Int J Comput Vis (2008) 76: 31–52
DOI 10.1007/s11263-007-0057-9

A General Method for Sensor Planning in Multi-Sensor Systems: Extension to Random Occlusion

Anurag Mittal · Larry S. Davis

Received: 15 April 2004 / Accepted: 3 May 2006 / Published online: 21 July 2007

Abstract Systems utilizing multiple sensors are required in many domains. In this paper, we specifically concern ourselves with applications where dynamic objects appear randomly and the system is employed to obtain some user-specified characteristics of such objects. For such systems, we deal with the tasks of determining measures for evaluating their performance and of determining good sensor configurations that would maximize such measures for better system performance.

We introduce a constraint in sensor planning that has not been addressed earlier: visibility in the presence of random occluding objects. occlusion causes random loss of object capture from certain necessitates the use of other sensors that have visibility of this object. Two techniques are developed to analyze such visibility constraints: a probabilistic approach to determine “average” visibility rates and a deterministic approach to address worst-case scenarios. Apart from this constraint, other important constraints to be considered include image resolution, field of view, capture orientation, and algorithmic constraints such as stereo matching and background appearance. Integration of such constraints is performed via the development of a probabilistic framework that allows one to reason about different occlusion events and integrates different multi-view capture and visibility constraints in a natural way. Integration of the thus obtained capture quality measure across the region of interest yields a measure for the effectiveness of a sensor configuration and maximization of such measure yields sensor configurations that are best suited for a given scenario.

The approach can be customized for use in many multi-sensor applications and our contribution is especially significant for those that involve randomly occurring objects capable of occluding each other. These include security systems for surveillance in public places, industrial automation and traffic monitoring. Several examples illustrate such versatility by application of our approach to a diverse set of different and sometimes multiple system objectives.

Keywords Sensor planning · Surveillance · Camera networks · Camera placement · Multi-camera vision

1 Introduction

Systems utilizing multiple visual sensors are required in many applications. In this paper, we consider applications where such sensors are employed to obtain information about dynamic objects that appear randomly in a monitored region. We assume that the probability distributions of the occurrence of such objects along with probability distributions of object characteristics such as appearance and geometry are known a priori. Also known are the characteristics of the static parts of the scene consisting of geometric and appearance models. Given such information, we address the task of determining quality measures for evaluating the performance of a vision system and of determining sensor configurations that would maximize such a quality measure.

Most of this work was done while A. Mittal was with Real-Time Vision and Modeling Department, Siemens Corporate Research, Princeton, NJ 08540.

A. Mittal (✉)
Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600036, India
e-mail: amittal@cse.iitm.ernet.in

L.S. Davis
Computer Science Department, University of Maryland, College Park, MD 20742, USA
e-mail: lsd@cs.umd.edu

Such analysis is applicable to several domains including surveillance and monitoring, industrial automation, transportation and automotive, and medical solutions. A common objective is to monitor a large area by having the sensors look at different parts of the scene; a typical system goal is to track each object within and across views (camera handoff) as it moves through the scene (Kettnaker and Zabih 1999; Cai and Aggarwal 1999; Collins et al. 2001). Another objective is to utilize multiple closely-spaced cameras for the purpose of accurate stereo matching and reconstruction (Darrell et al. 1998, 2001; Krumm et al. 2000). Such reconstructions can then be fused across the views in 3D space. A sensor configuration might be chosen that sacrifices matching accuracy for better visibility by utilizing widely separated cameras (Mittal and Davis 2002, 2003). Such an approach might be appropriate in more crowded scenes. Another alternative (Grimson et al. 1998; Kelly et al. 1995; Khan et al. 2001; Khan and Shah 2003) is to not match information directly across the views, but to merge the detections observed by multiple sensors in a consistent manner. These systems have different requirements and constraints. Here, we formulate a generic framework that incorporates a variety of such constraints with probabilistic visibility constraints that arise due to occlusion from other objects. Such framework enables analytical evaluation of the performance of a given vision system given the task requirement and maximization of such performance via better sensor placement.

1.1 Prior Work

Sensor planning has been studied extensively. Following (Maver and Bajcsy 1993) and (Tarabanis et al. 1995a), these methods can be classified based on the amount of information available about the scene: (1) No information is available, (2) A set of models for the objects that can occur in the scene are available, and (3) Complete geometric information is available.

1.1.1 Scene Reconstruction

The first set of methods, which may be called scene reconstruction or next view planning, attempt to build a model of the scene incrementally by successively sensing the unknown world from effective sensor configurations using the information acquired about the world up to this point (Miura and Ikeuchi 1995; Ye and Tsotsos 1999; Pito 1999; Cook et al. 1996; Roy et al. 2001; Maver and Bajcsy 1993; Lehel et al. 1999; Kutulakos and Dyer 1994; Armstrong and Antonis 2000; Krishnan and Ahuja 1996; Cameron and Durrant-Whyte 1990; Hager and Mintz 1991). The sensors are controlled based on several criteria such as occlusion, ability to view the largest unexplored region, ability to perform good stereo matching etc.

1.1.2 Model-Based Object Recognition

The second set of methods assume knowledge about the objects that can be present in the scene. The task, then, is to develop sensing strategies for model-based object recognition and localization (Grimson 1986; Hutchinson and Kak 1989; Kim et al. 1985; Magee and Nathan 1987; Deinzer et al. 2003; Roy et al. 2004). Sensing strategies are chosen that are most appropriate for identifying an object or its pose. Typically, such methods involve three steps: (1) Generation of the hypothesis remaining after an observation, (2) Evaluation of such hypothesis to generate information about the occluded parts of the scene, and (3) Determination of the next sensing configuration that best reduces the ambiguity about the object. Such a *hypothesize-and-verify* paradigm involves an expensive search in the sensor parameter space, and a discrete approximation of this space is typically employed. One such method is that of aspect graphs (Gigus et al. 1991; Petitjean et al. 1992; Gigus and Malik 1990; Cameron and Durrant-Whyte 1990; Hutchinson and Kak 1989) that capture the set of features of an object visible from a given viewpoint, grouping together viewpoints that have the same aspect into equivalence classes.

1.1.3 Scene Coverage

Methods that are directly related to ours are those that assume that complete geometric information is available and determine the location of static sensors so as to obtain the best views of a scene. This problem was originally posed and has been extensively considered in the computational geometry literature as the “art-gallery problem” (O’Rourke 1987; Shermer 1992; Urrutia 1997; Aggarwal 1984): Find the minimum set of points G in a Polygon P such that every point of P is visible from some point of G. Here, a simple definition of visibility is defined such that two points are called visible if the straight line segment between them lies entirely inside the polygon. Even this simpler problem was shown to be NP-hard by Lee and Lin (1986). However, Chvátal (1975) showed that the number of points of G will never exceed $\lfloor n/3 \rfloor$ for a simple polygon of n sides. Several researchers have demonstrated geometric combinatorial methods to obtain a “good” approximate solution to the problem (Ghosh 1987; Chin and Ntafos 1988). Furthermore, several extensions of this problem have been considered in the literature that generalize the problem for different types of guards and visibility definitions (Edelsbrunner et al. 1984; Culberson and Reckhow 1989; Kay and Guay 1970; Lingas 1982; Masek 1978; O’Rourke 1982). The reader is referred to (Shermer 1992) and (O’Rourke 1987) for surveys of work done in this field.

Several recent papers have incorporated additional sensor constraints such as incidence angle and range into the problem and reduce the resultant sensor planning problem to the well-known set-cover problem (González-Banos and Latombe 2001; Danner and Kavraki 2000; Gonzalez-Banos et al. 1998; Gonzalez-Banos and Latombe 1998): select a group of sets from a given collection of sets such that the union of such group equals a given set X. Such a set cover problem is again NP-hard and it is also well-known that the general version of the set cover problem cannot be approximated with a ratio better than $\log n$, where n is the size of the covered set X (Slavik 1997). However, for sets systems with a finite so-called VC-dimension d, polynomial time solutions exist that yield a set of size at most $O(d \cdot c \cdot \log c)$, where c is the size of the optimal set. González-Banos and Latombe (2001) use such results from VC-dimensionality in order to obtain a polynomial time algorithm for obtaining a sensor set of size at most $O(c \cdot \log(n+h) \cdot \log(c \log(n+h)))$, where n is the number of sides of a polygon, and h is the number of holes in it. Recent work by Isler et al. (2004) determines the VC-dimensionality of several set systems that are formed by utilizing different visibility and space (2D vs. 3D) assumptions. Such analysis can then be used to determine efficient approximation algorithms for these particular problems.

Several researchers (Cowan and Kovesi 1988; Stamos and Allen 1998; Reed and Allen 2000; Tarabanis et al. 1996; Maver and Bajcsy 1993; Yi et al. 1995; Spletzer and Taylor 2001; Wixson 1994; Abrams et al. 1999) have studied and incorporated more complex constraints based on several factors not limited to (1) resolution, (2) focus, (3) field of view, (4) visibility, (5) view angle, and (6) prohibited regions. The set of possible sensor configurations satisfying all such constraints for all the features in the scene is then determined. There are several different strategies for determining sensor parameter values. Several systems take a *generate-and-test* approach (Sakane et al. 1987, 1992; Yi et al. 1995), in which sensor configurations are generated and then evaluated with respect to the task constraints. order Another set of methods take a *synthesis* approach (Anderson 1982; Tarabanis et al. 1991, 1995b, 1996; Cowan 1988; Cowan and Bergman 1989; Cowan and Kovesi 1988; Stamos and Allen 1998), in which the task constraints are analytically analyzed, and the sensor parameter values that satisfy such analytical relationships are then determined. *Sensor simulation* is another approach utilized by some systems (Ikeuchi and Robert 1991; Raczkowsky and Mittenbuehler 1989). Such systems simulate the observed view given the description of objects, sensors and light sources and evaluate the task constraints in such views. Finally, there has been work that utilizes the *expert systems* paradigm, where expert knowledge of viewing and illumination techniques is used to provide advice about appropriate sensor configurations (Kitamura et al. 1990; Novini 1988). The reader is referred to the survey paper by (Tarabanis et al. 1995a) for further details.

Another related set of methods (Kang et al. 2000; Stuerzlinger 1999; Durand et al. 1997) has focused on finding good sensor positions for capturing a static scene from desirable viewpoints assuming that some geometric information about the scene is available. Bordering on the field of graphics, the main contribution of such methods is to develop efficient methods for determining the view of the scene from different viewpoints. The reader is referred to (Durand 1999) for a survey of such visibility problems that arise in different fields.

1.2 Motivation and Contributions

In addition to the "static" constraints that have been analyzed previously in the literature, there are additional constraints that arise when random occluding objects are present. Such constraints are essential to analyze since system performance is a function of object visibility. In people detection and tracking, for instance, handling occlusion typically requires the construction of motion models during visibility that can then be utilized to interpolate the missing object trajectories (MacCormick and Blake 2000; Zhao et al. 2001). However, if objects are occluded for a significant amount of time, the motion models become unreliable and the tracking has to be reinitialized. Using appearance models and temporal constraints, it might be possible to match these tracks and identify common objects (Khan and Shah 2003; Zhao and Nevatia 2004; Isard and MacCormick 2001). The accuracy of such a labeling, however, is again a function of the frequency and duration of the occlusion and deteriorates significantly with the increase in such duration. Thus, it is important to analyze such occlusion caused by other objects and the effect of such occlusion on system performance. The first part of the paper focuses on developing methods for the analysis of such visibility constraints arising due to the presence of random obstacles. Two types of methods are considered—probabilistic and worst-case (deterministic). The probabilistic approach analyzes visibility constraints for the "average" case, while the deterministic approach analyses worst-case scenarios.

System performance also depends on a number of other constraints such as image resolution, field of view and static obstacles as well as more complex algorithmic constraints such as stereo matching and background appearance. Integration of such constraints with probabilistic visibility constraints is considered in the next part of the paper. Most of the existing work has focused on allowing only specification of *hard* constraints, where any particular constraint has a binary decision regarding its satisfaction at a given location. In reality, many constraints are *soft*, in the sense that certain locations are better captured compared to others. Furthermore, a trade-off is typically involved between different requirements. For instance, a reduction in the distance from the camera enhances resolution but might increase the viewing angle from the camera and cause difficulties in stereo matching. The relative importance of different tradeoffs and the function integrating the different constraints is task-specific and needs to be specified according to the particular application. In this paper, we propose the use of such a function that specifies the quality of the object capture at a particular location from a given set of cameras provided that the object is visible from all of them. Then, a probabilistic framework is developed that allows one to reason about different occlusion events and integrates multi-view capture and visibility constraints in a natural way.

Integration of the capture quality measure across the region of interest yields a measure of the effectiveness of a given sensor configuration for the whole region. Then, the sensor planning problem can be formulated as the optimization of such a measure over the space of possible sensor configurations. Since exact optimization of such a criteria is an NP-hard problem, we propose methods that yield "good" configurations in a reasonable amount of time and are able to improve upon such solutions over time.

The above general method for sensor planning can be applied to many different systems in domains such as surveillance, traffic monitoring and industrial automation. Customization of the method for a given system requirement is performed by specification of the capture quality function that incorporates the different constraints specific to the system objective. The results section of the paper demonstrates the flexibility of the proposed approach in addressing several different system requirements.

The paper is organized as follows. Section 2 develops the theoretical framework for estimating the probability of visibility of an object at a given location in a scene for a certain configuration of sensors. Section 3 introduces some deterministic tools to analyze worst-case visibility scenarios. Section 4 describes the integration of static constraints with probabilistic visibility constraints to develop and then minimize a cost function in order to perform sensor planning in diverse environments. Section 5 concludes with model validation and planning experiments for a diverse set of synthetic and real scenes.

2 Probabilistic Visibility Analysis

In this section, we develop tools for evaluating the probability of visibility of an object from a given set of sensors in the presence of random occluding objects. Since this probability varies across space, it is estimated as a function of object position.

Since the particular application domain might contain either two or three dimensions, we consider the general case of an m dimensional space. Assume that we have a region $\mathcal{R} \subset \mathbb{R}^m$ of "volume" A observed by n sensors [Fig. 1] (The area of $\mathcal{R}$ if $m = 2$, and its volume if $m = 3$). Let $\mathcal{E}_i$ be the event that a target object $\mathcal{O}$ at location $\mathcal{L} \in \mathcal{R}$ in angular orientation $\boldsymbol{\theta}$ is visible from sensor i. The definition of such "visibility" can be defined according to the application (e.g. visibility of only a part of the object might be sufficient) and will be illustrated with an example subsequently. We will develop tools to estimate the following probabilities:

$$\begin{aligned}
&P(\mathcal{E}_i), \quad i = 1, \ldots, n, \\
&P(\mathcal{E}_i \cap \mathcal{E}_j), \quad i, j = 1, \ldots, n, \\
&\vdots \\
&P\left(\bigcap_i \mathcal{E}_i\right).
\end{aligned} \tag{1}$$

Although the reason for such estimation will become fully clear later, one can motivate it by the following observation: The probability that $\mathcal{O}$ is visible from at least one sensor may be expressed mathematically as the union $P(\bigcup_{i=1}^n \mathcal{E}_i)$ of such events, which may be expanded using the inclusion-exclusion principle as:

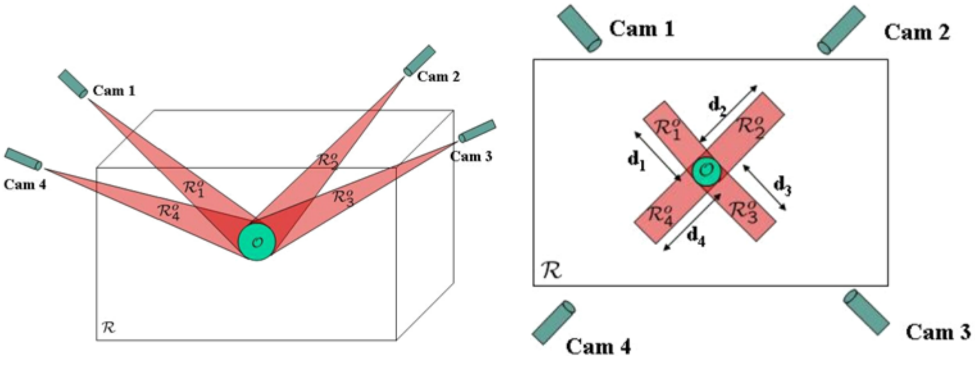
$$\begin{aligned}
P\left(\bigcup_i \mathcal{E}_i\right) &= \sum_{\forall i} P(\mathcal{E}_i) - \sum_{i<j} P(\mathcal{E}_i \cap \mathcal{E}_j) + \cdots \\
&\quad + (-1)^{n+1} P\left(\bigcap_i \mathcal{E}_i\right).
\end{aligned} \tag{2}$$

It is much easier to compute the terms on the RHS (right hand side) than the one on the LHS. The computation of such "intersection" terms is considered next.

In order to develop the analysis, we start with the case of a fixed number of objects in the scene. This will later be extended to the more general case of variable object densities. For ease of modeling, we assume that all objects are identical.

2.1 Fixed Number of Objects

Assume that there are a fixed number, k, of objects in the scene located randomly and uniformly in region $\mathcal{R}$. We first estimate $P(\mathcal{E}_i)$, which is the probability that $\mathcal{O}$ is visible from sensor i. Such visibility may be obstructed by the presence of another object in a certain "region of occlusion" denoted by $\mathcal{R}_i^o$ [Fig. 1]. Such a region of occlusion is dependent on the application as well as on the size and shape of the occluding object. For instance, requiring that all of an object be visible will yield a different region of occlusion than the requirement that only the object center is visible. In any case, given such a region, denote the volume of $\mathcal{R}_i^o$ by $\mathcal{A}_i^o$. Then, we need to estimate the probability that none of the k objects is present in this region of occlusion $\mathcal{R}_i^o$. Since there are k objects in the scene located independently



(a) (b)

Fig. 1 Scene Geometry for (**a**) 3D case, (**b**) 2.5D case, where the sensors have finite heights

of each other, the probability that none of them is present in the region of occlusion is $(1-\frac{A_i^o}{A})^k$. Thus:

$$P(\mathcal{E}_i) = \left(1 - \frac{A_i^o}{A}\right)^k. \tag{3}$$

However, this neglects the fact that two objects cannot overlap each other. In order to incorporate this condition, observe that the $(j+1)$-th object has a possible volume of only $\mathcal{A} - jA_{\mathrm{ob}}$ available to it, where A_{ob} is the volume of an occluding object.[1] Thus, (3) can be refined as

$$P(\mathcal{E}_i) = \prod_{j=0}^{k-1}\left(1 - \frac{A_i^o}{A - jA_{\mathrm{ob}}}\right). \tag{4}$$

This analysis can be generalized to other terms in (1). The probability that the object is visible from all of the sensors in a specified set $(i_1, i_2, \ldots, i_m)$ can be written as:

$$P\left(\bigcap_{i\in(i_1,i_2,\ldots,i_m)} \mathcal{E}_i\right) = \prod_{j=0}^{k-1}\left(1 - \frac{A^o_{(i_1,i_2,\ldots,i_m)}}{A - jA_{\mathrm{ob}}}\right) \tag{5}$$

where $A^o_{(i_1,\ldots,i_m)}$ is the volume of the combined region of occlusion $\mathcal{R}^o_{(i_1,\ldots,i_m)}$ for the sensor set $(i_1,\ldots,i_m)$ formed by the "geometric" union of the regions of occlusion $\mathcal{R}^o_{i_p}$ for the sensors in this set, i.e. $\mathcal{R}^o_{(i_1,\ldots,i_m)} = \bigcup_{p=1}^m \mathcal{R}^o_{i_p}$.

2.2 Uniform Object Density

A fixed assumption on the number of objects in a region is very restrictive. A more realistic assumption is that the objects have a certain density of occupancy. First, we consider the case of uniform object density in the region. This will then be extended to the more general case of non-uniform object density. The uniform density case can be treated as a generalization of the "k objects" case introduced in the previous section. To this end, we increase k and the volume A proportionately such that

$$k = \lambda A \tag{6}$$

where a constant object density λ is assumed. Equation (5) can then be written as

$$P\left(\bigcap_{i\in(i_1,\ldots,i_m)} \mathcal{E}_i\right) = \lim_{k\to\infty}\prod_{j=0}^{k-1}\left(1 - \frac{A^o_{(i_1,\ldots,i_m)}}{k/\lambda - jA_{\mathrm{ob}}}\right). \tag{7}$$

Defining

$$a = \frac{1}{\lambda A^o_{(i_1,\ldots,i_m)}}, \qquad b = \frac{A_{\mathrm{ob}}}{A^o_{(i_1,\ldots,i_m)}} \tag{8}$$

we obtain

$$P\left(\bigcap_{i\in(i_1,\ldots,i_m)} \mathcal{E}_i\right) = \lim_{k\to\infty}\prod_{j=0}^{k-1}\left(1 - \frac{1}{ka - jb}\right). \tag{9}$$

Denoting this limit by L and taking the logarithm of both sides yields:

$$\ln L = \lim_{k\to\infty}\sum_{j=0}^{k-1}\ln\left(1 - \frac{1}{ka - jb}\right).$$

This sum may be approximated via an integral:

$$\ln L \approx \lim_{k\to\infty}\int_0^k \ln\left(1 - \frac{1}{ka - xb}\right)\mathrm{d}x.$$

Such integration may be performed using the method of "integration by parts". Then, one obtains:

$$\ln L \approx \lim_{k\to\infty}\left(\frac{(ka - xb)}{b}\ln(ka - xb)\right.$$

[1] The prohibited volume is in fact larger. For example, for circular 2D objects, another object cannot be placed anywhere within a circle of radius $2r$ (rather than r) without intersecting the object. For simplicity, we redefine A_{ob} as the volume "covered" by the object. This is the volume of the prohibited region and in the 2D case, may be approximated as four times the actual area of the object.

$$-\frac{(ka-xb-1)}{b}\ln(ka-xb-1)\Bigg)\Bigg|_0^k.$$

After some calculations, one obtains:

$$L \approx \lim_{k\to\infty}\left(1-\frac{1}{k(a-b)}\right)^{-k(a-b)/b}\left(1-\frac{1}{ka}\right)^{k(a/b)} \times \left(\frac{k(a-b)-1}{ka-1}\right)^{1/b}.$$

Using some results on limits including the identity $\lim_{x\to\infty}(1+\frac{1}{x})^x = e$, we get:

$$L = P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)}\mathcal{E}_i\Bigg) \approx \left(1-\frac{b}{a}\right)^{1/b} = (1-\lambda A_{\text{ob}})^{(A^o_{(i_1,\dots,i_m)}/A_{\text{ob}})}. \qquad (10)$$

For obtaining this result, we approximated the sum via an integral. Using analysis very similar to the one presented here, it is not too difficult to show that the error in this approximation satisfies

$$e \le \int_0^k \left(\ln\left(1-\frac{1}{ka-xb}\right) - \ln\left(1-\frac{1}{ka-(x-1)b}\right)\right)\mathrm{d}x \to 0 \quad \text{as } k\to\infty.$$

We note at this point that the result obtained here is more accurate than the result presented in our earlier paper (Mittal and Davis 2004). However, for most cases where it can be safely assumed that $a \gg b$, it can be shown that the result in (Mittal and Davis 2004) is a close approximation to the current result. We also note in passing that the problem is related to the M/D/1/1 queuing model used in Queuing Theory (Kleinrock 1975).

So far, we assumed that all objects are identical. We now extend the analysis to the case of probabilistically varying shapes. We first note that in (7), the term jA_{ob} simply adds the contribution from the past j objects. Thus, more precisely, this term is equal to $\sum_i^j A^j_{\text{ob}}$. With sufficiently large j, which is the case when $k\to\infty$, one may approximate this term by $jA^{\text{avg}}_{\text{ob}}$. Then, the only variable left in this term is $A^o_{(i_1,\dots,i_m)}$. Such a region of occlusion is a function of the size of the occluding objects and given the size distribution of such objects, one may estimate the probability distribution $p_{A^o}()$ of $A^o_{(i_1,\dots,i_m)}$. Then, the average visibility probability may be computed as:

$$P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)}\mathcal{E}_i\Bigg) = \int_0^\infty (1-\lambda A^{\text{avg}}_{\text{ob}})^{(A^o_{(i_1,\dots,i_m)}/A^{\text{avg}}_{\text{ob}})} \times p_{A^o}(A^o_{(i_1,\dots,i_m)})\,\mathrm{d}A^o_{(i_1,\dots,i_m)}. \qquad (11)$$

In order to illustrate the computation of the density function $p_{A^o}()$, consider the case of cylindrical objects with radius r. In this case, the area of occlusion may be shown to be equal to $\sum_i 2rd_i$, where d_i's are the distances of occlusion from the object (see Fig. 2 and Sect. 2.4 for more details). Then, if the radii of the objects are normally distributed: $r \sim N(\mu_r,\sigma_r)$, the distribution of the area of occlusion is also normally distributed with mean $\sum_i 2\mu_r d_i$ and variance $\sum_i 4\sigma_r^2 d_i^2$. Using such a distribution function for $A^o_{(i_1,\dots,i_m)}$, one may compute the probability in (11).

2.3 Non-uniform Object Density

In general, the object density (λ) is a function of location. For example, the object density near a door might be higher. Moreover, the presence of an object at a location can influence the object density nearby since objects can appear in groups. We can integrate both of these object density factors with the help of a conditional density function $\lambda(\mathbf{x}_c|\mathbf{x}_O)$ that might be available to us. This density function gives the density at location $\mathbf{x}_c$ given that visibility is being calculated at location $\mathbf{x}_O$. Thus, this function captures the effect that the presence of the object at location $\mathbf{x}_O$ has on the density nearby.[2]

In order to develop the formulation for the case of non-uniform density, we note that the $(j+1)$-th object has a region available to it that is $\mathcal{R}$ minus the region occupied by the j previous objects. This object is located in this "available" region according to the density function $\lambda()$. The probability for this object to be present in the region of occlusion $R^o_{(i_1,\dots,i_m)}$ can then be calculated as the ratio of the average number of objects present in the region of occlusion to the average number of objects in the available region. Thus, one can write:

$$P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)}\mathcal{E}_i\Bigg) = \lim_{k\to\infty}\prod_{j=0}^{k-1}\left(1-\frac{\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}}\lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c}{\int_{\mathcal{R}-\mathcal{R}^j_{\text{ob}}}\lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c}\right) \qquad (12)$$

where $\mathcal{R}^j_{\text{ob}}$ is the region occupied by the previous j objects. Since the previous j objects are located randomly in $\mathcal{R}$, one can simplify:

$$\int_{\mathcal{R}-\mathcal{R}^j_{\text{ob}}}\lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c = \lambda_{\text{avg}}(A-jA_{\text{ob}})$$

[2] Such formulation only captures the first-order effect of the presence of an object. While higher order effects due to the presence of multiple objects can be considered, they are likely to be small.

where λ_{avg} is the average object density in the region. Using this equation in (12) and noting that $\lambda_{\mathrm{avg}} A = k$, we obtain:

$$P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i\Bigg) = \lim_{k\to\infty} \prod_{j=0}^{k-1} \Bigg(1 - \frac{\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}} \lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c}{k - j \cdot \lambda_{\mathrm{avg}} \cdot A_{\mathrm{ob}}}\Bigg). \tag{13}$$

Defining:

$$\begin{aligned} a &= \frac{1}{\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}} \lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c}, \\ b &= \frac{A_{\mathrm{ob}} \cdot \lambda_{\mathrm{avg}}}{\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}} \lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c}. \end{aligned} \tag{14}$$

Equation (13) may be put in the form of (9). As before, this can be simplified to obtain:

$$\begin{aligned} P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i\Bigg) &\approx \left(1 - \frac{b}{a}\right)^{1/b} \\ . \quad &= (1 - \lambda_{\mathrm{avg}} A_{\mathrm{ob}})^{(\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}} \lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c)/\lambda_{\mathrm{avg}} A_{\mathrm{ob}}} \end{aligned} \tag{15}$$

Furthermore, similar to the uniform density case, one may generalize this equation to the case of probabilistically varying object shape:

$$\begin{aligned} &P\Bigg(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i\Bigg) \\ &\quad = \int_0^\infty (1 - \lambda_{\mathrm{avg}} A^{\mathrm{avg}}_{\mathrm{ob}})^{(\int_{\mathcal{R}^o_{(i_1,\dots,i_m)}} \lambda(\mathbf{x}_c|\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_c)/\lambda_{\mathrm{avg}} A^{\mathrm{avg}}_{\mathrm{ob}}} \\ &\qquad \times p_{\mathcal{R}^o}(\mathcal{R}^o_{(i_1,\dots,i_m)})\,\mathrm{d}\mathcal{R}^o_{(i_1,\dots,i_m)} \end{aligned} \tag{16}$$

where $p_{\mathcal{R}^o}()$ is the probability density function for the distribution of the volume of the region of occlusion. This may again be computed from the distribution of the shape of the objects causing the occlusion.

2.4 Models for People Detection and Tracking

We have presented a general method for determining object visibility given the presence of random occluding objects. In this section, we consider model specification for the *2.5D* case of objects moving on a ground plane such that the sensors are placed at some known heights H_i from this plane. The objects are assumed to have the same horizontal profile at each height. Examples of such objects include cylinders, cubes, cuboids, and square prisms, and can adequately describe the objects of interest in many applications such as people detection and tracking. Let the area of their projection onto the ground plane be A_{ob}.

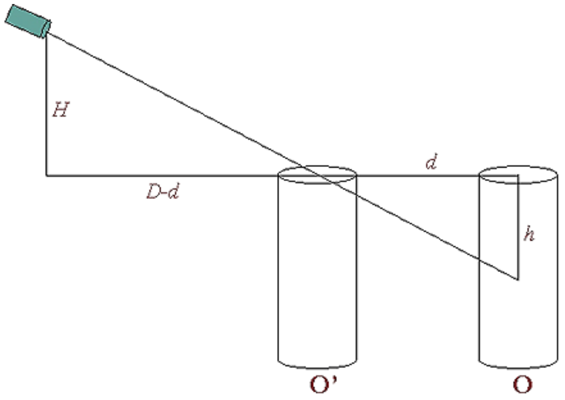


Fig. 2 The distance up to which an object can occlude another object is proportional to its distance from the sensor

A useful quantity may be defined for the objects by considering the projection of the object in a particular direction. We then define r as the average, over different directions, of the maximum distance from the centroid to the projected object points. For e.g., for cylinders, r is the radius of the cylinder; for a square prism with side $2s$, $r = \frac{1}{\pi/4}\int_0^{\pi/4} s\cos\theta\, d\theta = 2\sqrt{2}s/\pi$.

The visibility of an object may be defined according to the requirements of the particular application. For some applications, it may be desirable to view the entire object. For others, it may be sufficient to view the center line only. Furthermore, visibility for a certain height h from the top may be sufficient for some other applications like people detection and tracking. The occlusion region formed is a function of such visibility requirements. For the case of visibility of the center line and desired visibility only up to a length h from the top of the object, this region is a rectangle of width $2r$ and a distance d_i from the object, that is proportional to the object's distance from sensor i [Fig. 2]. Specifically,

$$d_i = (D_i - d_i)\mu_i = D_i \frac{\mu_i}{\mu_i + 1}, \quad \text{where } \mu_i = \frac{h}{H_i}. \tag{17}$$

Assuming that all object orientations are equally likely,[3] one may approximate the area of the region of occlusion R^o_i as $A^o_i \approx d_i(2r)$. Utilizing such models, it is possible to reason about the particular application of people detection and tracking for objects moving on a plane. Such a model will be utilized for the rest of the paper.

The analysis presented so far is probabilistic and provides "average" answers. In high security areas, worst-case analysis might be more appropriate. Such an analysis will be presented in the next section.

[3] It is possible to perform the analysis by integration over different object orientations. However, for ease of understanding, we will use this approximation.

3 Deterministic Worst-Case Visibility Analysis

The probabilistic analysis presented in the last section yields results in the average case. When targets are non-cooperative, a worst-case analysis is more appropriate. In this section, we analyze location-specific limitations of a given system in the worst-case. This analysis provides conditions that guarantee visibility regardless of object configuration and enables sensor placement such that such conditions are satisfied in a given region of interest. Towards this end, we provide the following results:

3.1 Point Objects

For point objects with negligible size, the following can be easily proved:

Theorem 3.1 *Part* 1: *Suppose there is an object $\mathcal{O}$ at location $\mathcal{L}$. If there are k point objects in the vicinity of $\mathcal{O}$, and n sensors have visibility of location $\mathcal{L}$ all with different lines of sight to $\mathcal{L}$, then $n > k$ is the necessary and sufficient condition to guarantee visibility for $\mathcal{O}$ from at least one sensor.*

Part 2: *If visibility from at least m sensors is required, then the condition to be satisfied is $n > k + m - 1$.*

Proof Part 1

(a) *Necessary*

Suppose $n \leq k$. Then, consider the following configuration. Place n objects such that each of them is obstructing one of the sensors (Fig. 3 shows the case for 6 sensors and 6 objects). In this situation, $\mathcal{O}$ is not visible from any of the sensors.

(b) *Sufficient*

Suppose $n > k$. $\mathcal{O}$ has n lines of sight to the sensors. However, there are only k objects that can obstruct these lines of sight. Since the objects are assumed to be point objects, they cannot obstruct more than one sensor. Therefore, by simple application of the pigeon-hole principle, there must be at least one sensor viewing $\mathcal{O}$.

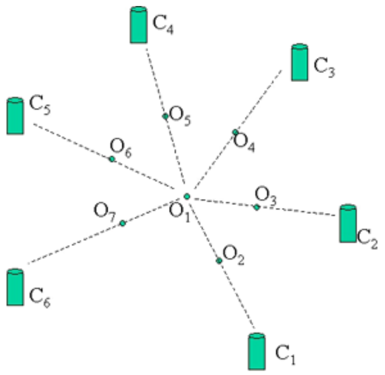


Fig. 3 Six sensors are insufficient in the presence of six potential obstructors

Part 2

(a) *Necessary*

Similar to the reasoning of Part 1(a) above, suppose $n \leq k + m - 1$. Place $p = \min(k, n)$ objects such that each obstructs one sensor. The number of sensors having a clear view of the object are then equal to $n - p$ which is less than m (follows easily from the condition $n \leq k + m - 1$).

(b) *Sufficient*

Suppose $n > k + m - 1$. $\mathcal{O}$ has n lines of sight to the sensors, k of which are possibly obstructed by other objects. Therefore, by the extended pigeon-hole principle, there must be at least $n - k \geq m$ sensors viewing $\mathcal{O}$. □

The actual arrangement of sensors does not matter as long as no two sensors are along the same line of sight from $\mathcal{L}$. This is due to the limitation of considering only point objects.

3.2 2D Finite Objects

Assume that we are given a distinguished point on the object such that object visibility is defined as the visibility of this point. This point may be defined arbitrarily. We will also assume that all objects are identical and define α as the maximum angle that any object can subtend at the distinguished point of any other object. For example, one can consider a flat world scenario where the objects and sensors are in 2D. In such a case, for cylinders with the center of the circular projection as the distinguished point, $\alpha = 60°$; for square prisms, $\alpha = 90°$ [Fig. 4]. Similarly, in $3D$, $\alpha = 60°$ for spherical objects and 60° for cubic ones.

Under these assumptions, the following results hold:

Theorem 3.2 *Part* 1: *Suppose there is an object $\mathcal{O}$ at location $\mathcal{L}$ and there are k identical objects with maximum subtending angle α in the vicinity of $\mathcal{O}$. Also, let n be the cardinality of the largest set of sensors such that all such sensors have visibility of location $\mathcal{L}$ and the angular separation between any two sensors in this set from the point of view $\mathcal{L}$ is at least α. Then, $n > k$ is a necessary and sufficient condition to guarantee visibility for $\mathcal{O}$ from at least one sensor.*

Part 2: *If we want visibility from at least m sensors, then $n \geq k + m - 1$ is the necessary and sufficient condition.*

Proof Part 1. The necessary condition follows directly from Theorem 3.1 Part 1(a). For the sufficiency condition, we note that the distinguished point of $\mathcal{O}$ can be obstructed by another object for a maximum viewing angle of α (Fig. 4). Therefore, if the sensors are separated by an angle $> \alpha$, no single object can obstruct more than one sensor. Since there are only k objects and $n > k$ sensors, by simple application of the pigeon-hole principle, there must be at least one sensor viewing $\mathcal{O}$.

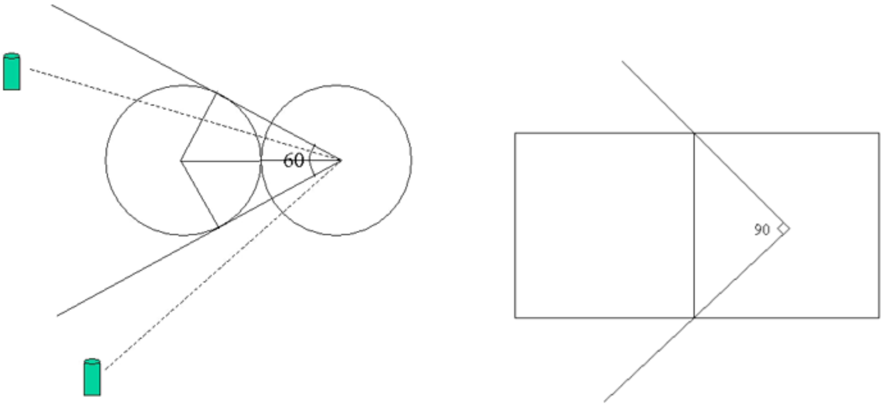


Fig. 4 $\alpha = 60°$ for identical cylinders, $\alpha = 90°$ for square prisms. An angular separation of α between the sensors from the point of view of the object ensures that one object can only obstruct one sensor

It may be noted that in the $2D$ case, n is never more than $\frac{2\pi}{\alpha}$ since it is not possible to place $n > \frac{2\pi}{\alpha}$ sensors such that there is an angular separation of at least α between them. In the $3D$ case, such maximum angle is $4\pi r^2/4\pi r^2 \sin^2(\alpha/4) = 1/\sin^2(\alpha/4)$, calculated as the surface area of a cube divided by the maximum surface area of a patch created by an object subtending an angle α at the center of the cube.

Part 2. The proof is similar to the proof in Part 1. □

An observation may be made here for the region "covered" by any two sensors such that an object will be guaranteed to have a minimum angular separation α between the views of the two sensors. Such a region is a circle passing through the centers of the two sensors such that the angle that the two centers subtend at the center of the circle is 2α [Fig. 5]. This result is derived from the condition that the angle subtended by a chord at any point on a circle is fixed and equal to half the angle subtended by it at the center.

4 Sensor Planning

In the previous sections, we have presented mechanisms for evaluation of the visibility constraints arising due to the presence of random occluders. Other "‘static" constraints also affect the view of the cameras and need to be considered in order to perform sensor planning. We next consider the incorporation of such constraints.

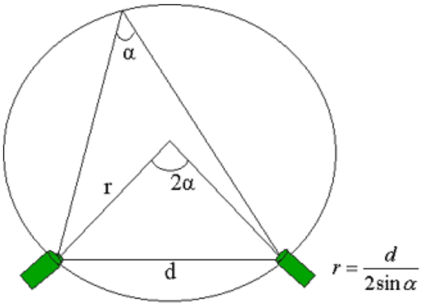


Fig. 5 Every location within the circle has an angle $> \alpha$ to the sensors

4.1 "Static" Constraints

Several stationary factors limit the view of any camera. We first describe such factors briefly and then discuss how they can be incorporated into a generic formulation that enables optimization of the sensor configuration with respect to a user-defined criteria. Such factors may be categorized as: *hard* constraints that *must* be satisfied at the given location for visibility, or *soft* constraints that may be measured in terms of a measure for capture quality.

1. Field of view: Cameras have a limited field of view. Such constraint may be specified in terms of maximum viewing angles from a central direction of the camera, and it can be verified whether a given location is viewable from a given camera.
2. Obstacles: Fixed *high* obstacles like pillars block the view of a camera. From a given location, it can be determined whether any obstacle blocks the view of a particular camera.
3. Prohibited areas: There might also exist prohibited areas where people are not able to walk. An example of such an area is a desk. These areas have a positive effect on the visibility in their vicinity since it is not possible for obstructing objects to be present within such regions.
4. Resolution: The resolution of an object in an image reduces as the object moves further away from the camera. Therefore, useful observations are possible only up to a certain distance from the camera. It is possible to specify such constraint as a *hard* one by specifying a maximum "resolution distance" from the camera. Alternately, such a constraint may be measured in terms of a quality measure (i.e. a *soft* constraint) that deteriorates as we move away from the camera.
5. Algorithmic constraints: Such constraints may involve inter-relationships between the views of several cameras.

Stereo matching across two or more cameras is an example of such a constraint and involves an integration of several factors including image resolution, the maximum distortion of a view that can occur from one camera to the other and the minimum angular separation that would guarantee a certain resolution in depth recovery. Such constraints may again be specified either as a *hard* or *soft*.

6. Viewing angle: An additional constraint exists for the maximum angle $\alpha_{\max}$ at which the observation of an object is meaningful. Such an observation may be the basis for performing some task such as object recognition. This constraint translates into a constraint on the minimum distance from the sensor to an object. This minimum distance guarantees the angle of observation to be smaller than $\alpha_{\max}$. Alternately, a quality measure may be defined that deteriorates as one moves closer to the camera center.

4.2 The Capture Quality

In order to determine the quality or goodness of any given sensor configuration, the "static" constraints need to be integrated into a single *capture quality* function $q_l(\theta)$ that measures how well a particular object at location l in angular orientation θ is captured by the given sensor configuration. Due to occlusions, however, such a quantity is a random variable that depends on the occurrence of events $\mathcal{E}_i$. Thus, one needs to specify the *capture quality* as a function of such events. More specifically, such function needs to be specified for all camera tuples that can be formed from the sensor set, i.e. one needs to determine $\{q_l(i,\theta)\}$, $i = 1, \dots, n$, $\{q_l(i,j,\theta)\}$, $i, j = 1, \dots, n$, and so on. Here, the m-tuple $q_l\ (i_1, \dots, i_m, \theta)$ refers to the capture quality obtained if an object at the location l in angular orientation θ is visible from *all* of the sensors in the m-tuple (i.e. the event $P(\bigcap_{i \in (i_1, \dots, i_m)} \mathcal{E}_i)$ occurs).

To give some insight into such specification, one can consider the case of stereo matching. Then, since visibility from at least two sensors would be required for matching, the capture quality $\{q_l(i,\theta)\}$, $i = 1, \dots, n$ would be zero. For the terms involving two sensors, several competing requirements need to be considered. It has been shown (Mulligan et al. 2001; Rodriguez and Aggarwal 1990; Kamgar-Parsi and Kamgar-Parsi 1989; Georgis et al. 1998; Blostein and Huang 1987) that under some simplifying assumptions, the error in the recovered depth due to image quantization is approximately proportional to $\delta z \approx z^2/bf$, where z is the distance from the cameras, b is the baseline distance between the cameras, and f is focal length. On the other hand, the angular distortion of the image of an object from one camera to the other may be approximated as $\theta_d \approx \tan^{-1}(b/z)$, and is directly related to the accuracy with which stereo matching may be performed. Furthermore, an increase in the distance from the cameras also decreases the projected size of the object, which might further decrease the accuracy of stereo matching. Thus, the accuracy of stereo matching first increases with the distances from the cameras, and then decreases, while the quantization error increases with such distances. Thus, a function that first increases and then decreases as a function of the distance from the cameras might be an appropriate choice for the quality function.

If a multi-camera algorithm is utilized, one may perform a similar (though more complex) analysis for terms involving more than two sensors. In the absence of such an algorithm, one possibility is to consider the quality of the best two pairs in the m-tuple as the quality of the m-tuple.

Thus, one needs to integrate several of the constraints previously described into a single quality function. As in the stereo matching example mentioned above, a trade-off between different constraints is typically involved and it is up to the user to specify functions that define the desired behavior in such conditions.

4.3 Integrating Static Constraints with Probabilistic Visibility

Given the capture quality measures for different m-tuples at a given location, we now present a framework that allows us to determine an overall measure for the capture quality of a sensor configuration at a given location such that the probabilities of visibility from different sensors are taken into consideration. We first partition the event space into the following disjoint sets [Fig. 6]:

No $\mathcal{E}_i$ occurs, with quality: 0
Only $\mathcal{E}_i$ occurs, with quality: $q(\mathcal{E}_i)$
Only $\mathcal{E}_i \cap \mathcal{E}_j$ occurs, with quality: $q(\mathcal{E}_i \cap \mathcal{E}_j)$
$\bigcap_i \mathcal{E}_i$ occurs, with quality: $q(\bigcap_i \mathcal{E}_i)$

Such separation allows one to specify the quality measure for each of such events separately. Then, the computation of probabilities for these disjoint events will yield a probability function for the capture quality at this location (Fig. 15 illustrates an example where the function is averaged over the entire region of interest). While it is possible to utilize such function directly and consider complex integration

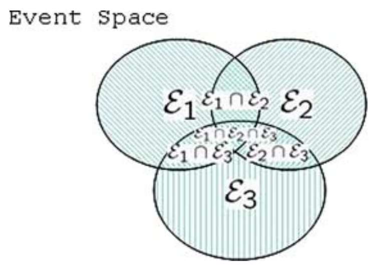

Event Space
$\mathcal{E}_1$
$\mathcal{E}_1 \cap \mathcal{E}_2$
$\mathcal{E}_2$
$\mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3$
$\mathcal{E}_1 \cap \mathcal{E}_3$
$\mathcal{E}_2 \cap \mathcal{E}_3$
$\mathcal{E}_3$

Fig. 6 The event space may be partitioned into disjoint event sets. Here, *only* $\mathcal{E}_i$, for instance, would only include event space that is not common with other events

measures, we assume for simplicity that the mean is a good measure for combining the quality from different events. In order to compute the *mean* capture quality, note that if the event space is partitioned into disjoint sets, the *mean* may be computed directly as the weighted average over all such sets, i.e. if events E_1 and E_2 are disjoint, one may compute: $q_{\text{avg}} = q(E_1) \cdot P(E_1) + q(E_2) \cdot P(E_2)$. So, the *mean* capture quality at a particular location for a particular object orientation θ can be calculated as:

$$\begin{aligned} q(\theta) &= \sum_{\forall i} q(\mathcal{E}_i, \theta) P(\text{Only } \mathcal{E}_i) \\ &\quad + \sum_{i<j} q(\mathcal{E}_i \cap \mathcal{E}_j, \theta) P(\text{Only } \mathcal{E}_i \cap \mathcal{E}_j) \\ &\quad + \cdots + q\left(\bigcap_i \mathcal{E}_i, \theta\right) P\left(\text{Only } \bigcap_i \mathcal{E}_i\right). \end{aligned}$$

This expression may be rearranged to obtain:

$$\begin{aligned} q(\theta) &= \sum_{\forall i} q^c(\mathcal{E}_i, \theta) P(\mathcal{E}_i) \\ &\quad - \sum_{i<j} q^c(\mathcal{E}_i \cap \mathcal{E}_j, \theta) P(\mathcal{E}_i \cap \mathcal{E}_j) \\ &\quad + \cdots + (-1)^{n+1} q^c\left(\bigcap_i \mathcal{E}_i, \theta\right) P\left(\bigcap_i \mathcal{E}_i\right) \end{aligned} \tag{18}$$

where $q^c(\bigcap_{i \in (i_1, \ldots, i_m)} \mathcal{E}_i, \theta)$ is defined as:

$$\begin{aligned} & q^c\left(\bigcap_{i \in (i_1, \ldots, i_m)} \mathcal{E}_i, \theta\right) \\ &\quad = \sum_{i \in (i_1, \ldots, i_m)} q(\mathcal{E}_i, \theta) - \sum_{i<j} q(\mathcal{E}_i \cap \mathcal{E}_j, \theta) \\ &\quad + \cdots + (-1)^{m+1} q\left(\bigcap_{i \in (i_1, \ldots, i_m)} \mathcal{E}_i, \theta\right). \end{aligned} \tag{19}$$

This analysis yields a capture quality measure for each location and each angular orientation for a given sensor configuration. Such quality measure needs to be integrated across the entire region of interest in order to obtain a quality measure for the given configuration. This integration is considered in the next section.

4.4 Integration of Quality Across Space

The analysis presented so far yields a function $q_{\mathbf{s}}(\mathbf{x}, \theta)$ that defines the capture quality of an object with orientation θ at location $\mathbf{x}$ given the sensor configuration defined by the parameter vector $\mathbf{s}$. The parameter vector may include, for instance, the location, viewing direction and zoom of each camera. Given such a function, one can define a suitable *cost* function to evaluate a given set of sensor parameters w.r.t. to the entire region to be viewed. Such sensor parameters may be constrained further due to other factors. For instance, there typically exists a physical limitation on the positioning of the cameras (walls, ceilings etc.). The sensor planning problem can then be formulated as a problem of constrained optimization of the cost function.

Several cost functions may be considered. Based on deterministic visibility analysis, one can consider a simple cost function that sums, over the region of interest $\mathcal{R}_i$, the number $N(\mathbf{x})$ of cameras that a location $\mathbf{x}$ is visible from:

$$C(\mathbf{s}) = -\sum_{\mathbf{x} \in \mathcal{R}_i} N(\mathbf{x}). \tag{20}$$

Using probabilistic analysis, a cost function can be defined that maximizes the minimum quality in the region:

$$C(\mathbf{s}) = -\min_{\mathbf{x} \in \mathcal{R}_i, \theta \in [0, \ldots, 2\pi]} q_{\mathbf{s}}(\mathbf{x}, \theta).$$

Another cost function, and perhaps the most plausible one in many situations, is to define the cost as the negative of the average capture quality in a given region of interest:

$$C(\mathbf{s}) = -\int_{\mathcal{R}_i} \int_0^{2\pi} \lambda(\mathbf{x}, \theta) q_{\mathbf{s}}(\mathbf{x}, \theta) \, d\theta \, d\mathbf{x}. \tag{21}$$

This cost function has been utilized for obtaining the results in this paper. Note that we have added an additional parameter θ to the object density function in order to incorporate information about object orientations into the density function. Since the orientation does not affect the occluding characteristics of an object, this parameter was integrated (and eliminated) for the visibility analysis presented previously.

4.5 Minimization of the Cost Function

While it may be possible to efficiently minimize the cost function when the specified constraints are simple (e.g. see González-Banos and Latombe 2001), minimization for the most general capture quality functions is a difficult and computationally expensive problem. For instance, the cost function obtained in (21) is quite complex and it can be shown that it is not differentiable. Furthermore, in most non-trivial cases, it has multiple local minima and possibly multiple global minima. Figure 7 illustrates the cost function for the scene shown in Fig. 12(a). Where, for illustration purposes, only two of the nine parameters have been varied. Even in this two dimensional space, there are two global minima and several local minima. Furthermore, the gradient is zero in some regions.

Due to these characteristics, some of the common optimization techniques like simple gradient descent or a "set

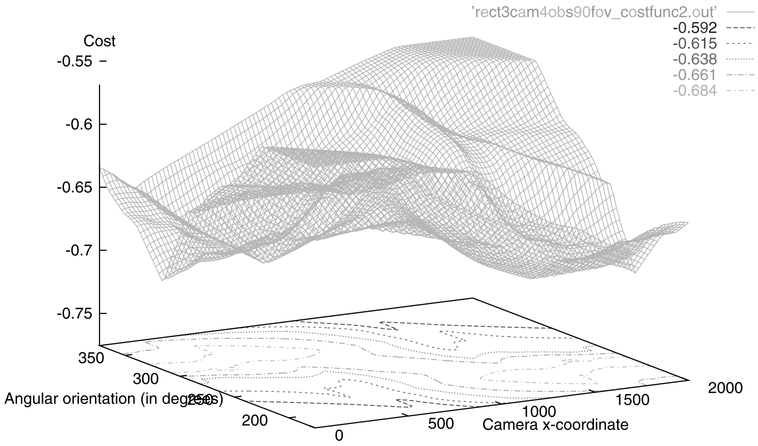


Fig. 7 The Cost Function for the scene in Fig. 12(a) where, for illustration purposes, only the x-coordinate and direction of the second camera have been varied

cover" formulation are not appropriate. Therefore, we consider global minimization techniques that can deal with complex cost functions (Shang 1997). Simulated Annealing and Genetic Algorithms are two classes of algorithms that have commonly been employed to handle such optimization problems. The nature of the cost function suggests that either of these two algorithms should provide an acceptable solution (Duda et al. 2001). For our experiments, we implemented simulated annealing using a sophisticated simulated re-annealing software *ASA* developed by Ingber (Ingber 1989).

Using this algorithm, we obtain extremely good sensor configurations in a reasonable amount of time (5 min—a couple of hours on a Pentium IV 2.2 GHz PC, depending on the desired accuracy of the result, the number of dimensions of the search space and complexity of the scene). For low dimensional spaces (<4), where it was feasible to verify the results using full search, it was found that the algorithm quickly converged to a global minimum. For moderate dimensions of the search space (<8), the algorithm was able to obtain a good solution, but only after some time. Although optimality of the solution could not be verified by full search, we believe the solutions to be close to the optimum since running the algorithm several times from different starting points and using different annealing parameters did not alter the final solution. For very high dimensional spaces (>8), although the algorithm provided reasonably good solutions very quickly, it sometimes took several hours to "jump" to a better solution.

5 Validation and Experiments

We have proposed analytical methods for computing the visibility characteristics of sensor configurations and integrated them with static constraints to provide a framework and an algorithm for recovering good sensor configurations with respect to certain quality measures. We first validate the analytical visibility models using real data. Then, we illustrate the applicability of the sensor planning algorithm by providing planning results for various scenes, synthetic and real.

5.1 Validation of the Visibility Model

In order to validate the analytical visibility analysis developed in this paper, we compare the predicted visibility with the visibility obtained for some real sequences [Fig. 8]. These sequences were captured in a laboratory environment using multiple cameras. Ground truth about person locations was established by using the M_2Tracker algorithm (Mittal and Davis 2003) that detects and tracks people automatically under occlusions using multiple cameras. This people location information was then used to determine the empirical visibility rate in the area where people were allowed to move (of approx. size 3 m × 3 m). Visibility rates were determined for the cases of visibility from k cameras, such that visibility from even one camera is sufficient. Visibility was defined as visibility of the center line of the person. This information was computed over 200 time steps and averaged over all possible (C_k^n) camera k-tuples, where n is the total number of cameras actually available. Different sequences were captured containing different number of people and statistics were obtained for each of them. This information was then compared with the theoretical visibility rate obtained using our models [Fig. 9]. Since a fixed number of people were restricted to move in the region, the analysis that uses a fixed number of objects was utilized for comparison purposes. Since the region is not too crowded, the visibility rates obtained using a uniform density assumption (with density computed as the number of people/area of the region) were quite close to the fixed objects assumption. As can be observed from the plot in Fig. 9, the predicted and actual visibility rates are quite close to each other, which validates the applicability of the analytical models developed in the paper.

Fig. 8 Some images from sequences used to validate the analytical visibility model

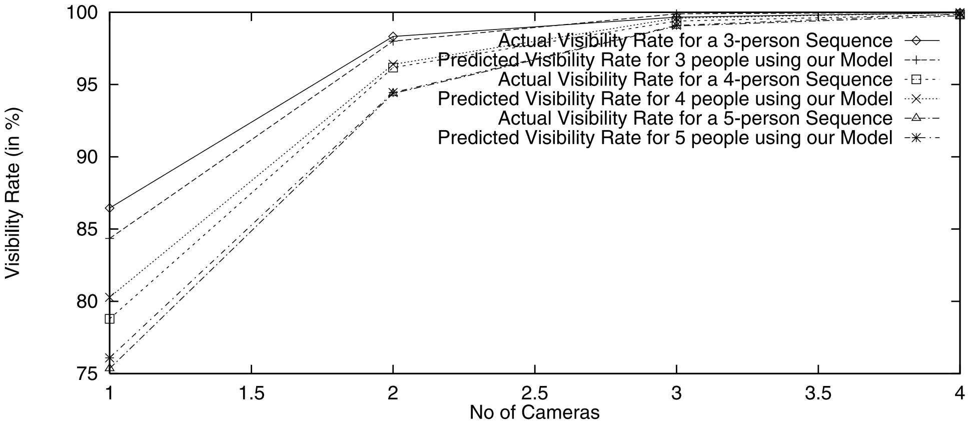


Fig. 9 Comparison of visibility rates obtained using our model with those obtained for real data

5.2 Sensor Planning Experiments

We now present results of the application of the sensor planning algorithm to various scenes. In order to illustrate the algorithm for complex scenes, we first consider synthetic examples. Then we show, for some simple real scenes, how the method may be utilized for sensor placement by utilization of information about object characteristics that may be obtained automatically by utilization of image-based detection and tracking algorithms.

5.2.1 Synthetic Examples

In the synthetic examples, we make the following assumptions. The sensors are mounted $H = 2.5$ m above the ground and have a field of view of 90°. We use a uniform object density $\lambda = 1\ \text{m}^{-2}$, object height = 150 cm, object radius $r = 15$ cm, minimum visibility height $h = 50$ cm and maximum visibility angle $\alpha_{\max} = 45°$. Furthermore, for ease of understanding, the first few examples will assume a simple quality function such that visibility from *any* direction is considered of equal utility and fixed thresholds are put on the visibility distance from the camera based on camera resolution ($maxdist_{\text{res}}$) and maximum viewing angle $\alpha_{\max}$ ($mindist_{\text{view}}$):

$$q_{\mathbf{x}}(\mathcal{E}_i, \theta) = \begin{cases} 1 & \text{if } mindist_{\text{view}} < dist(\mathbf{x}, cam) < maxdist_{\text{res}}, \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

Note that the parameter θ is neglected. Furthermore, for multiple sensor terms $q^c(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i, \theta)$, the quality is defined simply as the quality of the sensor having the best view:

$$q\left(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i, \theta\right) = \max_{i\in(i_1,\dots,i_m)} q(\mathcal{E}_i, \theta). \tag{23}$$

Under this assumption, it is easy to verify that the quantity q^c defined in (19) becomes:

$$q^c\left(\bigcap_{i\in(i_1,\dots,i_m)} \mathcal{E}_i, \theta\right) = \min_{i\in(i_1,\dots,i_m)} q(\mathcal{E}_i, \theta). \tag{24}$$

First, we consider a simple square area of size 10 m × 10 m and determine the number of cameras required for the scene. Figure 10 shows the mean quality maps obtained for the case of one, two, three and four sensors respectively. The maps are scaled such that $[0, 1]$ maps onto $[0, 255]$, thus creating a gray scale image. Brighter regions represent higher quality. Note how the mean capture quality decreases as we move away from a camera due to an increase in occlusion, in turn due to increase in the distance of occlusion d_i. The average capture quality obtained for the four cases were (a) 0.4296, (b) 0.672, (c) 0.8095, and (d) 0.888 respectively. This information can be used to select the appropriate number of cameras based on the application requirements.

In all the synthetic examples we consider next, we consider a rectangular room of size 10 m × 20 m. Figure 11

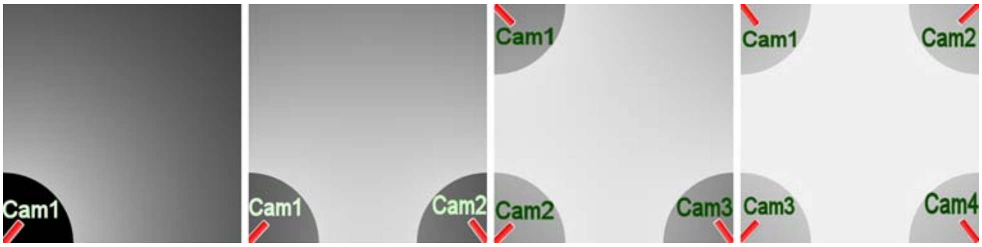


Fig. 10 Maps for the *mean* capture quality for 1, 2, 3 and 4 sensors in a square region. $H = 10$ m, $R = 50\text{ m} \times 50\text{ m}$, $\lambda = 1\text{ m}^{-2}$, $r = 15$ cm, $h = 50$ cm, and $\alpha_{\max} = 30°$. Note how the quality decreases as we move away from a camera due to an increase in occlusion caused by an increase in the distance of occlusion d_i (Fig. 2). The average capture qualities obtained were (a) 0.4296, (b) 0.672, (c) 0.8095, and (d) 0.888 respectively

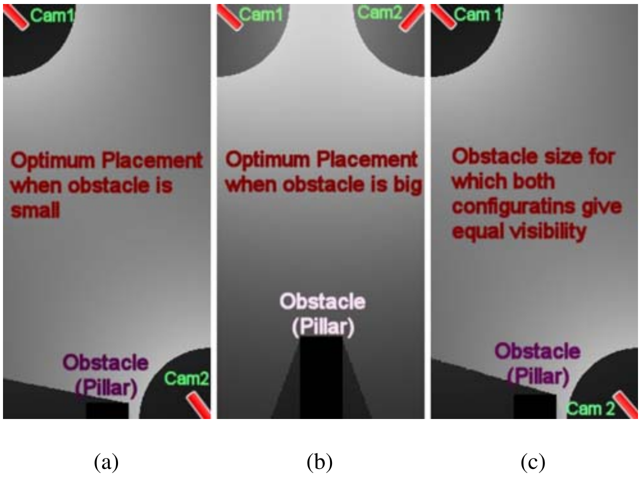

Cam1
Optimum Placement when obstacle is small
Obstacle (Pillar)
Cam2
Cam1
Cam2
Optimum Placement when obstacle is big
Obstacle (Pillar)
Cam 1
Obstacle size for which both configuratins give equal visibility
Obstacle (Pillar)
Cam 2
(a)
(b)
(c)

Fig. 11 Illustration of the effect of scene geometry on sensor placement. Optimum configuration when (**a**): obstacle size is small, (**b**): obstacle size is big, (**c**): obstacle size is such that both configurations are equally good

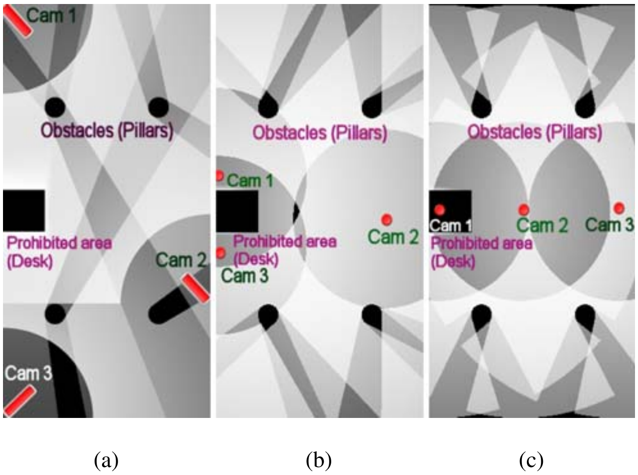


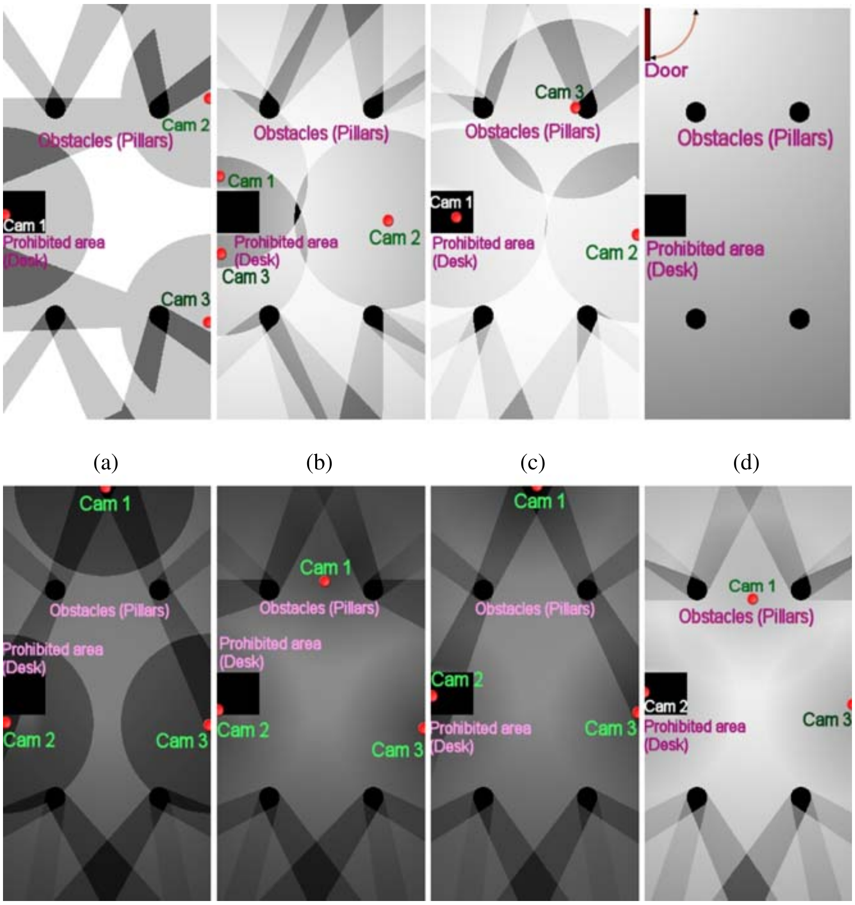
Fig. 12 Illustration of the effect of different camera specifications. With a uniform density assumption and visibility from *any* direction, the optimum configuration when the cameras have: (**a**): field of view of 90° and resolution up to 25 m, (**b**): 360° field of view (Omni-Camera), and resolution up to 25 m, (**c**): 360° field of view, but resolution only up to 10 m

illustrates the effect that an obstacle can have on camera placement. Using a maximum of two cameras having a field of view of 90°, the first configuration [a] was found to be optimum when the obstacle size was small ($<$60 cm). Configuration [b] was optimum when the object size was big ($>$60 cm). For the object size shown in configuration [c] ($\sim$60 cm), both configurations were equally good. Note that in both configurations all locations are visible from at least one camera. Therefore, current methods based solely on analysis of static obstacles would not be able to distinguish between the two.

Figure 12 illustrates how the camera specifications can significantly alter the optimum sensor configuration. Notice that the scene has both obstacles and prohibited areas. With three available cameras, configuration [a] was found to be optimum when the cameras have only 90° field of view but are able to "see" up to 25 m. With the same resolution, configuration [b] is optimum if the cameras have a 360° field of view (Omni-Camera, Nayar 1997; Peleg et al. 2001). If the resolution is lower so that cameras can "see" only up to 10 m, configuration [c] is optimum.

Figure 13 illustrates the effect of different assumptions about the objects and their visibility. With all other assumptions the same as above, configuration [a] was found to be optimum when the worst case analysis was utilized [see (20)]. On the other hand, a uniform object density assumption [see (21)] yielded configuration [b] as the optimum one. When an assumption of variable object densities was utilized such that the density is highest near the door and decreases linearly with the distance from it [d], configuration [c] was found to be the best. Note that a higher object density near the door leads to a repositioning of the cameras so that they can better capture this region.

So far, we have assumed a simple quality function [see (22 and (23)] that ignores the angular orientation θ of the objects and imposes fixed constraints on the camera resolution and viewing angle. We now illustrate how one may



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 13 Illustration of the effect of different object characteristics and visibility requirements. Optimum configuration using: (**a**): object visibility from *any* direction using worst-case analysis [see (20)], (**b**): object visibility from *any* direction using a uniform density [see (21)], (**c**): object visibility from *any* direction using variable densities [see (21)], for the object density shown in (**d**), (**e**): object visibility from *all* directions [see (25)], (**f**): object visibility from *all* directions, with a soft constraint on image resolution [see (26)], (**g**): object visibility from *all* directions, with soft constraint on resolution [see (26)], and using variable densities (**d**), (**h**): object visibility from *all* directions, with soft constraints on resolution and viewing angle [see (26) and (27)]

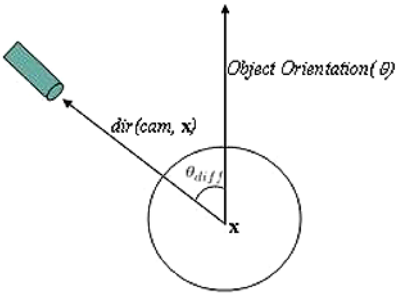


Fig. 14 Computation of the viewing angle θ_{diff}

change this function in order to incorporate more complex visibility requirements. Assuming that one requires visibility from *all* directions, one may alter the quality function as follows:

$$q_{\mathbf{x}}(\mathcal{E}_i,\theta)=\begin{cases}1 & \text{if } \theta_{\text{diff}}<\theta^{\max} \\ & \text{and } d^{\min}_{\text{view}}<dist(\mathbf{x},cam)<d^{\max}_{\text{res}} \\ 0 & \text{otherwise}\end{cases}, \tag{25}$$

where $\theta^{\max}$ is the maximum angular orientation at which the observation of the object is still considered useful, and $\theta_{\text{diff}}=abs(\theta-dir(cam,\mathbf{x}))$ such that $dir(cam,\mathbf{x})$ is the angular direction of the camera from the point of view $\mathbf{x}$ [Fig. 14]. Assuming a uniform density and the above definition of quality, with $\theta^{\max}=90°$, we obtain the sensor configuration shown in [e]. This may be compared with configuration [b]. Note that the cameras are now more spread out in order to capture the objects from many directions.

One may further expand the definition of the quality function in order to incorporate the *camera distance* constraints as soft constraints rather than hard ones. One possible assumption is that the quality decreases linearly with the camera distance when such distance is less than $d^{\min}_{\text{view}}$, and decrease exponentially when such distance is above $d^{\max}_{\text{res}}$:

$$q_{\mathbf{x}}(\mathcal{E}_i,\theta)=H(\theta_{\text{diff}})*\begin{cases}1 \quad \text{if } d^{\min}_{\text{view}}<dist(\mathbf{x},cam)<d^{\max}_{\text{res}}, \\ \frac{dist(\mathbf{x},cam)}{d^{\min}_{\text{view}}} \\ \quad \text{if } dist(\mathbf{x},cam)<d^{\min}_{\text{view}}, \\ \exp\left(-\frac{dist(\mathbf{x},cam)-d^{\max}_{\text{res}}}{d^{\max}_{\text{res}}}\right) \\ \quad \text{if } dist(\mathbf{x},cam)>d^{\max}_{\text{res}}\end{cases} \tag{26}$$

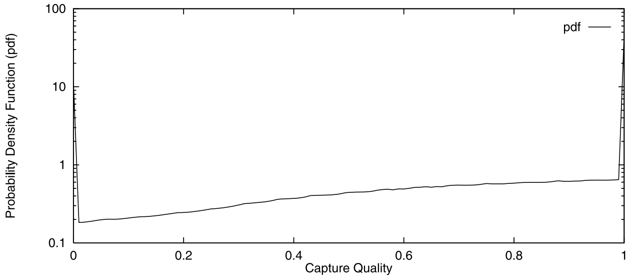


Fig. 15 The probability density function for the capture quality. Note the unusually high values for zero and one capture quality due to the possibilities of complete object occlusion and perfect capture in certain conditions

where $H(\theta_{\text{diff}}) = 1$ if $\theta_{\text{diff}} \geq \theta^{\max}$, $= 0$ otherwise. The sensor configuration obtained for such definition of the quality function is illustrated in [f]. Note that the cameras move inwards compared to configuration [e] because of the increased visibility in the regions close to a camera. Utilization of variable densities with such quality measure leads to configuration [g].

One may further allow a soft constraint on the viewing orientation. One possibility is to assume that the quality deteriorates linearly as the angular orientation θ_{diff} increases between a low and high value. Such factor may be incorporated into above the mentioned quality measure [see (26)] by specifying:

$$H(\theta_{\text{diff}}) = \begin{cases} 1 & \text{if } \theta_{\text{diff}} < \theta^{\min}, \\ \frac{\theta_{\text{diff}} - \theta^{\min}}{\theta^{\max} - \theta^{\min}} & \text{if } \theta^{\min} < \theta_{\text{diff}} < \theta^{\max}, \\ 0 & \text{if } \theta_{\text{diff}} > \theta^{\max}. \end{cases} \tag{27}$$

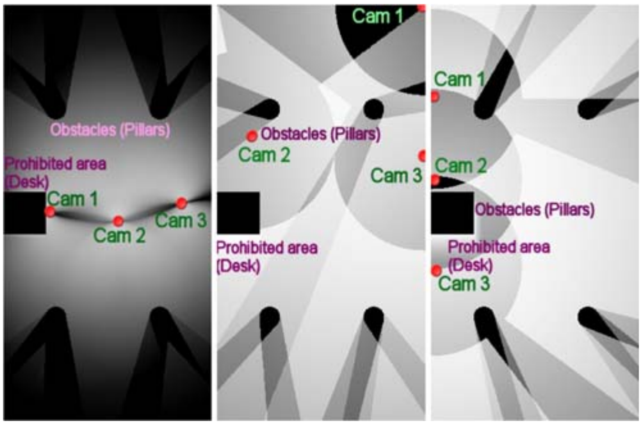
Such quality measure leads to the sensor configuration [h] when $\theta^{\min} = \pi/2$ and $\theta^{\max} = \pi$. Note that camera one moves further inwards compared to configuration [f] since the directional visibility requirement has been made a little less rigid. The probability distribution for the capture quality for this case is shown in Fig. 15. Using such information, one may be able to utilize more complex capture requirements. For instance, one may be able to specify that a certain percentile of the capture quality be maximized.

Next, we consider a stereo scenario in which matching across cameras and 3D reconstruction becomes an additional constraint. One can show that the error in triangulation for an omni-camera is proportional to:

$$e_{tr} \propto \sqrt{d_1^2 + d_2^2 + d_1 d_2 \cos(\alpha)} / \sin(\alpha) \tag{28}$$

where d_1 and d_2 are the distances of the object from the two cameras, and α is the angular separation between the two cameras as seen from the object. Although the error in matching is algorithm-dependent, a reasonable assumption is that:

$$e_m \propto d_1/\cos(\theta/2) + d_2/\cos(\theta/2). \tag{29}$$



(a) (b) (c)

Fig. 16 Illustration of integration of more complex algorithmic constraints. Configuration obtained using three omni-cameras, non-directional object visibility, uniform densities, and: (**a**): a stereo requirement (28), (29), (**b**): three omni-cameras, algorithmic constraint of no visibility with the top wall as background, (**c**): no visibility with the left wall as background

Considering a quality function that uses a weighted average of the two errors: $q = -(w_1 e_{tr} + w_2 e_m)$, configuration Fig. 16(a) was found to the best. Note that all the three cameras come closer to each other in order to be able to conduct stereo matching between any two of them.

In the final example for this scene, we consider a case where, because of algorithmic constraints, capture of an object with one of the walls as background is not useful. For instance, the wall may be painted a certain color and the objects may have a high probability of appearing in this color. Assuming that visibility with the top wall as background is not useful, we obtain configuration Fig. 16(b). The same constraint with the left wall yields configuration Fig. 16(c). Note that some cameras move close to the prohibited wall in order to avoid it as the background. Next, we consider a more complex scene where several constraints are to be satisfied simultaneously. In Fig. 17, the scene of a "museum" is shown where the entrance is on the left upper corner and the exit is on the bottom right corner. One is required to view the faces of people as they enter or exit the scene. For the rest of the area, 3D object localization is to be performed via stereo reconstruction. In the first part of the scene, the four cameras are trying to simultaneously satisfy the tasks of capturing the faces of the entering people in *ROI 1* and performing stereo reconstruction for the rest of the scene. In the middle portion, only stereo is to be performed. Finally, in the last part, the faces of the people leaving the scene in *ROI 2* are to be additionally captured. The difference in sensor placement for the three zones is interesting.

We have illustrated the applicability and generality of the sensor planning algorithm in various synthetic scenar-

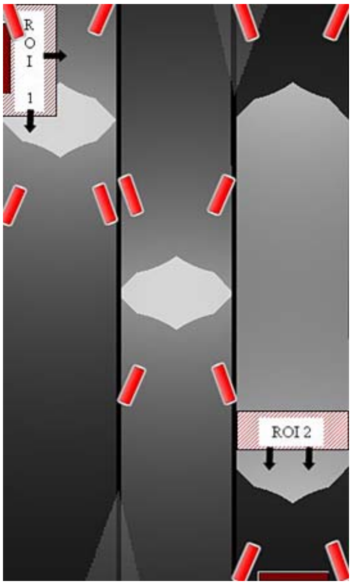


Fig. 17 Sensor Planning in a large "Museum", where several constraints are to be satisfied simultaneously

ios. Next, we will show results from the algorithm in some real scenes.

5.2.2 *Real Scenes*

We first present analysis of sensor placement for a real office room. The structure of the room is illustrated in Fig. 18(a). We used the following parameters—uniform density $\lambda = 0.25\ \text{m}^{-2}$, object height $= 170$ cm, $r = 23$ cm, $h = 40$ cm, and $\alpha_{\max} = 60°$. The cameras available to us had a field of view of 45° and needed to be mounted on the ceiling which is 2.5 m high. In order to view people's face as they enter the room, the quality function was chosen such that it includes only the "entering" object orientation. We first consider the case when there is no panel (separator). If only one camera is available, the best placement was found to be at location $(600, 600)$ at an angle of 135° (measured clockwise from the positive x-axis). If two cameras are available, the best configuration consists of one camera at $(0, 600)$ at an angle of 67.5° and the other camera at $(600, 600)$ at an angle of 132°. Figures 18(b) and (c) show the views from the cameras.

Next, we place a thin panel at location $(300, 300)$–$(600, 300)$. The optimum configuration of two cameras consists of a camera at $(0, 600)$ at an angle of 67.5° (same as before) and the other camera at $(180, 600)$ at an angle of 88°. Figures 18(d) and (e) show the views from the original and new location of the second camera. Next, we consider sensor planning in a small controlled environment [Fig. 20]. In the first experiment, face detection is maximized, while in the second one, we try to maximize person detection via background subtraction and grouping. We utilized an off-the-shelf face detector from OpenCV and characterized its performance over different camera distances and person orientations [Fig. 19]. This gives us the quality function that we need for our sensor planner. Cameras were then placed in the optimum sensor configuration thus obtained and face detection was performed on the video data. We also asked a test user to try to position the cameras manually and experiments were conducted with this configuration as well. Results of this experiment are presented in Figs. 20(a)–(f) and 21. In the next experiment, we tried to maximize person detection using background subtraction and grouping. An additional constraint we considered was that the appearance of one of the actors matched with one of the walls, thus making detection in front of it difficult. This condition was then integrated into the quality function. The results of this experiment are shown in Figs. 20(g)–(l) and 21. The actual rates were quite close to the predicted rates, the difference being possibly due to the small experimental data sizes used for the experiments and inaccuracies in the models utilized. Inspite of these differences, the relative performance of the different configurations was correctly predicted by the sensor planner, allowing for effective planning of the sensors.

In the next example, we consider camera placement in the lobby of a building, where the objective was to capture the faces of people as they enter[4] [Fig. 22]. Video was captured from an existing camera over a period of a couple of hours and a common background subtraction method (Stauffer and Grimson 2000) was utilized in order to detect foreground pixels. Spatial integration and reasoning on top of such pixel-level detection (Greiffenhagen et al. 2000; Paragios and Ramesh 2001) yields estimates of the position of the people on the ground plane. Such information was then averaged over time in order to determine the object densities at different portions on the plane. However, partial or total occlusions cannot be handled by a single camera and thus the algorithm fails to detect people that are occluded by other people. Other methods that utilize temporal information to track objects over time (Zhao and Nevatia 2003, 2004; Isard and MacCormick 2001; Elgammal and Davis 2001) or use multiple cameras to improve object visibility (Mittal and Davis 2003) could be utilized to improve such estimation. Furthermore, we found that there were long periods of inactivity followed by bursts of activity where several people appear suddenly in a group. Therefore, we considered only those portions of the video that contained some activity in order to determine the object densities.

[4] Thus, the quality function includes only "entering" object orientations near the door.

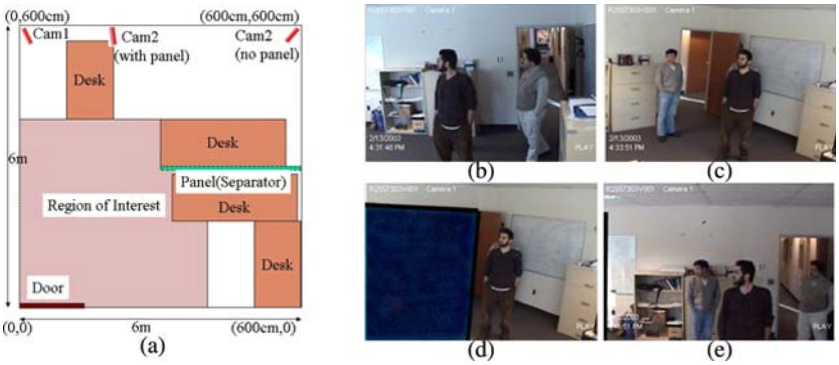


Fig. 18 (**a**) Plan view of a room used for a real experiment. (**b**) and (**c**) are the views from the optimum camera locations when there is no panel (obstacle). Note that, of the three people in the scene, one person is occluded in each view. However, all of them are visible from at least one of the views. Image (**d**) shows the view from the second camera in the presence of the panel. Now, one person is not visible in any view. To improve visibility, the second camera is moved to (180, 600). The view from this new location is shown in (**e**), where all people are visible again

Distance	Face Detection Rate
1.8 m–2.5 m	97.5%
2.5 m–3.1 m	94%
3.1 m–3.8 m	92.5%
3.8 m–4.5 m	85%
4.5 m–5.2 m	77%
5.2 m–6 m	40%
>6 m	0 %

Fig. 19 Empirical face detection rates for different distances from the cameras for the face detector from OpenCV. Additionally, detection rates reduced by about 30% from the frontal to the side view. This information is used by the sensor planner in the quality function

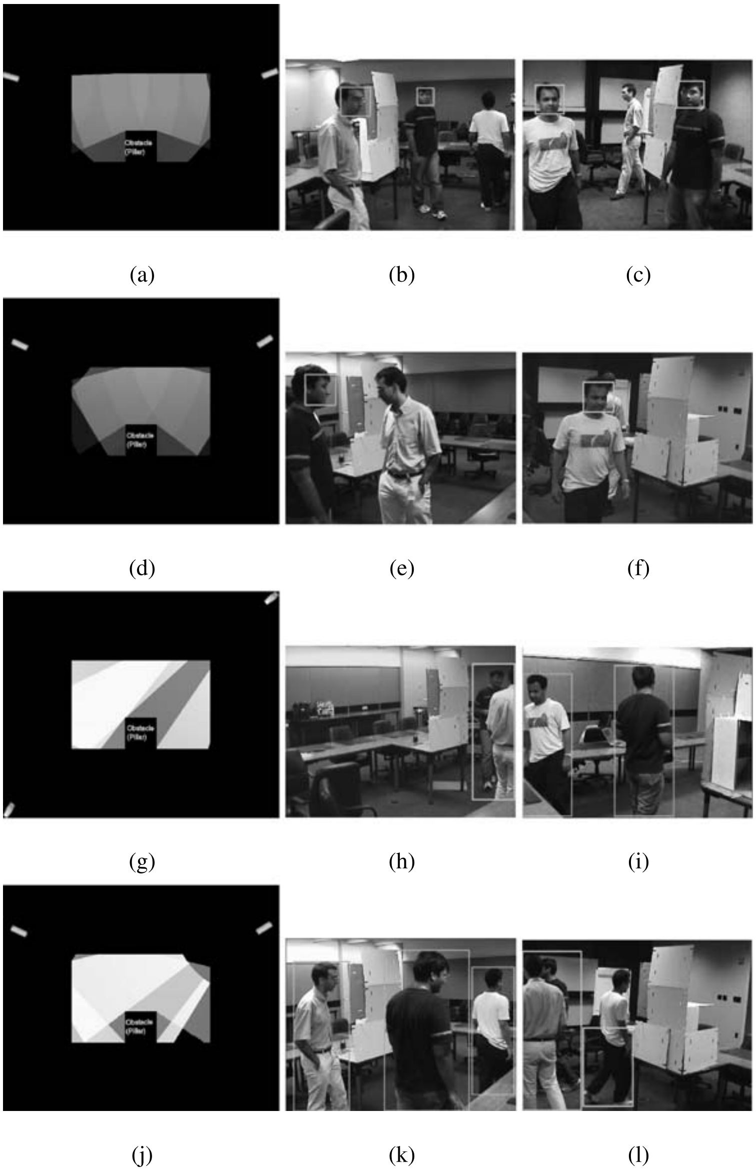
Utilizing such automatic algorithm, we were able to obtain the object density shown in Fig. 22(c). This object density was then utilized to identify a better location for the camera 22(d). The average visibility probability predicted was about 72%, while the actually obtained probability was 78%. In order to improve the visibility probability, if a second camera is also utilized, the two cameras in optimum configuration [e] achieve about 93% visibility (91% visibility predicted). Using two cameras, one may want to obtain 3D information via stereo matching. Utilization of such a constraint leads to the sensor configuration [f]. Note that the two cameras are much closer to each other in order to minimize the image distortion across the views. When the cameras were optimized for face detection, configuration [g] was obtained, while fixing the position of the camera but adjusting only the zoom and camera rotation led to configuration [h]. The images obtained from these configurations and the results of the face detector on such images is shown in images [i] and [j].

6 Discussion and Conclusion

We addressed the problem of sensor placement in multi-camera systems, especially those that are deployed to capture certain characteristics of dynamic objects that appear and move randomly in a specified region. While static constraints such as those due to image resolution, camera field-of-view, focus, static obstacles and the chosen algorithm have been well studied in the literature, occlusion due to the presence of other dynamic objects has not been considered. Such occlusion is random and necessitates a probabilistic formulation of the problem.

We developed two different approaches for modeling the constraints arising due to the presence of random occluding objects. The first approach utilizes a probabilistic framework to determine the average rates of visibility of an object from a given set of sensors. The second method evaluates worst-case scenarios and provides conditions that would guarantee visibility regardless of object configuration.

Integration of constraints due to occlusion from other objects with other types of constraints requires special care due to the random nature of the occlusion. On the other hand, all static constraints may be integrated into a single capture quality function that denotes the quality of the object acquisition when the object is visible from all of the sensors in a given set. This led us to the development of a framework for integration of probabilistic occlusion constraints with static constraints. Such an analysis yields information about the probability distribution of the capture quality at a particular location. Integration of this information over a given region of interest and maximization of appropriately chosen integrated measures over the space of possible camera parameters then leads to a method for determining good sensor locations. Selection of different measures for the quality function and different optimization criteria facilitate customization of the method to a variety of multi-sensor systems.


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Fig. 20 (**a**): Configuration of two cameras for optimum face detection. (**b**) and (**c**): Sample images captured from these camera locations. Note that some of the faces are not detected because of a large viewing angle or errors in the face detector. (**d**): Configuration selected by a human operator. (**e**) and (**f**): Sample images captured from this camera configuration. (**g**): Configuration of two cameras for person detection using background subtraction, where the right wall matches the color of people 33% of the time. (**h**) and (**i**): Sample images captured from the optimal camera locations. (**j**): Configuration selected by a human operator. (**k**) and (**l**): Sample images from the camera configuration in (**j**). Note how the top portion of one person is not detected due to similarity with the background

Fig. 21 Detection rates predicted by the algorithm compared with the actual rates obtained from experimental data

	Face Detection		Person Detection	
	w/ planning	w/o planning	w/ planning	w/o planning
Predicted	53.6%	48%	85%	81%
Actual	51.33%	42%	82%	76%

The utilization of the system for a given scenario requires that certain information is available. This includes not only the scene geometry and appearance, but also estimates of the probability distributions of the geometry and appearance of the objects that are expected to be present. In some scenarios, it may be possible to compute such distributions by utilization of advanced algorithms and additional cameras during training (Zhao and Nevatia 2003, 2004; Isard and MacCormick 2001; Elgammal and Davis 2001; Mittal and Davis 2003). Furthermore, if the object distributions are expected to be similar across a large area or across multiple locations, then it may be possible to compute such information from running such algorithms on only a part of the scene. In some other scenes, it may be sufficient to specify approximations to such distributions by hand.

The framework developed in the paper is perhaps most relevant to surveillance where many target areas are crowded and require multiple cameras for sufficient visibility and

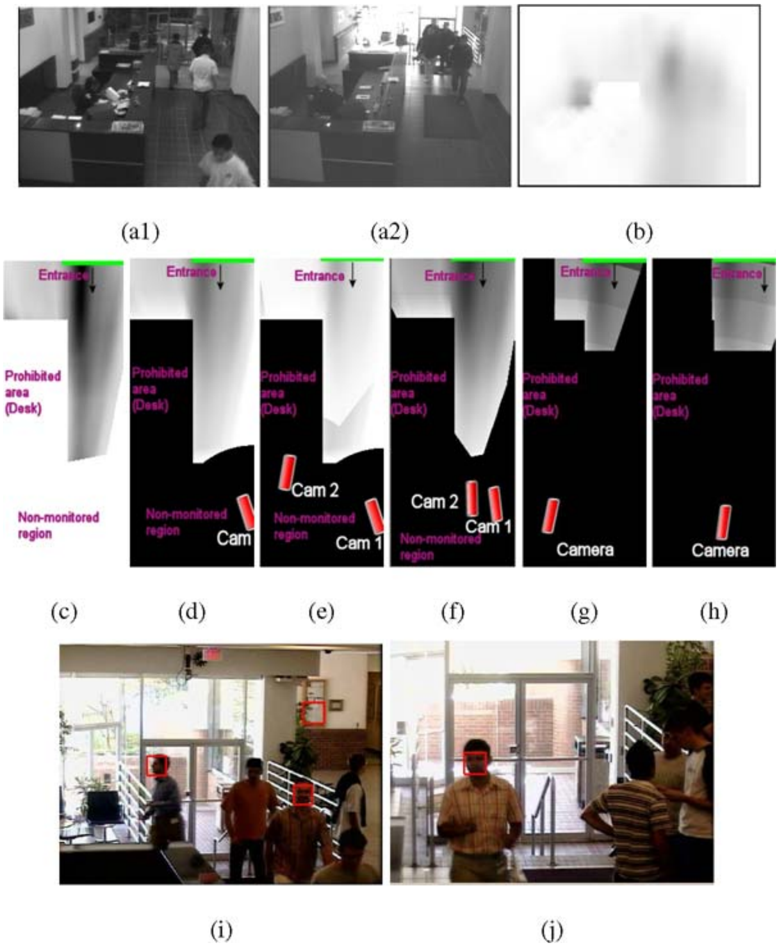


Fig. 22 Sensor placement in a lobby. (**a**): Two views from an original camera location at different times of the day. (**b**): Density map obtained via background subtraction (darker represents higher object density). (**c**): Mapping of the density map onto a plan view of the scene. (**d**): Optimal object visibility using one camera (72% visibility predicted, 78% obtained). (**e**): Optimal sensor placement using two cameras (91% visibility predicted, 93% obtained). (**f**): Optimal sensor placement using two cameras and a stereo requirement. (**g**): Optimization of face detection for people entering the building (46% detection predicted, 43% obtained). An example of face detection using this sensor setting is shown in (**i**). (**h**): Optimization of face detection when the position of the camera cannot be changed (but the direction and zoom can) (33% detection predicted, 35% obtained). Note that people turning right cannot be detected in this configuration. An example of face detection using this setting is shown in (**j**)

system performance. Another application domain is that of industrial automation where objects arrive randomly, for e.g. on a belt, and the vision system is utilized for providing real-time information for intelligent control of such objects. Other possible application domains include traffic monitoring and light control via "smart" vision sensor input and cameras mounted on cars for enhancing driver safety. paper could be cameras. Future work on this topic includes specification of more complex cost functions, investigation of more efficient methods for optimization of specific cost functions and better estimation of the visibility probability by considering the effect of long-term interaction between objects.

Acknowledgements We would like to thank Nikos Paragios for helpful comments on the manuscript and suggesting improvements and Visvanathan Ramesh for his support and helpful discussions on the topic.

References

Abrams, S., Allen, P. K., & Tarabanis, K. (1999). Computing camera viewpoints in a robot work-cell. *International Journal of Robotics Research*, *18*(3), 267–285.

Aggarwal, A. (1984). *The art gallery theorem: its variations, applications, and algorithmic aspects*. PhD thesis, Johns Hopkins University, Baltimore, MD.

Anderson, D. (1982). Efficient algorithms for automatic viewer orientation. *Computers & Graphics*, *9*(4), 407–413.

Armstrong, P., & Antonis, J. (2000). The construction of 3 dimensional models using an active vision system. In *European conference on computer vision* (Vol. II, pp. 182–196), Dublin, Ireland.

Blostein, S., & Huang, T. (1987). Error analysis in stereo determination of 3-D point positions. *Pattern Analysis and Machine Intelligence*, *9*(6), 752–766.

Cai, Q., & Aggarwal, J. (1999). Tracking human motion in structured environments using a distributed-camera system. *Pattern Analysis and Machine Intelligence*, *21*(11), 1241–1247.

Cameron, A., & Durrant-Whyte, H. (1990). A Bayesian approach to optimal sensor placement. *International Journal of Robotics Research*, *9*(5), 70–88.

Chin, W.-P., & Ntafos, S. (1988). Optimum watchman routes. *Information Processing Letters*, *28*, 39–44.

Chvátal, V. (1975). A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory Series B*, *18*, 39–41.

Collins, R., Lipton, A., Fujiyoshi, H., & Kanade, T. (2001). Algorithms for cooperative multi-sensor surveillance. *Proceedings of the IEEE*, *89*(10), 1456–1477.

Cook, D., Gmytrasiewicz, P., & Holder, L. (1996). Decision-theoretic cooperative sensor planning. *Pattern Analysis and Machine Intelligence*, *18*(10), 1013–1023.

Cowan, C. K. (1988). Model based synthesis of sensor location. In *IEEE conference on robotics and automation* (pp. 900–905).

Cowan, C. K., & Bergman, A. (1989). Determining the camera and light source location for a visual task. In *IEEE conference on robotics and automation* (pp. 509–514).

Cowan, C. K., & Kovesi, P. (1988). Automatic sensor placement from vision test requirements. *Pattern Analysis and Machine Intelligence*, *10*(3), 407–416.

Culberson, J., & Reckhow, R. (1988). Covering polygons is hard. In *29th symposium on foundations of computer science* (pp. 601–611).

Danner, T., & Kavraki, L. (2000). Randomized planning for short inspection paths. In *IEEE conference on robotics and automation* (pp. 971–976).

Darrell, T., Gordon, G., Harville, M., & Woodfill, J. (1998). Integrated person tracking using stereo, color, and pattern detection. In *IEEE international conference on computer vision and pattern recognition* (pp. 601–608), Santa Barbara, CA.

Darrell, T., Demirdjian, D., Checka, N., & Felzenszwalb, P. (2001). Plan-view trajectory estimation with dense stereo background models. In *IEEE international conference on computer vision* (Vol. II, pp. 628–635), Vancouver, Canada.

Deinzer, F., Denzler, J., & Niemann, H. (2003). Viewpoint selection—planning optimal sequences of views for object recognition. In *CAIP03* (pp. 65–73).

Duda, R., Hart, P., & Stork, D. (2001). *Pattern classification*. New York: Wiley.

Durand, F. (1999). *3D visibility: analytical study and applications*. PhD thesis, Université Joseph Fourier, Grenoble I. http://www-imagis.imag.fr.

Durand, F., Drettakis, G., & Puech, C. (1997). The 3D visibility complex: a unified data-structure for global visibility of scenes of polygons and smooth objects. In *9th Canadian conference on computational geometry*.

Edelsbrunner, H., Rourke, J., & Welzl, E. (1984). Stationing guards in rectilinear art galleries. *CVGIP* (pp. 167–176).

Elgammal, A., & Davis, L. (2001). Probabilistic framework for segmenting people under occlusion. In *IEEE international conference on computer vision* (Vol. II, pp. 145–152).

Georgis, N., Petrou, M., & Kittler, J. (1998). Error guided design of a 3D vision system. *Pattern Analysis and Machine Intelligence*, *20*(4), 366–379.

Ghosh, S. (1987). Approximation algorithms for art gallery problems. In *Canadian information processing society congress*.

Gigus, Z., & Malik, J. (1990). Computing the aspect graph for line drawings of polyhedral objects. *Pattern Analysis and Machine Intelligence*, *12*(2), 113–122.

Gigus, Z., Canny, J., & Seidel, R. (1991). Efficiently computing and representing aspect graphs of polyhedral objects. *Pattern Analysis and Machine Intelligence*, *13*(6), 542–551.

Gonzalez-Banos, H., & Latombe, J.-C. (1998). Planning robot motions for range-image acquisition and automatic 3D model construction. In *AAAI Fall symposium*.

González-Banos, H., & Latombe, J. (2001). A randomized art-gallery algorithm for sensor placement. In *SCG*, Medford, MA.

Gonzalez-Banos, H., Guibas, L., Latombe, J.-C., LaValle, S., Lin, D., Motwani, R., & Tomasi, C. (1998). Torsion planning with visibility constraints: Building autonomous observers. In *Robotics research—the eighth international symposium* (pp. 95–101).

Greiffenhagen, M., Ramesh, V., Comaniciu, D., & Niemann, H. (2000). Statistical modeling and performance characterization of a real-time dual camera surveillance system. In *IEEE international conference on computer vision and pattern recognition* (Vol. II, pp. 335–342), Hilton Head, SC.

Grimson, W. (1986). Sensing strategies for disambiguating among multiple objects in known poses. *IEEE Transactions on Robotics and Automation*, *2*(4), 196–213.

Grimson, W., Stauffer, C., Romano, R., & Lee, L. (1998). Using adaptive tracking to classify and monitor activities in a site. In *IEEE international conference on computer vision and pattern recognition*, Santa Barbara, CA.

Hager, G., & Mintz, M. (1991). Computational methods for task-directed sensor data fusion and sensor planning. *International Journal of Robotics Research*, *10*(4), 285–313.

Hutchinson, S., & Kak, A. (1989). Planning sensing strategies in robot work cell with multi-sensor capabilities. *IEEE Transactions on Robotics and Automation*, *5*(6), 765–783.

Ikeuchi, K., & Robert, J. (1991). Modeling sensor detectability with VANTAGE geometric/sensor modeler. *IEEE Transactions on Robotics and Automation*, *7*, 771–784.

Ingber, L. (1989). Very fast simulated re-annealing. *Mathematical Computer Modeling*, *12*, 967–973.

Isard, M., & MacCormick, J. (2001). BraMBLe: a Bayesian multiple-blob tracker. In *IEEE international conference on computer vision* (Vol. II, pp. 34–41), Vancouver, Canada.

Isler, V., Kannan, S., Daniilidis, K., & Valtr, P. (2004). VC-dimension of exterior visibility. *Pattern Analysis and Machine Intelligence*, *26*(5), 667–671.

Kamgar-Parsi, B., & Kamgar-Parsi, B. (1989). Evaluation of quantization error in computer vision. *Pattern Analysis and Machine Intelligence*, *11*(9), 929–940.

Kang, S., Seitz, S., & Sloan, P. (2000). Visual tunnel analysis for visibility prediction and camera planning. In *IEEE international conference on computer vision and pattern recognition* (Vol. II, pp. 195–202), Hilton Head, SC.

Kay, D., & Guay, M. (1970). Convexity and a certain property P_m. *Israel Journal of Mathematics*, *8*, 39–52.

Kelly, P. H., Katkere, A., Kuramura, D., Moezzi, S., Chatterjee, S., & Jain, R. (1995). An architecture for multiple perspective interactive video. In *Proceedings of the third ACM international conference on multimedia* (pp. 201–212).

Kettnaker, V., & Zabih, R. (1999). Counting people from multiple cameras. In *ICMCS* (Vol. II, pp. 253–259).

Khan, S., & Shah, M. (2003). Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. *Pattern Analysis and Machine Intelligence*, *25*(10), 1355–1360.

Khan, S., Javed, O., Rasheed, Z., & Shah, M. (2001). Human tracking in multiple cameras. In *IEEE international conference on computer vision* (Vol. I, pp. 331–336), Vancouver, Canada.

Kim, H., Jain, R., & Volz, R. (1985). Object recognition using multiple views. In *IEEE conference on robotics and automation* (pp. 28–33).

Kitamura, Y., Sato, H., & Tamura, H. (1990). An expert system for industrial machine vision. In *International conference on pattern recognition* (Vol. I, pp. 771–774).

Kleinrock, L. (1975). *Queuing systems, Vol. I: Theory*. New York: Wiley.

Krishnan, A., & Ahuja, N. (1996). Panoramic image acquisition. In *IEEE international conference on computer vision and pattern recognition* (pp. 379–384), San Francisco, CA.

Krumm, J., Harris, S., Meyers, B., Brumitt, B., Hale, M., & Shafer, S. (2000). Multi-camera multi-person tracking for easy living. In *Visual surveillance*.

Kutulakos, K., & Dyer, C. (1994). Recovering shape by purposive viewpoint adjustment. *International Journal of Computer Vision*, *12*(2–3), 113–136.

Lee, D., & Lin, A. (1986). Computational complexity of art gallery problems. *IEEE Transactions on Information Theory*, *32*, 276–282.

Lehel, P., Hemayed, E., & Farag, A. (1999). Sensor planning for a trinocular active vision system. In *IEEE international conference on computer vision and pattern recognition* (Vol. II, pp. 306–312), Ft. Collins, CO.

Lingas, A. (1982). The power of non-rectilinear holes. In *9th colloquium on automata, languages, and programming* (pp. 369–383).

MacCormick, J., & Blake, A. (2000). A probabilistic exclusion principle for tracking multiple objects. *International Journal of Computer Vision*, *39*(1), 57–71.

Magee, M., & Nathan, M. (1987). Spatial reasoning, sensor repositioning and disambiguation in 3D model based recognition. In *Workshop on spatial reasoning and multi-sensor fusion* (pp. 262–271).

Masek, W. (1978). *Some NP-complete set covering problems*. Technical report, manuscript, MIT.

Maver, J., & Bajcsy, R. (1993). Occlusions as a guide for planning the next view. *Pattern Analysis and Machine Intelligence*, *15*(5), 417–433.

Mittal, A., & Davis, L. (2002). M_2 tracker: a multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In *European conference on computer vision* (Vol. I, p. 18), Copenhagen, Denmark.

Mittal, A., & Davis, L. (2003). M_2 tracker: a multi-view approach to segmenting and tracking people in a cluttered scene. *International Journal of Computer Vision*, *51*(3), 189–203.

Mittal, A., & Davis, L. (2004). Visibility analysis and sensor planning in dynamic environments. In *European conference on computer vision* (Vol. III, p. 543), Prague, Czech Republic.

Miura, J., & Ikeuchi, K. (1995). Task-oriented generation of visual sensing strategies. In *IEEE international conference on computer vision* (pp. 1106–1113), Boston, MA.

Mulligan, J., Isler, V., & Daniilidis, K. (2001). Performance evaluation of stereo for tele-presence. In *IEEE international conference on computer vision* (Vol. II, pp. 558–565).

Nayar, S. K. (1997). Catadioptric omnidirectional camera. In *IEEE international conference on computer vision and pattern recognition*. Puerto Rico.

Novini, A. (1988). Lighting and optics expert system for machine vision. In *Optics, illumination, image sensing* (pp. 1005–1019).

O'Rourke, J. (1982). The complexity of computing minimum convex covers for polygons. In *20th Allerton conference*. (pp. 75–84), Monticello.

O'Rourke, J. (1987), *Art gallery theorems and algorithms*. Oxford: Oxford University Press.

Paragios, N., & Ramesh, V. (2001). A MRF-based approach for real-time subway monitoring. In *IEEE international conference on computer vision and pattern recognition* (Vol. I, pp. 1034–1040), Hawaii.

Peleg, S., Ben-Ezra, M., & Pritch, Y. (2001). Omnistereo: panoramic stereo imaging. *Pattern Analysis and Machine Intelligence*, *23*(3), 279–290.

Petitjean, S., Ponce, J., & Kriegman, D. (1992). Computing exact aspect graphs of curved objects: algebraic surfaces. *International Journal of Computer Vision*, *9*(3), 231–255.

Pito, R. (1999). A solution to the next best view problem for automated surface acquisition. *Pattern Analysis and Machine Intelligence*, *21*(10), 1016–1030.

Raczkowsky, J., & Mittenbuehler, K. H. (1989). Simulation of cameras in robot applications. In *Computer graphics applications* (pp. 16–25).

Reed, M. K., & Allen, P. K. (2000). Constraint-based sensor planning for scene modeling. *Pattern Analysis and Machine Intelligence*, *22*(12), 1460–1467.

Rodriguez, J., & Aggarwal, J. (1990). Stochastic analysis of stereo quantization error. *Pattern Analysis and Machine Intelligence*, *12*(5), 467–470.

Roy, S., Chaudhury, S., & Banerjee, S. (2001). Recognizing large 3-D objects through next view planning using an uncalibrated camera. In *IEEE international conference on computer vision* (Vol. II, pp. 276–281), Vancouver, Canada.

Roy, S., Chaudhury, S., & Banerjee, S. (2004). Active recognition through next view planning: a survey. *Pattern Recognition*, *37*(3), 429–446.

Sakane, S., Ishii, M., & Kakikura, M. (1987). Occlusion avoidance of visual sensors based on a hand eye action simulator system: HEAVEN. *Advanced Robotics*, *2*(2), 149–165.

Sakane, S., Niepold, R., Sato, T., & Shirai, Y. (1992). Illumination setup planning for a hand–eye system based on an environmental model. *Advanced Robotics*, *6*(4), 461–482.

Shang, Y. (1997). *Global search methods for solving nonlinear optimization problems*. PhD thesis, University of Illinois at Urbana-Champaign.

Shermer, T. (1992). Recent results in art galleries. *Proceedings of the IEEE*, *80*(9), 1384–1399.

Slavik, P. (1997). A tight analysis of the greedy algorithm for set cover. *Journal of Algorithms*, *25*, 237–254.

Spletzer, J., & Taylor, C. (2001). A framework for sensor planning and control with applications to vision guided multi-robot systems. In *CVPR*, Kauai, HI.

Stamos, I., & Allen, P. K. (1998). Interactive sensor planning. In *IEEE international conference on computer vision and pattern recognition* (pp. 489–494).

Stauffer, C., & Grimson, W. (2000). Learning patterns of activity using real-time tracking. *Pattern Analysis and Machine Intelligence*, *22*(8), 747–757.

Stuerzlinger, W. (1999). Imaging all visible surfaces. In *Graphics interface proceedings*. pp. 115–122. Los Altos: Kaufman.

Tarabanis, K., Tsai, R., & Kaul, A. (1991). Computing viewpoints that satisfy optical constraints. In *IEEE international conference on computer vision and pattern recognition* (pp. 152–158).

Tarabanis, K., Allen, P. K., & Tsai, R. (1995a). A survey of sensor planning in computer vision. *IEEE Transactions on Robotics and Automation*, *11*(1), 86–105.

Tarabanis, K., Tsai, R., & Allen, P. (1995b). The MVP sensor planning system for robotic vision tasks. *IEEE Transactions on Robotics and Automation*, *11*(1), 72–85.

Tarabanis, K., Tsai, R., & Kaul, A. (1996). Computing occlusion-free viewpoints. *Pattern Analysis and Machine Intelligence*, *18*(3), 279–292.

Urrutia, J. (1997). Art gallery and illumination problems. In *Handbook on computational geometry* (pp. 387–434). Amsterdam: Elsevier Science.

Wixson, L. (1994). Viewpoint selection for visual search tasks. In *IEEE international conference on computer vision and pattern recognition* (pp. 800–805).

Ye, Y., & Tsotsos, J. (1999). Sensor planning for 3D object search. *Computer Vision and Image Understanding*, *73*(2), 145–168.

Yi, S., Haralick, R., & Shapiro, L. (1995). Optimal sensor and light-source positioning for machine vision. *Computer Vision and Image Understanding*, *61*(1), 122–137.

Zhao, T., & Nevatia, R. (2003). Bayesian human segmentation in crowded situations. In *IEEE international conference on computer vision and pattern recognition* (Vol. II, pp. 459–466).

Zhao, T., & Nevatia, R. (2004). Tracking multiple humans in crowded environment. In *IEEE international conference on computer vision and pattern recognition* (Vol. II, pp. 406–413).

Zhao, T., Nevatia, R., & Lv, F. (2001). Segmentation and tracking of multiple humans in complex situations. In *IEEE international conference on computer vision and pattern recognition* (pp. 194–201), Kauai, HI.