Joint Measurement and Trajectory Recovery in Visible Light Communication

Dongfang Zheng, Gang Chen, and Jay A. Farrell, *Fellow, IEEE*

***Abstract*— The growing preponderance of light emitting diodes (LEDs) for lighting has motivated research into their dual use for visible light communication (VLC) and navigation. VLC extracts a bit sequence from a series of photodetector scans. Among this data is an LED ID that ensures the reliable data association in navigation and data communication. Recovering the LED data and ID requires the accurate prediction of each LED's projected position on the photodetector array to extract efficiently and reliably the LED ON–OFF status in each photodetector scan. Estimating the LED projected position is challenging because: 1) clutter and noise corrupt the measurements; 2) the LED status will be OFF in some scans; and 3) the predicted projection location sequence depends on the estimated rover state trajectory, which is uncertain and time varying. This paper presents a method to determine the q-most probable data and the LED position sequences simultaneously for a time window of data, using Bayesian multiple hypothesis tracking techniques by maximizing the posterior probabilities. This paper focuses on the VLC data and the LED position sequence extraction, which includes rover state estimation. Implementation of the multiple hypothesis tracking algorithm is illustrated by postprocessed experimental results.**

***Index Terms*— Bayesian multiple hypotheses tracking, data association, localization, navigation, vehicle state estimation, visible light communication.**

I. Introduction

LIGHT emitting diode (LED) illumination sources have long operational life, low-power consumption, and are mechanically robust, leading to their growing popularity. The LED light modulation frequency is so high (>300 MHz) that it can be undetectable by the human eye, yet detectable by photodiodes (PDs), which enables LEDs installed for illumination to be also used for communication and positioning [24].

A. Visible Light Communication

As their price declines, the number of LED light sources in our surroundings is growing quickly. LEDs are now used as stop lights, streetlights, advertising signs, airport taxi-way lighting, and indoor lighting. The growing popularity of LEDs for illumination provides new opportunities to use visible light communication (VLC) to augment or replace traditional radio frequency (RF) wireless communications. Compared with the RF spectrum, the visible light portion of the electromagnetic spectrum is unlicensed and unregulated, and the power consumption of LEDs is lower than that of other radio wave transmitters [20]. VLC has an inherent line-of-sight (LOS) property. This provides an inherent level of transmission security. On the other hand, the LOS property limits the VLC communication distance to ranges that are significantly shorter than for other forms of RF communication [21]. LOS detection also increases the complexities of receiving LED data when the PD array (PDA) is moving, since different LEDs enter and leave the field of view of the PDA and the LED projections on that PDA are moving.

Data transmission by visible light is not a new idea; consider smoke signals, fire-beacons, and lighthouses. Today, the characteristics of LEDs have greatly expanded the potential bandwidth and application areas for VLC. LED communication systems using PDs, cameras, or linear arrays have been introduced in [3], [30], [37], and [49]. Their applications include indoor information broadcasting systems via ceiling lamps [18], [27] and communication systems between cars via car headlights and taillights. Using visible light to transmit music has been demonstrated in [29].

B. Visible Light Positioning

The LOS property provides information about the PDA's relative position and rotation with respect to each detected LED. When the global position of each detected LED is known, the global position of the PDA can be estimated. Such feature-based navigation systems have utility in a variety of environments, especially indoors or in urban environments, where the current generation of global navigation satellite system receivers do not function well. Emergency guidance systems, equipment location, personnel location, indoor guidance for the visually impaired, and automated vehicles for office tasks are examples of indoor applications. Fig. 1(a) shows an indoor visible light communication and positioning system using the camera in a smartphone. For outdoor applications, the largest market would be automotive applications using intervehicle communications, for example, collision avoidance, platooning, automatic guided vehicle, and collaborative vehicle interactions. Fig. 1(b) shows a mobile collision avoidance system

Manuscript received February 3, 2016; accepted March 27, 2016. Date of publication May 12, 2016; date of current version December 14, 2016. Manuscript received in final form April 4, 2016. This work was performed within the Ubiquitous Communication by Light (UC-Light) Center, which is supported in part by the University of California Multicampus Research Program and Infinitives under Grant 142787. Recommended by Associate Editor M. Fujita.

The authors are with the Department of Electrical Engineering, University of California at Riverside, Riverside, CA 92521 USA (e-mail: dongfang.zheng@email.ucr.edu; gang.chen@ucr.edu; farrell@ece.ucr.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2016.2554062

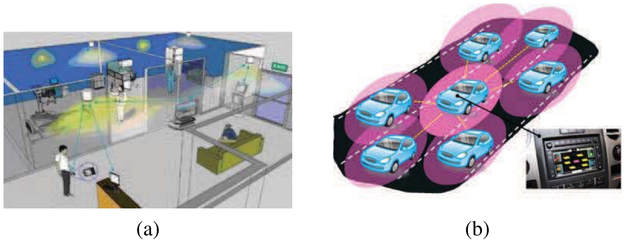


Fig. 1. VLC localization applications. (a) Indoor VLC localization using smartphones and computers. (b) Mobile collision avoidance using VLC relative localization.

wherein each vehicle computes its relative state with respect to its neighbors using LEDs and PDs installed in the vehicles. One of the most important advantages of LED features relative to passive features is that in VLC-based positioning, the unique ID of each LED transmitter can greatly enhance the reliability of the data association problem. Navigation systems using communication between LEDs and PDs have been discussed in [1], [26], [27], [43], [44], [49], [52], and [53].

For navigation, a single PD provides a simple hardware approach able to detect LED data at rates exceeding 300 MHz [24], but only offers the most basic level of positioning. This is a form of *fingerprinting* [17], [45]. Given a map of LED locations, when the PD detects the ith LED, it can look up that LED's location and know that it is in the same vicinity. This approach is binary, in the sense that the navigation either knows it is in a certain region, or it does not. If multiple LEDs are detected, then the estimated location can be further refined as the intersection of the LED detectable regions. However, multiple LEDs received through a single PD may be unidentifiable due to saturation and interference. Also, no methods have been published that estimate the full vehicle state vector using a single PD as the LED sensor. Due to the desire for high localization accuracy and full state estimation, this paper focuses on navigation systems using PDAs (e.g., camera or linear array).

Rectangular PDAs (i.e., cameras) provide each LED's projected position in the image as two angle-of-arrival (AOA) measurements. Such AOA measurements, combined suitably over time, allow accurate estimation of the vehicle state [35], [38], [46]. A camera also spatially separates the LED signals so that they do not interfere. However, the low-frame rate of standard cameras (e.g., 30 Hz) limits the achievable communication data rate. Standard commercial cameras have a fixed sequential pixel read approach. More expensive (>\$10 k) cameras suitable for research demonstrations, such as this, can be reprogrammed at the beginning of an experiment to read smaller rectangular portions of the camera PDA, so that the camera frame rate for this smaller region can be significantly increased (up to 100 k frames/s).[1]

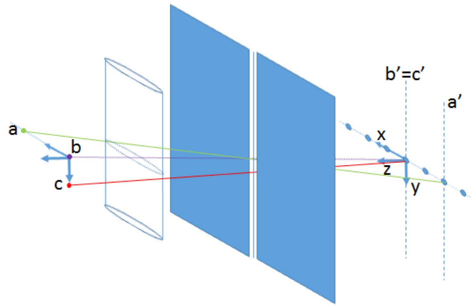


Fig. 2. Linear array physical setup.

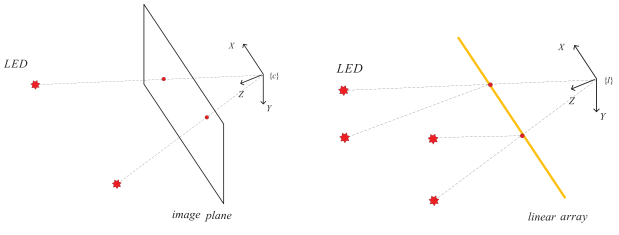


Fig. 3. (a) Camera and (b) linear array measurement models.

A linear array has a set of PDs arranged to form a straight line arranged behind a slit screen and a cylindrical lens as depicted in Fig. 2. The cylindrical lens and slit screen are oriented with their axis perpendicular to the axis of the linear array. The slit causes each light source to cast a bright line parallel with the y-axis onto the PD $x-y$ plane. In Fig. 2, points a, b, and c project to lines a', b', and c'. Because these lines intersect the x-axis containing the PDs, this physical arrangement causes LEDs separated sufficiently in the x-direction to project to distinct LEDs, as indicated by points a and b projecting to lines a' and b' in Fig. 2. Multiple LEDs on the plane defined by the slit (i.e., a line) and any one of the PDs (i.e., a point) will project to that PD and overlap (i.e., interfere) with each other. See lines b' and c' in Fig. 2.

While a spherical lens with a 2-D PDA in a camera, through the pinhole model yields an image plane model, a cylindrical lens with a 1-D PDA, through a narrow slit model yields an image line model. The measurement models for cameras and linear arrays are shown in Fig. 3. A linear array can be viewed as a small part of a camera, either a single row of pixels or as a single row that average all pixels in its (camera) column. Compared with a single PD or a camera, the linear array of PDs simultaneously enables a high sampling rate and accurate rover state estimation at low cost. The number of pixels in a linear array [32] is much smaller than that in a camera; therefore, for the same analog-to-digital conversion rate as a camera, each pixel can be sampled more frequently.

While a camera is able to distinguish LEDs that project to distinct image-plane $x - y$ locations, a linear array can only resolve LEDs that project to distinct locations along the linear array x-axis. Points with the same x and z coordinates, but differing in their y coordinates will project to the same linear array pixel. Therefore, for example, a linear array of PDs placed horizontally on a street would not be able to separately

[1] There is an important tradeoff to be noted in any VLC system. For an LED with a given brightness (average photons emitted per second) the number of photons received per sample will decrease inversely with the sampling rate, which can affect the brightness of the received signal. Eventually, as the sample rate increases, detection of the LED signal would become unreliable. Our experiments use standard commercial LEDs and this was not an issue at the communication rates used in our experiments.

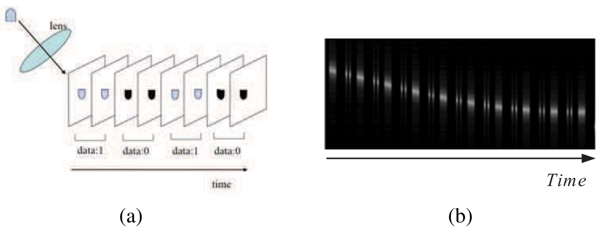


Fig. 4. LED data in a sequence of PD scans. (a) Depicted sequence of camera images of a scene containing a blinking LED [20]. (b) Experimental sequence of linear array scans (vertical slices) for a scene containing a blinking LED.

resolve the red and green signals from a stoplight if both were ON simultaneously. The linear array preserves the sensitivity to AOA around a single axis (i.e., linear array y-axis), so it is useful when the sensor moves in a 2-D plane (i.e., on a floor or road). Previous analysis of navigation and communication using a linear PDA, including simulation results, is presented in [49].

C. LED Detection and Data Recovery

This paper considers LED data modulated using as an ON–OFF keying scheme [25]. Recovering the LED data requires extracting the ON and OFF sequence of the LED from a record of consecutive scans of the PDA. The projection of an ON LED is a bright blob on the PDA. Depictions of signaling LEDs in a sequence of camera and linear array images are shown in Fig. 4. For 2-D image measurements, various feature detectors and descriptors, including corner (Förstner [15], Harris and Pike [19], Moravec [34], FAST [41], Shi and Tomasi [42]) and blob detectors (CENSURE [2], SURF [7], SIFT [31]), have been developed to find and describe features in images. These features can also be used to match the projections of the same feature in a temporal sequence of 2-D images. Other simpler techniques like searching for LEDs based on the intensity or color can also be used. Intensity and color-based methods are especially useful for linear array measurements where the existing corner or blob detectors cannot be used. Such LED detectors may extract several potential features in the image besides LEDs, resulting in clutter. To reduce the effort and time required to search for features, a typical vision-aided navigation system will not search the entire image; instead, it will define a smaller search area centered on the predicted feature projection location with the extent of the search area based on the projection uncertainty. However, due to the clutter measurements and noise, there may still be multiple potential measurements appearing in this search region. This can be the case even when the LED is OFF. The goal of the method developed herein is reliable data extraction and navigation even in such situations.

D. Multiple Hypothesis Data Association

Multiple detections existing within the search area—due to clutter, nonzero LED width, or specular reflections—yield a multiple hypothesis data association problem [13]. This data association problem could be solved by choosing the measurement that is closest to the predicted measurement position in the Mahalanobis sense [33]. This method does not work in this application, as it fails in the presence of noise or clutter when the LED is OFF. Another method that accounts for the null hypothesis (i.e., LED is OFF) can be implemented using the probabilistic data association filter [6]. Though this method works in theory, there is a very low probability to obtain a correct data sequence when it is long.

A Viterbi-based algorithm was introduced in [50]. The approach evaluates the probability of LED projections in a time sequence of linear array measurements and keeps the most probable projection sequence ending in each pixel. Based on the projection sequences that it maintains, the most probable LED projection sequence can be recovered backward through time and then the most probable data sequence can be extracted. The algorithm in [50] applies only to the linear arrays (not to cameras) and requires the assumption that the bandwidth of the motion of the measured projection sequence is much smaller than the image sample rate, for example, when the sensor is stationary. Although the LED switching frequency can be modulated up to hundreds of megahertz, sometimes to reduce cost it is preferable to use a low-rate sensor (i.e., 30 Hz) such as a webcam or a cellphone camera to receive the data using undersampling methods [40]. In such cases, the assumption for the algorithm in [50] is not valid. Finally, the Viterbi approach only provides the most probable sequence. The experiments have shown that the most likely sequence is not necessarily the correct sequence.

More sophisticated methods can be developed using multiple hypothesis tracking (MHT) (see [5], [28], [39]). MHT uses a deferred-decision approach in which the complete set of possible data associations within a sliding window is maintained, putting off hard decisions as long as possible [16]. Decisions are made by evaluating the probabilities of each sequence of data association hypotheses. In this application, each hypothesis sequence will include an estimated rover state trajectory. Since the number of possible sequences increases exponentially with the duration of the window, it is expensive to keep them all. It is also inefficient to compute the probabilities of all possible sequences and then discard most of them. To address these issues, an efficient hypothesis-oriented MHT was first proposed in [11]. The basic idea of this method was to keep only the q-best sequences, discarding the other sequences that have a lower probability.

From the above discussion, it should be clear that the communication and navigation applications are synergistic. For both the camera and linear array, the LED's unique ID would facilitate and ensure the reliability of the task of feature association. Measurement confirmation is straightforward by comparing the extracted and expected ID. Compared with other data association algorithms such as Mahalanobis distance [33] and template matching [10], using the unique ID is much more precise and reliable. The residual between the measured and expected LED detection locations is useful for the navigation state (e.g., position, attitude, and velocity) estimation. On the other hand, communication requires knowledge of which LED IDs to expect for efficiency of demodulation and knowledge of where each LED is expected to be in the image for efficiency in image processing. Accurate navigation enables both aspects. This paper fully develops and demonstrates the MHT approach for the VLC and navigation application.

E. Contributions and Outline

To the best of our knowledge, this is the first paper to apply an MHT tracking algorithm to joint LED measurement recovery and rover state estimation using PDAs (linear or rectangular). This paper also gives deep analysis about the relationship between LED data recovery and rover state estimation, noting that rover state estimation and LED data recovery are strongly coupled. Improvement of the solution for either problem enhances the solution of the other. From the localization (i.e., state estimation) viewpoint, this analysis leads to a reliable ability to use blinking LEDs as landmarks to accurately localize the rover's position. On the other hand, from the VLC viewpoint, this analysis provides a reliable and accurate solution for the problem of recovering LED data on a moving PDA.

The MHT-based LED data recovery method was introduced briefly in [51]. That article included only simulation results for a camera implementation. This paper extends this method for both linear and rectangular PDA sensors, fully discusses the probabilistic modeling, includes greater details about both LED data and vehicle state trajectory recovery, and includes real-world (postprocessed) experimental results using a camera. For each data association hypothesis probability, this paper explicitly analyzes the information about the number of measurements, which is not clearly explained in the MHT literature.

This paper is organized as follows. Section II defines the predicted region and data association hypothesis, and formulates the problem mathematically. Section III-A analyzes the posterior probability of the joint hypothesis. Section III-B introduces the implement method obtaining q-best hypotheses. Section IV discusses platform trajectory recovery. Section V gives the experimental results.

II. PROBLEM FORMULATION

This section presents the mathematical formulation of the problem. A specific example is presented in Section IV.

The PD is part of a rigid system, called the *rover*, with kinematic state (i.e., position, velocity, and attitude) denoted by $x(t) \in \mathbb{R}^n$. The rover state evolves according to

$$\dot{x} = f(x, u) \tag{1}$$

where f is a known nonlinear mapping and $u \in \mathbb{R}^m$ is the actual system input. In a navigation system, the input u represents the rover's motion information, which could be measured by an encoder or inertial measurement unit (IMU).

The lth LED at known and fixed location F_l projects onto the rover PD at position

$$z_l(t) = h(x(t), F_l) \tag{2}$$

at time t, where $z_l \in \mathbb{R}^p$ and the projection function h is defined in [49] (linear array) and [53] (camera). Accounting for additive measurement noise $n_{z_l}(t)$, the measurement of z_l is modeled as

$$\tilde{z}_l(t) = h(x(t), F_l) + n_{z_l}(t). \tag{3}$$

Each LED is switching its ON or OFF status to communicate information. The LED projected position can be detected only when the LED is in its ON state.

Given a time interval $\lambda \in [t_s, t]$, the projected position of the lth LED $z_l(\lambda)$ defines a trajectory across the PDA. Detection of the projected LED position depends on the LED ON or OFF status, as well as on the environmental conditions and interference from other light sources. The accuracy of each LED's recovered data stream is highly dependent on the accuracy of its projected PDA trajectory and the data association at each time step. Therefore, the data recovery and trajectory estimation problems are coupled. Improvement in the solution of either enhances the solution of the other.

The purpose of this paper is to develop an algorithm that simultaneously estimates the q-most likely data association sequences and rover state trajectories over the time interval $\lambda \in [t_s, t]$. Without loss of generality (WLOG), we define $t_s = 0$ and the samples times as $t_k = kT \in [t_s, t]$ for $k = 0, 1, \ldots, K$. After the data association step and the validity of the LED data, the feature association is established and the validated sequence of measurements $\tilde{z}_l(t_k)$ will be referred to as the *aiding measurements*. The state estimate corresponding to the most probable validated hypothesis is taken as the correct state estimation result.

A. Predicted Measurement Region

This paper uses the symbol $\hat{a}$ to represent the estimate of variable a. Given an initial distribution $x(0) \sim \mathcal{N}(x_0, P_0)$, the computer calculates an estimate $\hat{x}$ of x according to

$$\dot{\hat{x}}(t) = f(\hat{x}(t), \hat{u}(t)) \tag{4}$$

where $\hat{u}(t)$ is the estimate of $u(t)$ computed from its measurement $\tilde{u}(t)$. In the simplest case, the measurement is modeled as $\tilde{u}(t) = u(t) + \omega(t)$, where the process noise ω has the power spectrum density denoted by Q.

Define $\delta x(t) = x(t) - \hat{x}(t)$ as the state error. The error propagation model can be approximated by subtracting (4) from (1) and linearizing the result along the estimated trajectory

$$\delta\dot{x}(t) = F(t)\delta x(t) + G(t)\omega(t) \tag{5}$$

where

$$F(t) = \left.\frac{\partial f}{\partial x}\right|_{(\hat{x}(t),\hat{u}(t))} \quad \text{and} \quad G(t) = \left.\frac{\partial f}{\partial u}\right|_{(\hat{x}(t),\hat{u}(t))}.$$

The discrete-time equivalent error propagation model is

$$\delta x(k) = \Phi_{k-1}\delta x(k-1) + \omega_{k-1} \tag{6}$$

where $x(k)$ represents $x(t_k)$ and $\omega_{k-1} \sim \mathcal{N}(0, Q_{k-1})$. Computation of the state error transition matrix Φ_{k-1} and process noise covariance Q_{k-1} are discussed in [14, Sec. 4.7]. The state error covariance P_k evolves over time according to

$$P_k = \Phi_{k-1}P_{k-1}\Phi_{k-1}^{\top} + Q_{k-1}. \tag{7}$$

Given the state estimate $\hat{x}(k)$ at time-step $\{0, 1, ...K\}$, it is straightforward to compute both the predicted measurement position of the lth LED $\hat{z}_l(k)$ on the PDA and its error covariance $S_l(k)$

$$\hat{z}_l(k) = h(\hat{x}(k), F_l) \tag{8}$$

$$S_l(k) = H_{l,k} P_k H_{l,k}^\top + R_{l,k} \tag{9}$$

where $H_{l,k}$ is the linearized measurement matrix and $R_{l,k} = cov(n_{z_l}(k))$, both of which are defined in [52]. The quantities $\hat{z}_l(k)$ and $S_l(k)$ define a prior distribution for the LED trajectory on the PDA that can focus on the algorithm. A likely region for the lth LED on the PDA is

$$\begin{aligned} V_{l,\gamma}(k) &\triangleq \left\{z : (z - \hat{z}_l(k))^\top S_l^{-1}(k)(z - \hat{z}_l(k)) \le \gamma\right\} \\ &= \left\{z : r_l(k)^\top S_l^{-1}(k) r_l(k) \le \gamma\right\} \end{aligned} \tag{10}$$

where $r_l(k) = z(k) - \hat{z}_l(k)$ is the residual. The parameter γ, selected using an χ^2-distribution table, determines the probability that the real measurement falls in $V_{l,\gamma}(k)$. For the camera PDA, the region $V_{l,\gamma}(k)$ represents the interior of an ellipse. For the linear PDA, the region $V_{l,\gamma}(k)$ represents the interior of a line segment. WLOG, we only consider a single LED in its predicted region $V_{l,\gamma}$; therefore, we drop the l subscript.

Due to nonzero LED width and optical imperfections, LED projections on the PDA typically take the form of small convex clusters of the above threshold pixels. The center point of such clusters is selected as the candidate LED feature.

B. Data Association Hypotheses

Let $m_k \ge 0$ denote the number of detected LED features in the predicted region $V_\gamma(k)$ at time k. Define the measurement set at time-step k as

$$Z(k) \triangleq \{z_j(k)\}_{j=1}^{m_k}. \tag{11}$$

The symbol $z_j(k)$ denotes the projected location of the jth measurement *hypothesis*. Note that at most one measurement in $Z(k)$ was generated by the LED. The others are false detections due to noise, interference, or clutter. Since each or none of the measurements in $Z(k)$ could have originated from the LED, the number of data association hypotheses is $(m_k + 1)$. The null hypothesis (no detection) is indicated by $j = 0$. Therefore, $j \in \{0, \dots, m_k\}$.

Define

$$Z^k \triangleq \{Z(l)\}_{l=1}^k \tag{12}$$

as the set of measurements up to time-step k. For a time window with K time steps, the number of joint data association hypotheses is

$$L_K = \prod_{k=1}^K (m_k + 1). \tag{13}$$

Let $\theta^{k,\ell}$ represent a specific list of joint data association hypotheses up to time-step $k \le K$. The superscript $\ell \in \{1, \dots, L_k\}$ is the numeration index for the hypothesis sequence. Each hypothesis $\theta^{k,\ell}$ has the form

$$\theta^{k,\ell} = \{j_1, j_2, \dots, j_i, \dots, j_k\} \tag{14}$$

where $j_i \in \{0, \dots, m_i\}$. When $\theta^{k,\ell}(i) = j_i > 0$, it hypothesizes that at time-step i the j_ith measurement is the LED projection. When $\theta^{k,\ell}(i) = 0$, the hypothesis is that at time-step i none of the measurements originated from the LED. We say the joint hypothesis $\theta^{k,\ell}$ is an extension of $\theta^{k-1,s}$, if

$$\theta^{k,\ell} = \{\theta^{k-1,s}, j_k\} \tag{15}$$

where $\theta^{k,\ell}(k) = j_k \in \{0, \dots, m_k\}$. In this nomenclature, $\theta^{k-1,s}$ is the *parent hypothesis* of $\theta^{k,\ell}$. Herein, the indices ℓ and s in the superscript of θ are not necessarily identical, since the two hypotheses are enumerated independently at different time steps. Note that each joint data association hypothesis $\theta^{k,\ell}$, given that each has different measurements, corresponds to a distinct posterior estimate of the vehicle trajectory: $\hat{X}^{k,\ell} = \{\hat{x}^\ell(i)\}_{i=1,\dots,k}$ along with its covariance sequence $P^{k,\ell} = \{P_i^\ell\}_{i=1,\dots,k}$. Therefore, the algorithm presented herein, which keeps the q-most likely hypotheses will also propagate q state trajectories, one for each hypothesis.

At time k, each prior hypothesis $\theta^{k-1,s}$ provides a prior state estimate $\hat{x}^s(k-1)$ with error covariance matrix P_{k-1}^s, which are propagated to time k using (4) and (7), enabling computation of $V_{s,\gamma}(k)$, $Z(k)$, and m_k using (8)–(10). Each hypothesized measurement $z_j(k)$ corrects the hypothesized prior state using the standard Kalman filter measurement update to provide posterior estimates for $\hat{x}^\ell(k)$ and P_k^ℓ. The algorithm for maintaining only the q-best hypotheses will be presented in Section III-B.

Each data association sequence over a time window of length K defines an LED trajectory across the detector, which in turn decodes a sequence of detection $(j > 0)$ and nondetection $(j = 0)$ events into a bit sequence containing an ID, checksum, data, etc. The total number of time steps K and the number of hypotheses q are parameters in the algorithm selectable by the designer. The choice of K depends on the length (number of bits) in the LED data package and the sampling rate of the PD. If the length of LED data package is l_d and the sampling rate is n_s times of the data rate, the total number of time steps satisfies $K \ge n_s l_d$. Only after the LED status is recovered at each time instant in the sequence of K measurements, can the LED data be extracted. Fig. 4(b) shows the experimental data from a sequence of linear array measurements. The variation in the LED projected position due to rover motion, the ON–OFF status of the LED due to the data sequence, and the projection of the LED across multiple pixels are all clearly shown.

C. Technical Problem Statement

Given the vehicle motion model (1) and LED measurement model (2), and the measurement set Z^K defined in (12) and input set

$$U^K = \{u(l)\}_{l=0}^K$$

find the q-most probable joint data association hypotheses: $\Theta_q = \{\theta^{K,\ell}\}_{\ell=1}^q$.

After these hypotheses are found, their corresponding data sequences (containing LED IDs) are straightforward to recover. The predicted LED ID is available based on the state estimate, because the LED location is assumed to be known. By comparing each of the recovered and predicted IDs, the most probable element of Θ_q with the correct ID is selected and used both for the data extraction and the navigation state update. The advantage of finding the q-best hypotheses instead of only the most probable one is that having several candidate hypotheses yields a higher probability to recover at least one with the correct ID.

III. MHT-Based LED Data Recovery Method

This section evaluates the posterior probability mass function (PMF) of a joint data association hypothesis up to time-step k. The analysis builds on the method in [39] while adding details to clarify the measurement quantity information m_k. The presentation will assume that the measurements due to clutter are uniformly distributed in the predicted region V_γ.

A. Hypothesis Probability

The joint probability of an association hypothesis up to time-step $k \in \{1, \dots, K\}$ can be decomposed as

$$\begin{aligned} &p(\theta^{k,\ell} \mid Z^k, U^{k-1}) \\ &\quad = \frac{1}{c} p(Z(k) \mid \theta^{k,\ell}, Z^{k-1}, U^{k-1}) p(\theta^{k,\ell} \mid Z^{k-1}, U^{k-1}) \end{aligned} \tag{16}$$

where the normalization factor $c = p(Z(k) \mid Z^{k-1}, U^{k-1})$ is independent of the data association hypothesis $\theta^{k,\ell}$. The decomposition on the right-hand side of (16) is based on Bayes rule. The last term on the right-hand side can be further simplified by application of the general multiplication rule for conditional probability [i.e., $p(A, B) = p(A \mid B)p(B)$]

$$\begin{aligned} p(\theta^{k,\ell} \mid Z^{k-1}, U^{k-1}) &= p(\theta^{k,\ell}(k), \theta^{k-1,s} \mid Z^{k-1}, U^{k-1}) \\ &= p(\theta^{k,\ell}(k) \mid \theta^{k-1,s}, Z^{k-1}, U^{k-1}) \\ &\quad \times p(\theta^{k-1,s} \mid Z^{k-1}, U^{k-1}). \end{aligned}$$

Combining the last equation above with (16) yields the final desired result

$$\begin{aligned} p(\theta^{k,\ell} \mid Z^k, U^{k-1}) = &\frac{1}{c} p(Z(k) \mid \theta^{k,\ell}, Z^{k-1}, U^{k-1}) \\ &p(\theta^{k,\ell}(k) \mid \theta^{k-1,s}, Z^{k-1}, U^{k-1}) \\ &p(\theta^{k-1,s} \mid Z^{k-1}, U^{k-2}). \end{aligned} \tag{17}$$

When analyzing this joint probability, it is important to note that $m^k = \{m_i\}_{i=1}^k$ is not explicitly represented in (17). In [4], this quantity information m^k is considered to be part of the measurement set Z^k, while in [39] it is viewed to be in the hypothesis set $\theta^{k,\ell}$. The results given by the two approaches are identical except for a scale factor $p(m^k)$ that is independent of $\theta^{k,\ell}$, which can be seen from

$$p(\theta^{k,\ell}, m^k \mid Z^k, U^{k-1}) = p(\theta^{k,\ell} \mid Z^k, m^k, U^{k-1}) p(m^k).$$

We use the approach of [39], so that Z^k and $Z(k)$ will contain only the measurement position information.

The first term on the right-hand side of (17) is the joint prior probability of the current position measurements given the data association hypotheses, measurements, and inputs from all the previous time steps. The measurements in $Z(k)$ are assumed to be independent,[2] because they come from different sources, potentially one from the LED and the others from clutter, etc. With this independence assumption for the measurements in $Z(k)$, this term decomposes as

$$\begin{aligned} &p(Z(k) \mid \theta^{k,\ell}, Z^{k-1}, U^{k-1}) \\ &\quad = \prod_{j=1}^{m_k} p(z_j(k) \mid \theta^{k,\ell}, Z^{k-1}, U^{k-1}) = \prod_{j=1}^{m_k} f_z(j). \end{aligned} \tag{18}$$

Note that the quantity of measurements, m_k, is considered to be known when evaluating the probability of the measurement set $Z(k)$ in (18). With the stated assumption that measurements due to clutter (or false alarms) are uniformly distributed in the predicted region and the assumption that the LED projection measurement is corrupted by Gaussian noise, the function $f_z(j)$ in (18) is

$$f_z(j, j_k) = \begin{cases} \dfrac{1}{V_k}, & \text{for clutter} \\ & \text{(i.e., } j \neq j_k) \\ \mathcal{N}(z_j(k); \hat{z}^s(k), S^s(k)), & \text{for the LED} \\ & \text{(i.e., } j = j_k) \end{cases} \tag{19}$$

where V_k is the size (area or length) of the predicted region $V_{\gamma(k)}$ and $\mathcal{N}(z_j(k); \hat{z}^s(k), S^s(k))$ is the Gaussian distribution evaluated at $z_j(k)$ with expectation $\hat{z}^s(k)$ and covariance $S^s(k)$.

Because $\theta^{k,\ell}$ is an extension of $\theta^{k-1,s}$ as defined by (15), the conditioning variables allow the distributions in (18) to be written as

$$\prod_{j=1}^{m_k} p(z_j(k) \mid \theta^{k-1,s}, j_k, Z^{k-1}, U^{k-1}).$$

The variable j_k and the index j determine the form of f_j as defined in (19). The variables $\theta^{k-1,s}, Z^{k-1}$, and U^{k-1} are the prior information used to compute $\hat{x}^s(k-1)$, $\hat{z}^s(k)$, and $S^s(k)$, where the superscript s indicates its corresponding hypothesis $\theta^{k-1,s}$, which are computed based on (4)–(9) with measurement updates implemented by an optimal estimator (e.g., Kalman filter). More details for a specific example are discussed in Section IV.

Substituting (19) into (18), the first term of (17) is

$$\begin{aligned} &p(Z(k) \mid \theta^{k,\ell}, Z^{k-1}, U^{k-1}) \\ &\quad = \begin{cases} \dfrac{1}{V_k^{m_k}}, & \text{for } \theta^{k,\ell}(k) = 0 \\ \dfrac{\mathcal{N}(z_j(k); \hat{z}^s(k), S^s(k))}{V_k^{m_k - 1}}, & \text{for } \theta^{k,\ell}(k) \neq 0. \end{cases} \end{aligned} \tag{20}$$

The second term of (17) is the prior probability of the current association hypothesis $\theta^{k,\ell}(k)$ conditioned on $\theta^{k-1,s}$,

[2] LED reflections would violate this assumption, if they fall within V_γ. The following analysis ignores such reflections.

Z^{k-1}, and U^{k-1}. This term is computed as

$$p(\theta^{k,\ell}(k) \mid \theta^{k-1,s}, Z^{k-1}, U^{k-1}) = \begin{cases} (1-P_{\text{ON}})\mu_{\lambda V_k}(m_k), & \text{for } \theta^{k,\ell}(k)=0 \\ \dfrac{1}{m_k}P_{\text{ON}}\mu_{\lambda V_k}(m_k-1), & \text{for } \theta^{k,\ell}(k)=1\cdots m_k \end{cases} \tag{21}$$

where P_{ON} is the probability that the LED is ON and μ_F is the PMF of the number of false measurements due to clutter or noise. The probability P_{ON} depends on the coding scheme and information content that each LED sends. We assume P_{ON} is known with value 0.5. This assumption can be enforced using a Manchester code.

The existence of the term $\mu_{\lambda V_k}$ in (21) is due to the measurement quantity information m_k in $\theta^{k,\ell}(k)$. When $\theta^{k,\ell}(k)=0$, all the measurements in the predicted region are not from the LED, so the LED is OFF. We do not consider the case of missed detections. When $\theta^{k,\ell}(k) \neq 0$, the hypothesis is that the LED is ON and that it is associated with a specific measurement in the predicted region. Because the distribution is not conditioned on the current position measurement $Z(k)$, when evaluating this prior probability, each measurement has equal probability (i.e., $1/m_k$) to be the correct one.

The number of measurements due to clutter and false alarms is modeled as a Poisson process [5], so its distribution function μ_F at time-step k is

$$\mu_{\lambda V_k}(\phi) = \exp^{-\lambda V_k}\frac{(\lambda V_k)^{\phi}}{\phi!}. \tag{22}$$

The parameter λ represents the spatial density of clutter measurements. The term λV_k is the expected number of clutter measurements in the predicted region (i.e., $E\langle m_k\rangle = \lambda V_k + P_{\text{ON}}$). In some applications, the parameter λ is unknown. A common solution [4] is to use the true number of measurements m_k to replace its expected value, λV_k, in (22). This results in a nonparametric model of μ_{m_k}. Note that in this special case, it can be shown that $\mu_{m_k}(m_k) = \mu_{m_k}(m_k-1)$.

Let $\beta^{k,\ell} = p(\theta^{k,\ell} \mid Z^k, U^{k-1})$ denote the posterior PMF of $\theta^{k,\ell}$ on the left-hand side of (17). Substituting (20)–(22) into (17) and combining the constants into c', in the case that λ is known, the probability is

$$\beta^{k,\ell} = \begin{cases} \dfrac{1}{c'}\lambda(1-P_{\text{ON}})\beta^{k-1,s}, & \text{for } \theta(k)^{k,\ell}=0 \\ \dfrac{1}{c'}\mathcal{N}^{k-1,s}(z_j(k))P_{\text{ON}}\beta^{k-1,s}, & \text{for } \theta^{k,\ell}(k)=j\neq 0 \end{cases} \tag{23}$$

where $\mathcal{N}^{k-1,s}(z_j(k)) = \mathcal{N}(z_j(k); \hat{z}^s(k), S^s(k))$, and $c' = ce^{\lambda V_k}(m_k!)\lambda^{1-m_k}$. If λ is unknown, using the nonparametric model with μ_{m_k}, the probability is

$$\beta^{k,\ell} = \begin{cases} \dfrac{1}{c''}\dfrac{m_k}{V_k}(1-P_{\text{ON}})\beta^{k-1,s}, & \text{for } \theta^{k,\ell}(k)=0 \\ \dfrac{1}{c''}\mathcal{N}^{k-1,s}(z_j(k))P_{\text{ON}}\beta^{k-1,s}, & \text{for } \theta^{k,\ell}(k)=j\neq 0 \end{cases} \tag{24}$$

where $c'' = cV_k^{m_k-1}m_k$. Equation (23) or (24) shows how to compute the probability of joint data association incrementally for each hypothesis.

If all the former data association hypotheses in (23) are marginalized, the marginal likelihood that the jth measurement at time-step k has originated from the LED is

$$\mathcal{L}_{s,j}(k) = \mathcal{N}^{k-1,s}(z_j(k))P_{\text{ON}} \tag{25}$$

and the marginal likelihood function for the hypothesis that no measurement at time-step k is from the LED is

$$\mathcal{L}_{s,0}(k) = \lambda(1-P_{\text{ON}}). \tag{26}$$

Therefore, the probability that hypothesis $\theta^{k,\ell}(k) = j$ is the extension of joint hypothesis $\theta^{k-1,s}$ is

$$p(\theta^{k,\ell}(k) = j \mid Z^k, \theta^{k-1,s}, U^{k-1}) = \frac{\mathcal{L}_{s,j}(k)}{\sum_{i=0}^{m_k}\mathcal{L}_{s,i}(k)}. \tag{27}$$

At each time step, only the marginalized probability needs to be computed.

B. Computing the q-Best Hypotheses

The total number of possible hypotheses grows exponentially with the time-step K (13), while only one hypothesis is correct. Computing the probability of all possible hypotheses and then discarding most of them is inefficient. To overcome this difficulty, it is preferable to compute only those hypotheses having a relatively high probability. This is reasonable since the correct hypothesis (LED path) should always be among the most probable hypotheses. An efficient method to implement this approach was first introduced in [11]. The basic idea of this method is, at each time step, to only keep the q-best hypotheses, discarding the hypotheses that have a lower probability. This method employs Murty's algorithm [36] to find the jth best hypothesis solution.

Given the q-best hypotheses $\{\theta^{k-1,i}\}_{i=1}^{q}$, their corresponding probabilities $\{\beta^{k-1,i}\}_{i=1}^{q}$ up to the prior time $k-1$, and the measurement set $Z(k)$ at the current time-step k, the algorithm will generate the new q-best hypotheses $\{\theta^{k,i}\}_{i=1}^{q}$ up to the current time-step k with their updated probabilities. Define $\theta^{k-1,i}$ to be the parent hypothesis of $\theta^{k,j}$, if the latter is the extension of the former. Choosing the single hypothesis j_k at time-step k, $\theta^{k,*} = \{\theta^{k-1,i}, j_k\}$ is one possible extension of $\theta^{k-1,i}$, where $*$ is replaced by an integer to enumerate the extensions. The best extension of $\theta^{k-1,i}$ is

$$\theta^{k,i_1} = \{\theta^{k-1,i}, j_k\} \quad \text{where } j_k = \arg\max_{j}\mathcal{L}_{i,j}(k). \tag{28}$$

The notation i_1 in the superscript of θ^{k,i_1} means that this hypothesis is the most probable extension of $\theta^{k-1,i}$. The best extension corresponds to the hypothesis having the maximum marginal likelihood.

The following text describes the implementation details of the approach in [11]. To generate the q-best joint hypotheses up to the current time-step k, the first step is for each hypothesis $\theta^{k-1,i}$ to generate its best extension according to (25), (26), and (28). These new joint hypotheses are ordered according to their probabilities and are stored in the ordered list HYP-LIST. Their probabilities are calculated according to (23) and are stored in the PROB-LIST. The second step is for each hypothesis in HYP-LIST to use its parent hypothesis to generate the

TABLE I
q-BEST HYPOTHESES ALGORITHM

```
Input: $q$-best hypotheses up to former time step $\{\theta^{k-1,i}\}_{i=1}^{q}$,
       their corresponding probabilities $\{\beta^{k-1,i}\}_{i=1}^{q}$,
       the state estimates $\{\hat{x}^i(k-1)\}_{i=1}^{q}$ and covariances
       $\{P_{k-1}^{i}\}_{i=1}^{q}$,
       current measurement set $Z(k)$
Output: $q$-best hypotheses up to current time step $\{\theta^{k,i}\}_{i=1}^{q}$
       and their corresponding probabilities $\{\beta^{k,i}\}_{i=1}^{q}$,
       the current state estimates $\{\hat{x}^i(k)\}_{i=1}^{q}$ and
       covariances $\{P_k^i\}_{i=1}^{q}$,
1. Initialize HYP-LIST and PROB-LIST
       HYP-LIST$\triangleq \{\theta^{k,i_1}\}_{i=1}^{q}$
       $\theta^{k,i_1} \triangleq \{\theta_{i,1}(k), \theta^{k-1,i}\}$
       $\theta_{i,1}(k) = \arg\max_{\theta^{k,\ell}(k)} p(\theta^{k,\ell}(k) \mid Z^k, \theta^{k-1,s}, U^{k-1})$
2. Sort the hypotheses in HYP-LIST according to their
   probabilities
3. for $i = 1 : q$
       If the $j$-th best new hypothesis generated by $\theta^{k-1,i}$
       is still in HYP-LIST, then generate its $(j+1)$-th
       best new hypothesis.

       If the probability of the new hypothesis is higher
       than the lowest probability in PROB-LIST, then
       add it into the list and delete the hypothesis with
       the lowest probability.

       If not, break.
   end
```

jth ($j = 2, 3, \ldots$) best extension. If the probability of this extension is higher than the lowest probability in PROB-LIST, add it to HYP-LIST and the corresponding probability to PROB-LIST, and delete the hypothesis with the lowest probability from HYP-LIST and its corresponding probability from PROB-LIST. If the probability of this extension is lower than the lowest probability in PROB-LIST, stop generating new extensions by its parent hypothesis. After these processes conclude, the algorithm has produced the q-best hypotheses up to the current time step. Table I describes the algorithm.

Note that the factor $\mathcal{N}^{k-1,s}(z_j(k))$ in (23) is the Gaussian distribution evaluated at $z_j(k)$, with expectation $\hat{z}^s(k)$ and covariance $S^s(k)$. The process of computing the two parameters $\hat{z}^s(k)$ and $S^s(k)$ is discussed in Section III-C.

C. Hypothesis: Computed Quantities

As stated in Section II, rover state estimation and LED path recovery are coupled. When computing the probability of each LED path hypothesis from (23), the algorithm also computes various other useful items

$$\left\{\begin{array}{l}\hat{x}^{s+}(k-1)\\ \\ P_{k-1}^{s+}\end{array}\right. \Rightarrow \left\{\begin{array}{l}\hat{x}^{s-}(k)\\ \\ P_{k}^{s-}\end{array}\right. \Rightarrow \left\{\begin{array}{l}\hat{z}^{s}(k)\\ \\ S^{s}(k).\end{array}\right. \tag{29}$$

At the end of hypothesis generation for time-step $k-1$, for each hypothesis $\theta^{k-1,s}$ where $s = 1, \ldots, q$, we have the posterior state estimate $\hat{x}^{s+}(k-1)$ and error covariance matrix P_{k-1}^{s+}, where a $+$ in the superscript indicates a posterior estimate and a $-$ in the superscript will indicate a prior estimate. The first arrow in (29) represents the state and covariance temporal propagation processes generating the prior quantities at time k for hypothesis s. The state and error covariance are temporally propagated using (4) and (7). The second arrow in (29) represents the (prior) measurement prediction process using (8) and (9).

The probability of new hypothesis $\{\theta^{k-1,s}, j\}$ can be evaluated after computing $\mathcal{N}^{k-1,s}(z_j(k))$. Using the algorithm presented in Section III-B, the new q-best hypotheses up to the current time-step k are selected. Given each new hypothesis, for which $j \neq 0$, its posterior state estimate is updated according to the standard EKF measurement update process

$$\hat{x}^{+}(k) = \hat{x}^{-}(k) + P_k^{-} H_k^{\top} S^{-1}(z_j(k) - \hat{z}(k)) \tag{30}$$

$$P_k^{+} = P_k^{-} - P_k^{-} H_k^{\top} S^{-1} H_k P_k^{-}. \tag{31}$$

If $j = 0$, the LED is OFF and there is no LED position measurement at the current time-step k; therefore, the posterior state estimate will be identical to the prior.

IV. Trajectory Recovery: Encoder Model

In Section III, we presented and analyzed the LED data recovery algorithm with emphasis on the hypothesis probability evaluation equations and the hypothesis pruning method. When computing the hypothesis probability according to (23) or (24), the term $\mathcal{N}^{k-1,s}(z_j(k))$ defined in (20) needs to be computed. Note that this term depends on the state estimate $\hat{x}_{k-1}^{s}$ at time-step $k-1$ and its covariance P_{k-1}^{s}.

Section III-C discussed the relation between the hypothesis probability and the rover state estimate, but did not cover the detailed process of the state estimate computation. This section presents specific details for the computations of (4)–(9) of the presented approach for a land vehicle example. For illustrative purposes, postprocessed experimental data will be fully developed with application results in the following section.

A. Motion Sensor Model

For a land vehicle moving on a 2-D plane (i.e., building floor), the rover navigation state vector can be defined as $x_r = [n \;\; e \;\; \psi]^{\top}$, where $\{n, e\}$ represents the 2-D position coordinates in the navigation frame and ψ represents the rover's yaw angle (heading). The (unicyle model) kinematic model is

$$\dot{n} = \cos(\psi)u, \quad \dot{e} = \sin(\psi)u, \quad \dot{\psi} = \omega. \tag{32}$$

Equation (32) defines the function f from (1). The linear velocity u and angular rate ω are related to the wheel rotation rates $\dot{\phi}_L$ and $\dot{\phi}_R$ by

$$u = \frac{1}{2}(R_L\dot{\phi}_L + R_R\dot{\phi}_R), \quad \omega = \frac{1}{L}(R_L\dot{\phi}_L - R_R\dot{\phi}_R) \tag{33}$$

where L is the axle length and R_R and R_L are the wheel radii. The wheel rotation rates are computed from the encoders.

Given the (integer) wheel encoder measurements $e_L(t_k)$ and $e_R(t_k)$, [14, Sec. 9.2] shows that the navigation state can be propagated in discrete time using the following equations. The encoder increments are

$$\begin{aligned}\Delta e_L(k) &= e_L(t_k) - e_L(t_{k-1})\\ \Delta e_R(k) &= e_R(t_k) - e_R(t_{k-1}).\end{aligned}$$

Let $\Delta T_k = t_k - t_{k-1}$. Important criteria for accuracy are that the wheels do not slip, the encoder sample period ΔT_k be small, and that no counts are missed. Given the encoder increments, the discrete-time yaw update is

$$\hat{\psi}_k = \hat{\psi}_{k-1} + \frac{1}{L}\frac{2\pi}{C}(\hat{R}_L \Delta e_L(k) - \hat{R}_R \Delta e_R(k))$$

where C is the counts per wheel revolution. The discrete-time vehicle position update is

$$\hat{p}_k = \hat{p}_{k-1} + \begin{bmatrix}\cos(\hat{\psi}_k)\\ \sin(\hat{\psi}_k)\end{bmatrix}\frac{\pi}{C}[\hat{R}_L \Delta e_L(k) + \hat{R}_R \Delta e_R(k)]$$

where $\hat{p}_k = [\hat{n}_k, \hat{e}_k]^\top$.

The augmented state vector is $x = [n, e, \psi, R_L, R_R]^\top$ where we include the wheel radii so that they can be estimated to enhance the accuracy. With the error state vector defined as

$$\delta \boldsymbol{x} = [\delta n, \delta e, \delta\psi, \delta R_L, \delta R_R]^\top \tag{34}$$

linearization of the above discrete-time state equations around the estimated state yields the following linear error-state propagation model:

$$\delta x_{k+1} = \Phi_k \delta x_k + G_k \nu_k \tag{35}$$

where

$$\Phi_k = \begin{bmatrix} 1 & 0 & -s_\psi s_k & \dfrac{\pi\,\Delta e_L(k) c_\psi}{C} & \dfrac{\pi\,\Delta e_R(k) c_\psi}{C}\\ 0 & 1 & c_\psi s_k & \dfrac{\pi\,\Delta e_L(k) s_\psi}{C} & \dfrac{\pi\,\Delta e_R(k) s_\psi}{C}\\ 0 & 0 & 1 & \dfrac{2\pi}{LC}\Delta e_L(k) & -\dfrac{2\pi}{LC}\Delta e_R(k)\\ 0 & 0 & 0 & \lambda & 0\\ 0 & 0 & 0 & 0 & \lambda \end{bmatrix}$$

$$G_k = \begin{bmatrix} \dfrac{\pi}{C}\hat{R}_L c_\psi & \dfrac{\pi}{C}\hat{R}_R c_\psi & 0 & 0\\ \dfrac{\pi}{C}\hat{R}_L s_\psi & \dfrac{\pi}{C}\hat{R}_R s_\psi & 0 & 0\\ \dfrac{2\pi}{LC}\hat{R}_L & -\dfrac{2\pi}{LC}\hat{R}_R & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\nu_k = [\,\delta e_L(k) \quad \delta e_R(k) \quad \omega_L(k) \quad \omega_R(k)\,]^\top \tag{36}$$

where $s_k = (\pi/C)[\hat{R}_L \Delta e_L(k) + \hat{R}_R \Delta e_R(k)]$, $s_\psi = \sin(\hat{\psi}_k)$, and $c_\psi = \cos(\hat{\psi}_k)$.

The vector ν_k contains four driving noise terms. The symbols $\delta e_L(k)$ and $\delta e_R(k)$ represent errors in the measured encoder increments. Their distribution is unknown, but is assumed to be white and Gaussian with covariance σ_e^2. The symbols $\omega_L(k)$ and $\omega_R(k)$, each with covariance denoted by σ_ω^2, are the driving noise for the assumed Gauss–Markov wheel radii errors model

$$\begin{aligned}\delta R_L(k+1) &= \lambda \delta R_L(k) + \omega_L\\ \delta R_R(k+1) &= \lambda \delta R_R(k) + \omega_R.\end{aligned} \tag{37}$$

The parameter $\lambda \in (0, 1)$. If the uncertainty in the wheel radii has a covariance σ_R^2, then the parameters λ and σ_ω^2 can be selected such that $\sigma_\omega^2 = (1 - \lambda^2)\sigma_R^2$ (see [14, Sec. 9.2]). For example, assuming a correlation time of $T_c = 1000$ s for the wheel radii errors and a sample time of $T = 0.01$ s yields $\lambda = e^{-T/T_c} = 0.99999$. If the wheel radii uncertainty is $\sigma_r^2 = 0.0005$ m^2, then $\sigma_\omega^2 = 5 \times 10^{-9}$ m^2.

For six degree-of-freedom maneuvering, similar ideas can be developed for applications using an IMU [48].

B. Photodetector Array Models

All analysis up to this point applies to either a linear or a rectangular PDA. This section presents their models.

1) Camera Model: When using a camera to measure the LED projected positions, each LED brightly projects to a (usually convex) blob of pixels. The camera measurement model [53] accounts for camera distortion and intrinsic parameters. The distortion and intrinsic parameters can be compensated using standard offline calibration methods [9], [47]. Approaches to compute undistorted coordinates (u, v) from the actual measurements $(\tilde{u}, \tilde{v})$ are introduced in [22] and [53], allowing us to work with the undistorted camera measurement model which is described by

$$\boldsymbol{z} = \begin{bmatrix} u\\ v \end{bmatrix} = \begin{bmatrix} {}^c x / {}^c z\\ {}^c y / {}^c z \end{bmatrix} + \boldsymbol{n} \tag{38}$$

where $\boldsymbol{n}$ is the 2×1 image noise vector, with covariance matrix $\boldsymbol{R} = \sigma^2 \boldsymbol{I}_2$. The symbol ${}^c\boldsymbol{p}_{cL} = [{}^c x, {}^c y, {}^c z]^\top$ is used to represent the LED position coordinates in the camera frame, which is computed in real-time using the standard equations

$${}^c\boldsymbol{p}_{cL} = {}^c_b\boldsymbol{R}({}^b\boldsymbol{p}_{bL} - {}^b\boldsymbol{p}_{bc}) \tag{39}$$

$${}^b\boldsymbol{p}_{bL} = {}^b_n\boldsymbol{R}({}^n\boldsymbol{p}_{nL} - {}^n\boldsymbol{p}) \tag{40}$$

where ${}^b\boldsymbol{p}_{bL} \in \Re^3$ is the position of the LED relative to the rover represented in the body frame, ${}^n\boldsymbol{p}_{nL} \in \Re^3$ is the known and fixed position of the LED in the navigation frame. The direction cosine matrix ${}^b_n\boldsymbol{R} \in \Re^{3\times 3}$ rotates vectors from the navigation frame to the body frame. It can be computed from the Euler angles where the roll and pitch angles are zero due to the 2-D flat surface assumption and the yaw angle $\psi(t)$ is computed by the navigation system. The vector ${}^n\boldsymbol{p} = [n, e, z]$ is the time-varying rover position in the navigation frame. Because the vehicle is moving on a flat surface, the vertical position is known to have the value $z = 0$ in this 2-D application. The extrinsic parameters are ${}^b\boldsymbol{p}_{bc} \in \Re^3$ and ${}^c_b\boldsymbol{R} \in \Re^{3\times 3}$. Both the body-to-camera offset vector in the body frame ${}^b\boldsymbol{p}_{bc}$ and the rotation matrix ${}^c_b\boldsymbol{R}$ from body to camera frame are computed offline.

The linearized camera measurement error model is

$$\begin{aligned}\delta z &= \boldsymbol{H}\delta\boldsymbol{x} + \boldsymbol{n}\\ &= -\boldsymbol{J}\,{}^c_n\hat{\boldsymbol{R}}\;\delta^n\boldsymbol{p} + \boldsymbol{J}\,{}^c_b\boldsymbol{R}\,\lfloor {}^b\hat{\boldsymbol{p}}_{bL}\times\rfloor\delta\boldsymbol{\theta} + \boldsymbol{n}\end{aligned} \tag{41}$$

where

$$\boldsymbol{J} = \frac{1}{{}^c\hat{z}}\begin{bmatrix} 1 & 0 & -\dfrac{{}^c\hat{x}}{{}^c\hat{z}}\\ 0 & 1 & -\dfrac{{}^c\hat{y}}{{}^c\hat{z}} \end{bmatrix} \tag{42}$$

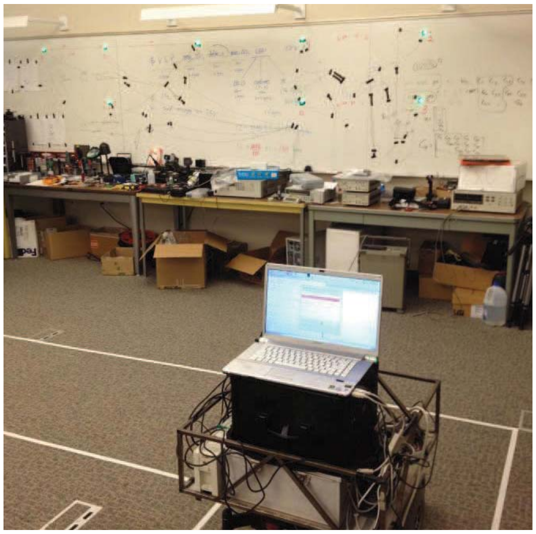


Fig. 5. Experiment platform moving within a safe operating area depicted by tape boundaries. The green spots on the white board are operating LEDs.

the notation $\lfloor \boldsymbol{a}\times \rfloor$ represents the skew-symmetric cross product matrix of the 3-D vector $\boldsymbol{a}$ and $\delta^n\boldsymbol{p}$ is the position error vector of which the only nonzero components are δn and δe. The symbol $\delta\boldsymbol{\theta}$ represents the attitude error vector for which the only nonzero term is $\delta\psi$. Therefore, with the rover error state as defined in (34), the camera measurement matrix is

$$\boldsymbol{H} = \boldsymbol{J}\left[-{}^c_n\hat{\boldsymbol{R}}(:,1:2) \quad {}^c_b\boldsymbol{R}\boldsymbol{J}_\psi{}^b\hat{\boldsymbol{p}}_{bL} \quad \boldsymbol{0}_{3\times 2}\right] \tag{43}$$

$$\boldsymbol{J}_\psi = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{44}$$

2) Linear PDA: A linear PDA measurement model is easily derived from the camera model. A linear PDA is equivalent to a single row of pixels in a camera, effectively providing only the $\tilde{u}$ measurement defined in (43). Therefore, its measurement model is

$$\tilde{z} = {}^cx/{}^cz + \boldsymbol{n} \tag{45}$$

where ${}^cx$ and ${}^cz$ have the same meaning as that in (38). The corresponding measurement matrix is the first row of $\boldsymbol{H}$ as defined in (43).

V. Experimental Results

This section presents an example postprocessed implementation of the algorithm using experimental data from a camera. The experiment uses a rover equipped with wheel encoders to test and demonstrate the algorithm. The encoder-based model specifics were discussed in Section IV.

The rover implements a trajectory tracking controller which is a (nonadaptive) version of the command filtered backstepping approach described in the example section of [12]. The rover moves in a $5m \times 5m$ area with 11 LEDs mounted on the four walls at known locations. The position of the rover in the navigation frame is represented by the vector ${}^n\boldsymbol{p} = (n, e, d)$, where $d = 0$. The experimental setup is shown in Fig. 5. Encoders attached to each front of the wheels measure the wheel rotation, which allows computation of the encoder increments $\Delta e_L(k)$, $\Delta e_R(k)$, rover speed u, and angular rate w. The origin of the body frame is the projection onto the floor of the center of the axle connecting the two front wheels. The camera's position ${}^b\boldsymbol{p}_{bc}$ and pose ${}^c_b\boldsymbol{R}$ relative to the body frame are calibrated offline. The frame rate of the camera in the experiment is 20 Hz. The parameter γ in (10) is selected such that the probability that the residual falls within $V_{l,\gamma}$ is 0.997. To produce the data for this section, the vehicle was driven through the lab environment. All the encoder data and candidate camera measurements [i.e., $Z(k)$ in (11)] were saved as a function of time, allowing the algorithm to be executed in MATLAB using postprocessing to construct the annotated figures in this section.

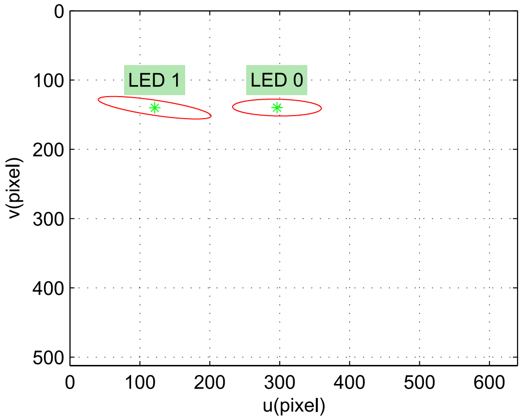


Fig. 6. Predicted LED positions in the image plane based on the prior rover state information: predicted positions (green stars) and 3-σ error ellipse regions (red ellipses).

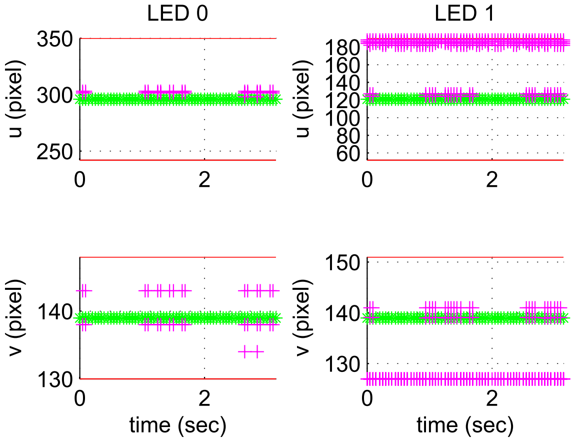
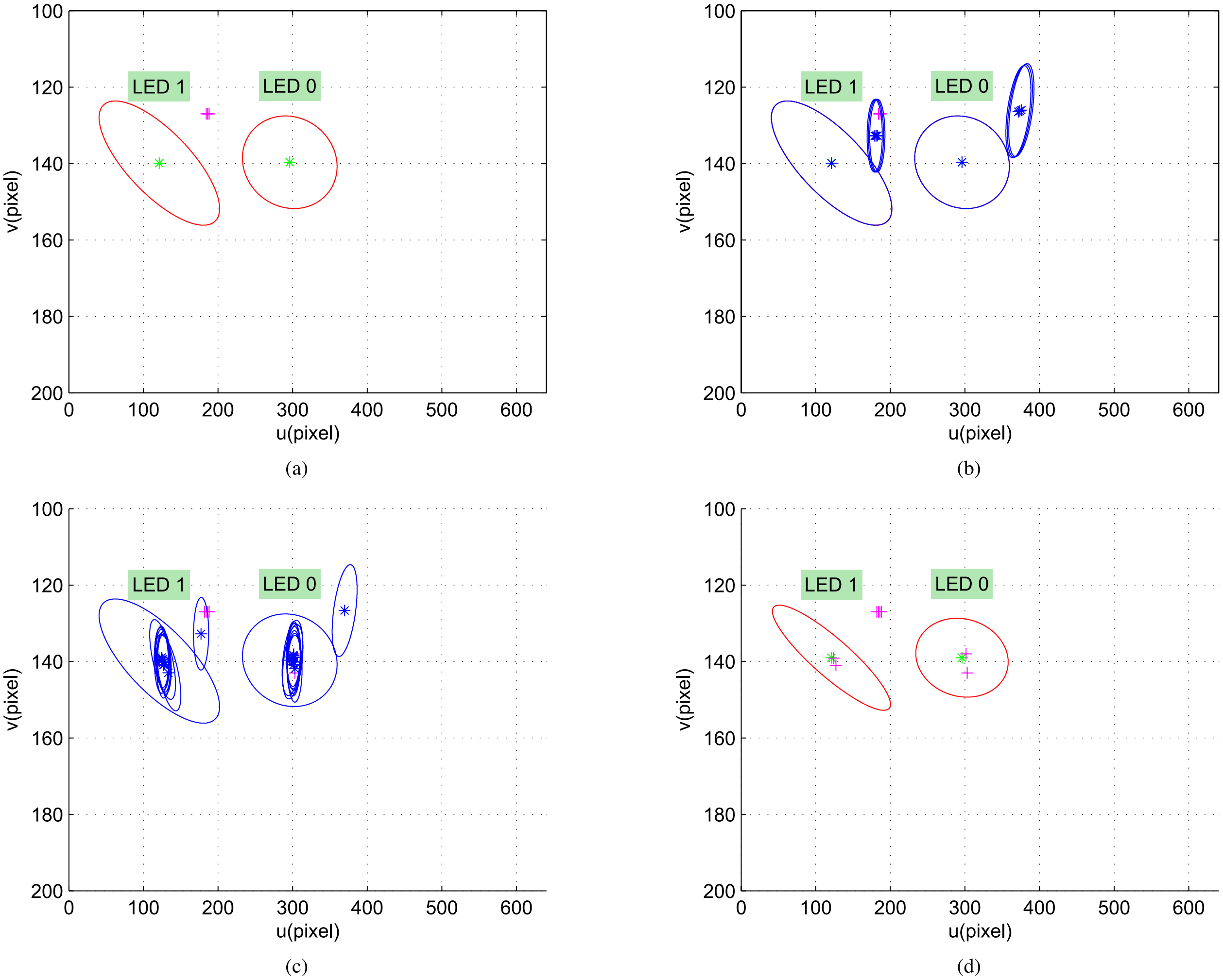


Fig. 7. Predicted and detected camera measurements for LED 0 and LED 1 in the first few seconds: predicted LED positions (green stars), 3-σ error regions (red lines), and LED measurements (magenta crosses).

A. Stationary

This section considers a stationary rover located near ${}^n\boldsymbol{p} = (-0.5, 0, 0)$ with a rotation angle $\psi = 0^o$. Therefore, the prior estimate of the rover state is $\hat{\boldsymbol{x}}_0 = [-0.5, 0, 0]^\top$. Because the rover is stationary, the inputs from the encoders are zero and the estimate of the vehicle state remains unchanged during the time update.



(a)

(b)

(c)

(d)

Fig. 8. Measurements and hypotheses at the first two steps. (a) Candidate measurements $Z(k)$ (magenta crosses), predicted LED locations (green stars), and uncertainty ellipse (red circles) at $t = 0$. (b) Predictions of two LED locations with uncertainty ellipses for each of the three state estimate hypotheses $\hat{x}^{\ell}(0)$ for $\ell = 1, 2, 3$. (c) Predictions of two LED locations with uncertainty ellipses for each of the ten top state estimate hypotheses $\hat{x}^{\ell}(0.05)$ for $\ell = 1, \ldots, 10$. (d) Null hypothesis LED predicted locations (green stars) with uncertainty ellipse (red circles) at time $t = 0.05$ and candidate LED detection locations (magenta crosses).

At each camera measurement time instant, only some of the LEDs will be within the FOV of the camera. An example of the predicted LED positions with 3-σ error covariance ellipses projected onto the image plane is shown in Fig. 6. The resolution of the image is 640 × 512. For the current rover state, only two LEDs are predicted to be in the image.

A time sequence of camera measurements is shown in Fig. 7. The left column is for LED 0 and the right column is for LED 1. Each LED candidate measurement has a value for both u and v, which correspond to the two rows. The green stars are the predicted LED pixel coordinates at each time step based on the rover state estimate and the known location of the LEDs. The magenta crosses are the detected candidate measurements [i.e., $Z(k)$ versus k]. These measurements are extracted by thresholding the pixel intensities that are within each LED's predicted region. Note that there are several time instances where more than one measurement candidate is within the predicted region. If an incorrect measurement candidate is selected, incorrect data extraction and possibly state divergence would result.

At the kth measurement time, the measurements of the potential LED projection locations that fall within the detection region are enumerated and stored in the measurement set $Z(k)$ defined in (11). The detection region is the minimum rectangle containing the predicted ellipse. Fig. 8(a) shows the detected and predicted measurements along with the prior uncertainty ellipse at time $t = 0$. There are two detected measurements (represented by two magenta +) in the image. They are so close together that they are not visually distinguishable. Although these two measurements are outside the two depicted 3-σ error covariance ellipses, they still fall within the detection region of LED 1, defined by the rectangle circumscribing the ellipse, so they will be considered as candidate measurements for LED 1. Since no measurements are in the detection region for LED 0, accounting for the null hypotheses, a total of three (one × three) data association hypotheses will be generated at $k = 0$.

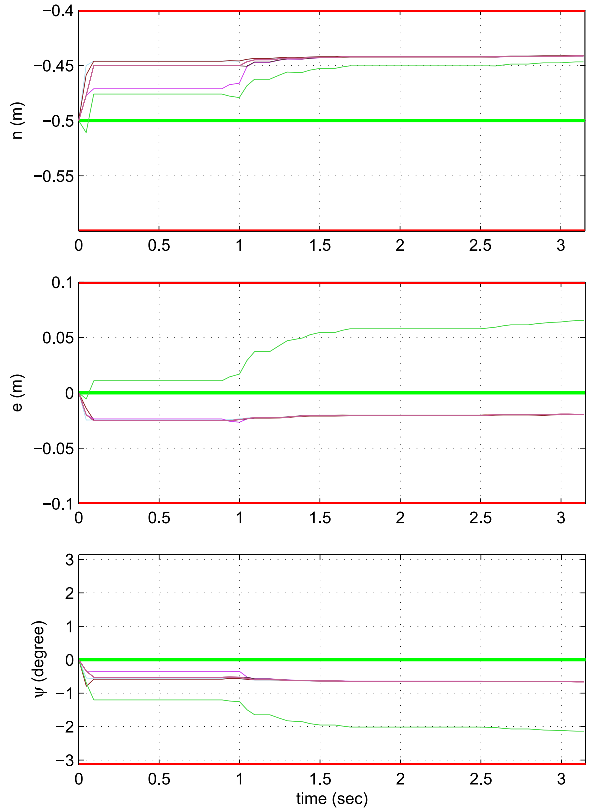


Fig. 9. Estimation results associated with each hypothesis sequence.

Using each data association hypothesis to update the state estimate at $k = 0$ yields three candidate state vector estimates: $\hat{x}^{\ell}(0)$ for $\ell = 1, 2, 3$. Each of these three state estimate hypotheses can repredict both the LED projection locations. The results are shown in Fig. 8(b). The blue star symbol represents the predicted LED projection location after updating the state by a hypothesis and the corresponding blue ellipse represents its uncertainty. Note that because each hypothesis updates the state estimate, each hypothesis will generate a new predicted LED position with an uncertainty ellipse for each LED. With three hypotheses and two LEDs, there are six blue stars and six ellipses in the upper right image. Not all of them are visually distinguishable. The null hypothesis left the state estimate and its uncertainty unchanged, so they are identical to the prior. The two measurement hypotheses have adjusted the state and error covariance, reducing the uncertainty, and moving the predicted measurements toward the actual measurements. Because the two measurement hypotheses were nearly the same, their predictions are also nearly the same, so that only four predictions are observed visually.

Fig. 8(d) shows the predicted, null hypothesis, LED projection locations at the second measurement time $t = 0.05$ s. There are two detected measurements for LED 0 (i.e., three hypotheses) and four detected measurements for LED 1 (i.e., five hypotheses). Therefore, at $t = 0.05$ s, we have 15 measurement hypotheses and three prior hypotheses, for a total of 45 hypotheses. At each time step, only the $q = 10$-most probable hypotheses are kept, as shown in Fig. 8(c), by their corresponding posterior predicted LED locations and error ellipses.

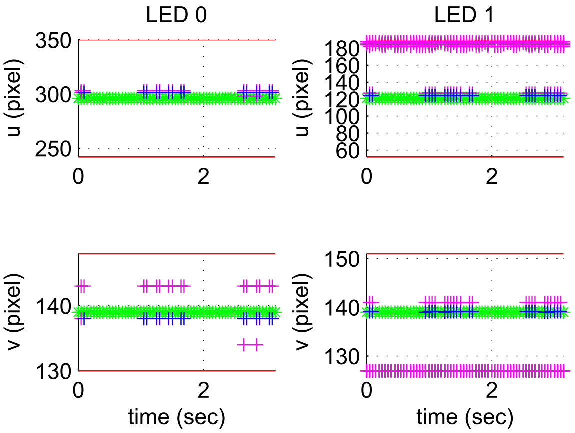


Fig. 10. Camera measurements with the most probable selection of the measurements at each time step.

The final navigation trajectory estimation results for the top $q = 10$ hypotheses over a 3.2-s window are shown in Fig. 9. The (wide) green line in the middle represents the assumed ground-truth position. The (wide) red lines depict three times the standard deviation around the assumed ground-truth location. The other different colored curves represent the estimated state corresponding to each hypothesis at each time step. Note these state estimates can be clearly distinguished in the images at the first few steps because each of them is calculated based on different data association hypotheses. After about 20 time steps (i.e., 1 s), these state estimates converge toward a few clusters of values. This is due to the pruning algorithm that keeps only the most probable state estimates. Note that the estimated positions are within a few centimeters of the hand measured (i.e., ground-truth) value of ${}^{n}\boldsymbol{p}$ stated at the beginning of this section.

Fig. 10 is a revised version of Fig. 7, where we have changed the coloring of the measurements that have been selected as the most probable hypothesis sequence from magenta to blue.

The $q = 10$ data sequences shown below are recovered for LED 0, whose true ID is {00000000}. When sending the ID, each LED will add a 4-b header {1010} in the front and another 4-b checksum in the back. For the correct data sequence, the checksum is {1010}. In this experiment, the camera works at a frame rate twice the LED data rate, which means that each bit is recovered from two consecutive sampled images.[3] The symbols $*$ in the sequences indicates that this bit could not be determined due to different recovered statii in the corresponding two consecutive steps. The numbers to the right of each sequence are the normalized probabilities. Given the set of hypotheses, each is searched first for a valid header. Once this is found, the corresponding 16-b data package checksum is checked. If verified, the ID is extracted and compared with the predicted LED ID. One such valid data packet for the second hypothesis is indicated with an underline. Due to the nonzero width of an LED, its projection onto the PDA may generate more than one closely clustered

[3] Each message contains 16 b. To sample each bit twice will require 32 images. At 20 images/s, one message arrives every 1.6 s. We save 3.2 s to ensure that a complete message is within the data window.

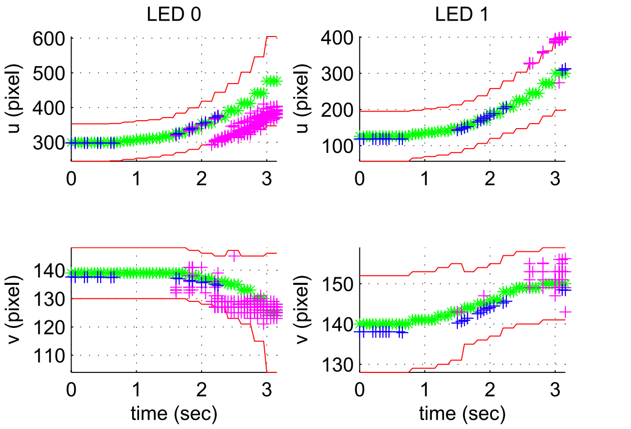


Fig. 11. Pixel coordinates for LED 0 and LED 1 in the image plane: predicted LED positions (green stars), 3-σ error regions (red curves), decoded most probable validated LED measurements (blue crosses), and other measurements (magenta crosses).

measurement, each communicating the same data sequence; therefore, several of the top candidates decode this valid data packet. However, if only the most probable hypothesis was extracted, the indeterminate first bit should cause concerns for decoding. Note that the second, third, and fifth hypotheses represent (repeating) valid extracted data packets. Among the correct data sequences, the second hypothesis with probability $p = 0.16$ is the most probable

$$\begin{aligned}
&\{\ast 0000000001010101000000000101010\}, && p = 0.24\\
&\{10000000001010\underline{1010000000001010}10\}, && p = 0.16\\
&\{10000000001010101000000000101010\}, && p = 0.13\\
&\{\ast 0000000001010101000000000101010\}, && p = 0.08\\
&\{00000000001010101000000000101010\}, && p = 0.07\\
&\{\ast 000000000 \ast 010101000000000101010\}, && p = 0.06\\
&\{\ast 0000000001010 \ast 01000000000101010\}, && p = 0.06\\
&\{\ast 000000000101010 \ast 000000000101010\}, && p = 0.06\\
&\{\ast 0000000001010101000000000010 \ast 010\}, && p = 0.06\\
&\{\ast 0000000001010101000000000 \ast 01010\}, && p = 0.06.
\end{aligned}$$

B. Moving

When the vehicle is moving, the outputs from the motion sensors (i.e., encoders) provide information to maintain an accurate state estimate through the navigation process described in Sections II–IV, which enables an accurate prediction of the LED projection locations in the image inspite of vehicle motion. This is shown in Fig. 11. Note that the computed LED projection locations (green stars) accurately predict the trends of the detected candidate PDA measurement locations (pink and blue crosses).

The algorithm for extracting the q-most likely sequences of LED measurements and then extracting the data is identical, whether or not the vehicle is moving. After applying the data recovery algorithm, the most probable data sequence passing the checksum test and having the correct ID is shown with blue cross symbols. Note that erroneous measurements (magenta crosses) near the predictions (green stars) have been correctly discarded.

The prior and recovered trajectories corresponding to the $q = 10$-most likely hypotheses are shown in Fig. 12. The wide green curve is the trajectory, starting from the prior and integrated by the encoder-based navigation, without any state updates from the LED measurements. Fig. 12 also shows the q-most likely trajectory estimates as narrower lines of various colors. For the majority of the time window, these alternative hypotheses generate very similar estimates; however, Fig. 12 does show time intervals when the alternate hypotheses contain different state estimates corresponding to their different measurement hypotheses.

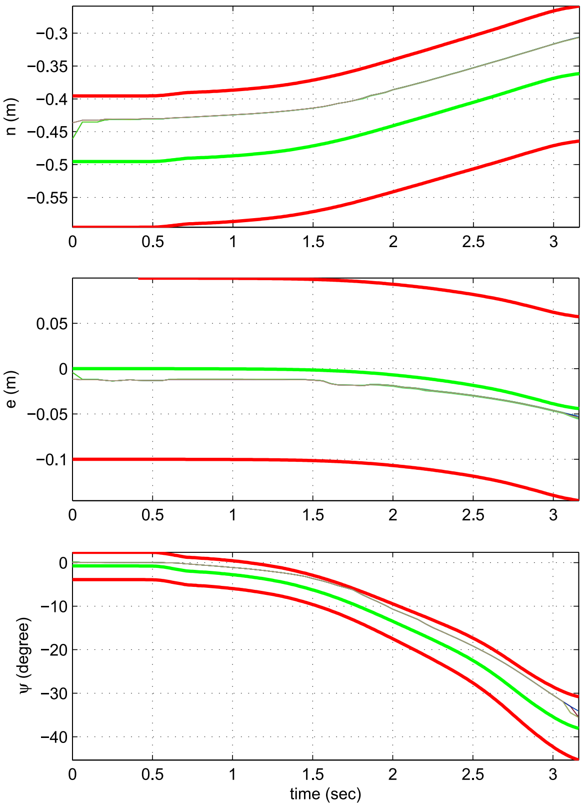


Fig. 12. Trajectory estimation results associated with each hypothesis: prior as advanced by encoder measurements (green lines), prior standard deviation (red lines) as advanced through time using (7), and posterior state estimates of each hypothesis (other colored lines).

Since the encoder measurements do not perfectly reproduce the rover linear speed and angular rate, error accumulates in the state estimates during the time updates and then is reduced in the appropriate directions during the measurement updates. This state error is characterized by the error covariance matrix. The $3 - \sigma$ error in the predicted measurement at each time is depicted by the red curve in Fig. 11.

The stationary and moving experiments were performed one after the other without moving the rover. Note that all the hypotheses in both experiments, except one of the hypotheses for the stationary case, compute a similar initial correction to the state from $(n, e) = (0.5, 0.0)$ toward $(-0.44, -0.02)$. This consistency of the corrections indicates that the prior position estimate was in error by a few centimeters, but is corrected consistently between the two independent experiments.

VI. Conclusion

This paper has developed and presented an algorithm to extract jointly the estimated vehicle trajectory and LED measurements on a PDA for the q-most likely hypotheses from a sequence of candidate LED position detections. The approach then extracts the data sequence for each of the q-most probable hypotheses. The algorithm works by recovering the LED trajectory across a (moving) array of PDs and is applicable to either a camera or a linear array. The choice of the parameters q and K affect the required real-time computation rate and algorithm performance. The algorithm requires the implementation of q Kalman filters. For $q = 1$, the approach is equivalent to that presented in [50], for which videos of real-time operations involving navigation, control, and image downloading are available at www.ee.ucr.edu/∼farrell/?page=content/PubSupp.html.

The algorithm herein has utility in VLC applications, particularly with the problem of accurately and reliably extracting data sequences communicated by LEDs to a camera or linear array in environments with various light sources. This approach is based on the multiple hypotheses tracking. The paper presented the background, algorithm, analysis, and discussion of the hypothesis probability model, and the measurement model, along with a detailed discussion of an experimental implementation.

This presentation of the algorithm has focused on processing of a single LED. The fact that it works equally well for multiple LEDs is demonstrated in the experiments. In the current approach, the hypotheses for each LED are considered as independent. When the uncertainty ellipses of multiple LEDs' overlap, this approach could allow the same detection to be simultaneously associated with multiple LEDs, which would be invalid. Future work could consider an approach that considers the detection associations of multiple LEDs jointly. When multiple LED data associations are jointly considered, each LED's data association hypothesis at a time step not only depends on its own former association hypotheses, but also on other LEDs' association hypotheses. This could greatly increase the complexity of the assignment of the measurements to the LEDs. This could be addressed by efficient assignment algorithms such as Auction [8] and JVC [23].

The proposed LED data recovery algorithm focuses on solving the problem of false LED detections. As we mentioned in Section III-A, the case of missed LED detections has not been considered and is an open topic for future research. When missed detections occur, but are not considered properly, inaccurate data will be recovered.

Finally, the main factor limiting the data extraction (i.e., bit) rate in the demonstrated experiments is the limitations of the PDAs (linear or camera) to which we currently have access. Commercially available products sample the arrays in a fixed sequential pattern. The camera that we used allows this pattern to be reprogrammed via a field-programmable gate array, but the reprogramming process is too slow to be effective in the real-time tracking of expected LED detection regions. The data rates could be greatly increased with random access pixel PDs. The authors would be glad to collaborate with any groups having such technology or access to linear PDAs.

Other interesting extensions include algorithms building on the simultaneous location and mapping literature to accommodate LEDs at unknown locations, as well as new methods to accommodate moving LEDs such as those on vehicles for computation of relative vehicle states.

Acknowledgment

The authors thank the reviewers of this paper whose constructive comments have enhanced the clarity of presentation.

References

[1] M. Z. Afgani, H. Haas, H. Elgala, and D. Knipp, "Visible light communication using OFDM," in *Proc. Int. Conf. Testbeds Res. Infrastruct. Develop. Netw. Communities*, 2006, pp. 129–134.

[2] M. Agrawal, K. Konolige, and M. R. Blas, "CenSurE: Center surround extremas for realtime feature detection and matching," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2008, pp. 102–115.

[3] S. Arai *et al.*, "Feasible study of road-to-vehicle communication system using LED array and high-speed camera," in *Proc. 15th World Congr. IT*, 2008, pp. 1–12.

[4] Y. Bar-Shalom, *Tracking and Data Association*. San Francisco, CA, USA: Academic, 1987.

[5] Y. Bar-Shalom, S. S. Blackman, and R. J. Fitzgerald, "The dimensionless score function for multiple hypothesis decision in tracking," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 4. Oct. 2005, pp. 3264–3269.

[6] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Syst.*, vol. 29, no. 6, pp. 82–100, Dec. 2009.

[7] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2006, pp. 404–417.

[8] D. P. Bertsekas, "The auction algorithm: A distributed relaxation method for the assignment problem," *Ann. Oper. Res.*, vol. 14, no. 1, pp. 105–123, Dec. 1988.

[9] J.-Y. Bouguet. (2010). *Camera Calibration Toolbox for MATLAB*. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/index.html

[10] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. New York, NY, USA: Wiley, 2009.

[11] I. J. Cox and S. L. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 2, pp. 138–150, Feb. 1996.

[12] W. Dong, J. A. Farrell, M. M. Polycarpou, V. Djapic, and M. Sharma, "Command filtered adaptive backstepping," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 3, pp. 566–580, May 2012.

[13] J. Elfring, R. Janssen, and R. van de Molengraft, "Data association and tracking: A literature survey," RoboEarth, Eindhoven Univ. Technol., Tech. Rep. FP7-ICT-248942, Apr. 2010.

[14] J. A. Farrell, *Aided Navigation: GPS With High Rate Sensors*. New York, NY, USA: McGraw-Hill, 2008.

[15] W. Förstner, "A feature based correspondence algorithm for image matching," *Int. Arch. Photogram. Remote Sens.*, vol. 26, no. 3, pp. 150–166, 1986.

[16] A. Frank, P. Smyth, and A. Ihler, "A graphical model representation of the track-oriented multiple hypothesis tracker," in *Proc. IEEE Statist. Signal Process. Workshop (SSP)*, Aug. 2012, pp. 768–771.

[17] A. R. Golding and N. Lesh, "Indoor navigation using a diverse set of cheap, wearable sensors," in *Proc. 3rd Int. Symp. Wearable Comput.*, Oct. 1999, pp. 29–36.

[18] J. Grubor, O. C. Jamett, J. W. Walewski, S. Randel, K.-D. Langer, "High-speed wireless indoor communication via visible light," in *ITG-Fachbericht-Breitbandversorgung in Deutschland-Vielfalt für Alle*. 2007, pp. 203–208.

[19] C. G. Harris and J. M. Pike, "3D positional integration from image sequences," *Image Vis. Comput.*, vol. 6, no. 2, pp. 87–90, May 1988.

[20] S. Haruyama, "Visible light communications: Recent activities in Japan," in *Smart Spaces: A Smart Lighting ERC Industry Academia Day at BU Photonics Center*, 2011.

[21] S. Haruyama, "Advances in visible light communication technologies," in *Proc. Eur. Conf. Exhibit. Opt. Commun.*, Sep. 2012, pp. 1–3.

[22] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proc. IEEE CVPR*, Jun. 1997, pp. 1106–1112.

[23] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, Dec. 1987.

[24] M. Kavehrad, "Broadband room service by light," *Sci. Amer.*, vol. 297, no. 1, pp. 82–87, 2007.

[25] K. Kiasaleh, "Performance analysis of free-space on-off-keying optical communication systems impaired by turbulence," *Proc. SPIE Free-Space Laser Commun. Technol. XIV*, vol. 4635, pp. 150–161, Apr. 2002, doi: 10.1117/12.464088.

[26] T. Komine and M. Nakagawa, "Integrated system of white LED visible-light communication and power-line communication," *IEEE Trans. Consum. Electron.*, vol. 49, no. 1, pp. 71–79, Feb. 2003.

[27] T. Komine and M. Nakagawa, "Fundamental analysis for visible-light communication system using LED lights," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 100–107, Feb. 2004.

[28] T. Kurien, "Issues in the design of practical multitarget tracking algorithms," in *Multitarget-Multisensor Tracking: Advanced Applications*, vol. 1. Norwood, MA, USA: Artech House, 1990, pp. 43–83.

[29] H. Le-Minh *et al.*, "Short-range visible light communications," *Wireless World Res. Forum*, vol. 21, Chennai, India, no. 15, pp. 1063–1065, 2007.

[30] J. Liu, W. Noonpakdee, H. Takano, and S. Shimamoto, "Foundational analysis of spatial optical wireless communication utilizing image sensor," in *Proc. IEEE Int. Conf. Imag. Syst. Techn.*, May 2011, pp. 205–209.

[31] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[32] V. Macellari, "CoSTEL: A computer peripheral remote sensing device for 3-dimensional monitoring of human motion," *Med. Biol. Eng. Comput.*, vol. 21, no. 3, pp. 311–318, May 1983.

[33] R. De Maesschalck, D. Jouan-Rimbaud, and D. L. Massart, "The Mahalanobis distance," *Chemometrics Intell. Lab. Syst.*, vol. 50, no. 1, pp. 1–18, Jan. 2000.

[34] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep. STAN-CS-80-813, 1980.

[35] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 3565–3572.

[36] K. G. Murty, "An algorithm for ranking all the assignments in order of increasing cost," *Oper. Res.*, vol. 16, pp. 682–687, May/Jun. 1968.

[37] T. Nagura, T. Yamazato, M. Katayama, T. Yendo, T. Fujii, and H. Okada, "Improved decoding methods of visible light communication system for ITS using LED array and high-speed camera," in *Proc. IEEE Veh. Technnol. Conf.*, May 2010, pp. 1–5.

[38] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proc. IEEE CVPR*, vol. 1. Jun./Jul. 2004, pp. I-652–I-659.

[39] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[40] R. Robert, "Intel labs camera communications (CamCom)," Intel Labs, Intel Corporation, Hillsboro, OR, USA, Tech. Rep., 2013.

[41] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2006, pp. 430–443.

[42] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE CVPR*, Jun. 1994, pp. 593–600.

[43] Y. Tanaka, T. Komine, S. Haruyama, and M. Nakagawa, "Indoor visible communication utilizing plural white LEDs as lighting," in *Proc. IEEE Int. Symp. Pers., Indoor Mobile Radio Commun.*, vol. 2. Sep./Oct. 2001, pp. F-81–F-85.

[44] Y. Tanaka, T. Komine, S. Haruyama, and M. Nakagawa, "Indoor visible light data transmission system utilizing white LED lights," *IEICE Trans. Commun.*, vol. E86-B, no. 8, pp. 2440–2454, 2003.

[45] V. Kulyukin, C. Gharpure, J. Nicholson, and S. Pavithran, "RFID in robot-assisted indoor navigation for the visually impaired," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 2. Sep./Oct. 2004, pp. 1979–1984.

[46] A. Vu, A. Ramanandan, A. Chen, J. A. Farrell, and M. Barth, "Real-time computer vision/DGPS-aided inertial navigation system for lane-level vehicle navigation," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 899–913, Jun. 2012.

[47] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 1. Sep. 1999, pp. 666–673.

[48] D. Zheng, "Joint visible light communication and navigation via LEDs," Ph.D. dissertation, Dept. Elect. Eng., Univ. California, Riverside, Riverside, CA, USA, Oct. 2014.

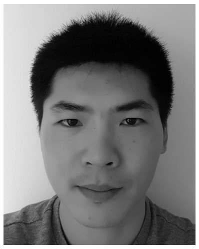
[49] D. Zheng, G. Chen, and J. A. Farrell, "Navigation using linear photo detector arrays," in *Proc. IEEE Int. Conf. Control Appl.*, Aug. 2013, pp. 533–538.

[50] D. Zheng, G. Chen, and J. A. Farrell, "An algorithm to recover an LED path," in *Proc. Eur. Control Conf.*, Jun. 2014, pp. 1631–1636.

[51] D. Zheng, G. Chen, and J. A. Farrell, "LED path recovery in a moving sensor," in *Proc. IFAC World Congr.*, 2014, pp. 8965–8970.

[52] D. Zheng, K. Cui, B. Bai, G. Chen, and J. A. Farrell, "Indoor localization based on LEDs," in *Proc. IEEE Int. Conf. Control Appl.*, Sep. 2011, pp. 573–578.

[53] D. Zheng, R. Vanitsthian, G. Chen, and J. A. Farrell, "LED-based initialization and navigation," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 6199–6205.



Dongfang Zheng received the B.S. degree in automation from the University of Science and Technology of China, Hefei, China, in 2009, and the Ph.D. degree in electrical engineering from the University of California at Riverside, Riverside, CA, USA, in 2014.

He was a Senior Software Engineer with Samsung Research America, Mountain View, CA, USA, where he was involved in the projects of indoor map generation and localization, and vision-based feature tracking. He is currently a Software Engineer with Google, Mountain View, where he is involved in the Project Tango. His current research interests include visual inertial navigation systems, concurrent odometry and mapping, semi-dense visual odometry, bundle adjustment and loop-closure detection in vision-based tracking systems, and online vision-based mapping algorithms.

Gang Chen received the Ph.D. degree in optical engineering from the Shanghai Institute of Optics and Fine Mechanics, Shanghai, China, in 2004.

He was an Assistant Professor and an Associate Professor with the Shanghai Institute of Optics and Fine Mechanics, CAS, from 1997 to 2006, where he was involved in free space ultraviolet and infrared communications, and optical fiber communication and sensing. Since 2006, he has been a Post-Doctoral Scholar with the Department of Electrical Engineering, University of California at Riverside (UCR), Riverside, CA, USA. In 2011, he was promoted to Associate Adjunct Professor with UCR. His current research interests include wireless optical communications and sensing system design.

Dr. Chen is a Member of SPIE.

Jay A. Farrell (F'16) received the B.S. degrees in physics and electrical engineering from Iowa State University, Ames, IA, USA, and the M.S. and Ph.D. degrees in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA.

He is currently a Professor and two-time Chair of the Department of Electrical and Computer Engineering with the University of California at Riverside, Riverside, CA, USA. He has authored over 200 technical publications and three books.

Prof. Farrell is a fellow of American Association for the Advancement of Science and a Distinguished Member of the IEEE Control Systems Society (CSS). At the Charles Stark Draper Laboratory, he received the Engineering Vice President's Best Technical Publication Award in 1990, and Recognition Awards for Outstanding Performance and Achievement in 1991 and 1993. He served as the Board of Governors of the IEEE CSS from 2003 to 2006 and from 2012 to 2014, the Vice President Finance and Vice President of Technical Activities, the CSS General Vice Chair of the IEEE CDC-ECC in 2011, the General Chair of the IEEE CDC in 2012, and the President in 2014. He was named a Global Navigation Satellite System Leader to Watch for 2009-2010 by *GPS World Magazine* in 2009 and a winner of the Connected Vehicle Technology Challenge by the U.S. Department of Transportation's Research and Innovative Technology Administration in 2011.