Optimal Node Selection for Target Localization in Wireless Camera Sensor Networks

Liang Liu, Xi Zhang, *Senior Member, IEEE*, and Huadong Ma, *Member, IEEE*

***Abstract*—This paper studies the node-selection problem for target localization in wireless camera sensor networks. The goal of node selection is to optimize the tradeoff between the energy consumption of wireless camera sensor networks and the quality of target localization. We propose a cooperative target localization algorithm, which is implemented by two phases: 1) target *detecting phase* and 2) target *locating phase*. For the target detecting phase, we develop a *probing environment and adaptive sleeping* (PEAS)-based density control algorithm to select the proper subset of deployed camera sensors for maintaining the desired density of nodes in the detecting mode. For the locating phase, we map the node-selection problem into an optimization problem and then propose an optimal node-selection algorithm to select a subset of camera sensors for estimating the location of a target while minimizing the energy cost. We conduct extensive experiments and simulations to validate and evaluate our proposed schemes.**

***Index Terms*—Density control, node selection, optimization problem, target localization, wireless camera sensor networks.**

I. Introduction

RECENT advances in the technologies of image sensors and embedded processors have enabled the deployment of large-scale wireless camera sensor networks [1]–[3] for various security and surveillance applications, as well as smart environment applications [4]–[6]. For most security and surveillance applications, the users are interested not only in the occurrence/existence of some watched events/targets but also in the locations of these events/targets. Therefore, localization capability is one of the most desirable characteristics of wireless camera sensor networks. In general, *localization* has two meanings: 1) self-localization of sensor nodes and 2) target localization. In this paper, we address the problem of target localization, and thus, throughout the rest of this paper, unless otherwise mentioned, *localization* refers to *target localization*.

The localization problem has received considerable attention in the area of wireless sensor networks [7]. Most existing localization algorithms for wireless sensor networks are based on the following sensing model—the sensing region is formed as a disk centered around the sensor where the parameters of a target/event linearly decay with the distance. However, the sensing model used in the wireless camera sensor networks is based on a sector sensing region and the *perspective projection* model. As a result, the comparison of those sensing models provides us with the opportunity to develop the novel localization approaches for wireless camera sensor networks. Thus, we need to design a vision-based localization algorithm that employs lightweight image processing and cooperation among camera sensors.

Unlike the vision-based localization algorithm in computer vision areas [8], the vision-based localization algorithm for wireless camera sensor networks imposes new challenges. The accuracy of localization can gradually be improved by selecting the most informative camera sensors until the required accuracy level of the target's location is achieved. This implies that the quality of localization improves with an increasing number of measurements from different camera sensors. Thus, from the perspective of localization, it is desirable to have many camera sensors involved in the process of localization. On the other hand, the limited energy is the major constraint of wireless camera sensor networks. Gaining measurements from many camera sensors and transmitting these measurements will reduce the lifetime of wireless camera sensor networks. Moreover, the bandwidth constraint also limits the number of obtained measurements. Therefore, to balance the tradeoff between the accuracy of localization and the cost of network energy, we need to properly select a small number of camera sensors that can provide the most informative measurements.

For the localization application, the operations of wireless camera sensor networks can be divided into two phases.

1) *Detecting phase:* Determine whether there exists any possible target in the field.
2) *Locating phase:* If any camera sensor detects the target, the camera sensors that can detect this target need to collaborate in accurately estimating the target location.

In addition, for a given time point, each camera sensor can operate in one of three different modes.

1) *Sleeping mode:* The camera sensor is almost shut down. The only activity of the camera sensor is to periodically detect the awaking messages.

Manuscript received November 22, 2008; revised March 24, 2009; accepted June 5, 2009. Date of publication September 4, 2009; date of current version September 17, 2010. This work was supported in part by the U.S. National Science Foundation CAREER Award under Grant ECS-0348694, by the National High Technology Research and Development Program of China under Grant 2009AA01Z305, and by the National Natural Science Foundation of China under Grant 60833009 and Grant 60925010. The review of this paper was coordinated by Dr. J. Misic.

L. Liu was with the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA. He is now with Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: liangliu82@gmail.com).

X. Zhang is with the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: xizhang@ece.tamu.edu).

H. Ma is with Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mhd@bupt.edu.cn).

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/TVT.2009.2031454

2) *Detecting mode:* The camera sensor captures the image frames with a *low* frequency and then determines whether there is any possible target.
3) *Locating mode:* The camera sensor captures the image frames with a *high* frequency and then sends these measurements to the fusion center node.[1] The camera sensor, which functions as the fusion center node, estimates the target location by using the measurements.

In the detecting phase, putting all the camera sensors in the detecting mode is too costly. At any given time, enabling a part of camera sensors in the detecting mode, which are called the *detecting camera sensors*, and the others in the sleeping mode, which are called the *sleeping camera sensors*, can efficiently prolong the lifetime of the wireless sensor networks. Therefore, it is necessary to choose a small set of detecting camera sensors from the deployed camera sensors to guarantee the quality of detection while minimizing the number of detecting camera sensors. In the locating phase, there are two problems that need to be solved: 1) how to find the camera sensors that can detect the target and 2) how to select the optimal set of camera sensors from the camera sensors that can detect the target to participate in the target localization process. We also call the camera sensors in the locating mode the *locating camera sensors*.

In this paper, we mainly focus on node selection, i.e., camera sensor selection, for localization in wireless camera sensor networks. The goal of node selection is to balance the tradeoff between the energy consumption of wireless camera sensor networks and the quality of localization. First, we propose a novel localization-oriented sensing model for camera sensors by taking the perspective projection and camera noisy models into account. Based on this sensing model, we develop a cooperative localization algorithm and then describe the corresponding procedure of wireless camera sensor networks. As to the detecting and locating phases, we design a two-step node-selection scheme. For the detecting phase, we analyze the relationships among detection-oriented coverage, called D-coverage for short, localization-oriented coverage, called L-coverage for short,[2] and the density of camera sensors. According to the expressions of D- and L-coverage probabilities, we propose a *probing environment and adaptive sleeping* [9] (PEAS)-based density control algorithm to select the requested subset of deployed camera sensors for maintaining the desired density of detecting nodes. For the locating phase, we first define cost and utility functions. The cost function is used to calculate the energy cost, including processing and communicating energy costs, of wireless camera sensor networks for localization. The utility function based on *continuous entropy* is used to quantify the contribution to localization accuracy made by the locating camera sensors. Using continuous entropy, we map the node-selection problem into an optimization problem and then develop an optimal selection algorithm to properly select a subset of camera sensors for cooperatively estimating the location of a target.

[1] In this paper, all nodes in the wireless camera sensor network are homogeneous. The camera sensor, which functions as the fusion center node, is elected by a certain rule.

[2] L-coverage is to strictly be defined in Section IV-A.

We summarize the main reasons for requiring node selection in wireless camera sensor networks in the list that follows.

1) With unlimited power supply, the more camera sensors work, the better the quality of localization. However, since the camera sensor has limited power, node selection is necessary for balancing the tradeoff between the accuracy of localization and the cost of network energy. Furthermore, node selection can homogenize the energy cost over all the camera sensors.
2) For the detecting phase, it is not necessary to put all the camera sensors in the detecting mode. Node selection can dynamically alternate camera sensors' modes to guarantee the quality of detection while maximizing the number of sleeping camera sensors.
3) For the locating phase, node selection can choose the most informative camera sensors to satisfy the accuracy requirement of localization while minimizing the energy cost.

The rest of this paper is organized as follows: Section II highlights the related works. Section III proposes our collaborative target localization algorithm and describes its corresponding procedure in wireless camera sensor networks. Section IV develops a PEAS-based node-selection scheme for the detecting phase. Section V defines cost and utility functions and then develops an optimal node-selection algorithm for the locating phase. Section VI conducts experiments to validate and evaluate our proposed scheme. This paper concludes with Section VII.

II. Related Works

The existing works about target localization in wireless sensor networks are mainly based on the disk sensing model and focus on the signal processing aspect. The authors of [10] proposed a framework for collaborative signal processing in distributed sensor networks and applied minimum square estimation to locate the target during the tracking process. The authors of [11] estimated the target location by incorporating the current measurement at a sensor with the past history at other sensors.

On the other hand, most previous research works of node selection aimed at saving energy and increasing scalability in wireless sensor networks. The authors of [12] developed a novel approach, called the information-driven sensor querying (IDSQ) algorithm, to select the "best" sensor measurement for updating the posterior belief state of an object. However, because the expected posterior distribution involves integrating over all possible measurements, this approach is difficult to be implemented. The authors of [13] proposed a mutual information-based scheme to address sensor node selection. They used the mutual information to quantify the expected posterior uncertainty. The feature of this scheme is the significant reduction of the computational complexity. The authors of [14] expanded the work in [13] and proposed a novel entropy-based sensor node-selection heuristic algorithm for target localization. This heuristic algorithm can select the suboptimal additional sensor subset without retrieving the measurements of all candidate sensors. The authors of [15] established an unscented Kalman filter framework to solve the problem of

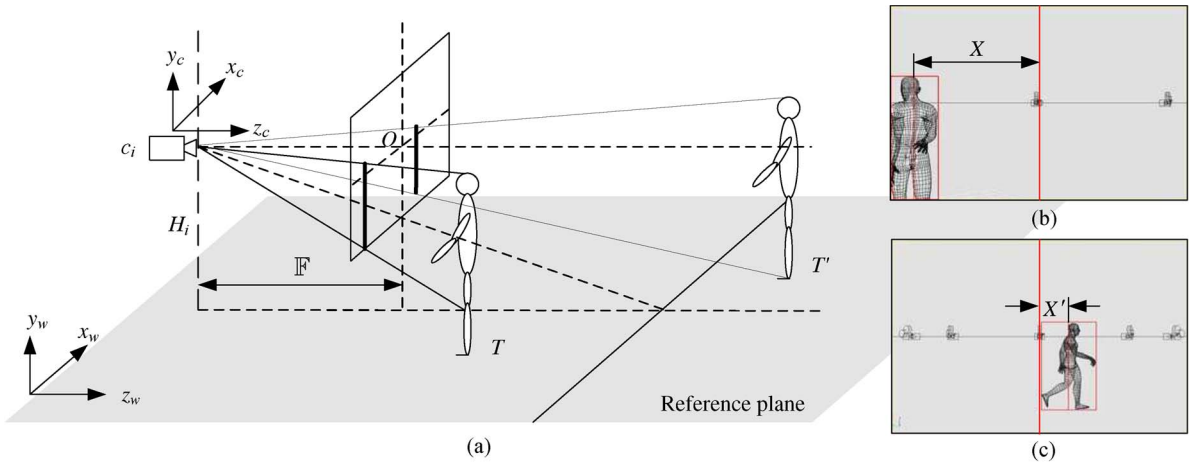


Fig. 1. (a) Perspective projection model of the camera sensor. (b) Picture of the target at T. The distance X from the vertical centerline of the target blob to the centerline of the picture is the observation measurement by this camera sensor when the target is at T. (c) Picture of the target at T'. X' is the corresponding observation measurement when the target is at T'.

TABLE I
PARAMETERS USED IN THE PROJECTION EQUATION

Parameter	Description
$(\mathbb{X}_i, \mathbb{Y}_i)$	The projection location of target in the image coordinates for camera sensor c_i
(x_t, z_t)	The location coordinates of target in the reference plane
(x_i, z_i)	The location coordinates of c_i in the reference plane
θ_i	Rotatable angle of c_i around x-axis
H_i	Height of c_i
$\mathbb{F}$	Focal length of camera sensor
s	Coefficient

optimal sensor selection. This scheme maximizes the information utility gained from a set of sensors subject to a constraint on the average energy consumption. In [16], the authors used a heuristic approach to select a subset of horizontal camera sensors for minimizing the visual hull of all objects in a scene. Their scheme adopts a heuristic method to compute the viewpoint scores and finds the optimal solution by using the combinatorial optimization techniques. The authors of [17] proposed a generic sensor model, where the measurements are interpreted as polygonal convex subsets of the plane. Their approach uses an approximation algorithm to minimize the error in estimating the target location. However, this work does not address the cost of using cameras.

From aforementioned related works, we can observe that the existing methods fail to satisfy the need of the tradeoff between the accuracy of vision-based localization and the cost for wireless camera sensor networks. To overcome this problem, we propose the optimal node selection scheme based on the collaborative target localization in camera sensor networks.

III. COLLABORATIVE TARGET LOCALIZATION IN CAMERA SENSOR NETWORKS

In this section, we propose a Bayesian-estimation-based localization scheme and then describe the collaborative target localization procedure in wireless camera sensor networks. To make the system model tractable, we need to make three reasonable assumptions.

A1. All camera sensors follow the same sensing model. We assume that the camera sensors are modeled by *perspective projection* and have the same shape as the field-of-view (FOV) region. Additionally, all noises are Gaussian noises with zero mean.

A2. The camera sensors can synchronously observe a moving target. The synchronization can be implemented by using the methods proposed in [18] and can guarantee adequate accuracy if the target moves with a limited speed.

A3. The message functions and transmissions introduce no information loss. In other words, quantization/modulation/encoding for measurements and the transmission channels are lossless.

A. Bayesian-Estimation-Based Localization Scheme

In the *computer vision* area, a point with the coordinates (x_t, y_t, z_t) can be mapped onto the image plane by using the *perspective projection* model. To ease presentation, we use the physical unit instead of the pixel unit in image coordinates. As shown in Fig. 1, the origin of image coordinates is the intersection of the optical axis and the image plane. For a given camera sensor c_i, the perspective projection equation [19] is shown as follows:

$$\begin{bmatrix} s\mathbb{X}_i \\ s\mathbb{Y}_i \\ s \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 & \sin\theta_i & -x_i \\ 0 & 1 & 0 & -H_i \\ \frac{\sin\theta_i}{\mathbb{F}} & 0 & -\frac{\cos\theta_i}{\mathbb{F}} & -\frac{z_i}{\mathbb{F}} \end{bmatrix} \begin{bmatrix} x_t \\ 0 \\ z_t \\ 1 \end{bmatrix}. \quad (1)$$

The descriptions of parameters in (1) are summarized in Table I and shown in Fig. 1. From Fig. 1, we can observe that there exist two types of projection for the target.

1) *Partial projection:* In the scenario illustrated in Fig. 1(a), when a target is at T (T is a point in the reference plane), the camera sensor can only capture the half-length image of the target [see Fig. 1(b)]. In other words, the corresponding $\mathbb{Y}_i$ is out of the image. According to (1), we have

$$\mathbb{X}_i = \frac{(\cos\theta_i x_t + \sin\theta_i z_t - x_i)\mathbb{F}}{\sin\theta_i x_t - \cos\theta_i z_t - z_i}. \tag{2}$$

2) *Complete projection:* As shown in Fig. 1(a), when a target is at T', the camera sensor can capture the full-length image of the target [see Fig. 1(c)]. The image coordinates of the target location can be calculated by

$$\begin{cases} \mathbb{X}_i = \frac{(\cos\theta_i x_t + \sin\theta_i z_t - x_i)\mathbb{F}}{\sin\theta_i x_t - \cos\theta_i z_t - z_i} \\ \mathbb{Y}_i = \frac{-H_i\mathbb{F}}{\sin\theta_i x_t - \cos\theta_i z_t - z_i} \end{cases}. \tag{3}$$

Generally speaking, it is difficult to obtain camera sensor c_i height H_i [see Fig. 1(a)]. Then, for simplicity, we use (2) as the perspective projection model instead of using (1).

Identifying the moving objects from a set of pictures or a video sequence is a fundamental and critical task in the target localization application of wireless camera sensor networks. When a camera sensor captures a frame, it can employ *background subtraction*[3] [20], [21] to remove the static background. As shown in Fig. 1(b) and (c), the area of an image frame where there is a significant difference between the observed and estimated images indicates the location of a moving object in this image plane. The area containing the change in the frame is further processed to find the horizontal shift, denoted by X, of the target's image from the center of the image plane. In our localization scheme, X is the measurement of the camera sensor, and only X is communicated to the central processor (sink node).

Generally speaking, the random measurement variable, denoted by X_i (the horizontal shift), for camera sensor c_i is not equal to the accurate $\mathbb{X}_i$ given by (2). This is because the perspective projection model in (2) is just an ideal model, and the measurement X_i can be corrupted by some additive noises in practice. Then, we have

$$X_i = \mathbb{X}_i + e_i \qquad \forall\, i \in \{1, \ldots, k\}$$

where e_i is the additive noise of X_i. The noise mainly comes from two aspects: 1) the sensing model of camera sensors and 2) the processing of background subtraction. Similar to [22], we also assume that the measurement error variance, which is by denoted σ_i^2, for c_i is of the following form:

$$\sigma_i^2 = \zeta d_i^2 + \sigma_p^2 + \sigma_s^2 \tag{4}$$

where d_i is the distance from c_i to the target. Making camera noise variance dependent on distance can efficiently model the weak perspective projection while allowing the use of (2). Our noise model also accounts for errors in the calibration of camera sensors. Errors in the location of c_i are taken into account in σ_p^2, and errors in the orientation are reflected in ζ. Moreover, the accuracy of the background subtraction method and the postures/motions of targets also cause errors, and these errors are contained in σ_s^2.

Therefore, we adopt the Gaussian error model to represent the relationship between X_i and the target location, denoted by $T(x_t, z_t)$, i.e., $e_i \sim N(0, \sigma_i)$. For an arbitrary value, denoted by $\mathcal{X}_i$, for random measurement variable X_i, the probability density function (pdf) for X_i is

$$f(\mathcal{X}_i|T) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(\mathcal{X}_i - \mathbb{X}_i)^2}{2\sigma_i^2}\right). \tag{5}$$

Let S be a deployment filed in the reference plane and $T \in S$ be the location of a target. Assume that the *a priori* probability distribution of T obeys the uniform distribution in S. Thus, for an arbitrary point $t(x, z)$ in the reference plane, the pdf of T is

$$f(t) = \begin{cases} \frac{1}{\|S\|}, & t \in S \\ 0, & t \notin S \end{cases} \tag{6}$$

where $\|S\|$ denotes the area of S.

If T can simultaneously be detected by k camera sensors, then k measurements are available. Let $\mathbf{X} = (\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_k)$ be an arbitrary point in the k-dimensional real-number space of $(X_1, X_2, \ldots, X_k)$, then

$$f(\mathbf{X}|t) = \prod_{i=1}^{k} f(\mathcal{X}_i|t) = \prod_{i=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(\mathcal{X}_i - \mathbb{X}_i)^2}{2\sigma_i^2}}. \tag{7}$$

According to the Bayesian formula and (6), we can get

$$f(t|\mathbf{X}) = \frac{f(\mathbf{X}|t)f(t)}{\int\int_S f(\mathbf{X}|t)f(t)dx\,dz} = \frac{f(\mathbf{X}|t)}{\int\int_S f(\mathbf{X}|t)dx\,dz}. \tag{8}$$

Let $\widehat{T}_k(\widehat{x}, \widehat{z})$ and $\widetilde{T}_k \triangleq |\widehat{T}_k - T|$ denote the estimate and the estimation error for a given $(X_1, X_2, \ldots, X_k)$, respectively. The estimation error $\widetilde{T}_k$ is calculated by

$$\widetilde{T}_k = |\widehat{T}_k - T| = \sqrt{(\widehat{x} - x_t)^2 + (\widehat{z} - z_t)^2} \tag{9}$$

which is the Euclidean distance between $\widehat{T}_k(\widehat{x}, \widehat{z})$ and $T(x_t, z_t)$. The mean square error (MSE) is a commonly used measure of estimator quality. A well-known Bayesian estimator can be applied to estimate $\widehat{T}_k$ while achieving the minimum MSE. Then, the minimum MSE estimate is

$$\widehat{T}_k(\widehat{x}, \widehat{z}) = \left(\int\int_S x f(t|\mathbf{X})dx\,dz, \int\int_S z f(t|\mathbf{X})dx\,dz\right). \tag{10}$$

When the distance between a target and c_i is too large, the background subtraction method cannot segment out the target. This implies that the camera sensor c_i cannot detect the target. Let r be the maximal detecting distance. Because $r \gg \mathbb{F}$, we employ a sector model to describe the sensing region of a camera sensor. Here, we use D_i to denote the sensing region of c_i. If a point belongs to D_i, then the point can be detected by c_i. As shown in Fig. 2, the sector model can be denoted by a 4-tuple $(L_i, r, \overrightarrow{V}_i, \alpha)$. $L_i(x_i, z_i)$ is the location of c_i. $\overrightarrow{V}_i$ is

[3]Background subtraction is a commonly used technique for segmenting out objects of interest in a scene for applications such as video surveillance.

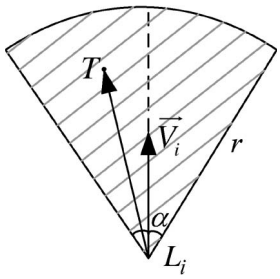


Fig. 2. Sector sensing model.

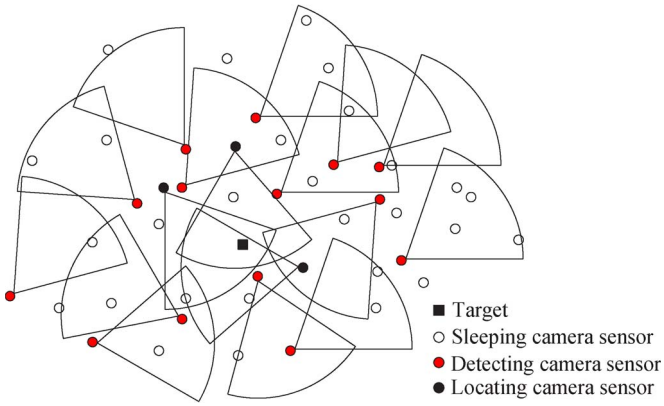


Fig. 3. Localization procedure of wireless camera sensor networks. There exist one target and three kinds of camera sensors, i.e., sleeping camera sensors, detecting camera sensors, and locating camera sensors.

the unit vector, which evenly splits the sensing sector into two halves, determining the sensing direction,[4] and α is the offset angle in the FOV on both sides of $\overrightarrow{V}_i$.

B. Localization Procedure of Wireless Camera Sensor Network

As shown in Fig. 3, N geographically distributed camera sensors, i.e., $C = \{c_1, c_2, \ldots, c_N\}$, are deployed in a surveillance region S. At the beginning, we need to decide the value of N, i.e., how many camera sensors should we deploy to satisfy the requirement of localization? L-coverage [23] is an important measure for the quality of localization. According to the cooperative localization scheme, a point that is detected by camera sensors does not imply that this point can be located. Then, to guarantee the quality of localization, we need to deploy much more camera sensors than the requirement of target detection.

In most surveillance applications, the targets infrequently appear with long intervals. If there is no target in S, then the camera sensor network is in the detecting phase, i.e., the task of camera sensors is detecting the target. Because the density of deployed camera sensors is much higher than the requirement of target detection, it is unnecessary to let all of the N camera sensors be in the detecting mode. Then, we should periodically select a set of detecting camera sensors to guarantee that most points in S can be detected by camera sensors and let the other camera sensors be in the sleeping mode to save energy.

[4] θ_i is the angle of $\overrightarrow{V}_i$.

If the target is detected by a camera sensor c_i, then the locating phase begins. The camera sensor c_i becomes the fusion center node and broadcasts its location (x_i, z_i), orientation angle θ_i, and measurement X_i to all the camera sensors in its communication range.[5] After receiving (x_i, z_i), θ_i, and X_i, each camera sensor calculates the probability of detecting the target. If the probability is below a predefined threshold, i.e., the camera sensor cannot detect the target with a high probability, then this camera sensor remains in the sleeping mode. Otherwise, the camera sensor becomes a candidate for the localization process. When a sleeping camera sensor becomes a candidate, this camera sensor is awaked to be in the detecting mode.

All the candidate camera sensors send their measurements to c_i. According to these measurements, c_i selects a set of camera sensors from the candidates to participate in the localization process. These selected candidate camera sensors periodically send the corresponding X's to c_i, and the others switch to the sleeping mode.

IV. Node Selection in the Detecting Phase

In the detecting phase, the goal of node selection is deciding the set of detecting camera sensors. There are two problems we need to address.

1) How to derive the density of deployed camera sensors and the density of detecting camera sensors for the desired L- and D-coverage probabilities, respectively.
2) How to design a density control scheme to determine the modes of camera sensors (sleeping or detecting) for maintaining the desired D-coverage probability.

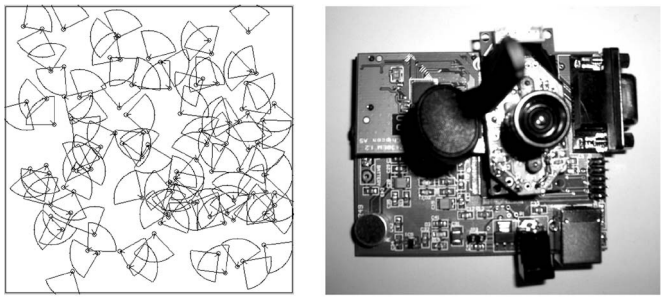
A. L-Coverage and D-Coverage Probabilities

In this paper, we consider the random deployment where camera sensors are randomly scattered within a vast 2-D geographical region, and their locations are uniformly and independently distributed in the region [24]. Such random deployment can be the result of certain deployment strategies. For example, sensors may be airdropped or launched via artillery in battlefields or unfriendly environments. Under this deployment strategy, the locations of camera sensors can be modeled by a 2-D stationary Poisson point process with intensity λ. This indicates that the number $N(S')$ of camera sensors in any subregion S' follows a Poisson distribution with a parameter $\lambda\|S'\|$, where $\|S'\|$ is the area of S'. Let k be a positive integer, the probability that $N(S')$ is equal to k is then given by

$$\Pr\{N(S') = k\} = \frac{(\lambda\|S'\|)^k}{k!} e^{-\lambda\|S'\|}. \tag{11}$$

Moreover, we assume that the orientation of each camera sensor is a random variable with the uniform distribution on $[0, 2\pi]$, i.e., $\theta \sim U(0, 2\pi)$. Fig. 4(a) and (b) illustrates a randomly scattered wireless camera sensor network with 100 nodes and a prototype camera sensor developed by our group, respectively.

[5] In general, the communication radius of all sensor nodes is assumed to be two times larger than the sensing radius. This implies that camera sensors that can detect the target must be in the communication range of c_i.



(a) (b)

Fig. 4. (a) One hundred camera sensors are deployed according to a 2-D Poisson process. (b) Prototype camera sensor.

In the literature, if a point is in a sensor's sensing range, then this point is covered by this sensor. This implies that the coverage in most existing works is D-coverage. If a point can be detected by at least K sensors, this point is K-covered. In the following lemma, we will derive the expression of the K-coverage probability for wireless camera sensor networks.

Lemma 1: Let T be an arbitrary point in S and N_T be the number of camera sensors that can detect T. If camera sensors are modeled by a 2-D stationary Poisson point process with intensity λ, then the probability that T is synchronously detected by k camera sensors is

$$\Pr\{N_T = k\} = \frac{(\lambda\alpha r^2)^k}{k!} e^{-\lambda\alpha r^2} \tag{12}$$

and the K-coverage probability, denoted by P_K, is

$$P_K = 1 - \sum_{i=0}^{K-1} \frac{(\lambda\alpha r^2)^i}{i!} e^{-\lambda\alpha r^2}. \tag{13}$$

Proof: The detailed proof is provided in Appendix A. ■

According to Lemma 1, the detecting coverage probability, denoted by P_d, is

$$P_d = P_1 = 1 - e^{-\lambda_d \alpha r^2} \tag{14}$$

where λ_d is the density of detecting camera sensors. Then, we also have

$$\lambda_d = \frac{\log(1 - P_d)}{-\alpha r^2}. \tag{15}$$

From the localization scheme in Section III, we can use the expected value of estimation error $\widetilde{T}_k$, denoted by δ_k, to measure how well the point T is located by k camera sensors, i.e.,

$$\delta_k \triangleq \mathrm{E}[\widetilde{T}_k] = \int_{\Re} \widetilde{T}_k f(\mathbf{X}|t) d\mathbf{X} \tag{16}$$

where $\widetilde{T}_k$ is given by (9), and $\Re$ is the real-number space of $(X_1, X_2, \ldots, X_k)$. The smaller the δ_k, the more reliable the estimated $\widehat{T}_k$. Assume that the accuracy of localization satisfies the requirement if δ_k is smaller than a predefined threshold ε. Therefore, a point is said to be L-covered if there exist k camera sensors that can estimate the location of this point, and the corresponding mean of estimation error $\mathrm{E}[\widetilde{T}_k] = \delta_k$ satisfies $\delta_k < \varepsilon$, where $0 < k \leq N$.

Let a be the ratio of ε to r, i.e., $a \triangleq \varepsilon / r$. We then define $\varphi(a) \triangleq \Pr\{\delta_2 < ar\}$, and furthermore, $a_t \triangleq \inf\{a \mid \varphi(a) \geq 0.8\}$. Then, we can derive an approximative expression of the L-coverage probability, denoted by P_l, as follows:

$$P_l \approx 1 - e^{-\lambda_l \alpha R^2} - \lambda_l \alpha R^2 e^{-\lambda_l \alpha R^2} \tag{17}$$

where λ_l is the density of deployed camera sensors, and

$$R = \begin{cases} r, & \text{if } \varepsilon > a_t r \\ \frac{\varepsilon}{a_t}, & \text{otherwise.} \end{cases} \tag{18}$$

The derivation of (17) is given in Appendix B.

From (17), we can obtain the corresponding density of camera sensors for a given L-coverage probability. Let ε_l be the desired value of P_l. Then, the density of deployed camera sensors is

$$\lambda_l = \frac{-1 - W\left(\frac{\varepsilon_l - 1}{e}\right)}{\alpha R^2} \tag{19}$$

where $W(\cdot)$ is the Lambert W-function.

B. Density Control for Detecting Camera Sensors

We use PEAS [9], which is a probing mechanism-based density control algorithm, to maintain a subset of camera sensors that is in the detecting mode while ensuring the desired D-coverage probability. The main procedure of this algorithm is as follows: After sensor deployment, all camera sensors are in the sleeping mode. Each sleeping camera sensor wakes up for an exponentially distributed period of time specified by the wake rate ϕ_t. When a sleeping camera sensor wakes up, it broadcasts a probing message, called PRB, within a certain probing range, denoted by r_c. If there exists a camera sensor that is in the detecting mode within the range r_c, then this detecting camera sensor broadcasts a reply message, called RPY, over the wireless channel. For the wake-up camera sensor, the received RPY message implies that there already exits a detecting camera sensor. Thus, if a wake-up camera sensor does not hear the RPY message within a given time interval, then this camera sensor assumes that there is no detecting camera sensor within the probing range r_c and then switches to the detecting mode. Otherwise, this camera sensor goes back to the sleeping mode.

In the aforementioned density control scheme, the probing range r_c and the wake rate ϕ_t are two important parameters for maintaining the density of detecting camera sensors at a desired value. We can use the method proposed in [9] to decide ϕ_t. However, this method cannot be applied to deriving r_c for wireless camera sensor networks, because the sector sensing model and the disk sensing model are different. Therefore, we derive the expression of r_c, which is given in the following lemma.

Lemma 2: Let ε_d be the desired D-coverage probability. If the deployment of wireless camera sensors follows the

2-D stationary Poisson point process with intensity λ_l, then the probing range r_c is determined by

$$r_c = \sqrt{\frac{-\alpha r^2}{\pi \log(1-\varepsilon_d)}}. \tag{20}$$

Proof: Let N' be the number of camera sensors deployed in a disk region with radius r_c. From the density control scheme, if a camera sensor is in the detecting mode, then there is no other detecting camera sensor in the disk centered around this camera sensor with r_c. This implies that

$$\frac{\lambda_d}{\lambda_l} = \mathrm{E}\left[\frac{1}{N'}\right]. \tag{21}$$

Because the camera sensors are modeled by a 2-D stationary Poisson point process with intensity λ_l, from (11), we have

$$\mathrm{E}[N'] = \sum_{k=0}^{\infty} k \Pr\{N' = k\} = \lambda_l \pi r_c^2.$$

Then, according to (21), the expression of r_c is

$$r_c = \sqrt{\frac{1}{\pi \lambda_d}}. \tag{22}$$

From (15), when $P_d = \varepsilon_d$, the corresponding λ_d is $[\log(1-\varepsilon_d)]/(-\alpha r^2)$. Then, substituting $\lambda_d = [\log(1-\varepsilon_d)]/(-\alpha r^2)$ into (22), we get (20). ■

V. Node Selection in the Locating Phase

In the locating phase, we first need to decide the set, denoted by C_c, of candidate camera sensors that can detect the target with a high probability. Ideally, we can obtain the maximum information gain when the fusion center node merges the measurements from all camera sensors in C_c, but this would be too costly. Our goal is to select the optimal set of camera sensors from C_c to obtain a precise estimate of the target location while minimizing the energy cost. In general, there exist two different criteria to define the optimal selection problem:

1) *maximum utility:* maximizes the accuracy of localization under the specified cost;
2) *minimum cost:* minimizes the cost to attain specified accuracy of localization.

Due to the constrained resource of wireless camera sensor networks, energy saving is one of the most important problems to be considered. Motivated by this, we study the selection of locating camera sensors by using the second criterion, i.e., minimum cost. Let 2^{C_c} denote the set of all subsets of C_c. To model this tradeoff between utility and cost, we need to define the following functions:

1) a utility function $\mathcal{U}: 2^{C_c} \to \mathbb{R}^+$, which quantifies the localization utility of measurements obtained by each $C_c' \subseteq C_c$;
2) a cost function $\mathcal{C}: 2^{C_c} \to \mathbb{R}^+$, which quantifies the energy cost of taking measurements from each $C_c' \subseteq C_c$.

Then, we can formulate the optimal selection problem as follows.

Locating node selection (LNS): Choose a subset $C_c' \subseteq C_c$, which minimizes $\mathcal{C}(C_c')$ subject to $\mathcal{U}(C_c') \geq \vartheta$, where ϑ is the predefined threshold for localization accuracy. Thus, LNS can be also expressed as

$$\mathrm{LNS}(C_c, \vartheta) = \arg \min_{C_c' \subseteq C_c,\, \mathcal{U}(C_c') \geq \vartheta} \mathcal{C}(C_c').$$

A. Candidate Camera Sensors

Assume that camera sensor c_0 is first to detect the target and that the corresponding measurement is X_0. Then, c_0 broadcasts its location (x_0, z_0), orientation angle θ_0, and X_0 in its communication range. Let c_i be an arbitrary camera sensor within the communication range of c_0. After receiving (x_0, z_0), θ_0, and X_0, sensor c_i can get $f(t|X_0)$ according to the Bayesian formula as follows:

$$f(t|X_0) = \frac{f(X_0|t)}{\int\int_S f(X_0|t)dx\,dz}. \tag{23}$$

Then, the probability, denoted by p_i, that c_i can detect the target is

$$p_i = \int\int_{D_i} f(t|X_0)dx\,dz \tag{24}$$

where D_i is the sensing region of c_i. If p_i exceeds a predefined threshold ξ, then c_i becomes a candidate camera sensor and sends its location, orientation, and measurement to c_0. This implies that the set of candidate camera sensors

$$C_c = \{c_i | p_i > \xi,\ |c_0 c_i| < 2r,\ 0 \leq i < N\} \tag{25}$$

where $|c_0 c_i|$ denotes the Euclidean distance between c_0 and c_i.

However, $p_i > \xi$ only implies that c_i can detect the target *with a high probability*. Therefore, it is possible that a few candidate camera sensors in C_c cannot detect the target, and a few camera sensors, which can detect the target, are not in C_c.

B. Utility Function

Let $\mathbf{X}$ be a measure vector of $C_c' \subseteq C_c$. The utility of C_c' can be defined as the uncertainty of the target location reduced by $\mathbf{X}$. We use *continuous entropy* and *mutual information* to quantify the uncertainty reduction for target localization. According to the expression of continuous entropy and (5), the continuous entropy of $f(t|\mathbf{X})$ is

$$h\left[f(t|\mathbf{X})\right] = -\int\int_S f(t|\mathbf{X}) \log f(t|\mathbf{X}) dx\,dz \tag{26}$$

where $h[\cdot]$ is the continuous entropy function.

Because the *a priori* probability distribution of the target location $f(t)$ is known, we can get the initial estimate of the

target location. Then, according to (4), we have

$$h\left[f(t)\right] = -\int\int_{S} f(t)\log f(t)dx\,dz = -\log\frac{1}{\|S\|}. \quad (27)$$

The mutual information, which is determined by

$$I(t;\mathbf{X}) = h\left[f(t)\right] - h\left[f(t|\mathbf{X})\right] \quad (28)$$

is the utility of C_c' for localization. Thus, we can define the utility function as follows:

$$\begin{aligned}\mathcal{U}(C_c') &= I(t;\mathbf{X}) \\ &= \log\frac{1}{\|S\|} - \int\int_{S} f(t|\mathbf{X})\log f(t|\mathbf{X})dx\,dz. \end{aligned} \quad (29)$$

C. Cost Function

For a locating camera sensor, the energy cost for location operations can be partitioned into two parts: 1) the energy cost for capturing and processing images, denoted by e_p; and 2) the energy cost for transmitting the measurement X, denoted by e_t. In this paper, we assume that all the camera sensors have the same energy cost for image capturing and processing, and measurement transmitting. For a wireless sensor network, the failure of several sensor nodes can affect the whole network topology. Thus, energy saving requires not only minimizing the total cost of the sensor network but also homogenizing the cost of the sensor nodes.

Let e_i be the remaining energy of sensor c_i. We define the cost function of c_i as the ratio between the total energy consumption of localization and the remaining energy, i.e.,

$$\mathcal{C}(c_i) = \begin{cases} \frac{e_p+e_t}{e_i}, & \text{if } c_i \text{ is not the fusion center node} \\ \frac{e_p}{e_i}, & \text{if } c_i \text{ is the fusion center node.} \end{cases} \quad (30)$$

For a set of camera sensors C_c', the cost value of this set of sensors is the maximum cost value in this set, i.e., the cost function is

$$\mathcal{C}\left(C_c'\right) = \max_{c_m \in C_c'} \mathcal{C}(c_m). \quad (31)$$

D. LNS Algorithm

If there exist more than one candidate camera sensors, then we sort these candidate camera sensors by their cost values and generate an ascending queue Q_c. Every element in Q_c is a subset of C_c, which is defined by (25). Let i_d be an index pointing at the elements in Q_c, and let $Q_c[i_d]$ be the i_dth element of Q_c. Set the initial value of i_d to be 0, i.e., i_d points at the head of Q_c. The head of Q_c only consists of the camera sensor with the minimum cost value. Equation (2) implies that it is impossible to get the target location by using one measurement, i.e., $\mathcal{U}(Q_c[0])$ cannot satisfy the requirement. Then, i_d points at the next element of Q_c. Because $Q_c[1]$ is also the set consisting of only one camera sensor, $\mathcal{U}(Q_c[1])$ cannot satisfy the requirement. From (31), $\mathcal{C}(Q_c[0]\bigcup Q_c[1]) = \mathcal{C}(Q_c[1]) \le \mathcal{C}(Q_c[2])$. Then, we insert $Q_c[0]\bigcup Q_c[1]$ into Q_c after $Q_c[1]$, i.e., $Q_c[2] = Q_c[0]\bigcup Q_c[1]$, and move the index i_d to the next element, i.e., $Q_c[2]$. If $\mathcal{U}(Q_c[2])$ is larger than the predefined threshold, denoted by ϑ, then $Q_c[2]$ is the optimal set we wanted. Otherwise, i_d moves to the next element of Q_c.

For the element $Q_c[i_d]$, if $Q_c[i_d]$ is the set that has only one camera sensor, then insert $Q_c[i_d]\bigcup Q_c[0]$, $Q_c[i_d]\bigcup Q_c[1], \ldots, Q_c[i_d]\bigcup Q_c[i_d-1]$ into Q_c after $Q_c[i_d]$ and move i_d to the next element. On the other hand, if $Q_c[i_d]$ is the set that has at least two camera sensors and $\mathcal{U}(Q_c[i_d])$ is larger than ϑ, then $Q_c[i_d]$ is the optimal set we look for. Otherwise, i_d moves to the next element of Q_c. The pseudocode of the LNS algorithm is summarized in Fig. 5.

```
01. Initialize Q_c array to null, i_d=0
02. if (|C_c| > 1) // There exit more than one candidate camera sensors.
03.   Ascending sort C_c by cost value and store into Q_c
04.   i_d=1  // Move the index to the second element of Q_c.
05.   while (Q_c[i_d] is not null)
06.     if (|Q_c[i_d]| > 1)
07.       if (U(Q_c[i_d]) > ϑ)
08.         return Q_c[i_d] // Q_c[i_d]: the optimal set of camera sensors.
09.     else  // Q_c[i_d] is the set consisting of one camera sensor.
10.       for i = 0 to i_d − 1
11.         Insert Q_c[i] ⋃ Q_c[i_d] into Q_c
12.     i_d = i_d + 1
13.   end while
```

Fig. 5. Selection algorithm for the locating camera sensors.

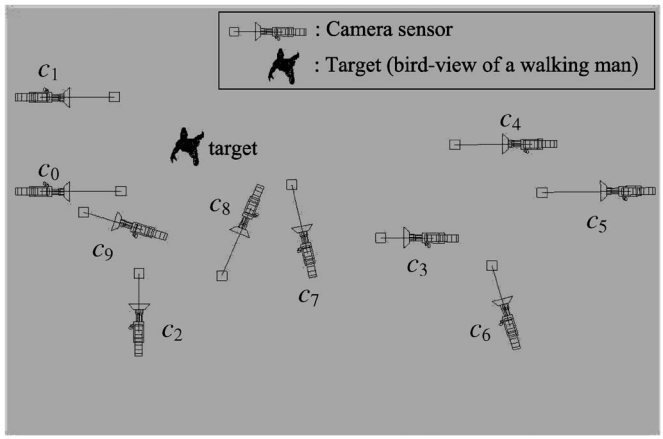


Fig. 6. Scene of a walking-man localization. We deploy ten camera sensors, i.e., c_0–c_9, in a rectangular surveillance field.

VI. Case Study and Simulations

To verify our proposed schemes and the derived relevant analytical analyses, we first utilize a case to illustrate the procedure of LNS in the locating phase and then show the effect of our proposed scheme on energy saving by extensive simulations.

A. Case Study of LNS

As shown in Fig. 6, we deploy ten camera sensors in a rectangular region. The values of related parameters[6] are as

[6]The values of parameters are based on a commonly used digital camera (Sony DSC-717F) and the related calibration process.

TABLE II
MEASUREMENTS X_i AND TARGET DETECTION PROBABILITIES p_i OF THE TEN CAMERA SENSORS. BECAUSE WE SET THE THRESHOLD OF THE DETECTING PROBABILITY TO BE 0.5, SOME X_i'S ARE NOT AVAILABLE

Camera sensor	(x_i, z_i)	Orientation angle	p_i	Measurement (X_i)
c_0	(0,0)	0°	1	-3.4375
c_1	(0,1300)	0°	0.8725	2.8531
c_2	(1300,-2600)	90°	0.5558	2.4063
c_3	(5200,-650)	180°	0.2646	-
c_4	(6500,650)	180°	0.5008	-0.1925
c_5	(7800,0)	180°	0.0012	-
c_6	(5850,-1950)	110°	0.0825	-
c_7	(3900,-650)	105°	0.6023	-
c_8	(3250,0)	240°	0	-
c_9	(1300,-650)	165°	0.0138	-

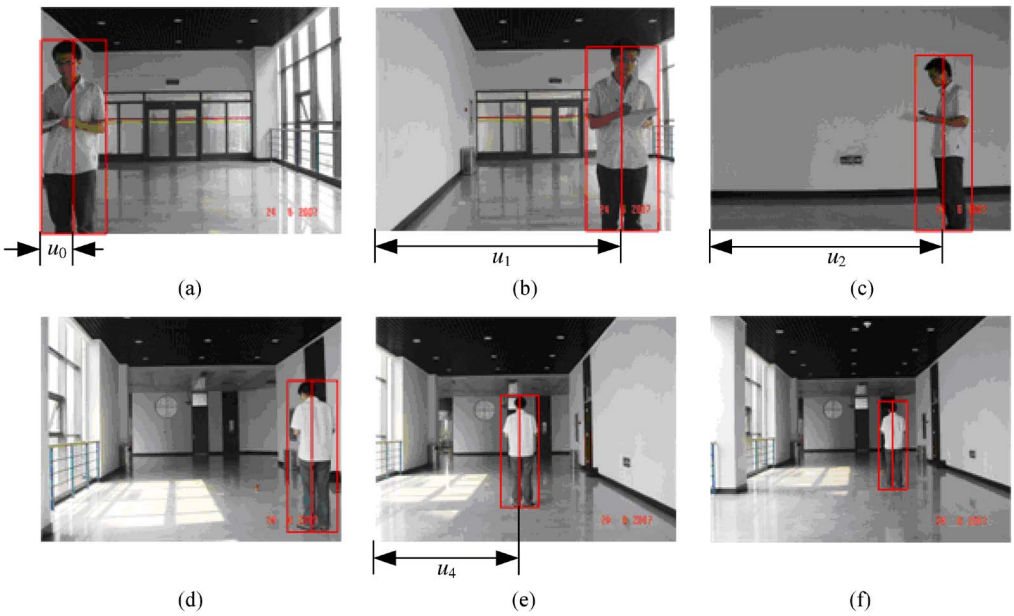


Fig. 7. Image frames captured by camera sensors c_0–c_5, respectively. u_i, $i = 0, 1, 2$, and 4 denote the horizontal pixel coordinates of the target. For the camera sensor c_3, u_3 is not available, because the target detection probability $p_3 = 0.2646 < 0.5$, and thus, c_3 is not the candidate node. Because the distance between c_5 and the target exceeds the sensing range r so that $p_5 = 0.0012 < 0.5$, c_5 is also not the candidate node, and thus, u_5 is not available.

follows: $\mathbb{F} = 9.45$ mm, $\alpha = 25°$, $r = 4000$ mm, $\zeta = 5 \times 10^{-8}$, $\sigma_p = 0.1$, $\sigma_s = 0.1$, $e_p = 1$, and $e_t = 2$. The locations and orientations of these camera sensors are listed in Table II.

Assume that the location of the target is (1950 mm, 650 mm) and that c_0 is the first to detect the target. From (23), we can obtain the pdf of $P(x_t, z_t|X_0)$ [see Fig. 8(a)]. According to (24), every camera sensor can calculate the probability of detecting the target. Their probabilities are also listed in Table II. In this paper, we assume that if the probability of detecting the target exceeds 0.5, the corresponding camera sensor becomes the candidate. Then, $C_c = \{c_0, c_1, c_2, c_4, c_7\}$. Moreover, Fig. 7 shows the target images captured by the six camera sensors, respectively. From Fig. 7(d), we can find that c_3 can detect the target. However, the corresponding probability $p_3 = 0.2646 < 0.5$, and thus, c_3 is not the candidate node. Meanwhile, c_7 is a candidate node, but it cannot detect the target. Fig. 7(f) shows that c_5 can detect the target, but p_5 is 0. This is because the distance between c_5 and the target exceeds r. Therefore, the final candidate set $C_c = \{c_0, c_1, c_2, c_4\}$.

Let u_i denote the horizontal pixel coordinates of the target for camera sensor c_i (see Fig. 7). In Fig. 7, $u_0 = 140$, $u_1 = 1055$, $u_2 = 990$, and $u_4 = 612$ are the pixel-coordinate-based measurements of c_0, c_1, c_2, and c_4, respectively. We first need to transform the horizontal pixel coordinates u_i into the real-world coordinates of the horizontal shifts X_i. Because the resolution of these camera sensors is 1280 × 960 and the size of the charge-coupled device is 8.8 mm × 6.6 mm, the transformation formula is as follows:

$$X_i = \left(u_i - \frac{1280}{2}\right) \times \frac{8.8}{1280}. \tag{32}$$

Using (32), we derive a number of the corresponding measurements X_i's, as summarized in Table II.

Set $e_0 = 25$, $e_1 = 30$, $e_2 = 80$, $e_4 = 20$, and $\vartheta = 7$. According to (30), we can get the initial Q_c as follows:

$$Q_c : \{c_2\}, \{c_0\}, \{c_1\}, \{c_4\}.$$

Because one camera sensor cannot satisfy the requirement of localization [$\mathcal{U}(\{c_0\}) = 4.9685$, see Fig. 8(a)], we insert $\{c_2, c_0\}$ into Q_c after $\{c_0\}$, which leads to the following queue:

$$Q_c : \{c_2\}, \{c_0\}, \{c_2, c_0\}, \{c_1\}, \{c_4\}.$$

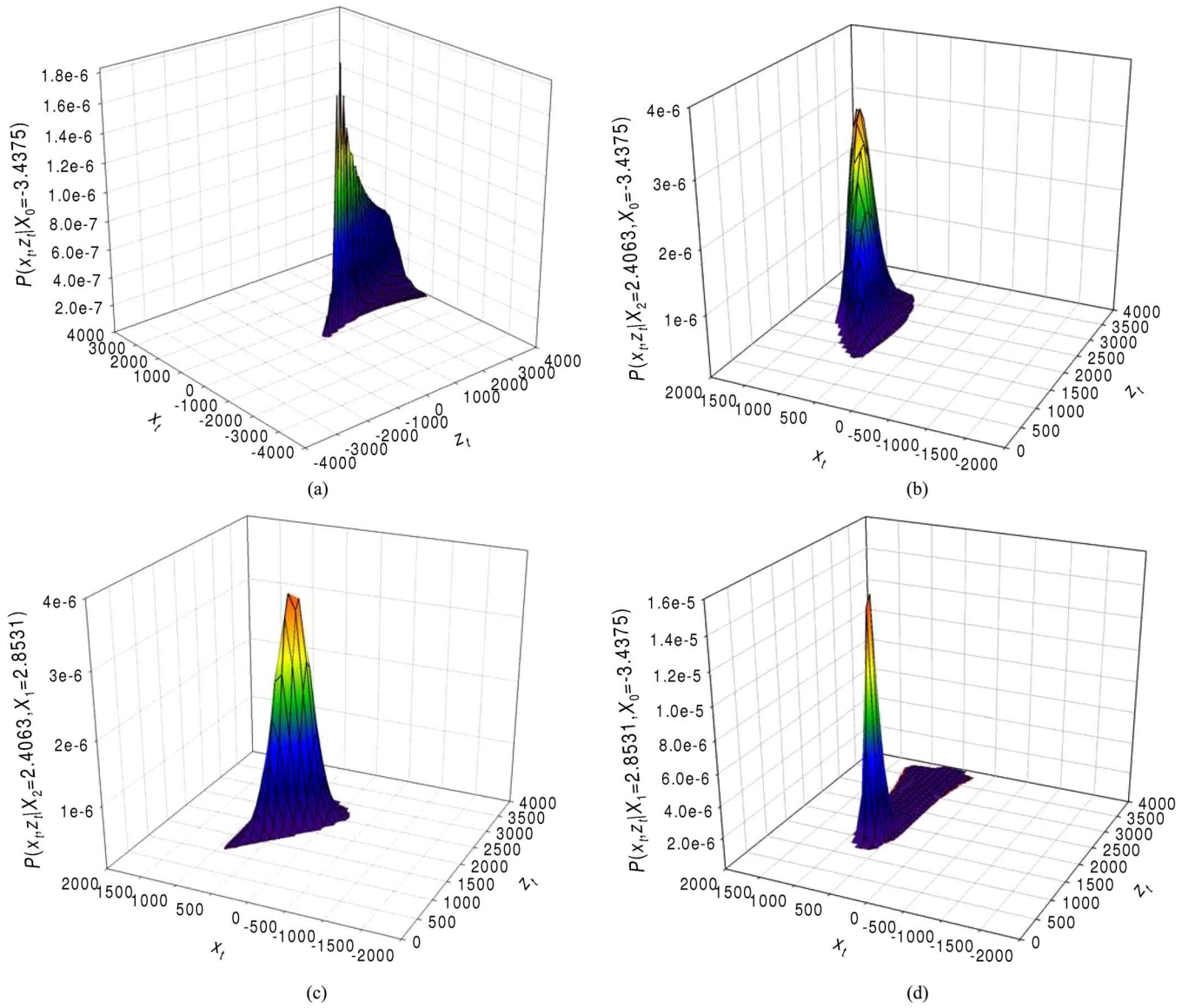


Fig. 8. Probability distribution of the target's location. (a) $P(x_t, z_t|X_0 = -3.4375)$. (b) $P(x_t, z_t|X_2 = 2.4063, X_0 = -3.4375)$. (c) $P(x_t, z_t|X_2 = 2.4063, X_1 = 2.8531)$. (d) $P(x_t, z_t|X_1 = 2.8531, X_0 = -3.4375)$.

Combing the measures of c_2 and c_0, we can get the pdf of $P(x_t, z_t|X_2 = 2.4063, X_0 = -3.4375)$ [see Fig. 8(b)]. The corresponding utility value is 6.0534, which cannot satisfy the requirement. Because the next element $\{c_1\}$ in Q_c cannot also satisfy the requirement, we insert $\{c_2, c_1\}, \{c_0, c_1\}, \{c_2, c_0, c_1\}$ into Q_c, i.e.,

$$Q_c : \{c_2\}, \{c_0\}, \{c_2, c_0\}, \{c_1\}, \{c_2, c_1\}, \{c_0, c_1\}, \{c_2, c_0, c_1\}, \{c_4\}.$$

Fig. 8(c) illustrates the pdf of $P(x_t, z_t|X_2 = 2.4063, X_1 = 2.8531)$. The corresponding utility value, i.e., 5.8614, cannot also satisfy the requirement. For the set $\{c_0, c_1\}$, the distribution of $P(x_t, z_t|X_1 = 2.8531, X_0 = -3.4375)$ is more concentrative than $P(x_t, z_t|X_2 = 2.4063, X_0 = -3.4375)$ and $P(x_t, z_t|X_2 = 2.4063, X_1 = 2.8531)$ [see Fig. 8(d)]. The corresponding utility value is equal to 7.0239, satisfying the requirement, which implies that $\{c_0, c_1\}$ is the optimal set of locating camera sensors.

B. Experimental Evaluations of Our Proposed Schemes

To perform empirical evaluations of our schemes, we have built up a simulation platform by VC++. The fixed parameters of the simulation platform are as follows: $S = 500 \times 500$, $r = 40$, $\alpha = \pi/6$, $\zeta = 5e - 4$, $\sigma_p = 0.1$, and $\sigma_s = 0.1$. The accuracy requirement of localization $\varepsilon = 4$. As shown in Appendix B, the corresponding $a_t = 0.09$. Because $\varepsilon > a_t r = 3.6$, $P_l \approx P_2$, we get

$$P_l \approx 1 - e^{-\lambda_l \alpha r^2} - \lambda_l \alpha r^2 e^{-\lambda_l \alpha r^2}.$$

In each simulation run, we randomly scatter a number of camera sensors according to a 2-D Poisson process with the mean equal to $\lambda \times 250\,000$ within S. The number of camera sensors N varies from 0 to 1000 per 100 steps. This implies that the density of camera sensors λ varies from 0 to 0.004 per 0.0004 step. A grid of 500×500 vertices is created for S. For a given λ, we generate a network topology. Assume that there are n vertices, which are covered by at least one camera sensor.

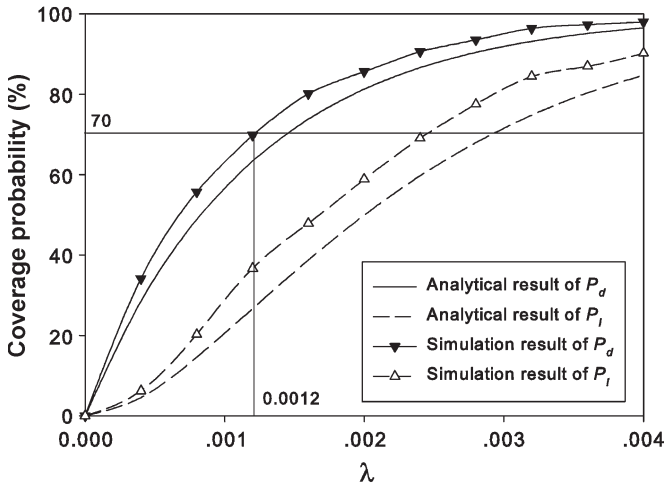


Fig. 9. L-coverage probability P_l and D-coverage probability P_d against the camera sensor density λ.

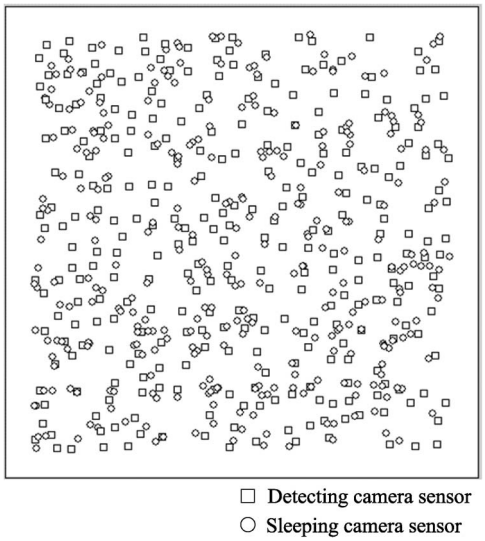


Fig. 10. Topology of the wireless sensor network with 600 camera sensors.

Then, the corresponding D-coverage probability is the ratio between n and 250 000. The aforementioned process is repeated 100 times to obtain the mean of the D-coverage probability for the given λ. Thus, the simulation result of P_d is $\mathrm{E}[n]/250\,000$. By using the similar method, we can also get the simulation results for P_l.

Fig. 9 plots the simulation and analytical results of P_d and P_l. From Fig. 9, we can have the observe the following.

1) For P_d and P_l, the simulation results are close to the corresponding analytical results.
2) To obtain a given coverage probability, the simulation result of λ should be slightly smaller than the corresponding analytical result.

We set $\varepsilon_d = \varepsilon_l = 0.7$. Substituting $\varepsilon_d = 0.7$ into (20), we have $r_c = 14$. According to the simulation results of P_l in Fig. 9, we generate 600 camera sensors, which leads to P_l that is approximately equal to 0.7. As shown in Fig. 10, when $r_c = 14$, the number of the detecting camera sensors is about 310, and P_d is about 0.72.

Assume that the initial energy of each camera sensor is 100; the energy consumption of the sleeping state can be neglected; and, for a fixed period τ, the energy consumption levels of the detecting and locating states are 0.1 and 1, respectively. For each τ, we randomly generate a target in S. Then, we apply three different schemes to the wireless camera sensor network.

1) **N** *scheme:* All the deployed camera sensors detect the target in the detecting phase, and all the camera sensors that can detect the target collaboratively estimate the location in the locating phase.
2) **D** *scheme:* In the detecting phase, the density control scheme selects the set of detecting camera sensors from deployed camera sensors; in the locating phase, all the camera sensors that can detect the target collaboratively estimate the location.
3) **L** *scheme:* This is the scheme proposed in this paper.

Fig. 11(a)–(f) shows the statistical results for the remaining energy of camera sensors. The x-axis denotes the remaining energy, and the y-axis denotes the ratio between the number of camera sensors with a given remaining energy and the total number of camera sensors. Fig. 11(a)–(f) shows that the energy cost of the **N** scheme is much higher than those of the **D** and **L** schemes, and this energy cost difference increases as the number of τ increases. The main reason is that almost half of the camera sensors are in the sleeping state in the **D** and **L** schemes. For a wireless camera sensor network, there is no event/target in most of the lifetime; thus, most energy is consumed for the detecting state.

When the number of τ is at the lower end, for most camera sensors, the remaining energy of the **L** scheme is the same as that for the **D** scheme. However, in the **D** scheme, there exist a few camera sensors that consume much more energy. For example, after 200τ, as shown in Fig. 11(b), the energy of 0.17% camera sensors are in [81, 83], and the energy of most camera sensors are in [87, 90]. This is because there are a few points in S that are covered by many camera sensors. As the number of τ increases, for most camera sensors, the remaining energy difference between the **L** scheme and the **D** scheme increases. As shown in Fig. 11(f), after 800τ, for the **L** scheme, the energy of 62.84% camera sensors are in the range of [57, 60], and for the **D** scheme only 37.5% camera sensors' energy falls in [57, 60].

VII. Conclusion

We have tackled the node-selection problem by balancing the tradeoff between the accuracy of target localization and the energy consumption in camera sensor networks. Based on the sensing model of camera sensors, we have proposed a cooperative localization algorithm, which is implemented by two phases: 1) detecting phase and 2) locating phase. As to the detecting and locating phases, we have designed a two-step node-selection scheme. For the detecting phase, we have developed a PEAS-based density control algorithm to select the proper subset of detecting camera sensors for maintaining the desired quality of detection. For the locating phase, we have mapped the LNS problem to an optimization problem and then

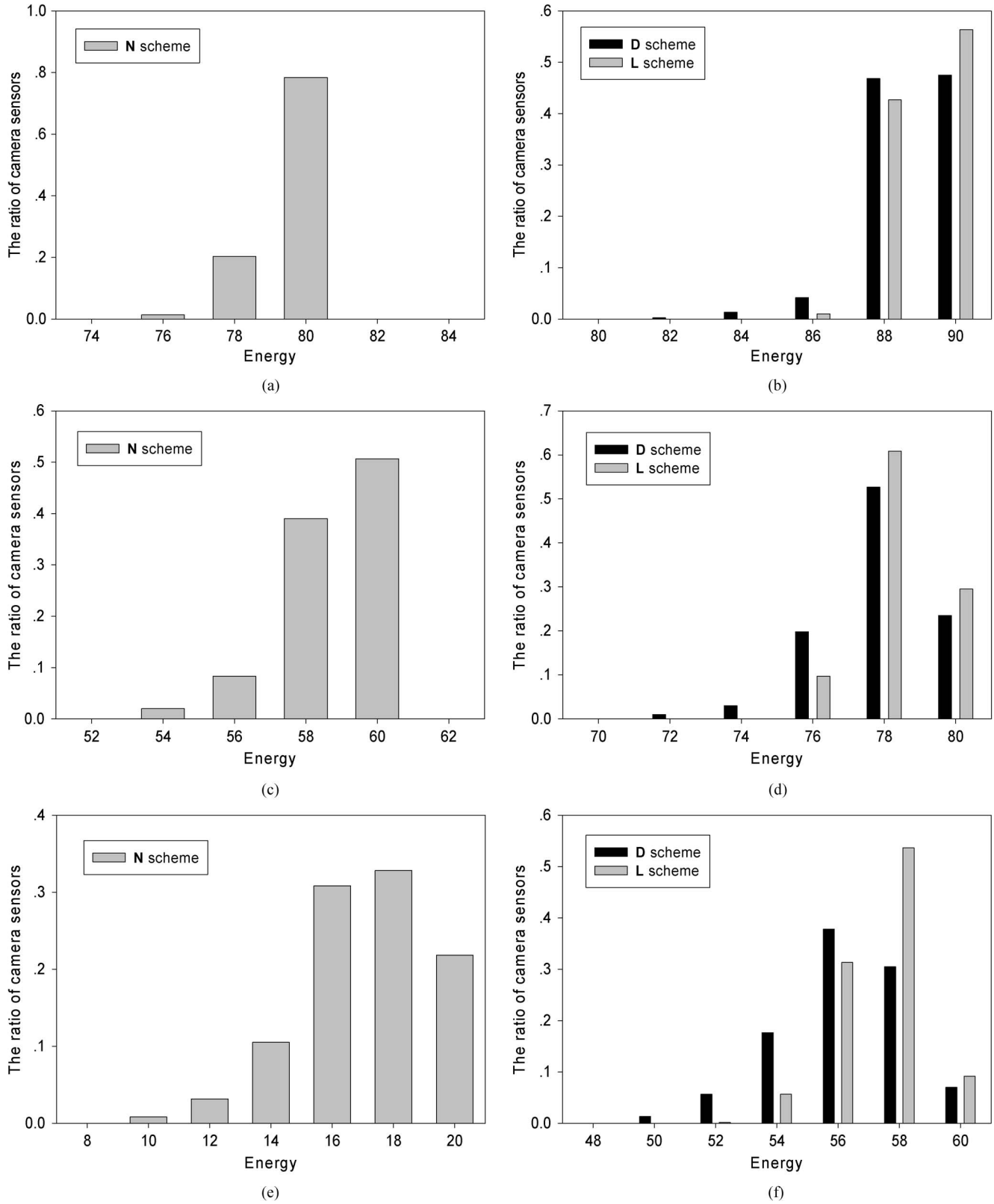


Fig. 11. Energy consumption comparisons among the **N**, **D**, and **L** schemes with different numbers of time periods. (a) **N** scheme, 200τ. (b) **N** and **L** schemes, 200τ. (c) **N** scheme, 400τ. (d) **N** and **L** schemes, 400τ. (e) **N** scheme, 800τ. (f) **N** and **L** schemes, 800τ.

proposed an optimal algorithm to select a set of camera sensors for estimating the location of target with the minimum cost. We have also conducted extensive experiments and simulations to validate and evaluate our proposed scheme.

APPENDIX A
PROOF OF LEMMA 1

Proof: From the sensing model of the wireless camera sensor networks, it is easy to know that, if a camera sensor can

detect a point T, then the location of this camera sensor must be in the disk, denoted by R, which is centered around T with radius r. On the other hand, not all camera sensors in R can detect T because of their orientations. Assume that there are n camera sensors in R, from (11), we have

$$\Pr\{N_R = n\} = \frac{(\lambda\pi r^2)^n}{n!} e^{-\lambda\pi r^2}$$

where N_R is the number of the cameras sensors within R. The probability that a camera sensor within R can detect T is α/π. Then, the conditional probability that $k(k \leq n)$ of these n camera sensors can detect T is

$$\Pr\{N_T = k | N_R = n\} = \left(\frac{\alpha}{\pi}\right)^k \left(1 - \frac{\alpha}{\pi}\right)^{n-k} \binom{n}{k}.$$

Thus, we have

$$\begin{aligned}\Pr\{N_T = k\} &= \sum_{n=k}^{\infty} \Pr\{N_R = n\} \Pr\{N_T = k | N_R = n\} \\ &= \sum_{n=k}^{\infty} \frac{(\lambda\pi r^2)^n}{n!} e^{-\lambda\pi r^2} \left(\frac{\alpha}{\pi}\right)^k \left(1 - \frac{\alpha}{\pi}\right)^{n-k} \binom{n}{k} \\ &= \frac{(\lambda\alpha r^2)^k}{k!} e^{-\lambda\alpha r^2}.\end{aligned}$$

From the definition of the K-coverage probability, we have

$$\begin{aligned}P_K &\triangleq \Pr\{N_T \geq K\} = \sum_{i=K}^{\infty} \Pr\{N_T = i\} \\ &= \sum_{i=K}^{\infty} \frac{(\lambda\alpha r^2)^i}{i!} e^{-\lambda\alpha r^2} = 1 - \sum_{i=0}^{K-1} \frac{(\lambda\alpha r^2)^i}{i!} e^{-\lambda\alpha r^2}.\end{aligned}$$

■

Appendix B
Derivation of (17)

Let $L(T)$ be the indicator function of whether a point T is L-covered or not, i.e.,

$$L(T) = \begin{cases} 1, & \text{if } T \text{ is } L\text{-covered} \\ 0, & \text{if } T \text{ is not } L\text{-covered.} \end{cases}$$

According to [25] and Fubini's theorem [26], if $\Pr\{L(T) = 1\}$ is constant for all $T \in S$, then the L-coverage probability is equal to the probability that T is L-covered, i.e.,

$$P_l = \Pr\{L(T) = 1\}. \tag{33}$$

A point $T \in S$ that is L-covered by k camera sensors implies that there exist k camera sensors that can detect T and that the corresponding δ_k of these k camera sensors is smaller than the predefined threshold ε. Referring to (2), we can obtain that it is impossible to get (x_t, y_t) by using only one X. This implies that the point that is detected by only one camera sensor is not L-covered. Let N_T be the number of camera sensors that can detect T. Then, we have

$$P_l = \sum_{k=2}^{\infty} \Pr\{N_T = k\} \Pr\{\delta_k < \varepsilon\}. \tag{34}$$

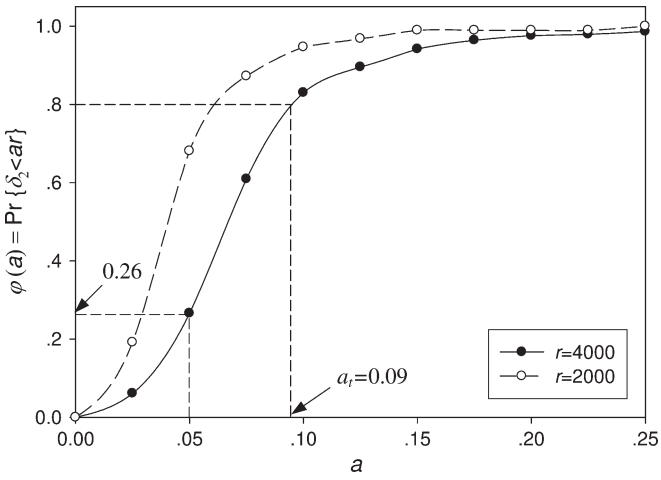


Fig. 12. Relationship between $\Pr\{\delta_2 < \varepsilon\}$ and a with different r. We set $\zeta = 5 \times 10^{-8}$, $\sigma_p = 0.1$, and $\sigma_s = 0.1$.

However, according to (16), it is difficult to derive the closed-form analytical expression for $\Pr\{\delta_k < \varepsilon\}$. Thus, we calculate $\Pr\{\delta_2 < \varepsilon\}$ by using Monte Carlo simulations.

We randomly deploy two camera sensors, i.e., c_1 and c_2, in the disk centered around T with radius r. Their orientations, i.e., θ_1 and θ_2, satisfy the random uniform distribution on $[\gamma_1 - \alpha, \gamma_1 + \alpha]$ and $[\gamma_2 - \alpha, \gamma_2 + \alpha]$, where γ_1 and γ_2 are the orientations of $\overrightarrow{L_1T}$ and $\overrightarrow{L_2T}$, respectively. Then, we can get the corresponding δ_2 according to (16). The aforementioned process is repeated 1000 times to obtain 1000 δ_2's. Let $a \triangleq \varepsilon/r$ vary from 0 to 0.25 per 0.025 step. For each value of a, we can summate the number, denoted by $N_{L,2}$, of δ_2's that are smaller than ε. Then, $\Pr\{\delta_2 < \varepsilon\}$ approximates the ratio of $N_{L,2}$ to 1000.

Define $\Pr\{\delta_2 < \varepsilon\} \triangleq \varphi(a)$ as the function of a, where $a \triangleq \varepsilon/r$. As shown in Fig. 12, we can get the plot of $\varphi(a)$ according to 1000 δ_2's of the corresponding Monte Carlo runs. When $r = 4000$, about 80% δ_2's are smaller than $r/10$. This implies that, if the requirement of localization accuracy is not very strict, then the probability that a point is L-covered by two camera sensors, i.e., $\Pr\{\delta_2 < \varepsilon\}$, is high. Furthermore, from Fig. 12, we can also observe that, for a fixed a, $\Pr\{\delta_2 < \varepsilon\}$ decreases as r increases.

When $k \geq 3$, it is complicated to derive δ_k according to (16), because the dimensions of $\Re$ are large. A property of $\Pr\{\delta_k < \varepsilon\}$ is that $\Pr\{\delta_k < \varepsilon\}$ increases as k increases, i.e., $\Pr\{\delta_k < \varepsilon\} < \Pr\{\delta_{k+1} < \varepsilon\}$. This is because using one more camera sensor for estimation reduces the estimation error. Then, when $\Pr\{\delta_2 < \varepsilon\}$ approaches 1, we can use P_2 as the approximation of P_l.

In this paper, we assume that, if $\Pr\{\delta_2 < \varepsilon\} \geq 0.8$, then $P_l \approx P_2$. Because $\Pr\{\delta_2 < \varepsilon\}$ monotonously increases as a

increases, we can obtain a threshold value of a, denoted by a_t, where $a_t = \inf\{a \mid \varphi(a) \geq 0.8\}$. Thus, if $\varepsilon \geq a_t r$, then

$$P_l \approx P_2 = 1 - e^{-\lambda_l \alpha r^2} - \lambda_l \alpha r^2 e^{-\lambda_l \alpha r^2}.$$

However, if $\varepsilon < a_t r$, i.e., $\Pr\{\delta_2 < \varepsilon\} < 0.8$, then the difference between P_l and P_2 cannot be neglected. As shown in Fig. 12, if $\varepsilon = 0.05r < a_t r$, then $\Pr\{\delta_2 < \varepsilon\} = 0.26$. Let us define

$$r' \triangleq \frac{\varepsilon}{a_t}.$$

Then, we can get the corresponding δ_2, $\Pr\{N_T = 2\}$, and P_K with sensing radius r', denoted by δ_2', $\Pr\{N_T' = 2\}$, and P_K', respectively. Because $\Pr\{\delta_2 < \varepsilon\}$ decreases as r increases, $\varphi'(a_t) > 0.8$, where $\varphi'(a)$ is the function of the relationship between $\Pr\{\delta_2' < \varepsilon\}$ and a. This implies that

$$P_l \approx P_2' = 1 - e^{-\lambda_l \alpha r'^2} - \lambda_l \alpha r'^2 e^{-\lambda_l \alpha r'^2}.$$

Therefore, we can derive the approximative expression for the relationship between P_l and λ as follows:

$$P_l \approx 1 - e^{-\lambda_l \alpha R^2} - \lambda_l \alpha R^2 e^{-\lambda_l \alpha R^2}$$

where

$$R = \begin{cases} r, & \text{if } \varepsilon > a_t r \\ \frac{\varepsilon}{a_t}, & \text{otherwise} \end{cases}$$

which is (17).

REFERENCES

[1] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: A multi-tier camera sensor network," in *Proc. 13th Annu. ACM Int. Conf. Multimed.*, Singapore, Nov. 2005, pp. 229–238.

[2] I. F. Akyildiz, T. Melodia, and K. Chowdhury, "Wireless multimedia sensor networks: A survey," *Comput. Netw.*, vol. 51, no. 4, pp. 921–960, Mar. 2007.

[3] H. Ma and D. Tao, "Multimedia sensor network and its research progresses," *J. Softw.*, vol. 17, no. 9, pp. 2013–2028, 2006.

[4] R. Holman, J. Stanley, and T. Ozkan-Haller, "Applying video sensor networks to nearshore environment monitoring," *Pervasive Comput.*, vol. 2, no. 4, pp. 14–21, Oct.–Dec. 2003.

[5] S. Denman, C. Fookes, J. Cook, C. Davoren, A. Mamic, G. Farquharson, D. Chen, B. Chen, and S. Sridharan, "Multi-view intelligent vehicle surveillance system," in *Proc. IEEE Int. Conf. AVSS*, Sydney, Australia, Nov. 2006, p. 26.

[6] M. Bramberger, A. Doblander, A. Maier, B. Rinner, and H. Schwabach, "Distributed embedded smart cameras for surveillance applications," *Comput.*, vol. 39, no. 2, pp. 68–75, Feb. 2006.

[7] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, Jul. 2007.

[8] W. M. Hu, T. N. Tan, L. Wang, and S. J. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 34, no. 3, pp. 334–352, Aug. 2004.

[9] F. Ye, S. Lu, and L. Zhang, *GRAdient Broadcast: A Robust, Long-Lived Large Sensor Network*, 2001. [Online]. Available: http://irl.cs.ucla.edu/papers/grab-tech-report.ps

[10] D. Li, K. Wong, Y. Hu, and A. Sayeed, "Detection, classification, tracking of targets in micro-sensor networks," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 17–29, Mar. 2002.

[11] J. Liu, J. Liu, J. Reich, P. Cheung, and F. Zhao, "Distributed group management for track initiation and maintenance in target localization applications," in *Proc. IPSN*, Apr. 2003, pp. 113–128.

[12] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *Int. J. High Perform. Comput. Appl.*, vol. 16, no. 3, pp. 293–313, 2002.

[13] E. Ertin, J. W. Fisher, III, and L. C. Potter, "Maximum mutual information principle for dynamic sensor query problems," in *Proc. IPSN*, Apr. 2003, pp. 405–416.

[14] H. Wang, K. Yao, G. Pottie, and D. Estrin, "Entropy-based sensor selection heuristic for localization," in *Proc. IPSN*, Apr. 2004, pp. 36–45.

[15] P. Pahalawatta, T. N. Pappas, and A. K. Katsaggelos, "Optimal sensor selection for video-based target tracking in a wireless sensor network," in *Proc. IEEE Int. Conf. Image Process.*, Singapore, 2004, pp. 3073–3076.

[16] D. B. R. Yang, J. W. Shin, A. O. Ercan, and L. J. Guibas, "Sensor tasking for occupancy reasoning in a network of cameras," in *Proc. BASENETS*, 2004. [Online]. Available: http://www.broadnets.org/2004/basenets.html

[17] V. Isler and R. Bajcsy, "The sensor selection problem for bounded uncertainty sensing models," in *Proc. Int. Symp. Inf. Process. Sens. Netw.*, Los Angeles, CA, 2005, pp. 151–158.

[18] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. Symp. Oper. Syst. Des. Implementation*, 2002, pp. 147–163.

[19] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing: Analysis and Machine Vision,* 2nd ed. London, U.K.: Chapman & Hall, 1995.

[20] M. Piccardi, "Background subtraction techniques: A review," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, The Hague, The Netherlands, Oct. 2004, pp. 3099–3104.

[21] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. S. Davis, "Real-time foreground–background segmentation using codebook model," *Real-Time Imaging*, vol. 11, no. 3, pp. 172–185, Jun. 2005.

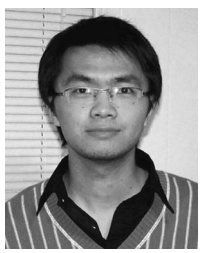
[22] A. O. Ercan, D. B.-R. Yang, A. El Gamal, and L. J. Guibas, "Optimal placement and selection of camera network nodes for target localization," in *Proc. DCOSS*, Jun. 2006, pp. 389–404.

[23] L. Liu, X. Zhang, and H. Ma, "Localization-oriented coverage based on Bayesian estimation in camera sensor networks," in *Proc. IEEE WOWMOM*, Jun. 2008, pp. 1–8.

[24] S. Kumar, T. H. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *Proc. ACM MobiCom*, Philadelphia, PA, 2004, pp. 144–158.

[25] B. Wang, K. C. Chua, V. Srinivasan, and W. Wang, "Information coverage in randomly deployed wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 8, pp. 2994–3004, Aug. 2007.

[26] G. B. Thomas and R. L. Finney, *Calculus and Analytic Geometry,* 8th ed. Reading, MA: Addison-Wesley, 1996.



Liang Liu was born in Chongqing, China, in 1982. He received the B.S. degree from South China University of Technology, Guangzhou, China, in 2004. He is currently working toward the Ph.D. degree with the Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, China.

He was a Visiting Ph.D. Student with the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station, during 2007–2008. His research interests are in the fields of wireless sensor networks, camera sensor networks, and information theory.

Xi Zhang (S'89–SM'98) received the B.S. and M.S. degrees from Xidian University, Xi'an, China, the M.S. degree from Lehigh University, Bethlehem, PA, both in electrical engineering and computer science, and the Ph.D. degree in electrical engineering and computer science (Electrical Engineering-Systems) from The University of Michigan, Ann Arbor.

He is currently an Associate Professor and the Founding Director of the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station. He was an Assistant Professor and the Founding Director of the Division of Computer Systems Engineering, Department of Electrical Engineering and Computer Science, Beijing Information Technology Engineering Institute, Beijing, China, from 1984 to 1989. He was a Research Fellow with the School of Electrical Engineering, University of Technology, Sydney, Australia, and the Department of Electrical and Computer Engineering, James Cook University, Townsville, Australia, under a Fellowship from the Chinese National Commission of Education. He worked as a Summer Intern with the Networks and Distributed Systems Research Department, AT&T Bell Laboratories, Murray Hill, NJ, and with AT&T Laboratories Research, Florham Park, NJ, in 1997. He has published more than 170 research papers in the areas of wireless networks and communications systems, mobile computing, network protocol design and modeling, statistical communications, random signal processing, information theory, and control theory and systems.

Prof. Zhang received the U.S. National Science Foundation CAREER Award in 2004 for his research in the areas of mobile wireless and multicast networking and systems. He received the Best Paper Awards at the IEEE WCNC 2010, the IEEE GLOBECOM 2009, and the IEEE GLOBECOM 2007, respectively. He also received the Texas Engineering Experiment Station Select Young Faculty Award for Excellence in Research Performance from the Dwight Look College of Engineering at Texas A&M University in 2006. He is currently serving as an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for the special issue on "wireless video transmissions," an Associate Editor for the IEEE COMMUNICATIONS LETTERS, a Guest Editor for the *IEEE Wireless Communications Magazine* for the special issue on "next generation of CDMA versus OFDMA for 4G wireless applications," an Editor for the *Wiley Journal on Wireless Communications and Mobile Computing*, an Editor for the *Journal of Computer Systems, Networking, and Communications*, an Associate Editor for the *Wiley Journal on Security and Communications Networks*, an Area Editor for the *Elsevier Journal on Computer Communications*, and a Guest Editor for the *Wiley Journal on Wireless Communications and Mobile Computing* for the special issue on "next generation wireless communications and mobile computing." He has frequently served as the Panelist on the U.S. National Science Foundation Research-Proposal Review Panels. He is serving or has served as the Technical Program Committee (TPC) Co-Chair for IEEE INFOCOM 2013, TPC Chair for IEEE GLOBECOM 2011, General Chair for the IEEE ICC 2011 Workshop on Advanced Networking and Smart-Services Based Clouding Computing, TPC Vice-Chair for IEEE INFOCOM 2010, TPC Co-Chair for the IEEE INFOCOM 2009 Mini-Conference, TPC Co-Chair for the IEEE GLOBECOM 2008 Wireless Communications Symposium, TPC Co-Chair for the IEEE ICC 2008 Information and Network Security Symposium, Symposium Chair for the IEEE/ACM International Cross-Layer Optimized Wireless Networks Symposium 2006, 2007, and 2008, respectively, the TPC Chair for IEEE/ACM IWCMC 2006, 2007, and 2008, respectively, the Demo/Poster Chair for IEEE INFOCOM 2008, the Student Travel Grants Co-Chair for IEEE INFOCOM 2007, General Chair for ACM QShine 2010, the Panel Co-Chair for IEEE ICCCN 2007, the Poster Chair for IEEE/ACM MSWiM 2007 and IEEE QShine 2006, Executive Committee Co-Chair for QShine, the Publicity Chair for IEEE/ACM QShine 2007 and IEEE WirelessCom 2005, and a Panelist on the Cross-Layer Optimized Wireless Networks and Multimedia Communications at IEEE ICCCN 2007 and WiFi-Hotspots/WLAN and QoS Panel at IEEE QShine 2004. He has served as a TPC member for more than 70 IEEE/ACM conferences, including IEEE INFOCOM, IEEE GLOBECOM, IEEE ICC, IEEE WCNC, IEEE VTC, IEEE/ACM QShine, IEEE WoWMoM, IEEE ICCCN, etc. He has presented technical tutorial lectures at the IEEE ICC and IEEE VTC conferences. He is a member of the Association for Computing Machinery.

Huadong Ma (M'99) received the B.S. degree in mathematics from Henan Normal University, Xinxiang, China, in 1984, the M.S. degree in computer science from Shenyang Institute of Computing Technology, Chinese Academy of Science, Shenyang, China, in 1990, and the Ph.D. degree in computer science from the Chinese Academy of Science, Beijing, China, in 1995.

He is a Professor and the Director of the Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia and the Chair of the Department of Computer Science and Technology, Beijing University of Posts and Telecommunications. He visited the United Nations University International Institute for Software Technology, Macao, China, as a Research Fellow in 1998 and 1999, respectively. From 1999 to 2000, he held a visiting position with the Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor. He was a Visiting Professor with The University of Texas at Arlington from July to September 2004 and a Visiting Professor with Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, from December 2006 to February 2007. His current research focuses on multimedia system and networking, sensor networks, and grid computing. He has published more than 100 papers and four books on the aforementioned fields.

Dr. Ma is member of the Association for Computing Machinery.