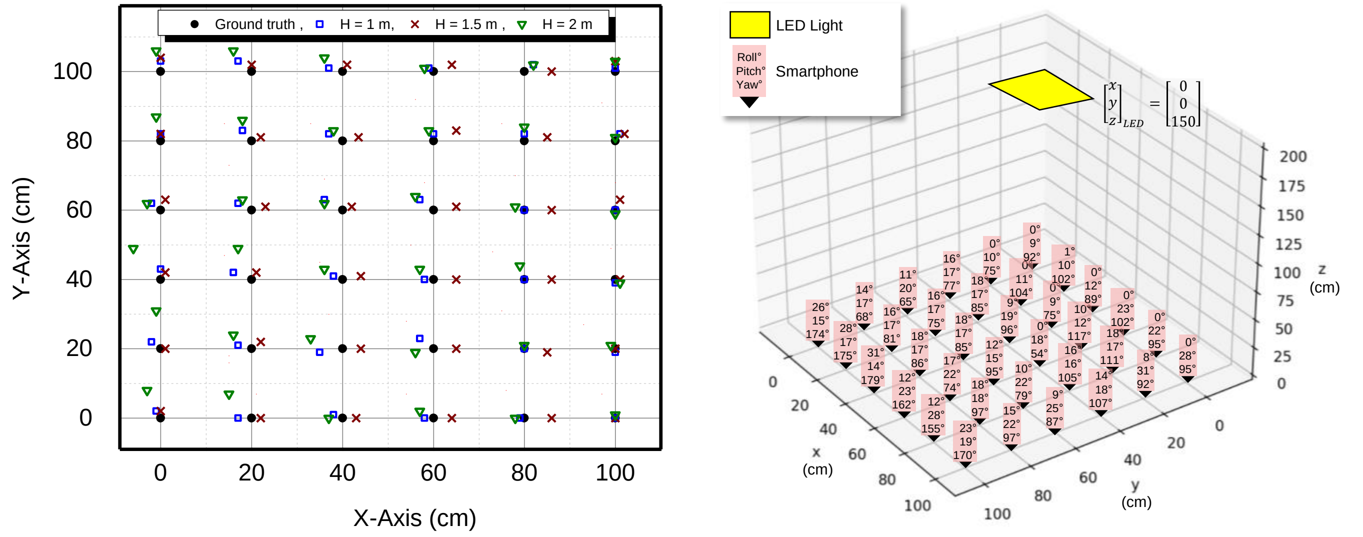
Processing Step	Computation Time (ms)
Preprocessing	6.5
Contour Detection	5
Polygon Fitting	0.5
PnP Computation	1
VLC ID Decoding	3
Total per Frame	16

D. Evaluation of Positioning Performance

The measurement setup with the types of LED lights used and a screenshot of the smartphone application is shown in Fig. 7. Several sets of measurements are conducted to verify the impact of background lights, display content, height, and light size on the positioning performance. During these measurements, we placed the smartphone on the grid with 20 cm spacing between adjacent points and note the position displayed on the smartphone application which shows a running average of 10 consecutive readings. At each point on the grid, as illustrated in Fig. 8 (b), the orientation of the smartphone is chosen arbitrarily with roll and pitch angles ranging from 0° – 30°, and yaw angle in the range of 0° – 180° while ensuring the LED light falls within the field-of-view of the camera. We first verify the robustness of the proposed system against the influence of background lights and the impact of display content on LED signage and displays. We use a 29 cm × 29 cm LED light panel with a height (H) of 1.5 m and take three measurements i.e., with background lights off, with background lights on, and with background lights on and using LED with display graphics as shown in Fig. 7. The measurement results, as shown in the cumulative density function (CDF) plot in Fig. 9, reveal that the three cases achieved nearly the same performance. The mean positioning error is measured to be

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980



(a) (b)

Fig. 8. VLP measurement results showing (a) positioning performance at various heights and (b) position and arbitrary orientation of smartphone under the LED light while conducting measurements

3.5 cm in the case of no background lights as compared to 3.7 cm with background lights on showing only a difference of 5%. This validates the capability of the system in correctly differentiating the VLC lights from the non-VLC lights and minimizing the influence of background lighting. In the case of LED with display graphics, the mean positioning error is 3.7 cm which is the same for the light panel without any graphics. This is because the display content only affects the VLC detection process by affecting the grayscale threshold which can be handled through adaptive threshold determination.

To verify the impact of height, we conduct three sets of measurements at the heights of 1 m, 1.5 m and 2 m, using the LED panel of size 29 cm × 29 cm while keeping the background lights on to further validate the robustness of the application. The measurement results, as shown in Fig. 8 (a), indicate that at higher heights the positioning error worsens which agrees with the simulation results. The CDF plot of the three sets of the height measurements is shown in Fig. 9 and the measured mean positioning errors using the 29 cm × 29 cm LED light panel for the three heights i.e., 2 m, 1.5 m and 1 m are 4.6 cm, 3.7 cm, and 2.4 cm, respectively. The increase in positioning error with increasing height is due to the dominance of image noise in the corner points when Euclidian distance among the corners is reduced [23]. To verify the impact of light size on positioning accuracy, we conduct another measurement with a light size of 19 cm × 19 cm at the height of 1.5 m which, when compared with the light size of 29 cm × 29 cm at the same height, showed an increase in positioning error, as shown in Fig. 9, with the mean value of 7.4 cm. The CDF of the positioning error of all the measurement results, shown in Fig. 9, confirms that the positioning error increases with decreasing light size or increasing height while the background lights and display content have little to no effect on the positioning accuracy.

Table IV shows a performance comparison of the related works on camera-based VLP systems. It can be observed that our proposed system achieves high-precision localization using a single rectangular LED and a comparatively much lower capture resolution, which helps achieve real-time performance.

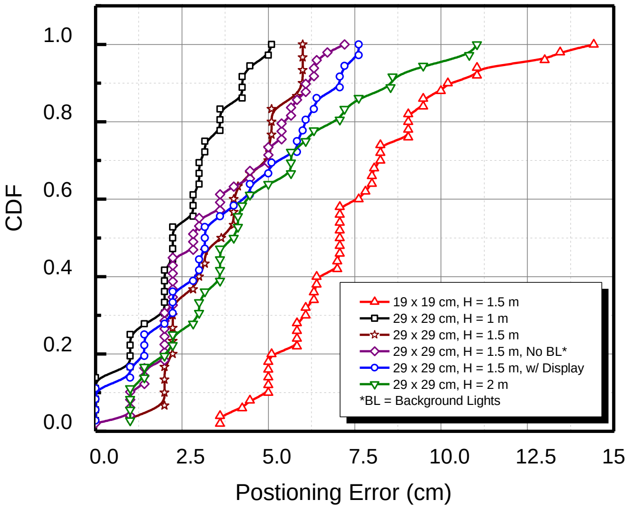


Fig. 9. Cumulative density function (CDF) of positioning error.

TABLE IV
COMPARISON OF CAMERA-BASED VLP SYSTEMS

Ref.	Method	Capture Resolution (pixels)	Real-time/ Offline	Uses Gyro	LEDs Shape	Mean Error (cm)
[12]	4-LED PnP	2.07 M	Matlab	No	○	6.6
[13]	Trilateration	0.35 M	Offline	No	○	5
[14]	Circle Projection	33.6 M	Offline	No	○	17.5
[15]	Projective Geometry + IMU	Not mentioned		Yes	○	11.2
[16]	Geometric Features	3.15 M	Offline	Yes	○	5.4
[17]	Boundary Fitting	2.07 M	Offline	Yes	○	3.9
This work	LED corners-based PnP	1.23 M	Real-time	No	□	4.6

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2022.3212980

In addition, compared to other single-LED-based systems, our method does not rely on a gyroscope for tilt compensation.

V. Conclusion

In this paper, we proposed a VLP system that uses a single rectangular LED and no gyroscope and applies a camera pose estimation algorithm on the four corners of the LED. We conducted extensive simulations and measurements to evaluate the systems performance for various heights, and light sizes. In addition, we demonstrated the robustness of our algorithm by measuring its performance in the presence of background lights, and using LEDs with display content. We implemented our algorithm on an Android smartphone to demonstrate efficient implementation, and achieved real-time performance of a 30 fps continuous positioning update. The system achieved a high-accuracy camera pose with an average positioning error of < 5 cm. The experimental validation and real-time performance proves that the proposed system can be readily deployed in practical settings, for instance, in industrial and commercial venues, to provide high-accuracy positioning for a wide range of industrial and consumer applications.

References

[1] A. Jovicic, J. Li and T. Richardson, "Visible light communication: opportunities, challenges and the path to market," in *IEEE Communications Magazine*, vol. 51, no. 12, pp. 26-32, December 2013.

[2] F. Zafari, I. Papapanagiotou, and K. Christidis, “Micro-location for Internet of Things equipped smart buildings,” *IEEE Internet Things J.,* vol. 3, no. 1, pp. 96–112, Feb. 2016.

[3] Y. Zhuang et al., “A survey of positioning systems using visible LED lights,” *IEEE Commun. Surv. Tuts.*, vol. 20, no. 3, pp. 1963–1988, Third Quarter 2018.

[4] N. Faulkner, F. Alam, M. Legg and S. Demidenko, "Watchers on the wall: Passive visible light-based positioning and tracking with embedded light-sensors on the wall," *IEEE Trans. Instrum. Meas*., vol. 69, no. 5, pp. 2522-2532, May 2020.

[5] A. H. A. Bakar, T. Glass, H. Y. Tee, F. Alam and M. Legg, "Accurate visible light positioning using multiple-photodiode receiver and machine learning," *IEEE Trans. Instrum. Meas*., vol. 70, pp. 1-12, 2021.

[6] S. H. Yang, H. S. Kim, Y. H. Son, and S. K. Han, “Three-dimensional visible light indoor localization using AOA and RSS with multiple optical receivers,” *J. Lightw. Technol*., vol. 32, no. 14, pp. 2480–2485, Jul. 15, 2014.

[7] C.-Y. Hong et al., “Angle-of-arrival (AOA) visible light positioning (VLP) system using solar cells with third-order regression and ridge regression algorithms,” *IEEE Photon. J.*, vol. 12, no. 3, pp. 1–5, Jun. 2020.

[8] T. Q. Wang al., “Position accuracy of time-of-arrival based ranging using visible light with application in indoor localization systems,” *J. Lightw. Technol.*, vol. 31, no. 20, pp. 3302–3308, Oct. 15, 2013.

[9] S.-Y. Jung, S. Hann, and C.-S. Park, “TDOA-based optical wireless indoor localization using LED ceiling lamps,” *IEEE Trans. Consum. Electron.*, vol. 57, no. 4, pp. 1592–1597, Nov. 2011.

[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, Sept. 2017.

[11] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, “Luxapose: Indoor positioning with mobile phones and visible light,” in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw*., 2014, pp. 447–458.

[12] Y. Li, Z. Ghassemlooy, X. Tang, B. Lin, and Y. Zhang, “A VLC smartphone camera based indoor positioning system,” *IEEE Photon. Technol. Lett*., vol. 30, no. 13, pp. 1171–1174, Jul. 1, 2018.

[13] B. Lin, Z. Ghassemlooy, C. Lin, X. Tang, Y. Li, and S. Zhang, “An indoor visible light positioning system based on optical camera communications,” *IEEE Photon. Technol. Lett.*, vol. 29, no. 7, pp. 579–582, Apr. 1, 2017.

[14] R. Zhang et al., “A single LED positioning system based on circle projection,” *IEEE Photon. J.*, vol. 9, no. 4, pp. 1–9, Aug. 2017.

[15] J. Hao, J. Chen, and R. Wang, “Visible light positioning using a single LED luminaire,” *IEEE Photon. J.*, vol. 11, no. 5, pp. 1–13, Oct. 2019.

[16] H. Cheng, C. Xiao, Y. Ji, J. Ni and T. Wang, "A single LED visible light positioning system based on geometric features and CMOS camera," in *IEEE Photon. Technol. Lett.*, vol. 32, no. 17, pp. 1097-1100, Sept.1, 2020.

[17] Y. Wang, B. Hussain and C. Patrick Yue, "Arbitrarily tilted receiver camera correction and partially blocked LED image compensation for indoor visible light positioning," *IEEE Sensors J.*, doi: 10.1109/JSEN.2021.3057103.

[18] Z. Yan, W. Guan, S. Wen, L. Huang and H. Song, "Multirobot cooperative localization based on visible light positioning and odometer," *IEEE Trans. Instrum. Meas*., vol. 70, pp. 1-8, 2021.

[19] W. Guan, L. Huang, B. Hussain and C. Patrick Yue, "Robust robotic localization using visible light positioning and inertial fusion," *IEEE Sensors J.*, doi: 10.1109/JSEN.2021.3053342.

[20] M. Maheepala, A. Z. Kouzani and M. A. Joordens, "Light-based indoor positioning systems: A review," *IEEE Sensors J.*, vol. 20, no. 8, pp. 3971-3995, 15 April15, 2020.

[21] B. Hussain, C. Qiu and C. P. Yue, "A universal VLC modulator for retrofitting LED lighting and signage," in *2019 IEEE 8th Global Conference on Consumer Electronics (GCCE)*, 2019.

[22] B. Hussain, Y. Wang, R. Chen, H. C. Cheng and C. P. Yue, "LiDR: Visible Light Communication-Assisted Dead Reckoning for Accurate Indoor Localization," *IEEE Internet Things J*, doi: 10.1109/JIOT.2022.3151664.

[23] S. Li, C. Xu, and M. Xie, “A robust O(n) solution to the perspective-n-point problem,” *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 34, no. 7, pp. 1444–1450, Jul. 2012.

[24] Limeng Pu, Rui Tian, Hsiao-Chun Wu and Kun Yan, "Novel object-size measurement using the digital camera," *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2016, pp. 543-548.

[25] Y. M. Mustafah, R. Noor, H. Hasbi and A. W. Azma, "Stereo vision images processing for real-time object distance and size measurements," 2012 *International Conference on Computer and Communication Engineering (ICCCE)*, 2012, pp. 659-663.

[26] S. Malik, G. Roth, and C. McDonald, “Robust 2D tracking for real-time augmented reality,” *Proc. Conf. Vision Interface*, vol. 1, no. 2, p. 12, 2002.

[27] M. Darcis, W. Swinkels, A. E. Güzel and L. Claesen, "PoseLab: A Levenberg-Marquardt Based Prototyping Environment for Camera Pose Estimation," *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2018, pp. 1-6.

[28] Buildings Department, The Government of the Hong Kong Special Administrative Region, Guidelines for using building information modelling in general building plans submission, 2019. [Online]. Available: https://www.bd.gov.hk/doc/en/resources/codes-and-references/code-and-design-manuals/BIMGBPS_e.pdf

[29] M. Holzl, R. Neumeier, and G. Ostermayer, “Analysis of compass sensor accuracy on several mobile devices in an industrial environment,” in *Computer Aided Systems Theory - EUROCAST 2013,* R. Moreno-D´ıaz, F. Pichler, and A. Quesada-Arencibia, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 381–389.

[30] N. Otsu, "A threshold selection method from gray-level histograms," in *IEEE Trans. Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62-66, Jan. 1979.

[31] Soille, P., Erosion and dilation, in Morphological Image Analysis. 2004, Springer. p. 63-103.

[32] S. Suzuki and K. Abe, “Topological structural analysis of digital binary images by border following,” *Comput. Vision Graphics Image Process*., vol. 30, no. 1, pp. 32–46, 1985.

[33] R. Urs, “An iterative procedure for the polygonal approximation of plane curves,” *Computer Graphics and Image Process*, vol. 1, 1972, pp. 244–256.

[34] B. Hussain, et al.: *JSAP-OSA Joint Symposia 2017* Abstracts, paper 6p_A409_6.

[35] Android Developers Guide, Camera2. [Online]. Available: https://developer.android.com/training/camera2

[36] Android Developers Guide, position sensors. [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_position

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: UCLA Library. Downloaded on October 10,2022 at 23:10:15 UTC from IEEE Xplore. Restrictions apply.