Sensor Scheduling For Lifetime Maximization in User-Centric Image Sensor Networks

Chao Yu, Gaurav Sharma
ECE Dept, University of Rochester,
Rochester NY 14627

ABSTRACT

We explore sensor scheduling strategies to maximize the operational lifetime of a user-centric image sensor network. Image sensors are deployed for gathering visual information over a monitored region. Users navigate within this region by specifying a time-varing desired viewpoint and the network responds with the requested visual data. By modeling the user's desired viewpoint in a probabilistic framework, we develop a stochastic formulation of the network lifetime and investigate the camera scheduling strategy that maximizes the expected value of network lifetime. By suitably abstracting the problem, we present a closed-form solution for the simplistic case when the monitored region is divided into two parts. Using asymptotic analysis, we then present a simple camera scheduling strategy for the general case that we conjecture to be optimal. Simulation results demonstrate a clear advantage of the proposed camera scheduling approach over previously considered alternatives.

1. INTRODUCTION

Image sensor networks have recently evoked intense research interest due to the increasing demand for applications such as security surveillance and disaster monitoring [1–3]. These sensor networks consist of portable wireless sensors with sensing and communication capabilities. Because the sensors are usually battery powered, power consumption imposes a critical constraint on the usability of these sensor networks along with other limiting factors such as memory and computational capabilities. We consider a *user-centric* application scenario as illustrated in Fig. 1, where image sensors are deployed to provide visual coverage over a monitored region. The network allows the user to navigate around the monitored region by specifying 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. In a densely deployed sensor network, the desired view overlaps with the fields of view (FoVs) of a number of cameras and one may select between the cameras providing coverage for a given region. We investigate camera selection strategies with a view to maximizing the lifetime of the network. Although described in the context of this specific application scenario, our analysis is general and adaptable to a broader class of sensor scheduling and general resource allocation problems.

User-centric sensor networks, such as the application scenario we consider, only collect and transmit data requested by users thus prolong the operational lifetime of the network. However, due to the stochastic nature of the user's request, an optimal strategy for resource allocation problems such as power allocation, sensor deployment and sensor scheduling in the network impose extra challenge compared to most of previous work that consider sensor networks deployed for deterministic tasks, e.g. sensor networks that collect complete data over a monitored area [4]. In related work [5], the authors address the problem of optimal assignment of cameras to monitor sub-regions of a monitored area in order to maximize the lifetime of the camera network, which provides full coverage of this monitored area. However, user interaction is not considered in the formulation. To account for user interactions, [6, 7] propose a heuristic approach by defining a *coverage cost* associated with each camera depending on the remaining energy of the camera and the coverage geometry. In this work, we provide a stochastic formulation for the sensor scheduling problem based on a probabilistic modeling of users' requests, and develop an optimal strategy to maximize the expected network lifetime.

This work is supported in part by the National Science Foundation under grant number ECS-0428157. Further author information: (Send correspondence to Chao Yu): E-mail: chyu@ece.rochester.edu, Telephone: 1 585 275 8122, Address: Electrical and Computer Engineering Department, University of Rochester, Rochester, NY, 14627-0126, USA, WWW: www.ece.rochester.edu/projects/iplab

Visual Communications and Image Processing 2009, edited by Majid Rabbani, Robert L. Stevenson, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 7257, 72570H · © 2009 SPIE-IS&T · CCC code: 0277-786X/08/$18 · doi: 10.1117/12.807958

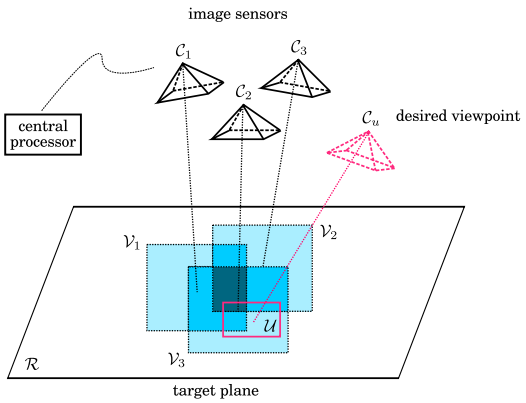


Figure 1. The target plane $\mathcal{R}$ is monitored by an image sensor network consists of cameras $\{\mathcal{C}\}_{i=1}^{N}$. $\mathcal{C}_i$ covers a sub-region $\mathcal{V}_i$ of the target plane. A central processor (CP) keeps record of the energy distribution, coverage geometry of the network, and receives the user's request. The CP selects a subset of cameras $\{\mathcal{C}\}_{i=1}^{N}$ to provide data and use received data to synthesize the user's desired view $\mathcal{U}$. Cameras are not allowed to communication with each other. The intensity of sub-regions indicates coverage information: regions that covered by more cameras appear darker.

In Section 2, we model users' requests as a random variable (r.v.) with a distribution that is either known as a priori or estimated from the record of prior requests, and formulate the network lifetime as a r.v. that depends on the current energy distribution in the network and the distribution of users' request. The optimal sensor (camera) scheduling strategy is defined as the one that maximizes the expected value of remaining network lifetime. In Section 3, we provide an abstraction of the sensor scheduling problem that allows us to analytically formulate the sensor lifetime in a more mathematically tractable fashion. We present exact, closed-form results for a simplistic scenario where the monitored area is divided into two parts and approximate results for general scenarios. The asymptotic analysis for this abstracted problem lead to a simple camera selection strategy. We describe detailed implementation of the application scenario we consider in Section 4 and present simulation setup and results in Section 5. Concluding remarks are given in Section 6.

2. PROBLEM FORMULATION

In this section, we formulate the camera selection problem and define the optimal strategy to maximize network lifetime.

The target plane $\mathcal{R}$ is monitored by N Cameras $\{\mathcal{C}_i\}_{i=1}^{N}$. Each camera $\mathcal{C}_i$ is batteried with energy w_i and covers a sub-region of $\mathcal{R}$ denoted by $\mathcal{V}_i$. We uniformly divide $\mathcal{R}$ into M_r blocks $\{\mathcal{R}_j, j \in [M_r]\}$, where $[a]$ represent the set $\{1, 2, \ldots, a\}$. We represent the coverage geometry of the cameras in terms of this discretized representation, and define $\mathbf{B}^r \in \mathbb{R}^{N \times M_r}$, where

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

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:

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

and denote the subset of cameras that cover block $\mathcal{R}_j$ by

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

The user specifies a desired viewpoint and the desired view $\mathcal{U}$ is also uniformly divided into M_u blocks $\{\mathcal{U}_j, j \in [M_u]\}$. The coverage geometry of $\mathcal{U}$ is similarly defined as (1) by $\mathbf{B}^u \in \mathbb{R}^{N \times M_u}$, and

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

and the set of cameras that cover $\mathcal{U}_j$ is denoted by

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

The discretization of $\mathcal{R}$,$\mathcal{U}$ yields sub-optimality, 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 $\mathcal{U}$ and $\{\mathcal{V}_i, \mathcal{R}_i\}_{i=1}^N$. In this section, we assume the coverage geometry represented by $\mathbf{B}^r, \mathbf{B}^u$ is known. Section 4 describes a practical approach to determine $\mathbf{B}^r, \mathbf{B}^u$ in the image sensor network we consider.

The network provides user the desired view in a block-by-block manner. For each block $\mathcal{U}_j, j \in [M_u]$ in the desired view $\mathcal{U}$, the network selects a camera $\mathcal{C}_s$ satisfying the coverage requirement that $\mathbf{B}^u_{s,j} = 1$ to transmit relevant data to the CP where an synthesized view of $\mathcal{U}_j$ is generated. We assume the energy required to transmit each block $\mathcal{R}_j$ is equal for all $j \in [M_r]$. The methods to extract relevant data and synthesize the desired view are described in Section 4.3.

We assume each block on the monitored plane is requested by the user independently throughout the operation of the network and the the probability that block $\mathcal{R}_j$ is requested is given by p_j, where $\sum_{j=1}^{M_r}(p_j) = 1$. Let w_j^t denote the energy of camera $\mathcal{C}_j$ at time t. The remaining lifetime L of the network 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

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

represents the distribution of users' requests and

$$\mathbf{w}^t = [w_1^t \; w_2^t \; \ldots \; w_{M_r}^t]^T$$

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

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

$$\mathbf{m}_s^{t+1} = \mathbf{B}^r \mathbf{w}_s^{t+1}, \tag{5}$$

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

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 $\mathbf{w}^{t+1}$ 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 on this approximation does not cause a serious compromise in optimality. Now if block $\mathcal{U}_j$ is requested at time t, the optimal camera selection strategy is to select a camera from $\lambda^u(j)$ so that the network has maximum expected lifetime with updated energy:

$$s^t = \underset{i \in \lambda^u(j)}{\operatorname{argmax}} \mathsf{E}[L(\mathbf{m}_i^{t+1}, \mathbf{p})] \tag{6}$$

So far, we have formulated the problem of optimal camera selection to maximize network lifetime based on the independence assumption of $\{\mathcal{R}_j\}_{j=1}^{M_r}$ being selected and transmitted. In Section 3, we address the optimization problem in (6) and present corresponding camera selection strategy.

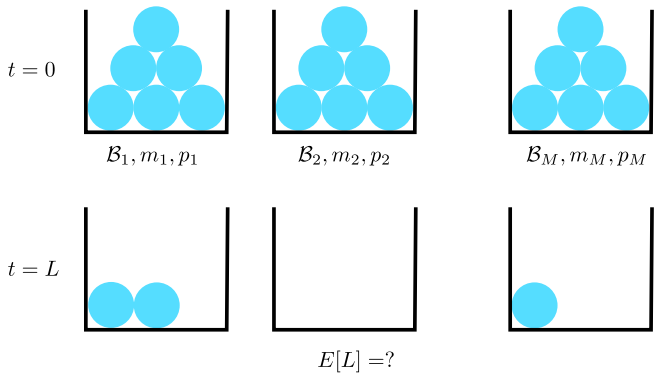


Figure 2. An abstraction of the sensor scheduling problem. M boxes $\mathcal{B}_i (i \in [M])$ contains m_i balls respectively. At each request, a ball is taken from $\mathcal{B}_i$ with probability p_i. After L requests, one of these boxes first become empty. We are concerned with $\mathsf{E}[L]$.

3. CAMERA SELECTION FOR LIFETIME MAXIMIZATION

3.1. Abstraction of Sensor Scheduling

Figure 2 illustrates an abstraction of the sensor scheduling problem: Consider M boxes $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_M$ respectively containing $m_1, m_2, \ldots, m_M$ 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, $\mathbf{m} = [m_1 \ m_2 \ \ldots \ m_M]^T$ represents the updated block-wise coverage energy $\mathbf{m}_s^{t+1}$ when camera $\mathcal{C}_s$ is selected, $\mathbf{p} = [p_1 \ p_2 \ \ldots \ p_M]^T$ represents the probabilities with which the blocks are requested (as before) and L denotes the remaining lifetime. For notational simplicity, we drop the superscript t, s in our discussion.

We first address a simplistic scenario where $M = 2$, for which an analytical solution of $\mathsf{E}[L(\mathbf{m}, \mathbf{p})]$ is obtained. In the general case that $M > 2$, we present a recursive approach to evaluate $\mathsf{E}[L(\mathbf{m}, \mathbf{p})]$. The computational load of this recursive approach is prohibitive, motivating us to further examine efficient approximations and the asymptotic behavior of $\mathsf{E}[L(\mathbf{m}, \mathbf{p})]$. Based on the asymptotic analysis of $\mathsf{E}[L(\mathbf{m}, \mathbf{p})]$, we develop a simple camera selection strategy which maximizes the expected network lifetime.

In the case $M = 2$, we assume $m_1 \leq m_2$ without loss of generality and present the following result,
Proposition 1: When $M = 2$, the distribution of L can be written as:

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

where $\alpha(\tau, b)$ represents the negative binomial distribution [8], and the p.m.f at k is

$$\alpha(k; \tau, b) = \begin{pmatrix} k + \tau - 1 \\ k \end{pmatrix} b^{\tau}(1 - b)^k \tag{8}$$

Let $F_\alpha(k; \tau, b) \stackrel{\text{def}}{=} \sum_{i=0}^{k} \alpha(k; \tau, b)$, the expectation of L can be obtained as

$$\mathsf{E}[L] = \beta(l - m_1, m_1, p) + \beta(l - m_2, m_2, 1 - p) \tag{9}$$

where

$$\beta(k, \tau, b) \stackrel{\text{def}}{=} \sum_{i=0}^{k} i\alpha(i; \tau, b) + \tau F_\alpha(k; \tau, b) \tag{10}$$

$$= \frac{nF_\alpha(k; \tau, b) - (k + 1)\alpha(k + 1; \tau, b)}{b} \tag{11}$$

Proof: When $M = 2$, we can write

$$\Pr(L = l) = \Pr(L = l, s^{-1} = 1) + \Pr(L = l, s^{-1} = 2) \tag{12}$$

where s^{-1} represents the index of box from which the last ball is selected. We note $\alpha(k; \tau, b)$, the p.m.f of the negative binomial distribution (8), represents the probability that in $(k + \tau)$ Bernoulli trials with a success probability b, k are failures and τ, including the last trial, are successful. (12) can then be interpreted as follows: $\alpha(l - m_1; m_1, p)$ represents the probability that $\mathcal{B}_1$ becomes empty at the l^{th} request, prior to which $(m_1 - 1)$ balls have been taken from $\mathcal{B}_1$ and $(l - m_1)$ taken from $\mathcal{B}_2$. $\alpha(l - m_2; m_2, 1 - p)$ can be similarly interpreted. The last ball must be taken from $\mathcal{B}_1$ for $m_1 \leq l < m_2$, and can be from either $\mathcal{B}_1$ or $\mathcal{B}_2$ for $m_2 \leq l < (m_1 + m_2)$.

Equation (9) is easily obtained by substituting (10) into the definition $\mathsf{E}[L] = \sum_{l=m_1}^{m_1+m_2-1} l\Pr(L = l)$. We next show the equality between (10) and (11). Let $q_k \stackrel{\text{def}}{=} \alpha(k; \tau, b)$ for short, the equality $\frac{q_{k+1}}{q_k} = \frac{\tau+k}{k+1}(1 - b)$ can be obtained by expanding q_k according to (8), we can further write

$$(k + 1)q_{k+1} = kq_k(1 - b) + q_k(1 - b)\tau. \tag{13}$$

Replacing k by i for $i \in [k]$, we obtain k equations which can be stacked together to obtain $\sum_{i=1}^{k} iq_i$ and eventually (11). ■

Note $F_\alpha(k; \tau, b)$ represents the cumulative distribution function (c.d.f) of the negative binomial distribution, which is characterized as a regularized incomplete beta function [9] and can be efficiently evaluated as a standard scientific routine. (9) and (11) allow exact evaluation of $\mathsf{E}[L]$. We next consider the case when $M > 2$ and assume $m_1 \leq m_2 \leq \ldots \leq m_M$ without loss of generality. Consider another experiment that uses the first $k(2 \leq k \leq M)$, boxes in our abstraction, each ball is requested from box $\mathcal{B}_i(i \in [k])$, with the normalized probability $p_k^i \stackrel{\text{def}}{=} \frac{p_i}{\sum_{i=1}^{k} p_i}$. In this experiment, let L_k denote the number of requests after which one of the boxes $\{\mathcal{B}_i\}_{i=1}^{k}$ becomes empty. We present the following method to recursively calculate the p.m.f of L_k from L_{k-1}.

Proposition 2: The p.m.f of $L_k(2 \leq k \leq M)$ can be recursively calculated from the p.m.f of L_{k-1},

$$\begin{aligned}\Pr(L_k = l) &= \mathbf{I}(m_1 \leq l \leq \pi(k)) \sum_{j=0}^{m_k-1} \left(\begin{array}{c} l-1 \\ j \end{array} \right) p_k^j (1 - p_k)^{l-j} \Pr(L_{k-1} = l - j) \\ &\quad + \mathbf{I}(m_k \leq l \leq \pi(k)) \left(\begin{array}{c} l-1 \\ l-m_k \end{array} \right) p_k^{m_k} (1 - p_k)^{l-m_k} \Pr(L_{k-1} > (l - m_k))\end{aligned} \tag{14}$$

where

$$\pi(k) \stackrel{\text{def}}{=} (\sum_{j=1}^{k} m_j - k + 1). \tag{15}$$

represents the maximum possible value of L.

Proof: We first rewrite

$$\Pr(L_k = l) = \Pr(L_k = l, s^{-1} \in [k - 1]) + \Pr(L_k = l, s^{-1} = k), \tag{16}$$

the first term of (16) can be written as

$$\begin{aligned}\Pr(L_k = l, s^{-1} \in [k - 1]) &= \sum_{j=0}^{m_k-1} \Pr(n_k^l = j, \sum_{i=1}^{k-1} n_i^l = l - j, L_{k-1} = l - j) \\ &= \sum_{j=0}^{m_k-1} \Pr(n_k^l = j, \sum_{i=1}^{k-1} n_i^l = l - j) \Pr(L_{k-1} = l - j)\end{aligned} \tag{17}$$

where n_k^l denotes the number of balls taken from $\mathcal{B}_k$ out of l requests. Equation (17) utilizes the Bayesian theorem and relies on the independence between the two terms therein. We observe each term of the summation in (17) corresponds to that in (14). The second term in (16) can be similarly interpreted. ■

We note (7) is a special case of (14) by observing

$$\Pr(L_1 = j) = \mathbf{I}(j = m).$$

Given (14), we can use $\mathsf{E}[L_k] = \sum_{l=m_1}^{\pi(k)} l\Pr(L_k = l)$ to directly evaluate the exact value of expected network lifetime, however, the computational load becomes prohibitive as k increases. We next present an alternative representation of $\mathsf{E}[L_k]$ which enables efficient approximation. We first denote the multinomial distribution by $\omega_\delta(k, \mathbf{b})$, where δ represents number of different possible results of a trail, k is the total number of trails and $\mathbf{b} = [b_1\ b_2\ \ldots\ b_\delta]$ represents probabilities of each possible result. The p.m.f of the multinomial distribution at $\mathbf{n}$ is $\omega_\delta(\mathbf{n}; k, \mathbf{b}) = \frac{k!}{n_1!n_2!\ldots n_\delta!} b_1^{n_1} b_2^{n_2} \cdots b_\delta^{n_\delta}$ for non-negative integer-valued $\mathbf{n}$ which satisfies $\sum_{i=1}^{\delta} n_i = k$, where n_i denotes the total number of the i^{th} result out of k trials. The c.d.f of $\omega_\delta(k, \mathbf{b})$ at τ can be written as $\Omega_\delta(\tau; k, \mathbf{b}) \stackrel{\text{def}}{=} \Pr(n_1 \le \tau_1, n_2 \le \tau_2, \ldots, n_\delta \le \tau_\delta)$.

Proposition 3: The expectation of L can be represented as

$$\mathsf{E}[L] = \sum_{l=m_1}^{\pi(M)} \Omega_M(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}) + (m_1 - 1)\Omega_M(\mathbf{m} - \mathbf{1}; m_1 - 1, \mathbf{p}) \tag{18}$$

where $\pi(\cdot)$ is defined in (15) and $\mathbf{1}$ denotes an $M \times 1$ vector, each of whose entries is unity.

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

$$\mathsf{E}(X) = \sum_{i=a}^{b} \Pr(X \ge i) + (a - 1)\Pr(X \ge a) \tag{19}$$

Equation (19) can be obtained by simply expanding $\Pr(X \ge i) = \sum_{j=i}^{b} \Pr(X = j)$ for each i then summing up the resulting equalities for $i = a, a + 1, \ldots, b$, and then reorganizing resulting terms on the right hand side.

Equation (19) applies to L since L takes positive integer values in the closed interval $[m_1, \pi(M)]$. We next note $\Pr(L \ge l)$ can be alternatively represented as $\Pr(n_i^{l-1} \le m_i - 1, i \in [M])$, indicating that after $(l - 1)$ requests, there is at least one ball left in each box. We can write

$$\Pr(L \ge l) = \Omega(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}). \tag{20}$$

Substituting (20) into (19), and reorganizing resulting terms, we obtain (18). ■

Equation (18) represents $\mathsf{E}(L)$ in terms of the c.d.f of a multinomial distribution, the latter can be efficiently approximated with high accuracy [10]. We thus obtain an approximation of $\mathsf{E}[L]$ as

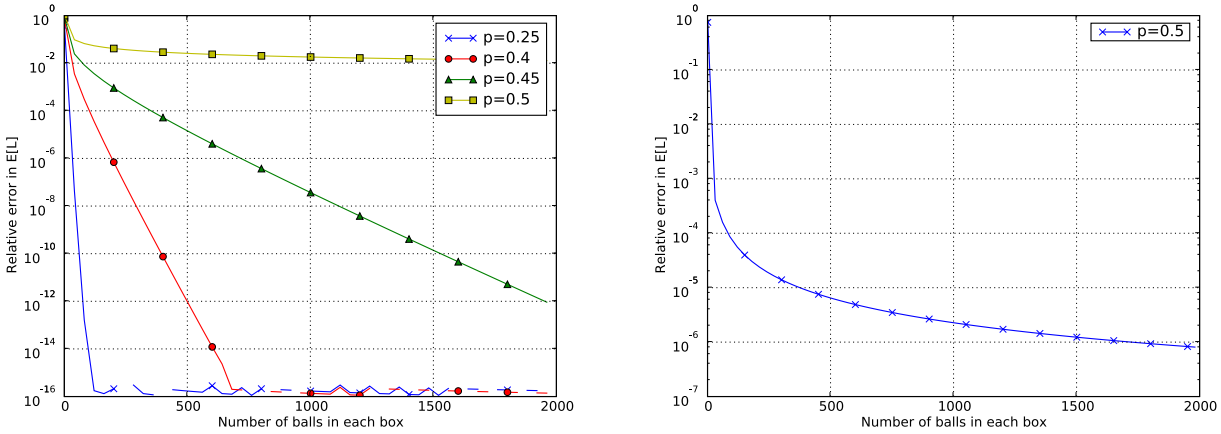
$$\tilde{\mathsf{E}}(L) = \sum_{l=m_1}^{\pi(M)} \tilde{\Omega}_M(\mathbf{m} - \mathbf{1}; l - 1, \mathbf{p}) + (m_1 - 1)\tilde{\Omega}_M(\mathbf{m} - \mathbf{1}; m_1 - 1, \mathbf{p}), \tag{21}$$

where $\tilde{\Omega}(\cdot)$ denotes approximation of $\Omega(\cdot)$. (21) enables accurate approximation of $\mathsf{E}[L]$ by $(\pi(M) - m_1 + 2)$ calls of the function $\tilde{\Omega}(\cdot)$, which achieves substantial acceleration compared to direct evaluation of $\mathsf{E}[L]$ based on (13).

We experimentally study the accuracy of this approximation and illustrate the result in Table 1. We observe from Table 1 that highly accurate approximation of $\mathsf{E}(L)$ can be achieved by using the proposed representation (18) and approximation of the c.d.f of a multinomial distribution [10]. However, (18) doesn't immediately leads to any intuition for a camera selection strategy. To this end, we proceed to investigate the asymptotic behavior of $\mathsf{E}[L]$ when $\mathbf{m}$ is sufficiently large.

	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

Table 1. An experimental evaluation of the approximation for (18) using [10]. 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 (18). *Approx:* approximate lifetime using (21). *Simulation:* The average lifetime resulting from 200 Monte Carlo simulations.



(a) (b)

Figure 3. An experimental evaluation of the approximation for $\mathsf{E}[L]$ using (22). 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 $\min(\frac{m_1}{p}, \frac{m_2}{1-p})$. (b): refined approximation for the case $\frac{m_1}{p} = \frac{m_2}{1-p}$ using the the bottom of (22).

Proposition 4: When $M = 2$ and m_1, m_2 are sufficiently large,

$$\mathsf{E}[L] \approx \begin{cases} \min(\frac{m_1}{p}, \frac{m_2}{1-p}) & \text{if } \frac{m_1}{p} \neq \frac{m_2}{1-p} \\ \frac{m_1}{p} + \sqrt{\frac{m_1(1-p)}{2\pi p^2}} + \sqrt{\frac{m_2 p}{2\pi(1-p)^2}} & \text{if } \frac{m_1}{p} = \frac{m_2}{1-p} \end{cases} \tag{22}$$

Due to length considerations, we skip the proof in this paper. Instead, we experimentally evaluate the accuracy of (22) and illustrate the results in Fig. 3. We observe that (22) achieves satisfactory accuracy and the relative error which is defined as $\frac{\mathsf{E}(L)-\tilde{\mathsf{E}}(L)}{\mathsf{E}(L)}$ where $\tilde{\mathsf{E}}(L)$ is the approximation using (22), converge to 0 at an exponential rate as the number of balls increases. We also observe in the case $\frac{m_1}{p} = \frac{m_2}{1-p}$, even the simple approximation $\min(\frac{m_1}{p}, \frac{m_2}{1-p})$ (the curve labeled by $p = 0.5$ in Fig. 3(a)) actually achieves satisfactory accuracy (1% relative error) for many practical applications. This observation leads us to the following approximation for the general cases where $M > 2$,

$$\mathsf{E}[L(\mathbf{m}, \mathbf{p})] \approx \min(\frac{m_1}{p_1}, \frac{m_2}{p_2}, \ldots, \frac{m_M}{p_M}) \tag{23}$$

This approximation is close provided the energy in each camera is large as compared to the energy for each access and provided the difference between the two smallest values in $\{\frac{m_i}{p_i}\}_{i=1}^N$ is not negligible. In this paper, we skip further analysis and refinement of (23) and proceed to propose and experimentally evaluate camera selection strategies based on this conjecture.

3.2. Camera Selection Strategy

Given the optimal criteria for camera selection defined in (6) and the approximation of expected network lifetime (23), we can write the optimal camera selection strategy when block $\mathcal{U}_j$ is requested by the user:

$$s = \underset{i \in \lambda^u(j)}{\text{argmax}} \{\min(\frac{m_{i,1}}{p_1}, \frac{m_{i,2}}{p_2}, \ldots, \frac{m_{i,M_r}}{p_{M_r}})\} \tag{24}$$

where we dropped the superscript t. An interpretation of (24) is to maximize the energy of the *hot-spot* block in the monitored plane. The hot-spot block refers to the block, the energy of which divided by the probability that this block is requested, has the minimum value in the (subregion of) monitored area.

In cases where the hot-spot block doesn't belong to any of the cameras that cover $\mathcal{U}_j$, the optimal strategy is as follows:

$$s = \underset{i \in \lambda^u(j)}{\text{argmax}} \{\min(\frac{m_{i,k}}{p_k}, k \in \kappa^r(i))\} \tag{25}$$

where $\kappa^r(i)$ denotes the set of blocks in the monitored region $\mathcal{R}$ covered by camera $\mathcal{C}_i$, i.e. $\kappa^r(i) \stackrel{\text{def}}{=} \{k | \mathbf{B}^r_{i,k}(i) = 1\}$. Equation (25) indicates that when block $\mathcal{U}_j$ is requested, we maximize the energy of the hot-spot block in the sub-region covered by the set of cameras that covers $\mathcal{U}_j$. We refer to the camera selection strategy described above as *OptCOV*.

OptCOV present several advantages: 1) OptCOV represent an optimized strategy based on a stochastic formulation of the network lifetime, thus we expect a longer lifetime by scheduling sensors using OptCOV. 2) (25) only requires an ordering operation which minimize the load on the CP and enables real-time applications. 3) User interactions are explicitly addressed in the model, thus the network can acquire or dynamically estimate information about users' requests and schedule sensors accordingly.

We introduce two other camera selection strategies for the purpose of comparison:

1. *View Angle Cost (MinANG):* The camera that has the most similar viewing directions with user's viewpoint at each block is selected. This approximately select the camera that achieves the highest quality provided that all the cameras are at (almost) same distance from the target plane so that image quality is affected primarily by viewing direction.

2. *Coverage Cost (CovCOST):* As in [7], we define a *coverage cost* ξ_i for camera $\mathcal{C}_i$ as the sum of contributions from all blocks covered by $\mathcal{C}_i$, and the contribution of each individual block is defined as the inverse of its energy:

$$\xi_i = \sum_{j \in \lambda^r(i)} \frac{1}{m_j} \tag{26}$$

   Given this definition of coverage cost, those cameras that have large overlapping FoVs with other cameras have small cost and therefore are selected more frequently than those cameras that solely cover a part of the monitored scene. Thus selection of cameras with minimal coverage cost results in a longer network lifetime.

The performance of all these strategies will be compared in Section 5.

4. SYSTEM SCENARIO

We describe the implementation of the overall system scenario as illustrated in Fig. 4. In the initialization stage, parameters of cameras are estimated. Based on these estimated parameters, the coverage information can be obtained. For each of desired viewpoints requested by users during the operation, the network synthesizes the desired view using data from a selected subset of the cameras.

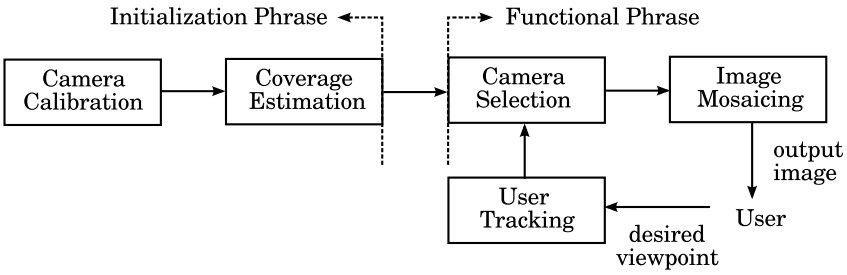


Figure 4. The operation of the network.

4.1. Initialization

Camera Calibration: Using a homogeneous representation, the image coordinate $\mathbf{x} = [x_1\ x_2\ x_3]^T$ of a $3D$ point $\mathbf{X} = [X_1\ X_2\ X_3\ 1]^T$ is given by $\mathbf{x} = \mathcal{P}\mathbf{X}$, where $\mathcal{P} \in \mathbb{R}^{3\times4}$ denotes the camera projection matrix [11] which is determined by intrinsic parameters (such as focal length) and extrinsic parameters (orientation and location) of the camera. The pixel coordinates of $\mathbf{x}$ can be obtained as $[\frac{x_1}{x_3}\ \frac{x_2}{x_3}]^T$. Plane-Based camera calibration techniques such as [12] can be used to estimate the parameters of cameras. Although the original algorithm in [12] is designed to calibrate a single camera, it can be extended to calibrate multiple cameras with varying intrinsic parameters [13].

Coverage Estimation: The coverage of each camera's view on a target plane can be obtained from the camera geometry. A world point is covered by a camera if the image coordinate of this point lies within the FoV of the camera. An example of coverage estimation can be found in Fig. 1. By considering the center of each block in the monitored plane $\mathcal{R}$ and desired view $\mathcal{U}$, we can respectively estimate the coverage matrix $\mathbf{B}^r, \mathbf{B}^u$, which serve as inputs to our camera scheduling strategy as described in Section 3.

4.2. Camera Scheduling

As described before, the desired view $\mathcal{U}$ is discretized into M_u blocks $\{\mathcal{U}\}_{i=1}^{M_u}$ and processed in sequence. For each block $\mathcal{U}_i$, we select camera $\mathcal{C}_s$ according to one of the three camera selection strategies (OptCOV, MinANG, CovCOST) in order to capture and transmit relevant data.

4.3. Image Mosaicing

Once the raw data for all the blocks in the desired view are transmitted from selected cameras and received by the CP, they must be transformed and mosaiced together to synthesize the desired view. General mosaicing of $3D$ scenes needs $3D$ geometric information which is hard to obtain. However, the mosaicing can be described by a homography [11] for the texture image of a planar surface (our case) or images captured by a rotating camera.

Without loss of generality, we assume $X_3 = 0$ for all world points $\mathbf{X}$ on the monitored plane and let $\mathbf{X}_h = [X_1\ X_2\ 1]^T$ where we neglect $X_3 = 0$, represents a world point on this 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 $\mathcal{H} \in \mathbb{R}^{3\times3}$ 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. 5, 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 $\mathbf{x}_1 = \mathcal{H}_1\mathcal{H}_2^{-1}\mathbf{x}_2$. This relation allows the view at user's desired viewpoint to be rendered from corresponding regions in the selected camera.

5. SIMULATIONS

We simulate a monitored plane of size $4m \times 3m$ (typical size of a wall). To simulate ad hoc deployment, N cameras are placed randomly within a $4m \times 3m$ field located $3m$ from the target plane, and the cameras are pointed toward the target plane with a random rotation within ±0.1 radian along each of the three axes to simulate practical variability in camera placement. All cameras (including the user's viewpoints) are assumed

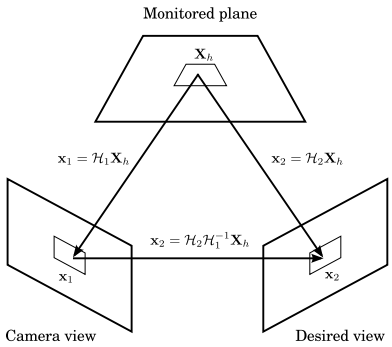


Figure 5. image coordinate of $\mathbf{X}_h$ in the second (desired) view $\mathbf{x}_2$ can be obtained from $\mathbf{x}_1$ through the homography.

to have images of 200×200 (in pixel units), with a focal length $f_0 = 218.75$ (in pixel units). For an image sensor with size $20mm \times 20mm$, this would correspond to focal length of $f_0 = 21.875mm$. The camera images are generated by using the scene geometry (homography) with the camera model of Section 4.3 when requested. Pixel values at non-integer locations are generated by bilinear interpolation.

The user's viewpoints are generated in our simulations from a random walk on a 16×16 grid in the plane of the cameras starting at the center $(x = 2m, y = 1.5m, z = 3m)$. Subsequent viewpoints are chosen from the neighboring 8 grid points and the current position with equal probability, and the user's views are assumed to be directed toward the target plane with random rotation within ± 0.1 radian along each of the three axes. We generate $M_u = 100$ user's viewpoints in each simulation. Each desired view is divided into 100 blocks and each block is synthesized by data transmitted from a selected camera.

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 [7]. We found that using a focal length $f = 218.75$ (in pixel units), a minimum of 18 cameras are necessary in order to ensure that the target plane is covered with a confidence of 99.5%.

We simulate a scenario that all cameras start with $3J$ of energy, which correspond to each camera being able to transmit 3 full frames of images. At each time moment, a desired view of 200×200 is requested. In order to increase coverage redundancy, we use twice the minimum number of cameras necessary for full coverage, i.e. $N = 36$. All results presented represent averages over 100 simulations.

Figure 5 compares the percentage coverage on the target plane over time for the different camera selection methods. The abscissa of Fig. 5 and Fig. 5 represent time, which is equivalent to the sequence number of the current request. We observe that OptCOV camera selection maintains significant higher percentage of coverage in the network. We define the network lifetime as the duration during which 95% of the monitored area is covered. Then OptCOV prolongs the network lifetime by a factor of 1.5 compared to CovCOST and 3 compared to MinANG. The improvement is achieved by allocating the energy consumption dynamically according to remaining energy in the cameras and the distribution of user's requests. We also observe the simple heuristic approach of CovCOST prolongs network lifetime by a factor of 2 compared to MinANG which completely ignores power considerations.

Figure 5 compares the PSNR of the mosaiced image. Initially, the image qualities are about the same while MinANG outperforms by about $1dB$ since the camera with closest viewing direction is selected. However, as time progresses MinANG loses coverage and the image quality degrades drastically and fails to meet user requirement (say below $35dB$). By preserving more camera based on anticipated coverage requirement, OptCOV provides high image quality over a longer duration and achieves higher level of user satisfaction. The performance of CovCOST lies in between MinANG and CovOPT. This clearly demonstrates the advantage of using OptCOV in an energy-constrained scheduling scenario.

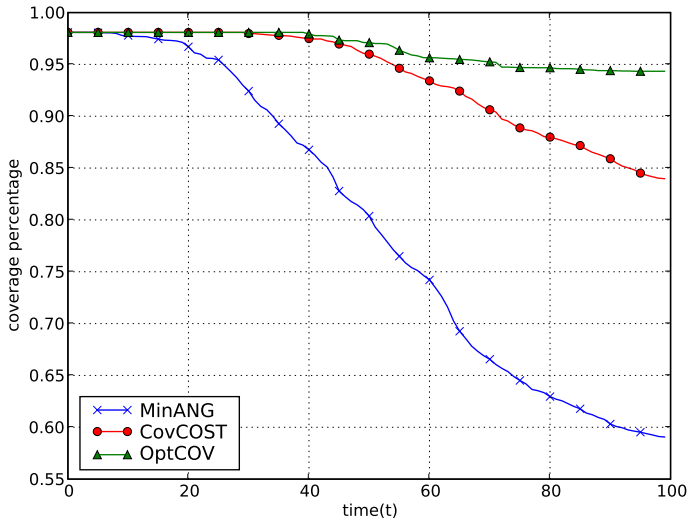


Figure 6. comparison of percentage coverage.

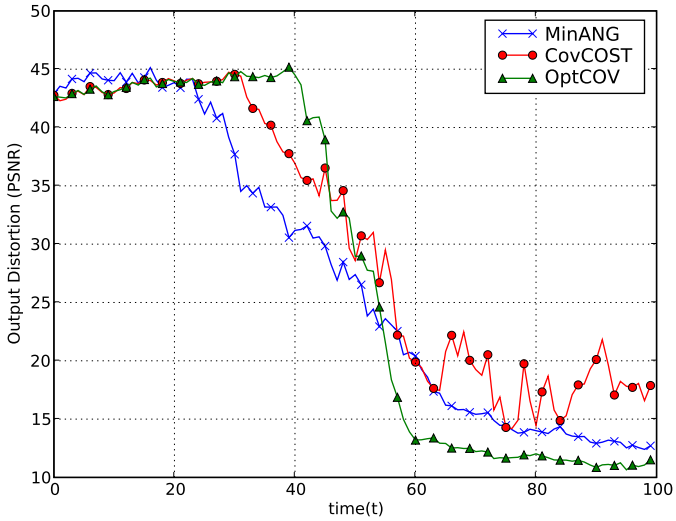


Figure 7. comparison of distortion in output image.

6. CONCLUSION

A stochastic model is developed for the lifetime of a battery-powered user-centric image sensor network and based on the model, sensor scheduling strategies are developed in order to maximize the operational network lifetime. Through a suitable abstraction of the problem, we obtain closed-form and approximate expression for the expected value of network lifetime, which form the basis of our proposed scheduling schemes. The computational load of the proposed sensor scheduling approach is very light, and user interactions are explicitly addressed. Simulation results demonstrate the utility of the proposed approach, offering a significant improvement in lifetime in comparison with alternative scheduling schemes.

Our analysis in this paper is presented in the context of the specific application scenario of a user-centric

image sensor network. However, we believe that the underlying abstraction and analysis are also applicable in a broader set of sensor scheduling and more general resource allocation problems.

REFERENCES

1. A. Keshavarz, A. Tabar, and H. Aghajan, "Distributed vision-based reasoning for smart home care," *ACM SenSys Workshop on Distributed Smart Cameras (DSC06)*, 2006.
2. P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: a multi-tier camera sensor network," *Proceedings of the 13th ACM Multimedia*, pp. 229–238, 2005.
3. M. Wu and C. Chen, "Collaborative Image Coding and Transmission over Wireless Sensor Networks," *EURASIP Journal on Advances in Signal Processing, special issue on visual sensor networks*, 2007.
4. C. Huang and Y. Tseng, "The Coverage Problem in a Wireless Sensor Network," *Mobile Networks and Applications*, vol. 10, no. 4, pp. 519–528, 2005.
5. J. Dagher, M. Marcellin, and M. Neifeld, "A method for coordinating the distributed transmission of imagery," *IEEE Trans. Image Proc.*, vol. 15, no. 7, 2006.
6. S. Soro and W. Heinzelman, "On the coverage problem in video-based wireless sensor networks," in *In Proceedings of the Second Workshop on Broadband Advanced Sensor Networks (BaseNets '05)*, 2005.
7. C. Yu, S. Soro, G. Sharma, and W. Heinzelman, "Lifetime-distortion trade-off in image sensor networks," in *Proc. IEEE Intl. Conf. Image Proc.*, vol. V, Sept. 2007, pp. 129–132.
8. J. M.Hilbe, *Negative Binomial Regression*. Cambridge University Press, 2007.
9. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge, U.K.: Cambridge University Press, 1992.
10. B. Levin, "A representation for multinomial cumulative distribution functions," *The Annals of Statistics*, vol. 9, no. 5, pp. 1123–1126, 1981.
11. R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. New York, NY, USA: Cambridge University Press, 2000.
12. Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *ICCV*, 1999, pp. 666–673.
13. C. Yu and G. Sharma, "Plane-based calibration of cameras with zoom variation," in *Proc. SPIE: Visual Communications and Image Processing*, J. G. Apostolopoulos and A. Said, Eds., vol. 6077, 15 - 19 Jan. 2006, pp. 607 710–1–607 710–9.