

# Camera Scheduling and Energy Allocation for Lifetime Maximization in User-Centric Visual Sensor Networks

Chao Yu, *Student Member, IEEE*, and Gaurav Sharma, *Senior Member, IEEE*

**Abstract**—We explore camera scheduling and energy allocation strategies for lifetime optimization in image sensor networks. For the application scenarios that we consider, visual coverage over a monitored region is obtained by deploying wireless, battery-powered image sensors. Each sensor camera provides coverage over a part of the monitored region and a central processor coordinates the sensors in order to gather required visual data. For the purpose of maximizing the network operational lifetime, we consider two problems in this setting: a) *camera scheduling*, i.e., the selection, among available possibilities, of a set of cameras providing the desired coverage at each time instance, and b) *energy allocation*, i.e., the distribution of total available energy between the camera sensor nodes. We model the network lifetime as a stochastic random variable that depends upon the coverage geometry for the sensors and the distribution of data requests over the monitored region, two key characteristics that distinguish our problem from other wireless sensor network applications. By suitably abstracting this model of network lifetime and utilizing asymptotic analysis, we propose lifetime-maximizing camera scheduling and energy allocation strategies. The effectiveness of the proposed camera scheduling and energy allocation strategies is validated by simulations.

**Index Terms**—Camera scheduling, energy allocation, image sensor networks, network lifetime, visual coverage.

## I. INTRODUCTION

WIRELESS visual/image sensor networks (VSN) have recently evoked intense research interest due to the increasing demand for applications such as security surveillance, smart home care, and environment monitoring [3]–[6]. These sensor networks provide visual coverage over a monitored region by deploying portable wireless sensors with imaging, signal processing, and communication capabilities. Because the sensors are usually battery powered, power consumption

imposes a critical constraint on the network lifetime of a VSN, which is defined as the duration of effective visual coverage over the monitored region. In this paper, we consider *user-centric* application scenarios where only part of the visual data, defined as the user’s “desired view,” is of interest at a given time instant, though the desired view varies with time and thereby involves the entire VSN. An example of such an application scenario is a visual surveillance network [7], [8] deployed for tracking and recording imagery of moving objects in a monitored region [9], [10]. In this scenario, the series of image requests arising from the object tracking can be modeled as a series of virtual “user” requests. Another sample application is a VSN deployed for tele-presence applications, where the desired view corresponds directly to the users’ requested view [11]. In these user-centric VSNs, the visual data of interest at a given time instant overlaps the fields of view (FoVs) of a number of cameras (sensors) and one may select among the cameras providing the requested coverage. We refer to this selection as the *camera scheduling* problem and investigate camera scheduling strategies with a view to maximizing the lifetime of the network. In some scenarios, the deployment of the cameras is constrained, and the allocation of available energy among these cameras can have a significant impact on the network lifetime. We therefore also investigate *energy allocation* to distribute the total available energy among the sensor nodes. Our abstraction of the user-centric VSN setting in terms of a time varying “desired view” allows us to focus on the camera scheduling and energy allocation problems without bringing in other application specific aspects of these VSNs, which have a limited impact on these specific problems.

Camera scheduling and energy allocation in user-centric VSNs are challenging because of two reasons. On the one hand, meaningful definitions of lifetime for a VSN must take into account the *visual coverage* provided by the network. On the other hand, the stochastic nature of data requests needs to be suitably addressed. The main contribution of this paper is a stochastic formulation for the expected value for the network lifetime addressing both of these aspects. Visual coverage information of the network is incorporated into the formulation of network lifetime, wherein the data requests are additionally modeled as a random variable (r.v.) with a distribution that is either known *a priori* or estimated from the record of prior requests. Using an abstraction for this formulation, we obtain expressions for the expected network lifetime and develop computationally efficient approximations and suitable lifetime-maximizing sensor scheduling strategies. Using the

Manuscript received March 20, 2009; revised March 09, 2010. First published March 29, 2010; current version published July 16, 2010. This work was supported in part by the National Science Foundation under grant number ECS-0428157. Parts of this work have been presented at Visual Communications and Image Processing (VCIP), January 2009, San Jose CA, and at the IEEE International Conference on Image Processing (ICIP), November 2009, Cairo, Egypt. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Amy R. Reibman.

C. Yu is with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627 USA (e-mail: chyu@ece.rochester.edu; devincy@gmail.com).

G. Sharma is with the Electrical and Computer Engineering Department and the Department of Biostatistics and Computational Biology, University of Rochester, Rochester, NY 14627 USA (e-mail: gaurav.sharma@rochester.edu; g.sharma@ieee.org).

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/TIP.2010.2046794

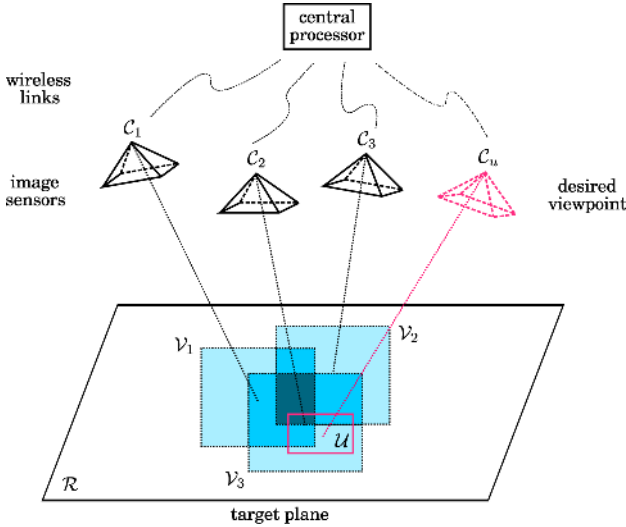


Fig. 1. Target plane  $\mathcal{R}$  is monitored by a VSN consists of cameras  $\{C_j\}_{j=1}^N$ .  $C_j$  covers a subregion  $\mathcal{V}_j$  of the target plane. A central processor (CP) keeps record of the energy distribution, coverage geometry of the network, and receives the user requests. For each user's desired view  $\mathcal{U}$ , the CP selects a subset of cameras to provide data and synthesizes the desired view  $\mathcal{U}$ . No direct communication is feasible between the cameras. The intensity of subregions indicates coverage information: regions covered by more cameras appear darker.

same analysis, the energy allocation problem is formulated as a max-min optimization problem that aims to maximize the duration of coverage for the most critical part of the monitored region for which the available energy is the least. By transforming the min-max optimization into an equivalent linear programming (LP) problem, we present a computationally efficient solution for the energy allocation problem.

Strategies for optimizing network lifetime in generic wireless sensor networks (WSNs) have been previously considered in the literature [12]–[15]. The coverage model for VSNs, however, is drastically different from the common WSN model of circular coverage centered at the sensor. For instance, physically adjacent cameras in a VSN may have completely nonoverlapping FoVs. Second, in typical uses of VSNs, only a subset of the data is of interest, e.g., in a surveillance network, coverage may only be desired for the moving objects. The analysis of lifetime in user-centric VSNs must therefore consider stochastic data requests, which are typically absent in conventional WSNs. In the context of a VSN, [16] addresses the problem of optimal assignment of cameras to monitor subregions of a monitored area in order to maximize the lifetime of the camera network. However, user interactions are not considered. To account for user interactions, [17] proposes a heuristic approach for camera scheduling by defining a cost function associated with each camera depending upon the remaining energy of the camera and the coverage geometry. In this paper, we extend and complement this prior work by developing a mathematical model that leads to analysis and simulation results which provide additional insight.

Though our formulation is valid for several classes of VSNs, for concrete discussion, here we consider an application scenario illustrated in Fig. 1, where image sensors are deployed to provide visual coverage over a monitored region. The network allows users to navigate around the monitored region by spec-

TABLE I  
LIST OF SYMBOLS

Symbol	Meaning
$\mathcal{R}$	monitored area
$\mathcal{U}$	the desired view
$C_j$	the $j^{\text{th}}$ camera
$\mathbf{B}^r, \mathbf{B}^u$	coverage matrices for $\mathcal{R}$ and $\mathcal{U}$
$M_r, M_u$	number of blocks in the discretization of $\mathcal{R}$ and $\mathcal{U}$ , respectively
$\lambda^r(i)$	subset of cameras that cover a block $\mathcal{R}_i$ in $\mathcal{R}$
$\lambda^u(i)$	subset of cameras that cover a block $\mathcal{U}_i$ in $\mathcal{U}$
$w_j (w_j^t)$	the energy of camera $C_j$ (at time $t$ )
$p_i$	the probability that $\mathcal{R}_i$ is requested by the user
$L$	network lifetime
$m_i$	coverage energy of the block $\mathcal{R}_i$
$\mathbf{I}(\cdot)$	the indicator function
$\alpha(\cdot)$	the p.m.f. of a negative binomial distribution
$\omega(\cdot)$	the p.m.f. of a multinomial distribution
$\Omega(\cdot)$	the c.d.f. of a multinomial distribution

ifying a desired viewpoint (position and direction) that varies over time. The user's viewpoint determines the part of the scene that should be captured and transmitted to the user. The desired view at the viewpoint is synthesized at a central processor (CP) by combining parts of the image sent from selected cameras.

This paper is organized as follows. Section II presents a stochastic formulation for the expected network lifetime in a user-centric VSN and formulates the optimal camera scheduling strategy to maximize network lifetime. In Section III, we provide an abstraction of the problem, and present exact, approximate, and asymptotic analysis for estimating the network lifetime. A lifetime-maximizing camera selection strategy is developed in Section IV. We next propose our energy allocation strategy in Section V for maximizing the approximated lifetime. Detailed implementation of the application scenario is described in Section VI. Finally, we describe simulation setup and results in Section VII and conclude the paper in Section VIII. For readers' convenience, a list of symbols is included as Table I.

## II. PROBLEM FORMULATION

We consider the network as illustrated in Fig. 1. For simplicity, the monitored region is assumed to be a planar surface, a situation that may occur for aerial surveillance where the 3D scene can be well approximated by a 2-D plane when viewed at a large distance. The target plane  $\mathcal{R}$  is monitored by  $N$  cameras  $\{C_j\}_{j=1}^N$ . Each camera  $C_j$  has a battery with energy  $w_j$  and covers a subregion of  $\mathcal{R}$  denoted by  $\mathcal{V}_j$ . We uniformly divide  $\mathcal{R}$  into  $M_r$  blocks, represented by  $\{\mathcal{R}_i, i \in [M_r]\}$  where  $[a]$  represents the set  $\{1, 2, \dots, a-1, a\}$ . We represent the coverage geometry of these cameras in terms of this discretized representation, and define a *coverage matrix*  $\mathbf{B}^r \in \mathbb{R}^{M_r \times N}$  as

$$\mathbf{B}_{i,j}^r \stackrel{\text{def}}{=} \mathbf{I}(\mathcal{R}_i \subseteq \mathcal{V}_j) \quad (1)$$

where subscripts  $i, j$  respectively denotes the row and column index of the matrix and  $a \subseteq b$  indicates region(set)  $a$  lies within region(set)  $b$ , we also use  $\mathbf{I}(\mathcal{A})$  to represent the indicator function

$$\mathbf{I}(\mathcal{A}) = \begin{cases} 1, & \text{if } \mathcal{A} \text{ is true} \\ 0, & \text{otherwise.} \end{cases}$$

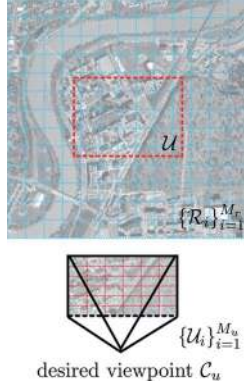


Fig. 2. Discretization of the target plane  $\mathcal{R}$  and desired view  $\mathcal{U}$ . Camera views  $\{\mathcal{V}_j\}_{j=1}^N$  are similarly discretized. The texture image on  $\mathcal{R}$  is used in our simulations.

The subset of cameras that cover block  $\mathcal{R}_i$  is represented by

$$\lambda^r(i) \stackrel{\text{def}}{=} \{j | \mathbf{B}_{i,j}^r = 1, j \in [N]\}. \quad (2)$$

The user specifies a desired viewpoint and accordingly a desired view  $\mathcal{U}$  on the target plane  $\mathcal{R}$ . This desired view  $\mathcal{U}$  is also uniformly divided into  $M_u$  blocks, represented by  $\{\mathcal{U}_i, i \in [M_u]\}$ . The discretization of  $\mathcal{R}$  and  $\mathcal{U}$  is illustrated in Fig. 2. The coverage geometry of  $\mathcal{U}$  is similarly defined as (1) by a coverage matrix  $\mathbf{B}^u \in \mathbb{R}^{M_u \times N}$ , where

$$\mathbf{B}_{i,j}^u \stackrel{\text{def}}{=} \mathbf{I}(\mathcal{U}_i \subseteq \mathcal{V}_j) \quad (3)$$

and the subset of cameras that cover the block  $\mathcal{U}_i$  is denoted by

$$\lambda^u(i) \stackrel{\text{def}}{=} \{j | \mathbf{B}_{i,j}^u = 1, j \in [N]\}. \quad (4)$$

The discretization of  $\mathcal{R}$ ,  $\mathcal{U}$  yields suboptimality, finer discretization results in better performance at the expense of higher computational load. Also note that we consider uniform discretization for ease of description. Alternatively,  $\mathcal{R}$ ,  $\mathcal{U}$  can be divided according to different levels of intersections between  $\mathcal{U}$ ,  $\mathcal{R}$  and  $\{\mathcal{V}_j\}_{j=1}^N$ . We assume the coverage geometry represented by  $\mathbf{B}^r$ ,  $\mathbf{B}^u$  is known.<sup>1</sup>

The network provides the user-desired view in a block-by-block manner. For each block  $\mathcal{U}_i, i \in [M_u]$  in the desired view, the CP selects a camera  $\mathcal{C}_s$  satisfying the coverage requirement, i.e., having  $\mathbf{B}_{i,s}^u = 1$ , to transmit relevant data to the CP where an synthesized view  $\tilde{\mathcal{U}}_i$  is generated. We assume the energy required in order to transmit each block  $\mathcal{U}_i$  to the CP is equal for all  $i \in [M_u]$ .

We assume each block  $\mathcal{R}_i, i \in [M_r]$ , on the monitored plane is requested by the user independently throughout the operation of the network and the probability that the block  $\mathcal{R}_i$  is requested is given by  $p_i$ , where  $\sum_{i=1}^{M_r} p_i = 1$ . Let  $w_j^t$  denote the energy of camera  $\mathcal{C}_j$  at time  $t$ . The remaining lifetime of the network, denoted by  $L$ , at time  $t$  is a r.v. with a probability mass function (p.m.f) determined by  $\{\mathbf{p}, \mathbf{w}^t, \mathbf{B}^r\}$  (note we consider  $L$  as a discrete r.v. in this paper), where

$$\mathbf{p} = [p_1 \ p_2 \ \dots \ p_{M_r}]^T$$

<sup>1</sup>Section VI describes a practical approach to determine  $\mathbf{B}^r, \mathbf{B}^u$  in the VSN we consider.

represents the probability distribution (p.m.f.) of users' requests and

$$\mathbf{w}^t = [w_1^t \ w_2^t \ \dots \ w_N^t]^T$$

denotes the energy distribution, i.e.,  $w_j^t$  is the energy at the  $j$ th camera node at time  $t$ . We denote by  $\mathbb{E}[L(\mathbf{p}, \mathbf{w}^t, \mathbf{B}^r)]$  the expectation of  $L$ , where  $\mathbb{E}[\cdot]$  denotes the expectation operator. At time  $t$ , if camera  $\mathcal{C}_s$  is selected to record and transmit data,  $\mathbf{w}^t$  is updated to  $\mathbf{w}_s^{t+1}$ . The optimal camera selection strategy at time  $t$  is defined as the strategy that maximizes the expected remaining lifetime of the network with respect to the updated energy, i.e.,  $\mathbb{E}[L(\mathbf{p}, \mathbf{w}_s^{t+1}, \mathbf{B}^r)]$ .

We next map the energies of the cameras onto the monitored region and define the *coverage energy* of a block  $\mathcal{R}_i$  as the sum of the energies of all the cameras that cover  $\mathcal{R}_i$ . To this end, we define

$$\mathbf{m}_s^{t+1} = \mathbf{B}^r \mathbf{w}_s^{t+1} \quad (5)$$

thus,  $\mathbf{m}_s^{t+1} \in \mathbb{R}^{M_r}$  and the  $i$ th entry  $m_{i,s}^{t+1}$  represents the coverage energy of  $\mathcal{R}_i$  at time  $(t+1)$ . Specifically, the coverage energy of  $\mathcal{R}_i$  becomes zero when  $w_j^{t+1} = 0$  for all cameras  $\{\mathcal{C}_j, j \in \lambda^r(i)\}$ . We refer to the coverage energy of a block as the energy of the block for brevity.

In order to obtain a useful and tractable formulation of our problem, we approximate the remaining lifetime as a function of  $\{\mathbf{m}_s^{t+1}, \mathbf{p}\}$ . Note that in this process, we have collapsed the dependency of  $L$  on the two parameters i.e., the updated camera energies  $\mathbf{w}_s^{t+1}$  at time  $t+1$  and the coverage matrix  $\mathbf{B}^r$  into the single parameter  $\mathbf{m}_s^{t+1}$ . In this process, we are neglecting the fact that the change in the energy of the selected camera  $\mathcal{C}_s$  will in fact change the energy distribution not only over the block being requested, which we shall account for, but also over the other blocks for which  $\mathcal{C}_s$  provides coverage. Since the parameters  $\{\mathbf{m}_s^{t+1}\}$  are updated afresh at each time step by utilizing (5), scheduling based upon this approximation does not cause a serious compromise in optimality. Now if block  $\mathcal{U}_i$  is requested at time  $t$ , the optimal camera selection strategy is to select a camera from  $\lambda^u(i)$  so that the network has maximum expected lifetime with the updated energy allocation. Mathematically, the optimal camera index is given by

$$s^t = \arg \max_{j \in \lambda^u(i)} \mathbb{E}[L(\mathbf{m}_j^{t+1}, \mathbf{p})]. \quad (6)$$

In order to obtain a solution for (6), we proceed by analyzing the expected network lifetime in(6).

### III. EXPECTED NETWORK LIFETIME

Fig. 3 illustrates an abstraction of the sensor scheduling problem: Consider  $M$  boxes  $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_M$  respectively containing  $m_1, m_2, \dots, m_M$  balls. At each (discrete) time instant, a ball is requested from one of these boxes where the probability of the request from  $\mathcal{B}_i$  is  $p_i$  for some  $0 < p_i < 1$  and  $\sum_{i=1}^M p_i = 1$ . We are concerned with the number of requests  $L$  after which one of these boxes first becomes empty. This abstraction models our scheduling problem of (6) where  $M$  corresponds to the number of blocks in the monitored region,  $\mathbf{m} = [m_1 \ m_2 \ \dots \ m_M]^T$  represents the updated

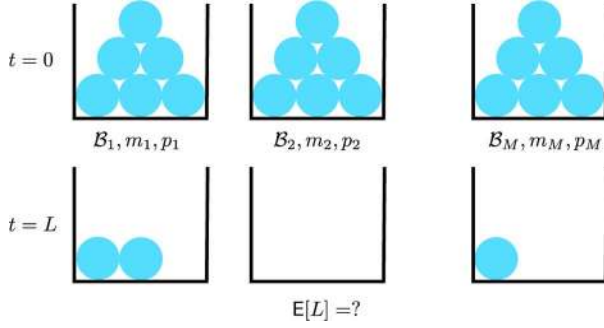


Fig. 3. Abstracted representation of the network lifetime.  $M$  boxes  $\mathcal{B}_i (i \in [M])$  contains  $m_i$  balls respectively. At each request, a ball is taken from  $\mathcal{B}_i$  with probability  $p_i$ . After  $L$  requests, one of these boxes first become empty.  $E[L]$  corresponds to the expected network lifetime.

block-wise coverage energy  $\mathbf{m}_s^{t+1}$  when camera  $\mathcal{C}_s$  is selected,  $\mathbf{p} = [p_1 \ p_2 \ \dots \ p_M]^T$  represent the probabilities with which the blocks are requested (as before) and  $L$  denotes the remaining lifetime. For the camera scheduling problem,  $\mathbf{p}$  may also be dynamically estimated during the network operation. For notational simplicity, we drop the superscript  $t, s$  in our discussion, and write the expectation  $E[L(\mathbf{m}, \mathbf{p})]$  as  $E[L]$ .

For a simplistic scenario where  $M = 2$ , the distribution for  $L$  and the expectation  $E[L]$  are obtained analytically in Proposition 1. In the general case that  $M > 2$ , we present a recursive approach to exactly evaluate  $E[L]$  in Proposition 2. The computational load of this recursive approach is prohibitive as  $M$  increases, motivating us to investigate efficient approximations of  $E[L]$  in Proposition 3 and its asymptotic behavior in Proposition 4. Based upon the asymptotic analysis of  $E[L]$ , we develop our camera scheduling and energy allocation strategies which maximizes the expected network lifetime. The proofs for Propositions 1–4 can be found in Appendices I–IV, respectively.

Let  $\alpha(k; j, b)$  denote the p.m.f of a negative binomial distribution [18] which characterizes the number of failed Bernoulli trials prior to  $j$  successful trials. Specifically,  $\alpha(k; j, b)$  represents the probability that in  $(k+j)$  Bernoulli trials,  $k$  are failures and the other  $j$ , including the last trial, are successful, where the probability of each trial being successful is  $b$ . Using this notation, we obtain the following proposition.

*Proposition 1:* For  $M = 2$ , the p.m.f of  $L$  can be written as (7), shown at the bottom of the page, where  $p$  and  $(1-p)$  denote

the probabilities  $p_1$  and  $p_2$ , respectively. The expectation of  $L$  can be obtained as

$$E[L] = \beta(m_2 - 1; m_1, p) + \beta(m_1 - 1; m_2, 1 - p) \quad (8)$$

where

$$\alpha(k; j, b) = \binom{k+j-1}{k} b^j (1-b)^k, \text{ for } k \geq 0 \quad (9)$$

$$F_\alpha(k; j, b) \stackrel{\text{def}}{=} \sum_{i=0}^k \alpha(i; j, b) \quad (10)$$

$$\beta(k; j, b) = \frac{j F_\alpha(k; j, b) - (k+1) \alpha(k+1; j, b)}{b} \quad (11)$$

□

where  $F_\alpha(k; j, b)$  represents the corresponding cumulative distribution function (c.d.f) of the negative binomial distribution. Also note  $F_\alpha(k; j, b)$  is characterized as a regularized incomplete beta function [19], which along with the p.m.f  $\alpha(k; j, b)$ , is available in standard scientific software packages. Thus,  $E[L]$  can be directly evaluated using (8).

We next consider the case where  $M > 2$  and obtain a recursion for the p.m.f of  $L$  proceeding as follows. We consider a sequence of  $M$  experiments, where in the  $k$ th experiment ( $2 \leq k \leq M$ ) only the first  $k$  boxes  $\{\mathcal{B}_i\}_{i=1}^k$  are utilized in the experiment, with a ball requested from the  $i$ th box with normalized probability  $p'_i = p_i / \sum_{i=1}^k p_i$ . In the  $k$ th experiment, let  $L_k$  denote the number of requests after which one of the boxes  $\{\mathcal{B}_i\}_{i=1}^k$  first becomes empty, then immediately we see  $L = L_M$ . The following proposition allows us to recursively calculate the p.m.f of  $L_k$  ( $2 \leq k \leq M$ ) from the p.m.f of  $L_{k-1}$ .

*Proposition 2:* See equation (12) at the bottom of the page, where  $\mathbf{m}_{[k-1]}$  denotes the vector  $[m_1, m_2, \dots, m_{k-1}]^T$ , and

$$\pi(k) \stackrel{\text{def}}{=} \left( \sum_{j=1}^k m_j - k + 1 \right) \quad (13)$$

represents the maximum possible value of  $L_k$ . □

By observing that  $\Pr(L_1 = j) = \mathbf{I}(j = m_1)$ , it can be verified that (7) is a special case of (12). The expected network lifetime  $E[L] = \sum_{l=\min(\mathbf{m})}^{\pi(M)} l \Pr(L = l)$  can be directly calculated using the p.m.f of  $L$ . However, in order to obtain the p.m.f of

$$\begin{aligned} \Pr(L = l) &= \mathbf{I}(m_1 \leq l \leq (m_1 + m_2 - 1)) \alpha(l - m_1; m_1, p) \\ &\quad + \mathbf{I}(m_2 \leq l \leq (m_1 + m_2 - 1)) \alpha(l - m_2; m_2, 1 - p) \end{aligned} \quad (7)$$

$$\begin{aligned} \Pr(L_k = l) &= \mathbf{I}(\min(\mathbf{m}_{[k-1]}) \leq l \leq \pi(k)) \sum_{j=0}^{m_k-1} \alpha(l - j; j, p'_k) \Pr(L_{k-1} = l - j) \\ &\quad + \mathbf{I}(m_k \leq l \leq \pi(k)) \alpha(l - m_k; m_k, p'_k) \Pr(L_{k-1} > (l - m_k)) \end{aligned} \quad (12)$$

$L$ , the p.m.f of  $\{L_2, L_3, \dots, L_{M-1}\}$  has to be recursively calculated, therefore, the computational load becomes prohibitive as  $M$  increases, especially when the values in the vector  $\mathbf{m}$  are large.

In order to obtain an efficient approximation of  $E[L]$  for the general case  $M > 2$ , we present an alternative representation of  $E[L]$  in terms of the c.d.f of a multinomial distribution. Let  $\Omega(\mathbf{n}; k, \mathbf{b}, j)$  denotes the c.d.f of a multinomial distribution, and  $\omega(\mathbf{n}; k, \mathbf{b}, j)$  denotes the p.m.f. of this multinomial distribution

$$\omega(\mathbf{n}; k, \mathbf{b}, j) = \frac{k!}{n_1! n_2! \dots n_j!} b_1^{n_1} b_2^{n_2} \dots b_j^{n_j}$$

for non-negative integer-valued vector  $\mathbf{n}$  which satisfies  $\sum_{i=1}^j n_i = k$ , where  $j$  represents number of different possible results of a trail,  $k$  is the total number of trails, and  $\mathbf{b} = [b_1 \ b_2 \ \dots \ b_j]$  represents probabilities of each possible result, and  $n_i$  denotes the total number of the  $i$ th result out of  $k$  trials.

*Proposition 3:* The expectation of  $L$  can be represented as

$$E[L] = \sum_{l=\min(\mathbf{m})}^{\pi(M)} \Omega(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}, M) + (\min(\mathbf{m}) - 1) \Omega(\mathbf{m} - \mathbf{1}; \min(\mathbf{m}) - 1, \mathbf{p}, M) \quad (14)$$

where  $\pi(\cdot)$  is defined in (13),  $\mathbf{1}$  denotes an  $M \times 1$  vector, each of whose entries is unity.  $\square$

Equation (14) allows direct evaluation of  $E[L]$  without recursively calculating the p.m.f of  $L$ . Furthermore,  $\Omega(\mathbf{n}; k, \mathbf{b}, j)$  has an accurate approximation which can be relatively efficiently evaluated using the method in [20]. We thus obtain an approximation of  $E[L]$  as

$$\tilde{E}(L) = \sum_{l=\min(\mathbf{m})}^{\pi(M)} \tilde{\Omega}(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}, M) + (\min(\mathbf{m}) - 1) \tilde{\Omega}(\mathbf{m} - \mathbf{1}; \min(\mathbf{m}) - 1, \mathbf{p}, M) \quad (15)$$

where  $\tilde{\Omega}(\cdot)$  denotes approximation of  $\Omega(\cdot)$  as described in [20]. The accuracy of the approximation based upon (15) is considered in Table II, where the results indicate that the proposed approximation of (15) is accurate. The computational advantage of the approximation can be seen via an example for  $M = 20$ ,  $m_i = 50$  for all  $i$ , and uniformly generated  $p_i \in [0, 1]$  ( $1 \leq i \leq M$ ), normalized to have unit sum. In this case, calculating the approximate network lifetime using (15) takes 1.4 s, while obtaining the exact value using the recursion (12) requires 107.7 s.<sup>2</sup>

<sup>2</sup>Both simulations are implemented in Matlab™, and executed on a workstation with an Intel Pentium(R) IV 3.0 GHz CPU and 1 GB memory, other simulations are run on the same platform.

TABLE II

EXPERIMENTAL EVALUATION OF THE APPROXIMATION FOR (14) USING [20]. THREE BOXES CONTAIN  $(m, m, 2m)$  BALLS RESPECTIVELY. THE PROBABILITIES THAT A BALL BEING REQUESTED FROM ONE OF THESE BOXES ARE CORRESPONDINGLY  $(0.25, 0.25, 0.5)$ . *EXACT*: THE EXACT LIFETIME CALCULATED BY EXHAUSTIVELY CALCULATING THE C.D.F OF THE MULTINOMIAL DISTRIBUTION IN (14). *APPROX*: APPROXIMATE LIFETIME USING (15). *SIMULATION*: THE AVERAGE LIFETIME RESULTING FROM 200 MONTE CARLO SIMULATIONS

	Exact	Approx	Simulation
$m = 5$	13.55	13.59	13.33
$m = 10$	30.65	30.54	30.42
$m = 20$	66.59	66.29	66.86
$m = 30$	103.47	103.04	102.91

However, (15) still requires considerable computation, and it is not immediately clear how to develop a camera selection strategy based upon (15). To this end, we proceed to investigate the asymptotic behavior of  $E[L]$  when  $\mathbf{m}$  is sufficiently large.

*Proposition 4:* For  $M = 2$  and  $m_1, m_2$  sufficiently large, (16) holds (see bottom of the page), where  $\mathcal{N}(\cdot)$  denotes the probability density function (p.d.f) of a Gaussian distribution, i.e.,  $\mathcal{N}(k; j, b) \stackrel{\text{def}}{=} (1/\sqrt{2\pi b}) \exp(-(k-j)^2/(2b))$   $\square$

We experimentally evaluate the accuracy of (16) and illustrate the results in Fig. 4. The result is measured by the relative error, which is defined as

$$\epsilon(L) \stackrel{\text{def}}{=} \frac{E[L] - \tilde{E}[L]}{E[L]}$$

where  $E[L]$  is the (approximate) exact value obtained from (15) and  $\tilde{E}[L]$  is the asymptotic approximation using (16). We observe from Fig. 4 that (16) achieves high accuracy, the relative error converges to 0 at an exponential rate as the number of balls increases. In particular, we observe from Fig. 4(b) that in the case  $m_1/p = m_2/(1-p)$ , the refined approximation in (16) achieves high accuracy, however, the simple approximation  $\min(m_1/p, m_2/(1-p))$ , represented by the curve labeled by  $p = 0.5$  in Fig. 4(a), also achieves satisfactory accuracy (1% relative error) for this case where  $m_1/p = m_2/(1-p)$ . This observation leads us to propose the following approximation for the general cases where  $M > 2$

$$E[L(\mathbf{m}, \mathbf{p})] \approx \min\left(\frac{m_1}{p_1}, \frac{m_2}{p_2}, \dots, \frac{m_M}{p_M}\right). \quad (17)$$

Following analysis similar to Appendix IV, it can be seen that approximation using (17) is close and the relative error  $\epsilon(L)$  reduces at an exponential rate as  $\mathbf{m}$  increase, provided  $\mathbf{m}$  is sufficiently large and the difference between the two smallest values in  $\{m_i/p_i, i \in [M]\}$  is not negligible. We proceed to propose and experimentally evaluate camera selection strategies based upon this approximation.

$$E[L] \approx \begin{cases} \min\left(\frac{m_1}{p}, \frac{m_2}{1-p}\right) & \text{if } \frac{m_1}{p} \neq \frac{m_2}{1-p} \\ \frac{m_1}{p} - \frac{(m_1+1)}{p} \mathcal{N}\left(m_1+1; \frac{m_1(1-p)}{p}, \frac{m_1(1-p)}{p^2}\right) - \frac{(m_2+1)}{1-p} \mathcal{N}\left(m_2+1; \frac{m_2 p}{1-p}, \frac{m_2 p}{(1-p)^2}\right) & \text{if } \frac{m_1}{p} = \frac{m_2}{1-p} \end{cases} \quad (16)$$

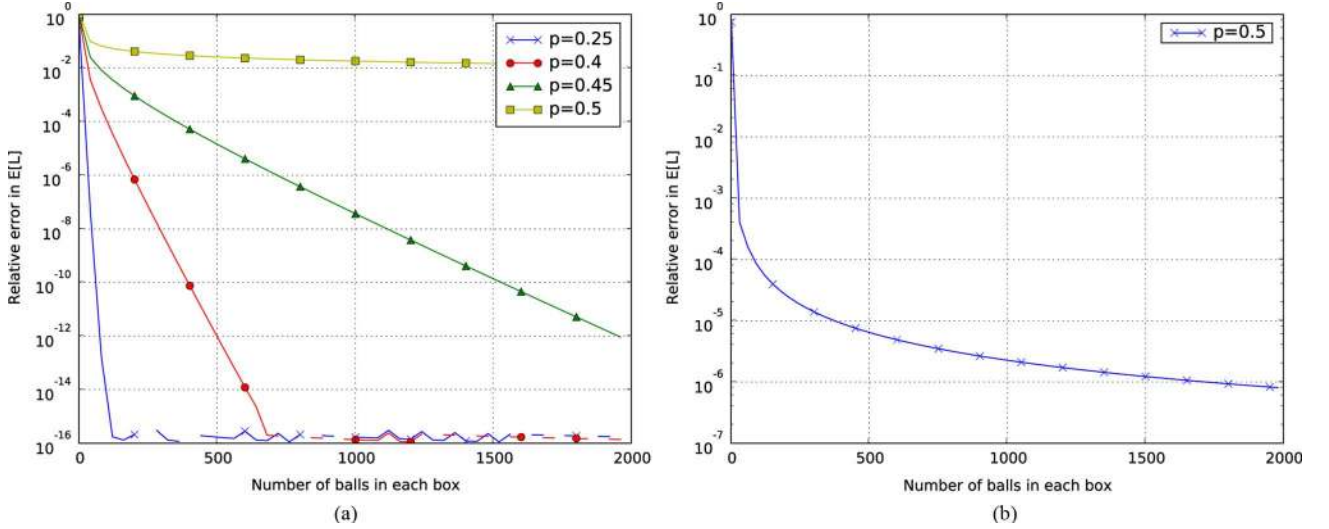


Fig. 4. Experimental evaluation of the approximation for  $E[L]$  using (16). Two boxes  $\mathcal{B}_1, \mathcal{B}_2$  contain the same number of balls  $m_1 = m_2$ , each experiment takes a ball from  $\mathcal{B}_1, \mathcal{B}_2$  with probability  $p$  and  $(1-p)$ , respectively. Abscissa represents the value of  $m_1, m_2$ , ordinate represents the relative error. (a) Simple approximation using  $\min(m_1/p, m_2/(1-p))$ . (b) Refined approximation for the case  $m_1/p = m_2/(1-p)$  using the second expression on the right hand side of (16).

#### IV. CAMERA SELECTION STRATEGY

Given the approximation of expected network lifetime (17), the optimal camera selection strategy when block  $\mathcal{U}_i$  is requested by the user, as previously defined in (6), can be rewritten as

$$s = \arg \max_{j \in \lambda^u(i)} \left\{ \min \left( \frac{m_{j,1}}{p_1}, \frac{m_{j,2}}{p_2}, \dots, \frac{m_{j,M_r}}{p_{M_r}} \right) \right\} \quad (18)$$

where for notational simplicity, we drop the superscript  $t$ , and  $m_{j,k}$  denotes the updated coverage energy at block  $\mathcal{R}_k$  when camera  $\mathcal{C}_j$  is selected. Equation (18) can be intuitively interpreted as maximizing the normalized energy of the *hot-spot* block in the monitored plane. The hot-spot block refers to the block on the target plane  $\mathcal{R}$ , the energy of which divided by the probability that this block is requested, has the minimum value in the (subregion of) monitored area.

In practice, the hot-spot block may not belong to any of the cameras that cover  $\mathcal{U}_i$ , the optimal strategy is consequently modified as follows:

$$s = \arg \max_{j \in \lambda^u(i)} \left\{ \min \left( \frac{m_{j,k}}{p_k}, k \in \kappa^r(j) \right) \right\} \quad (19)$$

where  $\kappa^r(j)$  denotes the set of blocks in the monitored region  $\mathcal{R}$  covered by camera  $\mathcal{C}_j$ , i.e.,  $\kappa^r(j) \stackrel{\text{def}}{=} \{k | \mathbf{B}_{k,j}^r = 1\}$ . Equation (19) indicates that when block  $\mathcal{U}_i$  is requested, we maximize the energy of the hot-spot block in the subregion covered by the set of cameras that covers  $\mathcal{U}_i$ . We refer to the camera selection strategy described previously as *OptCOV*. When the probabilities  $\{p_i\}_{i=1}^{M_r}$  are equal, i.e., the monitored region is uniformly requested, *OptCOV* reduces to the heuristic previously proposed in [17]. Our mathematical framework for analysis of the expected network lifetime, thus, analytically justifies and generalizes the heuristic of [17].

*OptCOV* exhibits several advantages: 1) *OptCOV* represents an optimal strategy based upon a stochastic formulation of the

network lifetime, thus, it is expected to sustain a longer lifetime compared to alternative heuristics. 2) Scheduling based upon (19) only requires an ordering operation which imposes minimal load on the CP and enables real-time applications. 3) User interactions are explicitly addressed in the model, thus, the network can schedule sensors based upon information about the distribution of user requests, which can be estimated if not known as *a priori*. The performance of *OptCOV* is compared against other alternatives in Section VII.

#### V. ENERGY ALLOCATION: A MIN-MAX FORMULATION

The formulation of expected network lifetime also naturally leads to an energy allocation strategy in situations where the total available energy  $w_{tot}$  can be dynamically allocated among the camera sensors at fixed locations at the time of deployment. The optimal energy allocation  $\mathbf{w}^*$  maximizes the expected network lifetime given by (17), and is therefore obtained as the solution of the following max-min optimization problem

$$\begin{aligned} \max_{\mathbf{w}} \quad & \min_i \{f_i\}_{i=1}^{M_r} \\ \text{s.t.} \quad & \sum_{i=1}^N w_i = w_{tot} \\ & \mathbf{f} = \mathbf{P}\mathbf{B}^r \mathbf{w} \\ & w_i \geq 0 \end{aligned} \quad (20)$$

where  $f_i \stackrel{\text{def}}{=} m_i/p_i$  for  $i = 1, 2, \dots, M_r$  and  $\mathbf{f} = [f_1, f_2, \dots, f_{M_r}]$ ,  $\mathbf{P}$  is the diagonal matrix formed by the vector  $[1/p_1, 1/p_2, \dots, 1/p_{M_r}]$ .

It can be seen that (20) corresponds to maximization of a concave function. Although numerical routines are available to directly address the optimization problem (20), it is not well-behaved, partly because the objective function in (20) is not differentiable everywhere. To address this, we next transform the optimization in (20) into a LP problem, for which efficient algorithms can achieve accurate solution in polynomial time

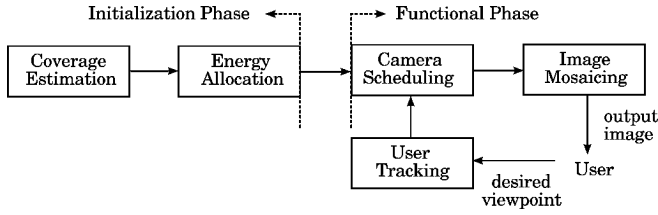


Fig. 5. Visual sensor network operation.

[21]. Introducing a new variable  $t$ , we rewrite the max-min optimization problem of (20) as the equivalent linear programming problem

$$\begin{aligned} \min_{\mathbf{w}} \quad & t \\ \text{s.t.} \quad & \sum_{i=1}^N w_i = w_{tot} \\ & \mathbf{f} = \mathbf{P}\mathbf{B}^r \mathbf{w} \\ & -f_i \leq t, w_i \geq 0, \text{ for } i = 1, 2, \dots, M_r. \end{aligned} \quad (21)$$

The LP (21) can be rewritten in standard LP form by defining  $\mathbf{u} \stackrel{\text{def}}{=} [\mathbf{w}^T, t]^T$ , thus, the objective function can be written as  $t = \mathbf{h}^T \mathbf{u}$  where  $\mathbf{h} \stackrel{\text{def}}{=} [\mathbf{0}^T, 1]^T$  and  $\mathbf{0}$  denotes a  $N \times 1$  vector with value 0. The constraints in (21) can also be rewritten in terms of  $\mathbf{u}$ . The optimal values of  $\mathbf{w}^*$ ,  $t^*$  are obtained simultaneously by solving the LP problem for  $\mathbf{u}^* = [\mathbf{w}^{*T}, t^*]$ .

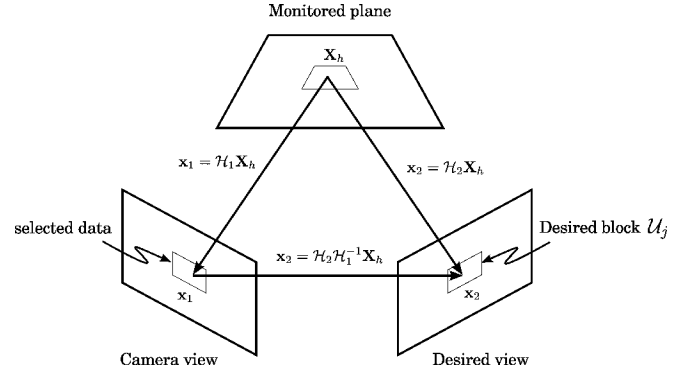
## VI. SYSTEM SCENARIO

To evaluate our algorithms, we simulate the network scenario illustrated in Fig. 1. The operation of this network is illustrated in Fig. 5. In the initialization stage, the coverage information for the image sensors is obtained, which enables computation of the coverage matrices  $\mathbf{B}^u$  and  $\mathbf{B}^r$  defined in Section II. In the functional stage, for each desired view  $\mathcal{U}$  requested by users, the network selects a subset of the cameras to provide the visual data. The synthesized view  $\tilde{\mathcal{U}}$  is generated at the CP using received data.

### A. Initialization Phase

1) *Coverage Estimation*: The coverage information can be obtained after estimating the intrinsic parameters (such as focal length) and extrinsic parameters (orientation and location) of the cameras. We use the plane-based *camera calibration* methods described in [22], [23] to estimate these parameters. Given the calibrated camera geometry, the pixel coordinate  $\mathbf{x}$  on the image focal plane can be obtained for a world point  $\mathbf{X}$ . A world point is covered by a camera if the image coordinate of this point lies within the FoV of the camera. In our implementation, a square block  $\{\mathcal{R}_i\}_{i=1}^{M_r}$  on the target plane (and similarly  $\{\mathcal{U}_i\}_{i=1}^{M_u}$ ) is covered by a camera if the four corners of this block are covered by the camera. The estimated coverage information are represented by the coverage matrices  $\mathbf{B}^r, \mathbf{B}^u$ , which are used for camera scheduling and energy allocation.

2) *Discretization Parameters*: The parameter  $M_u$  determines the number of blocks into which the desired view is discretized. Since a block of the desired view is considered covered by a camera if the camera provides coverage for the entire block, a fine discretization (large  $M_u$ ) ensures that the

Fig. 6. Coordinates  $\mathbf{x}_1$  and  $\mathbf{x}_2$  in first and second camera views, respectively, correspond to a common point  $\mathbf{X}_h$  in the scene. The point  $\mathbf{x}_2$  can be obtained from  $\mathbf{x}_1$  through the homography.

coverage estimation is accurate and adequate flexibility is available in scheduling.  $M_u$  should therefore be chosen so that with high probability each of discretized blocks in the desired view lies within at least a single camera field of view. An increase in  $M_u$ , however, also comes with an increased computational cost so the choice of  $M_u$  represents a tradeoff between accuracy/flexibility and computational complexity.

Once  $M_u$  is determined, the parameter  $M_r$  that determines the discretization of the coverage region can be selected such that the size of a block  $\mathcal{R}_i$  on the coverage region  $\mathcal{R}$  is close to the projection of a block  $\mathcal{U}_j$  on the desired view. This ensures accuracy for the proposed abstraction of Section III where each transmission of a desired view block consumes a similar amount of energy, equivalent to one ball in the abstraction. While a larger value of  $M_r$  may be chosen, this levies a computational penalty while offering little benefit.

3) *Energy Allocation*: Given the estimated coverage matrix  $\mathbf{B}^r$ , the distribution of user requests  $\mathbf{p}$ , and the total available energy, the optimized energy allocation strategy is obtained by solving the LP problem (21).

### B. Functional Phase

1) *Camera Scheduling*: Given the desired viewpoint, the desired view  $\mathcal{U}$  is first discretized into  $M_u$  blocks  $\{\mathcal{U}_i\}_{i=1}^{M_u}$ . For each block  $\mathcal{U}_i$ , the CP selects a camera  $\mathcal{C}_s$  using a camera scheduling strategy to capture and transmit relevant data as described next.

2) *Image Mosaicking*: Only relevant data in the selected camera is transmitted to the CP, which is then mosaicked to synthesize the desired view. The selection of relevant data and mosaicking can be described by a homography [24] for the texture image of a planar surface (our case). We denote by  $\mathbf{X}_h$  the homogeneous representation of a world point on a plane. The image coordinate  $\mathbf{x}$  can be written as:  $\mathbf{x} \sim \mathcal{H}\mathbf{X}_h$ , where  $\sim$  indicates equality up to a scale factor and  $\mathcal{H} \in \mathbb{R}^{3 \times 3}$  is a matrix denoting the homography between the camera plane and the target plane.  $\mathcal{H}$  can be calculated from the parameters of calibrated cameras. As illustrated in Fig. 6, two projections  $\mathbf{x}_1, \mathbf{x}_2$  of world point  $\mathbf{X}_h$  are connected as  $\mathcal{H}_1^{-1}\mathbf{x}_1 = \mathbf{X}_h = \mathcal{H}_2^{-1}\mathbf{x}_2$ , thus,  $\mathbf{x}_2$  can be obtained from  $\mathbf{x}_1$  as  $\mathbf{x}_2 = \mathcal{H}_2\mathcal{H}_1^{-1}\mathbf{x}_1$ . This relation allows the user's view to be rendered from corresponding regions in the selected camera as illustrated in Fig. 6.

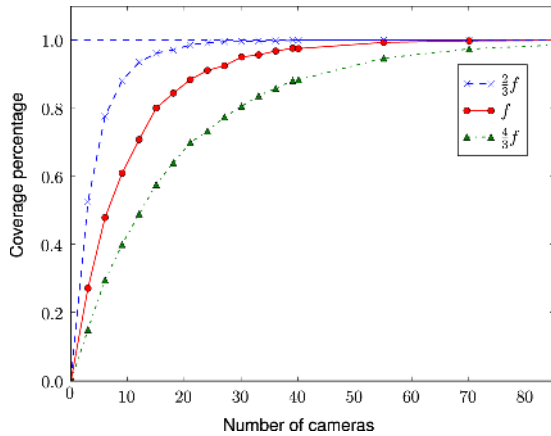


Fig. 7. Average percentage of initial coverage on the target plane vs. the number of cameras for focal lengths  $2/3f$ ,  $f$ ,  $4/3f$ .

## VII. SIMULATIONS

We perform simulations to validate the proposed camera scheduling and energy allocation schemes. In our simulations, the target plane is of size  $4\text{ m} \times 3\text{ m}$  (typical size of a wall) and  $N$  cameras are placed randomly within a  $4\text{ m} \times 3\text{ m}$  field located  $3\text{ m}$  from the target plane, and the cameras are pointed toward the target plane with a random rotation within  $\pm 0.1$  radian along each of the three axes to simulate practical variability in camera placement. Users' viewpoints are generated according to a separable Gaussian distribution with its maxima at the center of the wall on which the cameras are mounted, with a standard deviation of  $0.3\text{ m}$  along the two dimensions. The probabilistic distribution of each block on the monitored plane being requested, i.e.,  $\mathbf{p}$ , is obtained by a Monte-Carlo simulation. All cameras (including the user's viewpoints) are assumed to have images of  $200 \times 200$  (in pixel units), with a focal length  $f = 427.5$  (in pixel units).<sup>3</sup> The simulated camera images are rendered according to the homography. In order to simulate the image capture process in a physical camera, where optical blurring eliminates potential aliasing during sampling, we first generate an upsampled image at this camera according to the scene geometry, blur the image by a Gaussian filter, then downsample to obtain the "camera" image.<sup>4</sup>

We first conduct a Monte Carlo simulation in order to determine the number of cameras required in order to provide adequate coverage of the target plane [25]. Fig. 7 shows the average (over 100 simulations) coverage percentage achieved over the target plane upon initial deployment of the cameras as a function of the number of cameras for different focal lengths. From the figure it can be seen that for the focal length  $f$ , a minimum of 50 cameras are necessary in order to ensure that, on average, 99% of the target plane is covered in the beginning. In order to provide enough coverage redundancy, we therefore use  $N = 100$  cameras for our simulations. We set  $M_r = 400$  and  $M_u = 100$  in our simulations based upon the guidelines in Section VI and justified by simulations that are presented subsequently.

<sup>3</sup>For an image sensor with size  $35\text{ mm} \times 35\text{ mm}$ , this would correspond to a focal length of  $f = 75\text{ mm}$

<sup>4</sup>Based upon empirical evaluation, we select an upsampling/downsampling factor of 4 and the size of the Gaussian filter as  $21 \times 21$  in the upsampled grid.

### A. Camera Scheduling

We simulate a scenario where all cameras start with  $2J$  of energy, which correspond to each camera being able to transmit 2 full frames of images. We generate 200 user viewpoints in each simulation and present averages over 100 simulations.

We consider another two alternative camera scheduling strategies, and compare the three options:

- 1) *OptCOV*: the proposed lifetime-maximizing strategy.
- 2) *View Angle Cost (MinANG)*: From among the cameras providing coverage of a desired block, the camera that is closest in angle to the user-desired viewpoint is selected [25]. This heuristic is intended to approximate camera selection based upon optimizing the image quality since all cameras are at (almost) the same distance from the target plane, image quality is affected primarily by the viewing direction.
- 3) *Random Selection (RAND)*: One camera is randomly selected from the subset of cameras providing coverage for the requested block.

Fig. 8(a) compares the percentage coverage on the target plane over time for the different camera selection methods. The average coverage percentage, across the multiple simulation runs, is plotted in the figures against the network operating time.<sup>5</sup> We observe that the OptCOV camera scheduling strategy maintains a higher percentage of coverage over the monitored region for a longer time duration than alternative scheduling strategies. If the network lifetime is defined as the duration during which 95% of the monitored area is covered, OptCOV prolongs the network lifetime by a factor of 1.5 compared to RAND and by a factor of 2 compared to MinANG. An interesting observation is the sharp drop of coverage at  $t = 130$  for OptCOV. The proposed OptCOV strategy prolongs the network lifetime by allocating the energy consumption evenly (normalized by the requesting probability) across the network. However, OptCOV does not reduce the energy consumption. The total available energy in the camera network is the same for all scheduling strategies. When OptCOV is used, multiple cameras run out of energy around the same time [ $t = 130$  in the simulation of Fig. 8(a)], resulting in a sharp drop of coverage beyond this time. In contrast, MinANG and RAND behave in a more random fashion with respect to energy initialization and therefore fall off more slowly in the coverage.

Fig. 8(b) compares the PSNR of the mosaicked image. Initially, the image quality is similar for the different strategies, and MinANG offers minor improvement because of the selection of the camera with closest viewing direction. By preserving more cameras based upon the anticipated coverage requirement, OptCOV provides high image quality over a longer duration. Fig. 8(a) and (b) clearly demonstrate the advantage of using OptCOV in an energy-constrained scheduling scenario. The tradeoff between the network lifetime and the quality of reconstructed images is investigated further in [25].

Comparing Fig. 8(a) and (b), an interesting observation is that between  $t = 130$  and  $t = 150$ , OptCOV still provides higher image quality. In this duration, although MinANG and RAND provide higher visual coverage, this visual coverage is not optimally distributed according to the distribution of user's requests, and therefore these strategies still offer a lower image quality.

<sup>5</sup>The error bars in these and subsequent figures indicate one standard deviation limits.



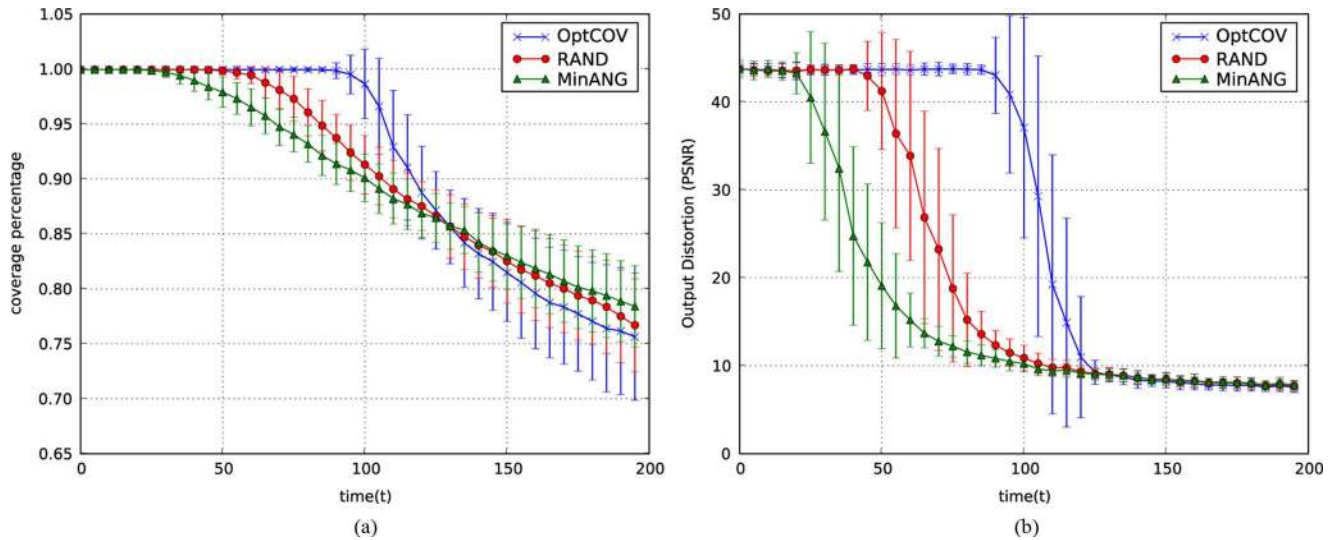


Fig. 8. Comparison of the proposed scheduling method (OptCOV) against random scheduling (RAND) and view angle based scheduling (MinANG) (a) Coverage percentage and (b) Distortion, as functions of time.

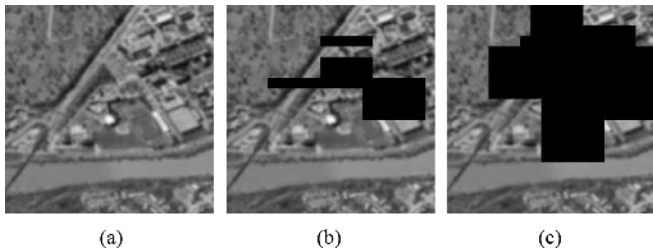


Fig. 9. Snapshots of mosaicked images in one simulation at the moment of  $0.9L_{opt}$ , where  $L_{opt}$  denotes the network lifetime using OptCOV scheduling. (a) OptCOV. (b) RAND. (c) MinANG. Black regions in (b) and (c) correspond to blocks for which coverage is lost.

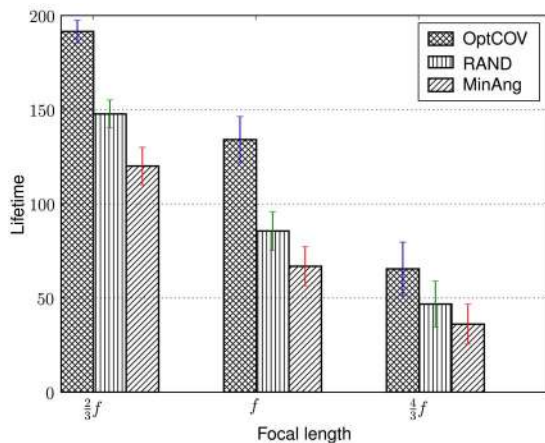


Fig. 10. Comparison of lifetime with different scheduling strategies for different camera focal lengths.

Fig. 9 shows snapshots of the mosaicked output using different camera scheduling strategies for one simulation at a time instant corresponding to 90% of the lifetime for the OptCOV strategy. The snapshots provide visual validation of the results of Fig. 8(a) and (b)—the black regions in Fig. 9(b) and (c) arise from loss of coverage for the RAND and MinANG schemes.

Fig. 10 examines the network lifetime when the focal lengths of the cameras are varied. We consider three focal

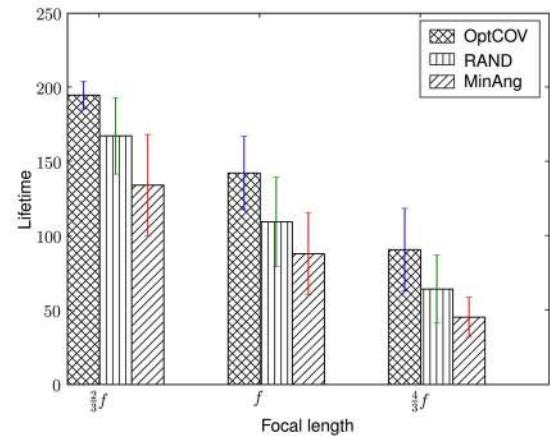


Fig. 11. Comparison of visual coverage for different scheduling strategies when users' viewpoints are generated as a random walk on the camera plane.

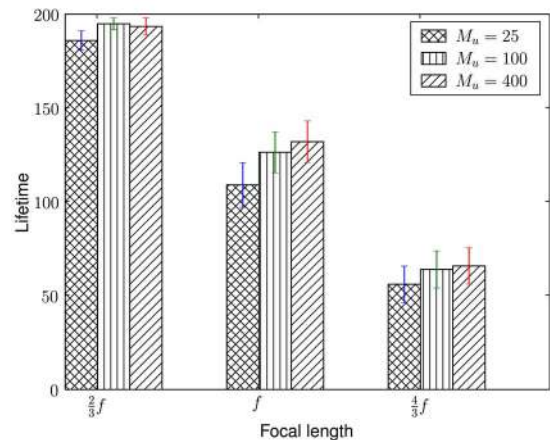


Fig. 12. Comparison of lifetime with different discretization levels ( $M_u$ ) for different camera focal lengths.

lengths  $\frac{2}{3}f$ ,  $f$ , and  $\frac{4}{3}f$ . We observe from Fig. 10 that the network lifetime for all scheduling strategies decreases as the focal length increases, which is expected since larger focal

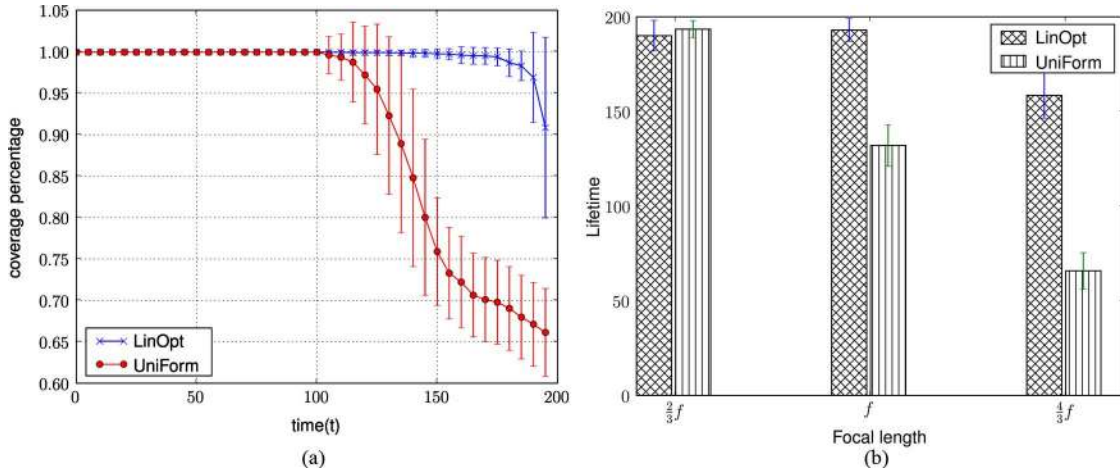


Fig. 13. (a) Comparison of coverage with different energy allocation schemes. (b) Comparison of network lifetime with different energy allocation schemes for three different camera focal length settings.

lengths result in smaller FoV. It can be seen that in two extreme cases: a) where each camera covers the whole monitored area, and b) where the FoVs of the cameras don't overlap at all, all camera scheduling strategies result in the same network lifetime.

We also consider a simulation scenario where the users' viewpoints are generated by a random walk on a  $16 \times 16$  grid in the plane of the cameras starting at a random grid node. Subsequent viewpoints are chosen from the neighboring 8 grid points and the current position (9 choices in total) with equal probability. This is intended to approximate the process of tracking a moving object in the covered region. The stationary distribution for this random walk [26] provides the probabilities for visiting each node. These are uniform over the internal nodes and lower for the peripheral nodes. The conclusions based upon Fig. 11 remain similar to those for Fig. 8, the previous simulation.

We next consider the impact of the discretization by examining how the choice of  $M_u$  impacts the lifetime. In Fig. 12, we compare the network lifetime when three different values of  $M_u$ : 25, 100 and 400. Results show that small  $M_u$  causes loss of performance due to coarser approximation of the visual coverage region, and values of  $M_u$  over 100 provides only marginal improvement. Therefore we use  $M_u = 100$  and a corresponding value of  $M_r = 400$  which approximately ensures the size of a block  $\mathcal{R}_i$  on the coverage region  $\mathcal{R}$  is close to the projection of a block  $\mathcal{U}_j$  on the desired view.

### B. Energy Allocation

In this simulation, we apply the OptCOV camera scheduling strategy and compare two energy allocation schemes:

- Optimized allocation by solving (21) using LP (LinOpt).
- Uniform distribution of the total available energy  $w_{tot}$  among all the cameras (UniForm).

Fig. 13(a) compares the percentage coverage on the target plane over time for the energy allocation schemes mentioned previously. The figure clearly illustrates the benefit of the proposed optimized energy allocation. LinOpt prolongs the network lifetime by almost 50% compared to the unoptimized Uni-

Form scheme. Direct solution of the max-min problem (20) yields performance very close to LinOpt (as expected) but require much more computation (2.3 s v.s. 157 s for Matlab™ based implementations).

For both energy allocation schemes, Fig. 13(b) examines the effect of focal length on the network lifetime. We observe from Fig. 13(b) that energy allocation using LinOpt is more effective as the focal length increases. When the focal length is small, LinOpt and UniForm have similar performance. This behavior can be anticipated because as the focal length decreases, the fields of view of the cameras become enlarged resulting in greater overlap and consequently less benefit is obtained from the optimal allocation. In the extreme case, when each camera covers the entire target area, the lifetime is independent of energy allocation.

Recalling Fig. 10, we observe the following *duality* between camera scheduling and energy allocation. When the focal length is relatively small, i.e.,  $2/3f$ , optimized camera scheduling is more effective in prolonging the network lifetime, however, as the focal length increases, optimized energy allocation prolongs network lifetime more significantly. When the cameras have no overlap, only energy allocation is effective in prolonging network lifetime.

## VIII. CONCLUSION AND DISCUSSION

We provide a stochastic model for the operational lifetime of a visual sensor network (VSN) and obtain exact and approximate expressions for the expected network lifetime via a suitable abstraction. Computationally efficient sensor scheduling and energy allocation strategies targeted toward maximizing the expected lifetime can be developed using the analysis. These strategies account for the visual coverage geometry and the stochastic nature of data requests, and thereby offer significant improvements in lifetime compared with alternative schemes.

We conclude this paper with several remarks. First, although we describe a simple application scenario, the analysis can be extended to other VSN applications, e.g., a surveillance system designed to capture moving objects. In this scenario, the cov-

erage information  $\mathbf{B}^r, \mathbf{B}^u$  can be estimated by suitably discretizing the monitored area (e.g., 3-D space can be divided into small cubes). The rest of the analysis is identical with that presented in this paper.

Second, our optimal scheduling and energy allocation strategies are based upon an approximate analysis, where we ignored the fact that the image blocks are not being requested independently due to the overlap of camera FoVs and the user's behaviors, however, the compromise of optimality is not severe in practice. The camera scheduling strategy we propose is performed at each time instant with fresh parameters, thereby preventing the propagation of suboptimality. For the energy allocation, the suboptimality is also mitigated when the focal length is reasonably large as demonstrated by simulations.

We consider strategies to prolong the visual coverage of VSNs in this paper. However, image quality is another aspect of the usability of VSNs that merits further study in follow-on work. For our simulations, we consider scenarios where the focal lengths of the cameras are the same. This ensures that all cameras provide similar visual quality and therefore scheduling may be performed purely based upon coverage lifetime considerations. If the cameras have heterogeneous focal lengths, further modeling of the image quality is required in order to design scheduling and energy allocation strategies, which considers the communication constraints and the geometric transformations required for the image data.

In the present paper, we do not address the problem of camera placement, assuming instead that the cameras are placed randomly and are not manipulable. Situations where the placement is manipulable have been previously considered in the context of generic WSNs [27] and VSNs [28], [29]. Due to the large number of degrees of freedom for camera placement, i.e., location and orientation, various restrictions are applied to make the problem tractable. In [28], the cameras are aligned horizontally and in [29], the cameras are in the same plane and all other parameters are discrete. The stochastic lifetime model that we propose in this paper may also be extended to incorporate camera placement and the joint optimization of placement and energy allocation. This, however, is quite challenging (particularly because many local optima can be expected in camera placement parameters) and is beyond the scope of this paper.

#### APPENDIX I

##### PROOF OF PROPOSITION 1

For  $M = 2$ , we can write

$$\Pr(L = l) = \Pr(L = l, s^l = 1) + \Pr(L = l, s^l = 2) \quad (22)$$

where  $s^l$  represents the index of box from which the  $l$ th ball is selected, thus,  $\Pr(L = l, s^l = 1)$  represents the probability that  $\mathcal{B}_1$  becomes empty at the  $l$ th request. Therefore, in the first  $(l - 1)$  requests,  $(m_1 - 1)$  balls have been taken from  $\mathcal{B}_1$  and  $(l - m_1)$  taken from  $\mathcal{B}_2$ , we obtain (23), shown at the bottom of the page, where (23) follows from the definition of  $\alpha(\cdot)$  in (9). We can similarly represent  $\Pr(L = l, s^l = 2)$  and obtain (7).

Let  $\beta(k; j, b) \stackrel{\text{def}}{=} \sum_{l=j}^{k+j} l\alpha(l - j; j, b)$ , we can write

$$\begin{aligned} E(L) &= \sum_{l=m_1}^{m_1+m_2-1} l\alpha(l - m_1; m_1, p) \\ &\quad + \sum_{l=m_2}^{m_1+m_2-1} l\alpha(l - m_2; m_2, 1 - p) \\ &= \beta(m_2 - 1; m_1, p) + \beta(m_1 - 1; m_2, 1 - p). \end{aligned} \quad (24)$$

Recalling  $F_\alpha(k; j, b) \stackrel{\text{def}}{=} \sum_{i=0}^k \alpha(i; j, b)$ , we next rewrite  $\beta(k; j, b)$

$$\beta(k; j, b) = \sum_{i=0}^k i\alpha(i; j, b) + jF_\alpha(k; j, b). \quad (25)$$

Denoting, in short,  $q_k \stackrel{\text{def}}{=} \alpha(k; j, b)$ , from (9) we can obtain  $q_{k+1}/q_k = (j + k)/(k + 1)(1 - b)$ , which yields

$$(k + 1)q_{k+1} = kq_k(1 - b) + q_k(1 - b)j. \quad (26)$$

Replacing  $k$  by  $i$  for  $i \in \{0, 1, 2, \dots, k - 1\}$ , we obtain  $k$  in the same form as (26). Stacking together these  $k$  equations and reorganizing the resulting terms, we obtain the expression for  $\sum_{i=0}^k i q_i$  and

$$\beta(k; j, b) = \frac{jF_\alpha(k; j, b) - (k + 1)\alpha(k + 1; j, b)}{b} \quad (27)$$

which completes the proof.

#### APPENDIX II

##### PROOF OF PROPOSITION 2

We first write

$$\Pr(L_k = l) = \Pr(L_k = l, s^l \in [k - 1]) + \Pr(L_k = l, s^l = k) \quad (28)$$

the first term on the RHS can be written as (29)–(32), shown at the bottom of the next page, where  $n_i^l$  denotes the number of

$$\begin{aligned} \Pr(L = l, s^l = 1) &= \mathbf{I}(m_1 \leq l \leq (m_1 + m_2 - 1)) \binom{l - 1}{m_1 - 1} p^{m_1} (1 - p)^{l - m_1} \\ &= \mathbf{I}(m_1 \leq l \leq (m_1 + m_2 - 1)) \alpha(l - m_1; m_1, p) \end{aligned} \quad (23)$$

balls taken from the  $t$ th box in the first  $l$  requests. Equation (31) is obtained from the Bayes rule, (32) is obtained by observing that the conditional probability in (31) conforms to a negative binomial distribution. Similarly, we obtain (33) and (34), shown at the bottom of the page.

The proof is completed by substituting (32) and (34) into (28), and incorporating the range of valid values of  $l$ . The term  $\pi(k)$  is defined in (13), and denotes the maximum possible lifetime for the  $k$ th experiment, which occurs when all boxes  $\{\mathcal{B}_i\}_{i=1}^k$  have one ball remaining before the  $\pi(k)$ th request.

### APPENDIX III

#### PROOF OF PROPOSITION 3

We first observe the following property for a r.v.  $X$  that takes on non-negative integer values in the closed interval  $[a, b]$ :

$$\mathbb{E}[X] = \sum_{i=a}^b \Pr(X \geq i) + (a-1)\Pr(X \geq a) \quad (35)$$

Equation (35) can be obtained by using the expansion  $\Pr(X \geq i) = \sum_{j=i}^b \Pr(X = j)$  for each  $i \in \{a, a+1, \dots, b\}$ , and reorganizing the resulting terms in (35). Using (35) for our r.v.  $L$ , we obtain

$$\mathbb{E}[L] = \sum_{l=\min(\mathbf{m})}^{\pi(M)} \Pr(L \geq l) + (\min(\mathbf{m}) - 1)\Pr(L \geq \min(\mathbf{m}) - 1). \quad (36)$$

We next note  $\Pr(L \geq l)$  can be alternatively represented as  $\Pr(n_i^{l-1} \leq m_i - 1, i \in [M])$ , indicating that after  $(l-1)$  requests, there is at least one ball left in each box. We can write

$$\Pr(L \geq l) = \Omega_M(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}). \quad (37)$$

Substituting (37) into (36), we obtain (14).

### APPENDIX IV

#### PROOF OF PROPOSITION 4

It is well known that a negative binomial distributed r.v. can be viewed as the sum of a sequence of i.i.d geometrically distributed r.v.s, i.e.,  $X = \sum_{i=1}^j Y_i$ , where  $X$  has a p.m.f  $\Pr(X = x) = \alpha(x; j, b)$  and each  $Y_i$  is geometric with a p.m.f  $\Pr(Y_i = k) = (1-b)^{k-1}b \stackrel{\text{def}}{=} \eta(k; b)$  for  $i \in [j]$ . The term  $\eta(k; b)$  denotes the probability that  $k$  failures happen before one success in  $(k+1)$  Bernoulli trials where  $b$  is the probability of success in each trial. The mean and variance of  $Y_i$  are  $(1-b)/b$  and  $(1-b)/b^2$ , respectively. According to the central limit theorem [30], the summation of  $j$  i.i.d (geometrically in our case) distributed r.v.s conforms to the Gaussian distribution when  $j$  is sufficiently large we, thus, obtain

$$\alpha(x; j, b) \approx \mathcal{N}\left(x; \frac{j(1-b)}{b}, \frac{j(1-b)}{b^2}\right). \quad (38)$$

$$\Pr(L_k = l, s^l \in [k-1]) = \sum_{j=0}^{m_k-1} \Pr(n_k^{l-1} = j, s^l \in [k-1], L_k = l) \quad (29)$$

$$= \sum_{j=0}^{m_k-1} \Pr\left(n_k^{l-1} = j, \sum_{i=1}^{k-1} n_i^{l-1} = l - j - 1, s^l \in [k-1], L_{k-1} = l - j\right) \quad (30)$$

$$= \sum_{j=0}^{m_k-1} \Pr\left(n_k^{l-1} = j, \sum_{i=1}^{k-1} n_i^{l-1} = l - j - 1, s^l \in [k-1] \middle| L_{k-1} = l - j\right) \times \Pr(L_{k-1} = l - j) \quad (31)$$

$$= \sum_{j=0}^{m_k-1} \binom{l-1}{j} (p'_k)^j (1-p'_k)^{l-j} \Pr(L_{k-1} = l - j) \quad (32)$$

$$\Pr(L_k = l, s^l = k) = \Pr(n_k^{l-1} = m_k - 1, s^l = k, L_{k-1} > l - m_k) \quad (33)$$

$$= \Pr(n_k^{l-1} = m_k - 1, s^l = k | L_{k-1} > l - m_k) \Pr(L_{k-1} > l - m_k) = \binom{l-1}{l-m_k} (p'_k)^{m_k} (1-p'_k)^{l-m_k} \Pr(L_{k-1} > l - m_k). \quad (34)$$

$$\begin{aligned} \Pr(s^t = 1) &= \sum_{k=0}^{m_2-1} \Pr(n_1^{k+m_1-1} = m_1 - 1, n_2^{k+m_1-1} = k) \Pr(s^{k+m_1} = 1) \\ &= \sum_{k=0}^{m_2-1} \alpha(k; m_1, p). \end{aligned} \quad (40)$$

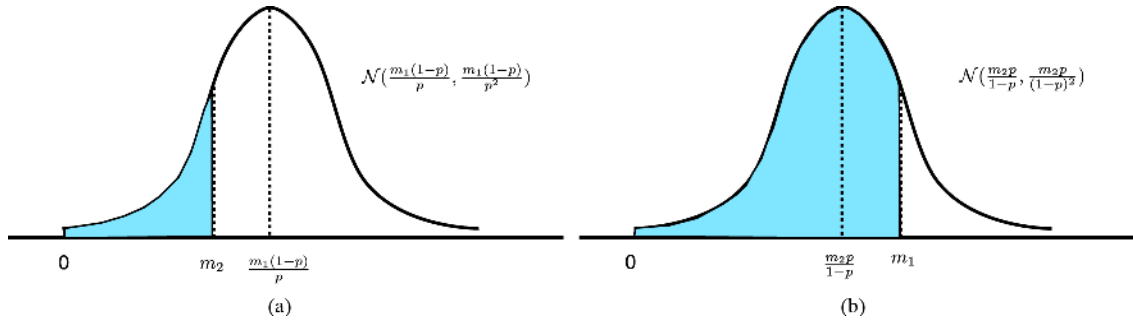


Fig. 14. (a)  $\Pr(s^t = 1) = \sum_{k=0}^{m_2-1} \alpha(k; m_1, p)$  is approximated by the area of the shaded region, where  $\alpha(k; m_1, p)$  is approximated by the integral from  $k$  to  $k + 1$ . (b)  $\Pr(s^t = 2)$  is similarly approximated.

$$\beta(m_2 - 1; m_1, p) = \frac{m_1 F_\alpha(m_2 - 1; m_1, p) - (m_1 + 1)\alpha(m_1 + 1; m_1, p)}{p} \quad (45)$$

Equation (38) can be alternatively represented in the standard form  $b/\sqrt{j(1-b)}(X - j(1-b)/b) \sim \mathcal{N}(0, 1)$ . We next write  $E[L]$  as

$$E(L) = E(L|s^t = 1)\Pr(s^t = 1) + E(L|s^t = 2)\Pr(s^t = 2) \quad (39)$$

where  $s^t$  indicates the index of the box from which the last ball is taken. It can be shown that (40) holds (see bottom of the previous page). We next represent  $\Pr(s^t = 1)$  by an integral of the Gaussian p.d.f using the approximation (38) as illustrated in Fig. 14.

In the case where  $m_1/p > m_2/(1-p)$ , we obtain

$$\Pr(s^t = 1) \approx \int_{-\infty}^{m_2} \mathcal{N}\left(k; \frac{m_1(1-p)}{p}, \frac{m_1(1-p)}{p^2}\right) dk \quad (41)$$

$$= 1 - Q\left(\frac{m_2 p - m_1(1-p)}{\sqrt{m_1(1-p)}}\right) \quad (42)$$

$$= Q\left(\frac{m_1(1-p) - m_2 p}{\sqrt{m_1(1-p)}}\right) \quad (43)$$

$$\leq \frac{1}{2} \exp\left(-\frac{m_1(1-p) - m_2 p}{2\sqrt{m_1(1-p)}}\right)^2 \quad (44)$$

where (42) is obtained by using the  $Q$ -function  $Q(x) = 1/\sqrt{2\pi} \int_x^\infty e^{-1/2x^2}$ , (43) uses the property  $Q(-x) = 1 - Q(x)$ , and (44) is obtained by utilizing the up-bound of  $Q$ -function that  $Q(x) \leq 1/2e^{-x^2/2}$  for  $x \geq 0$ . We observe from (44) that  $\Pr(s^t = 1) \rightarrow 0$  provided  $m_1, m_2$  are sufficiently large and  $m_1/p > m_2/(1-p)$  as we consider. As a result, (39) can be simplified as  $E(L) \approx E(L|s^t = 2)$ . Recall that  $\Pr(L = l|s^t = 2) = \alpha(l - m_2; m_2, 1-p)$ , thus,  $(L - m_2)$  has a negative binomial distribution with expectation  $m_2 p/(1-p)$ , and it follows that  $E[L] \approx m_2/(1-p)$ . Considering the symmetric case  $m_1/p < m_2/(1-p)$ , we obtain in general  $E[L] \approx \min(m_1/p, m_2/(1-p))$  when  $m_1/p \neq m_2/(1-p)$ .

In the case that  $m_1/p = m_2/(1-p)$ , we rewrite  $E(L) = \beta(m_2 - 1; m_1, p) + \beta(m_1 - 1; m_2, (1-p))$ , where

$\beta(\cdot)$  is defined in (27). Next consider (45), shown at the top of the page, we observe  $F_\alpha(m_2 - 1; m_1, p) = \Pr(s^t = 1) \approx 0.5$  when  $m_1/p = m_2/(1-p)$  and  $m_1, m_2$  are sufficiently large, therefore  $\beta(m_2 - 1; m_1, p) \approx 0.5m_1/p - ((m_1 + 1)/p)\mathcal{N}(m_1 + 1; m_1(1-p)/p, m_1(1-p)/p^2)$ . Following similar analysis for  $\beta(m_1 - 1; m_2, (1-p))$ , we obtain (16).

#### ACKNOWLEDGMENT

The authors would like to thank Prof. W. Heinzelman for useful discussions and the anonymous reviewers for several valuable comments.

#### REFERENCES

- [1] C. Yu and G. Sharma, M. Rabbani and R. L. Stevenson, Eds., "Sensor scheduling for lifetime maximization in user-centric image sensor networks," in *Proc. SPIE: Visual Communications and Image Processing (VCIP)*, Jan. 2009, vol. 7257, p. 7257-0H-1-12.
- [2] C. Yu and G. Sharma, "Optimized energy allocation in battery powered image sensor networks," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 2009, pp. 3461-3464.
- [3] I. Akyildiz, T. Melodia, and K. Chowdhury, "A survey on wireless multimedia sensor networks," *Comput. Netw.*, vol. 51, no. 4, pp. 921-960, 2007.
- [4] A. Keshavarz, A. Tabar, and H. Aghajan, "Distributed vision-based reasoning for smart home care," in *Proc. ACM SenSys Workshop on Distributed Smart Cameras*, 2006.
- [5] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: A multi-tier camera sensor network," *Proc. 13th ACM Multimedia*, pp. 229-238, 2005.
- [6] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Adv. Multimedia*, 2009, 640386.
- [7] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade, "Algorithms for cooperative multisensor surveillance," *Proc. IEEE*, vol. 89, no. 10, pp. 1456-1477, Oct. 2001.
- [8] Y. Tian, L. Brown, A. Hampapur, M. Lu, A. Senior, and C. Shu, "IBM smart surveillance system (S3): Event based video surveillance system with an open and extensible framework," *Mach. Vis. Appl.*, vol. 19, no. 5, pp. 315-327, 2008.
- [9] B. Song and A. Roy-Chowdhury, "Stochastic adaptive tracking in a camera network," in *Proc. Int. Conf. Computer Vision*, 2007, pp. 1-8.
- [10] W. Lin, M. Sun, R. Poovandran, and Z. Zhang, "Human activity recognition for video surveillance," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1128-1139, Aug. 2008.
- [11] Z. Yang, W. Wu, K. Nahrstedt, G. Kurillo, and R. Bajcsy, "Viewcast: View dissemination and management for multi-party 3d tele-immersive environments," in *Proc. ACM 15th Int. Conf. Multimedia*, 2007, pp. 882-891.

- [12] H. Zhang and J. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," in *Proc. Ad Hoc & Sensor Wireless Networks*, 2005, vol. 1, no. 1–2, pp. 89–123.
- [13] C. Huang and Y. Tseng, "The coverage problem in a wireless sensor network," *Mobile Netw. Appl.*, vol. 10, no. 4, pp. 519–528, 2005.
- [14] Y. Chen and Q. Zhao, "On the lifetime of wireless sensor networks," *IEEE Commun. Lett.*, vol. 9, no. 11, pp. 976–978, Nov. 2005.
- [15] M. Cardei and J. Wu, "Energy-efficient coverage problems in wireless ad-hoc sensor networks," *Comput. Commun.*, vol. 29, no. 4, pp. 413–420, 2006.
- [16] J. Dagher, M. Marcellin, and M. Neifeld, "A method for coordinating the distributed transmission of imagery," *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 1705–1717, Jul. 2006.
- [17] S. Soro and W. Heinzelman, "On the coverage problem in video-based wireless sensor networks," in *Proc. 2nd Workshop on Broadband Advanced Sensor Networks (BaseNets '05)*, 2005, pp. 932–939.
- [18] J. M. Hilbe, *Negative Binomial Regression*. Cambridge, U.K.: Cambridge Univ. Press, 2007.
- [19] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992.
- [20] B. Levin, "A representation for multinomial cumulative distribution functions," *Ann. Statist.*, vol. 9, no. 5, pp. 1123–1126, 1981.
- [21] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1989.
- [22] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. IEEE Int. Conf. Computer Vision*, 1999, pp. 666–673.
- [23] C. Yu and G. Sharma, J. G. Apostolopoulos and A. Said, Eds., "Plane-based calibration of cameras with zoom variation," in *Proc. SPIE: Visual Communications and Image Processing*, Jan. 15–19, 2006, vol. 6077, pp. 607710-1–607710-9.
- [24] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
- [25] C. Yu, S. Soro, G. Sharma, and W. Heinzelman, "Lifetime-distortion trade-off in image sensor networks," in *Proc. IEEE Int. Conf. Image Proc.*, Sep. 2007, vol. V, pp. 129–132.
- [26] T. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ: Wiley, 2006.
- [27] Y. Wang, C. Hu, and Y. Tseng, "Efficient placement and dispatch of sensors in a wireless sensor network," *IEEE Trans. Mobile Comput.*, vol. 7, no. 2, pp. 262–274, 2008.
- [28] A. Ercan, D. Yang, A. El Gamal, and L. Guibas, "Optimal placement and selection of camera network nodes for target localization," *Lecture Notes Comput. Sci.*, vol. 4026, p. 389, 2006.
- [29] J. Zhao, S. Cheung, and T. Nguyen, "Optimal camera network configurations for visual tagging," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 4, pp. 464–479, Aug. 2008.
- [30] A. Papoulis and S. U. Pillai, *Probability Random Variables and Stochastic Processes*, 4th ed. New York: McGraw-Hill, 2002.



**Chao Yu** (S'05) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China in 2004, the M.S. degree in electrical and computer engineering from University of Rochester, Rochester, NY, in 2005, and is currently pursuing the Ph.D. degree in electrical and computer engineering from the University of Rochester.

His research interests lie in the area of signal and image/video processing, and specifically in distributed signal processing, distributed image/video compression and processing, and resource allocation

in (visual) sensor networks.

Mr. Yu is a recipient of the best paper awards at the SPIE Visual Communications and Image Processing (VCIP) conference, San Jose, CA, 2009.



**Gaurav Sharma** (SM'00) received the B.E. degree in electronics and communication engineering from Indian Institute of Technology Roorkee (formerly University of Roorkee), India, in 1990; the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bangalore, India, in 1992; and the M.S. degree in applied mathematics and the Ph.D. degree in electrical and computer engineering from North Carolina State University, Raleigh, in 1995 and 1996, respectively.

From August 1992 through August 1996, he was a research assistant at the Center for Advanced Computing and Communications, ECE Department, NCSU. From August 1996 through August 2003, he was with Xerox Research and Technology, Webster, NY, initially as a member of research staff and subsequently at the position of principal scientist. Since Fall 2003, he is an Associate Professor at the University of Rochester in the Department of Electrical and Computer Engineering and in the Department of Biostatistics and Computational Biology. He is also the Director for the Center for Electronic Imaging Systems (CEIS), a New York state funded center for promoting joint university-industry research and technology development, which is housed at the University of Rochester. His research interests include multimedia security and watermarking, color science and imaging, genomic signal processing, and image processing for visual sensor networks. He is the editor of the *Color Imaging Handbook* (CRC, 2003).

Dr. Sharma is a member of Sigma Xi, Phi Kappa Phi, Pi Mu Epsilon, IS&T, and the signal processing and communications societies of the IEEE. He was the 2007 chair for the Rochester section of the IEEE and served as the 2003 chair for the Rochester chapter of the IEEE Signal Processing Society. He currently serves as the vice chair for the IEEE Signal Processing Society's Image and multidimensional signal processing (IMDSP) technical committee and is an advisory member of the IEEE Standing committee on Industry DSP. He currently serves as an associate editor the *Journal of Electronic Imaging* and in the past has served as an associate editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING and the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.