Camera Network Node Selection for Target Localization in the Presence of Occlusions

A. O. Ercan
Department of Electrical Engineering
Stanford University
aliercan@stanford.edu

A. El Gamal
Department of Electrical Engineering
Stanford University
abbas@ee.stanford.edu

L. J. Guibas
Department of Computer Science
Stanford University
guibas@cs.stanford.edu

Abstract

A camera network node subset selection methodology for target localization in the presence of static and moving occluders is described. It is assumed that the locations of the static occluders are known, but that only prior statistics for the positions of the object and the moving occluders are available. This occluder information is captured in the camera measurement via an indicator random variable that takes the value 1 if the camera can see the object and 0, otherwise. The minimum MSE of the best linear estimate of object position based on camera measurements is then used as a metric for selection. It is shown through simulations and experimentally that a greedy selection heuristic performs close to optimal and outperforms other heuristics.

Keywords

Wireless sensor network, camera network, selection, target localization, occlusion.

1 Introduction

There is a growing need to develop low cost wireless networks of cameras with automated detection capabilities, e.g., [6]. The main challenge in building such networks is the high data rate of video cameras. On the one hand sending all the data, even after performing standard compression, is very costly in transmission energy, and on the other, performing sophisticated vision processing at each node to substantially reduce transmission rate requires high processing energy. To address these challenges, a task-driven approach, in which simple local processing is performed at each node to extract the essential information needed for the network to collaboratively perform the task, has been proposed, e.g., [5]. Communication and computation cost can also be reduced by dynamically selecting the best subset of camera nodes to collaboratively perform the task. Such selection allows for efficient sensing with little performance degradation relative to using all the cameras, and makes it possible to scale to the network to a large numbers of nodes.

The selection problem has been studied in the sensor network literature, where a utility metric of the task is optimized over subsets of the desired size. Examples of such metrics include, information theoretic quantities such as Fischer Information Matrix and mutual information, e.g., [3,4], coverage [9], and occupancy map [7,12]. Other researchers considered general utility functions and used their properties for optimal selection [1,2]. Camera selection has also been a topic of interest in computer vision and graphics. In viewpoint selection scene models and metrics such as the number of faces or voxels seen have been used [8,11].

The work in this paper follows the framework in [5], where the selection for single point target localization in a camera network is investigated. In that work, noisy camera measurements and an object prior are assumed and the minimum mean squared error (MSE) of the best linear estimate of the object position in 2-D is used as a metric for selection. As selection is a combinatorial problem, a semi-definite programming approximation is proposed and shown to achieve close to optimal solutions with low computational burden. A simple heuristic for dealing with limited camera field of view (FOV) and static occluders are briefly discussed but are not tested in simulations or experimentally. More importantly, no dynamic (moving) occluders are considered in [5].

In this paper, we show how static and dynamic occlusions and limited FOV can be incorporated in the framework of [5]. We assume that simple local processing whereby each image is reduced to a scan-line is performed, and only the center of the detected object from each camera node is communicated to its cluster head. The minimum MSE of the best linear estimate of the point object position that incorporates the occlusions and limited FOV is used for selection. Given the noisy camera measurements, the object prior, the dynamic occluder priors, the static occluder information and the FOVs of the cameras, a greedy heuristic is used for selection. We show that this simple heuristic performs close to optimal and outperforms naive heuristics such as, picking the closest subset of cameras or a uniformly spaced subset.

The rest of the paper is organized as follows: In Section 2, we introduce the setup and camera model, define the selection metric and explain how it can be efficiently computed. In Section 3, we compare the performance of the greedy selection heuristic to other heuristics and to the optimal solution, both in simulation and experimentally.

2 Problem Formulation

We consider the setup illustrated in Fig. 1 in which N cameras are aimed roughly horizontally around a room. Although an overhead camera would have a less occluded view than a horizontally placed one, it generally has a more limited view of the scene and is often impractical to deploy. Additionally targets may be easier to identify in a horizontal view. The cameras are assumed to be fixed and their lo-

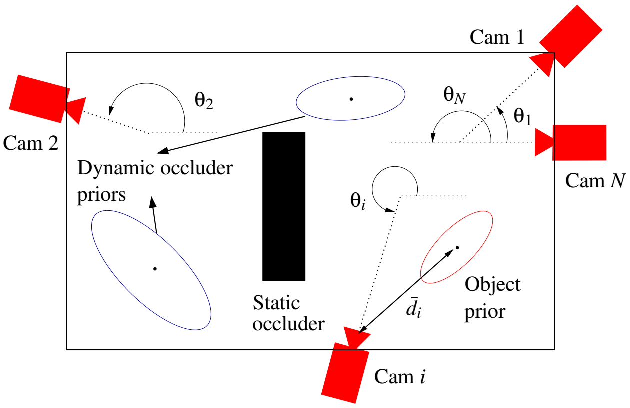


Figure 1. Illustration of the setup.

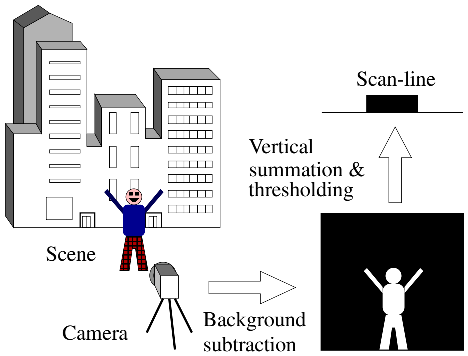


Figure 2. Local processing at each camera node.

cations and orientations are known to some accuracy. The camera network's task is to localize an object in the presence of static occlusions (such as partitions, tables, etc) and other moving objects. We assume the object to localize to be a point object. This is reasonable because the object may be distinguished from occluders using certain point features. We assume M other moving objects, each modeled as cylinder of diameter D. The position of each object is assumed to be the center of its cylinder. From now on, we shall refer to the object to localize as the "object" and the other moving objects as "dynamic occluders."

Our focus is on selecting the best subset of nodes of size $k < N$ to perform the localization. We assume that selection is performed in a short enough time that all objects can be considered still during the process. What differentiates a dynamic from a static object is the degree of knowledge about their positions. We assume the positions and the shapes of the static occluders to be completely known in advance. On the other hand, we assume that only some statistics of the object and dynamic occluder positions are known. Such information can be made available through a higher level application such as tracking that is performed by the camera network. Specifically, we assume that the object position x is a Gaussian random vector with mean μ and covariance matrix Σ_x, and the position of dynamic occluder $s \in \{1,2,\ldots M\}$, x_s, to be also Gaussian with mean μ_s and covariance matrix Σ_s. Further, the positions of the object and dynamic occluders are assumed to be mutually independent.

As in [5], we assume simple background subtraction is performed locally at each camera node. Since the horizontal position of the object in each camera's image plane is the most relevant information to 2-D localization, the background subtracted images are vertically summed and thresholded to obtain a "scan-line" (see Fig. 2). We assume that the camera nodes can distinguish between the object and the occluders. This can be done, for example, through feature detection, e.g., [10]. Only the center of the object in the scan-line is sent to the cluster-head. If a camera can "see" the object, we assume the linear noisy camera model in [5]. On the other hand, if the camera cannot see the object because of occlusions, it assumes that the object is at its mean, which provides no new information. Mathematically, we define for camera $i = 1,2,\ldots,N$, the random variable

$$\eta_i \equiv \begin{cases} 1, & \text{if camera } i \text{ sees the object} \\ 0, & \text{otherwise.} \end{cases}$$

The camera measurement model including occlusions is then defined as

$$z_i = \eta_i \left(a_i^T (x-\mu) + v_i\right) + a_i^T \mu, \; i = 1,2,\ldots,N, \qquad (1)$$

where $a_i^T = [-\sin(\theta_i) \; \cos(\theta_i)]$, v_i is camera i's measurement error and is assumed to be Gaussian with zero mean and variance $\sigma_{vi}^2 = \sigma_{oi}^2 + \zeta_i \bar{d}_i^{\,2}$, where $\bar{d}_i$ is the distance from camera i to the mean of the object position μ (see Fig. 1). The v_is are assumed to be mutually independent and also independent of the object and dynamic occluder positions.

We formulate the problem of camera node selection for target localization in the framework of linear estimation (LE) [5]. Given the object and dynamic occluder priors, the positions and shapes of the static occluders, the camera FOVs and the camera noise parameters, we use the minimum MSE of the best linear estimate of the object position as a metric for localization error. The best camera node subset is defined as the subset that minimizes this metric. Note that we do not propose or assume that the actual localization is performed using LE. Once the best subset is chosen, the selected camera nodes are queried for measurements and an appropriate localization method such as a Bayesian estimator, which can handle a more accurate camera model may be used.

To compute the MSE, define for cameras $i, j \in \{1,2,\ldots,N\}$

$$p_{ij}(x) \quad \equiv \quad \Pr\{\eta_i = 1, \eta_j = 1 | x\}, \qquad (2)$$

Then it can be shown that the MSE of the linear estimator assuming the camera model in (1) is given by

$$\begin{aligned} \text{MSE} &= \text{Tr}\left(\Sigma_x - \Sigma_{zx}^T \Sigma_z^{-1} \Sigma_{zx}\right) \qquad (3) \\ \Sigma_{zx}(i) &= a_i^T \left[\text{E}_x\left(p_{ii}(x)\tilde{x}\tilde{x}^T\right)\right] \\ \rho(i) &\equiv a_i^T \left[\text{E}_x\left(p_{ii}(x)\tilde{x}\right)\right] \\ \Sigma_z(i,j) &= a_i^T \text{E}_x\left(p_{ij}(x)\tilde{x}\tilde{x}^T\right) a_j - \rho(i)\rho(j) \\ &\quad + \begin{cases} \text{E}_x\left(p_{ii}(x)\right)\sigma_{v_i}^2 & i = j \\ 0 & i \neq j \end{cases}, \end{aligned}$$

where $\tilde{x} \equiv x - \mu$. The MSE for a subset $S \subset \{1, 2, \ldots, N\}$, $\mathrm{MSE}(S)$, is defined as in (3) but with only the camera nodes in S included.

2.1 Computing MSE(S):

Since selection is envisioned to be performed at each cluster head, it is important that $\mathrm{MSE}(S)$ can be efficiently computed and optimized. In order to do this, we need to compute the probabilities $p_{ij}(x)$ and then evaluate the expectations over x.

First we ignore the static occluders and limited camera FOV and only consider the dynamic occluders. Now, consider

$$\begin{aligned}
p_{ii}(x) &= \Pr\{\eta_i = 1 | x\} \\
&= \Pr\{\mathrm{cam}_i \text{ is not occluded } | x\} \\
&= \Pr\left\{ \bigcap_{s=1}^{M} (\text{object } s \text{ does not occlude } \mathrm{cam}_i) \middle| x \right\} \\
&\overset{(a)}{=} \prod_{s=1}^{M} \Pr\{\mathrm{obj}_s \text{ does not occlude } \mathrm{cam}_i | x\} \\
&= \prod_{s=1}^{M} (1 - q_i^s(x)), \qquad (4)
\end{aligned}$$

where step (a) follows by the assumption that the occluder positions are independent, and $q_i^s(x) = \Pr\{\text{Object } s \text{ occludes } \mathrm{cam}_i | x\}$.

To compute $q_i^s(x)$, refer to Fig. 3. Without loss of generality we assume that camera i is at the origin. We assume that the dynamic occluder diameter D is small compared to the occluder standard deviations. Note that object s occludes point x at camera i if its center is inside the rectangle $A_i(x)$. With these assumptions, we can approximate $q_i^s(x)$ by

$$\begin{aligned}
& q_i^s(x) \\
&= \int_{A_i(x)} \frac{1}{2\pi\sqrt{|\Sigma_s|}} \exp\left(-\frac{1}{2}(x' - \mu_s)^T \Sigma_s^{-1} (x' - \mu_s) \right) dx' \\
&\overset{(b)}{\approx} \frac{D}{2\sigma_s} \sqrt{\frac{\alpha}{2\pi\|v\|^2}} \exp\left(-\frac{\alpha(\mu_{sy}\cos(\theta_{si}) - \mu_{sx}\sin(\theta_{si})^2}{2\sigma_s^2\|v\|^2} \right) \\
&\quad \left[\mathrm{erf}\left(\frac{1}{\sqrt{2}\sigma_s} \frac{\|x\|\,\|v\|^2 - \mu_s^T u}{\|v\|} \right) + \mathrm{erf}\left(\frac{1}{\sqrt{2}\sigma_s} \frac{\mu_s^T u}{\|v\|} \right) \right],
\end{aligned}$$

where $u = [\cos(\theta_{si}) \; \alpha\sin(\theta_{si})]^T$, $v = [\cos(\theta_{si}) \; \sqrt{\alpha}\sin(\theta_{si})]^T$, σ_s^2 and σ_s^2/α, $\alpha \geq 1$, are the eigenvalues of the covariance matrix Σ_s of the position of occluder s, and step (b) follows by the assumption of small dynamic occluders. We integrate the Gaussian along the long edge of $A_i(x)$ and assume it to be constant along the short edge. Next, we consider computing

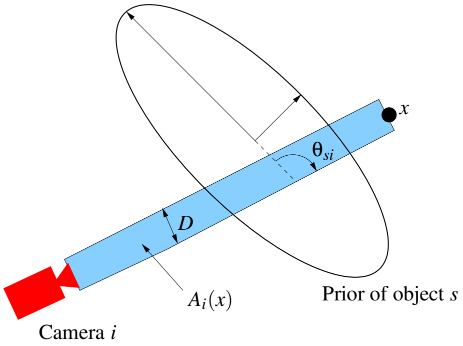


Figure 3. Computing $q_i^s(x)$.

$p_{ij}(x)$, for $i \neq j$

$$\begin{aligned}
p_{ij}(x) &= \Pr\left\{ \bigcap_{s=1}^{M} (\text{object } s \notin A_i(x) \text{ and object } s \notin A_j(x)) \middle| x \right\} \\
&= \prod_{s=1}^{M} \Pr\{\text{object } s \notin A_i(x) \text{ and object } s \notin A_j(x) | x\} \\
&= \prod_{s=1}^{M} \left(1 - \Pr\{\text{object } s \in A_i(x) \text{ or object } s \in A_j(x) | x\}\right) \\
&\overset{(c)}{\approx} \prod_{s=1}^{M} \left(1 - \Pr\{\text{object } s \in A_i(x)\} - \Pr\{\text{object } s \in A_j(x) | x\}\right) \\
&= \prod_{s=1}^{M} \left(1 - q_i^s(x) - q_j^s(x)\right), \qquad (5)
\end{aligned}$$

where (c) follows from the assumption of small D and the reasonable assumption that cameras i and j are not too close so that the overlap between $A_i(x)$ and $A_j(x)$ is negligible.

To compute the expectations in (3), we fit a grid of points over the 3-σ ellipse of the Gaussian pdf. The p_{ij}s are computed over these points as explained above. We then perform a 2-D numerical integration over the grid.

2.2 Adding Static Occluders and Limited FOV

The effects of static occluders and limited FOV can be readily included in computing $p_{ii}(x)$ and $p_{ij}(x)$. Let $I_i(x)$ be the indicator function of the points visible to camera i when static occluders and the limited FOV are present (see Fig. 4), then it is easy to show that

$$\begin{aligned}
p_{ii}(x) &= \Pr\left\{ (I_i(x) = 1) \text{ and } \bigcap_{s=1}^{M} (\text{object } s \notin A_i(x)) \middle| x \right\} \\
&= I_i(x) \prod_{s=1}^{M} (1 - q_i^s(x)), \text{ and similarly} \qquad (6)
\end{aligned}$$

$$p_{ij}(x) = I_i(x) I_j(x) \prod_{s=1}^{M} \left(1 - q_i^s(x) - q_j^s(x)\right). \qquad (7)$$

3 Selection

The selection problem involves minimizing $\mathrm{MSE}(S)$ subject to $|S| = k$. This is combinatorial and requires $O(N^k)$

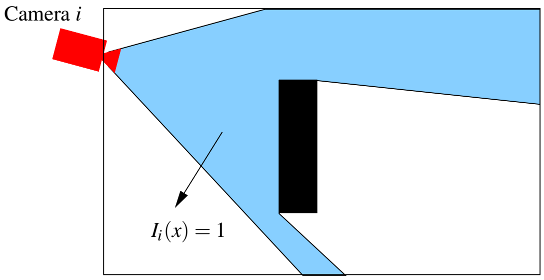


Figure 4. Illustration of $I_i(x)$.

trials if brute-force search is used. This can be too costly in a wireless camera network setting. Note that in [5], we found a closed form expression for MSE(S) and used a semi-definite programming (SDP) heuristic to perform the selection. As the occlusions at different cameras (represented by the η_is) are not independent from each other or from x, we cannot reduce the MSE to a simple closed form and use the same SDP heuristic. Instead, we use the greedy selection algorithm in Fig. 5. Fortunately, as we shall demonstrate in the following subsections, this greedy heuristic yields close to optimal results.

It can be shown that the computational complexity of the greedy algorithm is $O(k^2MNL + k^4N)$, where k is the subset size, M is the number of dynamic occluders, N is the number of cameras, and L is the number of grid points used to evaluate the expectations in (3).

3.1 Simulation Results

We performed Monte-Carlo simulations to compare the performance of the greedy approach to the optimal brute-force enumeration as well as to the heuristics:

- *Uniform:* Pick uniformly placed cameras.
- *Closest:* Pick the closest cameras to the object mean.

Fig. 6 compares the RMS localization error of the four selection procedures for a typical simulation run with $k = 3$ to 9 camera nodes out of 30 cameras uniformly placed around a circular room with 3 static occluders and 10 dynamic occluders. We first randomly chose the object and dynamic occluder prior parameters as follows. We chose the means randomly and independently. The eigenvalues of Σ_x are set equal to 100 and 12.5. The larger eigenvalue of Σ_s is set equal to 100 and the smaller one is chosen at random. We then applied random rotations to all priors. We performed the selection using the four aforementioned procedures. We then dropped the object and the dynamic occluders at random according to the selected priors 5000 times and localized the object with the selected camera nodes. This procedure was repeated 20 times. As seen in Fig. 6, the error for the greedy approach completely overlaps with that of brute-force enumeration and outperforms the other heuristics.

Note that, even if a selection algorithm makes bad decisions, e.g., selects cameras that are all occluded, the worst the central processor (cluster head) can do is predict that the object position is at its mean. Because of this, the difference in performance between the above procedures is not too large. However, in an application such as tracking, these errors could build up over time and may in fact result in completely losing the object.

```
Algorithm: Greedy camera node selection
Inputs: Object's prior ($\mu$, $\Sigma_x$);
Dynamic occluders' priors ($\mu_s, \Sigma_s, s \in \{1,\dots,M\}$);
Shapes and positions of static occluders;
Camera positions and orientations ($\theta_i$, $i \in \{1,\dots,N\}$);
FOVs of the cameras;
Number of camera nodes to select ($k$).
Output: Best subset ($S$).
Choose a grid of points $x_l$ centered
at $\mu$ covering 3-$\sigma$ ellipse of $\Sigma_x$,
$l \in \{1,\dots,L\}$
$S := \emptyset$
for $c = 1 \dots k$
    $E := \infty$
    for $i = 1 \dots N$
        if $i \notin S$
            $S := S \cup \{i\}$
            Compute $p_{ij}(x_l)$, $j \in S$
            $e := \mathrm{MSE}(S)$
            if $e < E$
                $E := e, b := i$
            end if
         $S := S \backslash \{i\}$
     end if
 end for
$S := S \cup \{b\}$
end for
```

Figure 5. The greedy camera node selection algorithm.

Fig. 7 depicts an example selection for $k = 3$. Note that even though the selection using the closest heuristic seems to be quite natural because it avoids occlusions with high probability, the greedy, which selects the same nodes as the brute force in this case, better localizes the object along the major axis of its prior, where the uncertainty about its position is higher.

3.2 Experimental Results

We tested our selection algorithm in an experimental setup consisting of 16 web cameras placed around a $22' \times 19'$ room. The horizontal FOV of the cameras used is 47°, and they all look toward the center of the room. The relative positions and orientations of the cameras in the room can be seen in Fig. 8(a). The cameras are hooked up to a PC via an IEEE 1394 (FireWire) interface and can provide 8-bit 3-channel (RGB) raw video at 15 Frames/s. The PC connected to a camera models a camera node with processing and communication capabilities. Each PC is connected to 2 cameras, but the data from each camera is processed independently. The data is then sent to a central PC (cluster head), where further processing is performed.

The simple processing described in Section 2 is performed by the selected nodes and the scan-lines are sent to the cluster head where localization is performed. The object was randomly placed 100 times according to the prior shown in Fig. 8(a). The object to localize is the tip of the tripod as

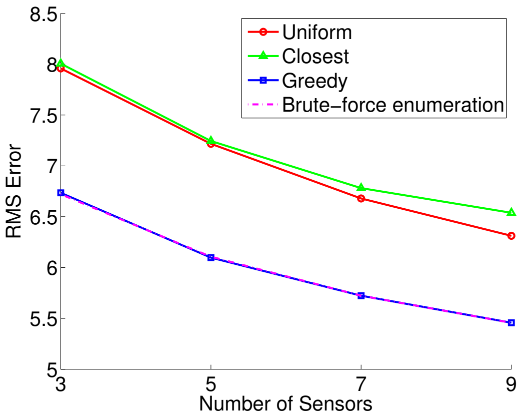


Figure 6. Simulation results: localization performance for different selection heuristics. Camera FOVs are 60°, room radius is 100 units, $D = 2$, $\sigma_{oi} = 5$, $\zeta_i = 0.01$.

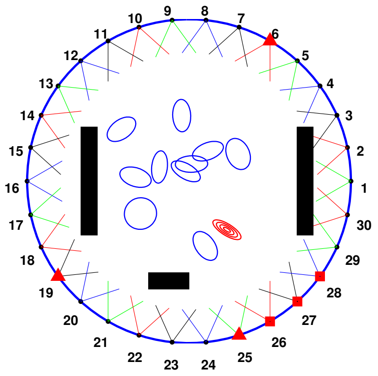
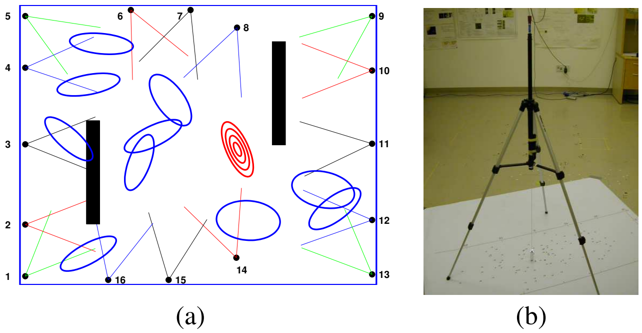


Figure 7. An example selection for $k = 3$. The prior with multiple contours is for the object. The priors with single contour are for the dynamic occluders. The black rectangles are static occluders. Closest heuristic selects cameras marked with squares. Greedy and brute-force select the cameras marked with triangles.

shown in Fig. 8(b). We added 2 static and 10 dynamic "virtual" occluders to the experimental data. We randomly selected priors for the dynamic occluders as before. For each placement of the object, we randomly placed the dynamic occluders, according to these priors. After the camera nodes are selected and queried for measurements, we threw away the measurements from the cameras that would have been occluded, had there been real occluders. The selection procedures were applied with $k = 2$ to 7 camera nodes and the object was localized with the selected nodes for 100 placements using linear estimation. This procedure was repeated 100 times using different random priors for the dynamic occluders. The virtual static occluders have fixed locations throughout. Fig. 9 compares the RMS localization error of the four selection procedures averaged over $100 \times 100 = 10,000$ runs. As can be seen from the figure, the greedy approach again outperforms the other 2 heuristics and performs very close to brute-force enumeration. The experiments confirm that our selection algorithm is useful using real cameras with highly non-linear measurements.



(a) (b)

Figure 8. Experimental setup. (a) The prior with multiple contours is for the object. The priors with single contour are for the dynamic occluders. The black rectangles are static occluders. Cones show FOVs of cameras. (b) The object to be localized.

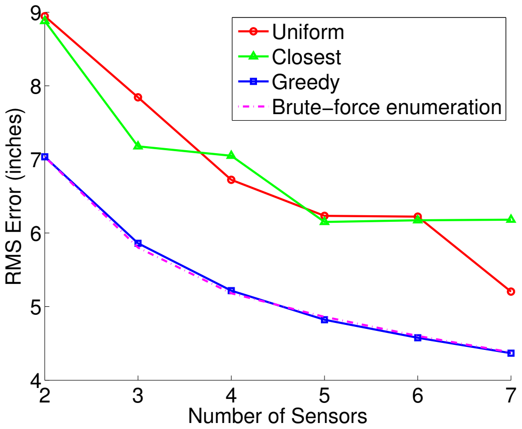


Figure 9. Experimental Results. $\mu = [163.8, 111.4]^T$ **inches (origin is the lower left corner of the room in Fig. 8(a)),** $\Sigma_x = (35.5, -43.5; \ -43.5, 126.5)$ **square inches,** $D = 12$ **inches. The measured noise parameters for the cameras are** $\zeta_i = 0.0012$ **and** $\sigma_{oi} = 1$ **inch.**

4 Conclusion

The paper develops a camera network node selection methodology for target localization in the presence of static and moving occluders. The minimum MSE of the best linear estimate of object position based on camera measurements is used as a metric for selection. It is shown through simulations and experimentally that a greedy selection heuristic performs close to optimal.

Acknowledgments

The authors would like to thank Prof. Jack Wenstrand for helpful discussions. This work is partially supported under DARPA Microsystems Technology Office Award No. N66001-02-1-8940. Leonidas Guibas wishes to acknowledge the support of DoD Multidisciplinary University Research Initiative (MURI) program administered by the Office of Naval Research under Grant N00014-00-1-0637, NSF grant CNS-0435111, and ARO grant W911NF-04-R-0005.

5 References

[1] F. Bian, D. Kempe, and R. Govindan. Utility-based sensor selection. In *Proceedings of IPSN*, pages 11–18, April 2006.

[2] J. Byers and G. Nasser. Utility-based decision-making in wireless sensor networks. Technical Report 2000-014, 1 2000.

[3] M. Chu, H. Haussecker, and F. Zhao. Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. *The International Journal of High Performance Computing Applications*, 16(3):293–313, 2002.

[4] A. Doucet, B.-N. Vo, C. Andrieu, and M. Davy. Particle filtering for multi-target tracking and sensor management. In *Proceedings of the Fifth International Conference on Information Fusion*, pages 474–481, 2002.

[5] A. O. Ercan, D. B.-R. Yang, A. E. Gamal, and L. J. Guibas. Optimal placement and selection of camera network nodes for target localization. In *Proceedings of IEEE International Conference on Distributed Computing in Sensor Systems*, June 2006.

[6] R. Holman and T. Ozkan-Haller. Applying video sensor networks to nearshore environment monitoring. *IEEE Pervasive Computing*, 2(4):14–21, 2003.

[7] V. Isler and R. Bajcsy. The sensor selection problem for bounded uncertainty sensing models. In *Proceedings of IPSN*, pages 151–158, April 2005.

[8] D. Roberts and A. Marshall. Viewpoint selection for complete surface coverage of three dimensional objects. In *Proceedings of the British Machine Vision Conference*, September 1998.

[9] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *Proceedings of IEEE International Conference on Communications*, June 2001.

[10] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

[11] P.-P. Vazquez, M. Feixas, M. Sbert, and W. Heidrich. Viewpoint selection using viewpoint entropy. In *Proceedings of the Vision Modeling and Visualization'01*, November 2001.

[12] D. B.-R. Yang, J.-W. Shin, A. O. Ercan, and L. J. Guibas. Sensor tasking for occupancy reasoning in a network of cameras. In *BASENETS*, October 2004.