Visibility Analysis and Sensor Planning in Dynamic Environments

Anurag Mittal[1] and Larry S. Davis[2]

[1] Real-Time Vision and Modeling, Siemens Corporate Research, Princeton, NJ 08540.
anurag@scr.siemens.com
[2] Computer Science Department, University of Maryland, College Park, MD 20742.
lsd@cs.umd.edu

Abstract. We analyze visibility from static sensors in a dynamic scene with moving obstacles (people). Such analysis is considered in a probabilistic sense in the context of multiple sensors, so that visibility from even one sensor might be sufficient. Additionally, we analyze worst-case scenarios for high-security areas where targets are non-cooperative. Such visibility analysis provides important performance characterization of multi-camera systems. Furthermore, maximization of visibility in a given region of interest yields the optimum number and placement of cameras in the scene. Our analysis has applications in surveillance - manual or automated - and can be utilized for sensor planning in places like museums, shopping malls, subway stations and parking lots. We present several example scenes - simulated and real - for which interesting camera configurations were obtained using the formal analysis developed in the paper.

1 Introduction

We present a method for sensor planning that is able to determine the required number and placement of static cameras (sensors) in a dynamic scene. Such analysis has previously been presented for the case of static scenes where the constraints and obstacles are static. However, in many applications, apart from these static constraints, there exists occlusion due to dynamic objects (people) in the scene. In this paper, we incorporate these dynamic visibility constraints into the sensor planning task. These constraints are analyzed in a probabilistic sense in the context of multiple sensors. Furthermore, we develop tools for analyzing worst-case visibility scenarios that are more meaningful for high-security areas where targets are non-cooperative.

Our analysis is useful for both manned and automated vision systems. In manned systems where security personnel are looking at the video stream, it is essential that the personnel have visibility of the people in the scene. In automated systems, where advanced algorithms are used to detect and track multiple people from multiple cameras, our analysis can be used to place the cameras in an optimum configuration.

Automated Multi-camera vision systems have been developed using a wide range of camera arrangements. For better stereo matching, some systems[1] use closely-spaced cameras. Others [2,3] adopt the opposite arrangement of widely separated cameras for maximum visibility. Others [4] use a hybrid approach. Still others [5,6,7], use multiple cameras for the main purpose of increasing the field of view. In all these systems, there is

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3021, pp. 175–189, 2004.

a need for analyzing the camera arrangement for optimum placement. In many cases, our method can be utilized without any alteration. In systems that have additional algorithmic requirements (e.g. stereo matching), further constraints - *hard* or *soft* - can be specified so that the optimum camera configuration satisfies (*hard*) and is optimum (*soft*) w.r.t. these additional constraints.

In addition to providing the optimum configuration, our analysis can provide a *gold standard* for evaluating the performance of these systems under the chosen configuration. This is because our analysis provides the theoretical limit of detectability. No algorithm can surpass such a limit since the data is missing from the images. Thus, one can determine as to how much of the error in a system is due to missing data, and how much of it is due to the chosen algorithm.

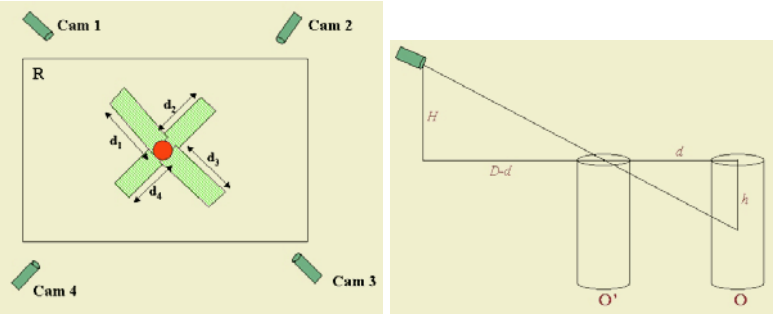
Sensor planning has been researched quite extensively, especially in the robotics community, and there are several different variations depending on the application. One set of methods use an active camera mounted on a robot. The objective then is to move the camera to the best location in the next view based on the information captured uptil now. These methods are called next view planning[8,9,10]. Another set of methods obtain a model (either 2D or 3D) of a scene by optimum movement of the camera [11,12]. Such model acquisition imposes certain constraints on the camera positions, and satisfaction of these constraints guarantees optimum and stable acquisition.

Methods that are directly related to ours are those that determine the location of static cameras so as to obtain the best views of a scene. This problem was originally considered in the computational geometry literature as the art-gallery problem [13]. The solutions in this domain utilize simple 2D or 3D scene models and simple assumptions on the cameras and occlusion in order to develop theoretical results and efficient algorithms to determine good sensor configurations (although the NP-hard nature of the problem typically necessitates an approximate solution). Several researchers [14,15,16,17] 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. In addition to these "static" constraints, there exist additional "visibility" constraints imposed by the presence of dynamic obstacles. Such constraints have not been analyzed earlier and their incorporation into the sensor planning task constitutes the novel aspect of our work.

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 development of a cost function and its minimization in order to perform sensor planning in complex environments. Section 5 concludes the paper with some simulated and real experiments.

2 Probabilistic Visibility Analysis

In this section, we analyze probabilistically the visibility constraints in a multi-camera setting. Specifically, we develop tools for evaluating the probability of visibility of an object from at least one sensor. Since this probability varies across space, this probability is recovered for each possible object position.



(a) (b)

Fig. 1. (a) Scene Geometry used for stochastic reasoning. (b) The distance up to which an object can occlude another object is proportional to its distance from the sensor.

2.1 Visibility from at Least One Sensor

Assume that we have a region $\mathcal{R}$ of area A observed by n sensors [Fig. 1 (a)]. Let $\mathcal{E}_i$ be the event that a target object $\mathcal{O}$ at location $\mathcal{L}$ is visible from sensor i. The probability that $\mathcal{O}$ is visible from at least one sensor can be expressed mathematically as the union $P(\bigcup_{i=1}^{n} \mathcal{E}_i)$ of these events, and it can be expanded using the inclusion-exclusion principle as:

$$P(\bigcup_{i} \mathcal{E}_i) = \sum_{\forall i} P(\mathcal{E}_i) - \sum_{i<j} P(\mathcal{E}_i \cap \mathcal{E}_j) + \quad \cdots \quad + (-1)^{n+1} P(\bigcap_{i} \mathcal{E}_i) \tag{1}$$

The motivation for this expansion is that it is easier to compute the terms on the RHS (right hand side) compared to the one on the LHS.

In order to facilitate the introduction of the approach to be followed in computing $P(\bigcup_{i=1}^{n} \mathcal{E}_i)$, we consider the specific case of objects moving on a ground plane. The objects are also 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}. Furthermore, we assume that the sensors are placed at some known heights H_i from this plane. Also, we define visibility to mean that the center line of the object (corresponding to the centroid in the horizontal profile) is visible for at least some length h from the top of the object (in people tracking, this might correspond to viewing the face).

A useful quantity can 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; for square prism with side $2s$, $r = \frac{1}{\pi/4} \int_0^{\pi/4} scos\theta \; d\theta = 2\sqrt{2}s/\pi$. The quantity r will be useful in calculating the average occluding region of an object. Furthermore, it can easily be shown that the distance d_i up to which an object can occlude another object is proportional to its distance D_i from sensor i [Fig. 1 (b)]. Mathematically,

$$d_i = (D_i - d_i)\mu_i = D_i \frac{\mu_i}{\mu_i + 1}, \quad \text{where} \quad \mu_i = \frac{h}{H_i} \tag{2}$$

Fixed Number of Objects. In order to develop the analysis, we start with the case of a fixed number k of objects in the scene under the assumption that they are located randomly and uniformly in region $\mathcal{R}$. This will be extended to the more general case of object densities in subsequent sections.

Under this assumption, we first estimate $P(\mathcal{E}_i)$, which refers to the probability that none of the k objects is present in the region of occlusion $\mathcal{R}_i^o$ for camera i. Assuming that all object orientations are equally likely [1], one may approximate the area of this region of occlusion as $A_i^o \approx d_i(2r)$. Then, the probability for a single object to *not* be present in this region of occlusion is $\left(1 - \frac{A_i^o}{A}\right)$. Since there are k objects in the scene located independently of each other, the probability that none of them is present in the region of occlusion is $\left(1 - \frac{A_i^o}{A}\right)^k$. Thus:

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

In order to provide this formulation, we have neglected the fact that two objects cannot overlap each other. In order to incorporate this condition, we observe that the $(j+1)$-th object has a possible area of only $\mathcal{A} - jA_{ob}$ available to it [2]. Thus, Equation 3 can be refined as

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

This analysis can be generalized to other terms in Equation 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 determined as:

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

where $A^o_{(i_1,\ldots i_m)}$ is the area 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. $\mathcal{R}^o_{(i_1,\ldots i_m)} = \bigcup_{p=1}^{m} \mathcal{R}^o_{i_p}$.

Uniform Object Density. A fixed assumption on the number of objects in a region is clearly inadequate. 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

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

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

will be extended to the more general case of non-uniform object density in the next section. 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 area A proportionately such that

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

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

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

We define:

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

Here, a captures the effect of the presence of objects and b is a *correction* to such effect due to the finite object size. Then, we obtain:

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

Combining terms for j and $k-j$, we get

$$\left(1 - \frac{1}{ka - jb}\right)\left(1 - \frac{1}{ka - (k-j)b}\right) = \frac{k^2a^2 - k^2ab - 2ka + j(k-j)b^2 + bk + 1}{k^2a^2 - k^2ab + j(k-j)b^2}$$

Assuming $a \gg b$ (i.e. the object density λ is much smaller than $1/A_{ob}$, the object density if the area is fully packed), we can neglect terms involving b^2. Then, the above term can be written as

$$\approx \left(1 - \frac{1}{k}\left(\frac{2a-b}{a^2-ab}\right)\right)$$

There are $k/2$ such terms in Equation 9. Therefore,

$$P(\bigcap_{i\in(i_1,\ldots i_m)} \mathcal{E}_i) \approx \lim_{k\to\infty} \left(1 - \frac{1}{k}\left(\frac{2a-b}{a^2-ab}\right)\right)^{k/2}$$

Using the identity $lim_{x\to\infty}(1+\frac{1}{x})^x = e$, we get

$$P(\bigcap_{i\in(i_1,\ldots i_m)} \mathcal{E}_i) \approx e^{-\frac{2a-b}{2a(a-b)}} \tag{10}$$

Non-uniform Object Density. In general, the object density (λ) is a function of the location. For example, the object density near a door might be higher. Moreover, the presence of an object at a location influences the object density nearby since objects tend to appear in groups. We can integrate both of these influences on the object density 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 is able to capture the effect that the presence of the object at location $\mathbf{x}_O$ has on the density nearby[3].

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 $R^o_{(i_1,\ldots i_m)}$ can then be calculated as the ratio of the average number of people present in the region of occlusion to the average number of people in the available region. Thus, one can write:

$$P(\bigcap_{i\in(i_1,\ldots i_m)} \mathcal{E}_i) = \lim_{k\to\infty} \prod_{j=0}^{k-1} \left(1 - \frac{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c}}{\int_{\mathcal{R}-\mathcal{R}^j_{ob}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c}}\right) \tag{11}$$

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

$$\int_{\mathcal{R}-\mathcal{R}^j_{ob}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c} = \lambda_{avg}(A - jA_{ob})$$

where λ_{avg} is the average object density in the region. Using this simplification in Equation 11 and noting that $\lambda_{avg}A = k$, we obtain:

$$P(\bigcap_{i\in(i_1,\ldots i_m)} \mathcal{E}_i) = \lim_{k\to\infty} \prod_{j=0}^{k-1} \left(1 - \frac{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c}}{k - j\cdot\lambda_{avg}\cdot A_{ob}}\right) \tag{12}$$

Defining:

$$a = \frac{1}{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c}}, \quad b = \frac{A_{ob}\cdot\lambda_{avg}}{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\,\mathrm{d}\mathbf{x_c}}, \tag{13}$$

Equation 12 may again be put in the form of Equation 9. As before, this may be simplified to obtain the expression in Equation 10.

2.2 Visibility from Multiple Sensors

In many applications, it is desirable to view an object from more than one sensor. Stereo reconstruction/depth recovery is an example where the requirement of visibility from at

[3] 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.

least two sensors is to be satisfied. In order to evaluate the probability of visibility from at least two sensors, one can evaluate:

$$P(\bigcup_{(i<j)} (\mathcal{E}_i \cap \mathcal{E}_j)) \tag{14}$$

This term can be expanded exactly like Equation 1 treating each term $(\mathcal{E}_i \cap \mathcal{E}_j)$ as a single entity. All the terms on the RHS will then have only intersections in them which are easy to compute using the formulation developed in the previous sections.

2.3 Additional Constraints

Other "static" constraints also affect the view of a particular camera. Therefore, the visibility probability needs to be calculated after incorporating these additional constraints. This is easily achieved in our scheme since the visibility constraints are analyzed at individual locations and additional constraints can also be verified at these locations. The constraints that have been incorporated in our system include:

1. FIELD OF VIEW: Cameras have a limited field of view. At each location, it can be verified whether that location is within the field of view of a particular camera.
2. OBSTACLES: Fixed *high* obstacles like pillars cause occlusions in certain areas. From a given location, it needs to 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, meaningful observations are possible only up to a certain distance from the camera. It can easily be verified whether the location is within a certain "resolution distance" from the camera.
5. ALGORITHMIC CONSTRAINTS: There are several algorithmic constraints that may exist. For example, stereo matching across two (or more) cameras imposes a constraint on the maximum distortion of the view that can occur from one camera to the other. This constraint can be expressed in terms of the angular separation between the camera centers from the point of view of the object. It can be easily be verified whether this constraint is satisfied at a particular location.
6. VIEWING ANGLE: An additional constraint exists for the maximum angle α_{max} at which the observation of an object is meaningful. Such observation can be the basis for performing some other tasks like object recognition.This constraint translates into a constraint on the minimum distance from the sensor that an object must be. This minimum distance guarantees the angle of observation to be smaller than α_{max}.

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

3 Worst-Case Visibility Analysis

In this section, we present some simple results for 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. Since the analysis is quite simple, we will only briefly describe these results. We propose:

Theorem 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* $\mathcal{L}$, *then* $n > k + m - 1$ *is the necessary and sufficient condition to guarantee visibility for* $\mathcal{O}$ *from at least* m *sensors.*

Proof. (a) *Necessary*: Suppose $n <= k+m-1$. 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 $n <= k+m-1$). (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 $n - k >= m$ sensors viewing $\mathcal{O}$.

This result holds for point objects only. It can be extended to finite objects if certain assumptions are made. One can assume a flat world scenario where the objects and the sensors are in 2D. Also assume that we are given a point of interest in the object such that object visibility is defined as the visibility of this point of interest. This point can be defined arbitrarily. Let us also define an angle α as the maximum angle that any object can subtend at the point of interest of any other object. For example, for identical cylinders with the center as the point of interest, $\alpha = 60^\circ$. For identical square prisms, $\alpha = 90^\circ$. Under these assumptions, the above result holds if we take n to be the number of sensors that have visibility of location $\mathcal{L}$ such that the angular separation between any two sensors, from the point of view of $\mathcal{L}$, is at least α. Also, n must be less than $2\pi/\alpha$ since it is not possible to place $n > 2\pi/\alpha$ sensors such that there is an angular separation of at least α between them.

For a given camera configuration, one can determine the number of cameras that each location of interest has visibility to. This will yield the maximum number of people that can be present in the vicinity of the person and still guarantee visibility for him.

4 Sensor Planning

The visibility analysis presented in section 2 yields a function $p_{\mathbf{s}}(\mathbf{x})$, that refers to the probability that an object located at location $\mathbf{x}$ is visible from at least one of the sensors that have the parameter vector $\mathbf{s}$. Such 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 in order to evaluate a given set of sensor parameters. Such sensor parameters may be further constrained 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. Such optimization will yield the optimum sensor parameters according to the specified cost function.

4.1 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 $\mathbf{x}$ has visibility to:

$$C(\mathbf{s}) = -\sum_{\mathbf{x}\in\mathcal{R}_i} N(\mathbf{x}) \tag{15}$$

Using probabilistic analysis, one may define a cost function that minimizes the maximum occlusion probability in the region:

$$C(\mathbf{s}) = \max_{\mathbf{x}\in\mathcal{R}_i}(1 - p_{\mathbf{s}}(\mathbf{x}))$$

Another cost function, and perhaps the most reasonable one in many situations, is to define the cost as the negative of the average number of visible people in a given region of interest:

$$C(\mathbf{s}) = -\int_{\mathcal{R}_i} \lambda(\mathbf{x}) p_{\mathbf{s}}(\mathbf{x})\, \mathrm{d}\mathbf{x} \tag{16}$$

This cost function has been utilized for obtaining the results in this paper.

It is also possible to integrate other constraints into the cost function. For instance, some of the constraints in section 2.3 may be specified as *soft* constraints rather than *hard* constraints (for e.g. resolution, viewing angle and algorithmic constraints). According to the application, any arbitrary function of the constraints may be considered:

$$C(\mathbf{s}) = f(c_1, \ldots c_J, \lambda(), \mathcal{R}_i)$$

where $c_j, j = 1 \ldots J$ are the different constraints to be satisfied.

4.2 Minimization of the Cost Function

The cost function defined by Equation 16 (as also other suitable ones) is non-linear 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. Fig. 2 illustrates the cost function for the scene shown in Fig. 4 (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 of the cost function, it is not possible to minimize it using simple gradient-based methods that can only find the local minimum of a well-behaved "convex" function. Global minimization methods that can deal with complex cost functions are necessary [18]. Simulated Annealing and Genetic Algorithms are two classes of algorithms that may be considered. The nature of the cost function suggests that either of these two algorithms should provide an acceptable solution[19]. For our experiments, we implemented a simulated annealing scheme using a highly sophisticated simulated re-annealing software *ASA* developed by L. Ingber [20].

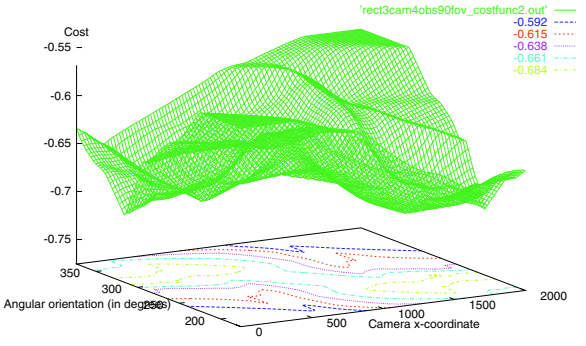


Fig. 2. The Cost Function for the scene in Fig. [4 (a)] where, for illustration purposes, only the x-coordinate and direction of the second camera have been varied.

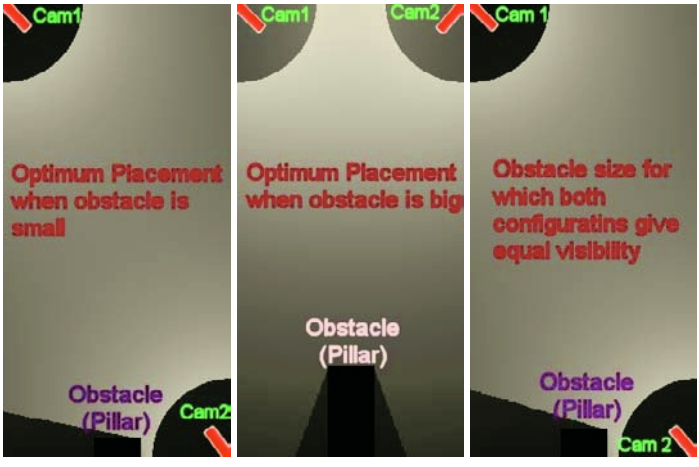
Using this algorithm, we were able to obtain extremely good sensor configurations in a reasonable amount of time (5min - a couple of hours on a Pentium IV 2.2GHz PC, depending on the desired accuracy of the result, the number of dimensions of the search space and complexity of the scene). For low dimensional 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 again able to obtain the optimum solution, but only after some time. Although the optimality of the solution could not be verified by full search, we assumed such solution to be optimum since running the algorithm several times from different starting points and different annealing parameters did not alter the final solution. For very high dimensional spaces (> 8), although the algorithm provided "good" solutions very quickly, it took several hours to converge to the best one. Some of the "optimal" solutions thus obtained will be illustrated in the next section.

5 Simulations and Experiments

We have proposed a stochastic algorithm for recovering the optimal sensor configuration with respect to certain visibility requirements. In order to validate the proposed method, we provide results of the algorithm for various scenes, synthetic and real.

5.1 Synthetic Experiments

In all the synthetic examples we consider next, we take a rectangular room of size 10mX20m. The sensors were restricted to be mounted H = 2.5m above the ground and have a field of view of 90 °. We use a uniform object density $\lambda = 1m^{-2}$, object height = 150cm, object radius r=15cm, minimum visibility height h=50cm and maximum visibility angle $\alpha_{max} = 45°$. The illustrations shown are visibility maps scaled such that



(a) (b) (c)

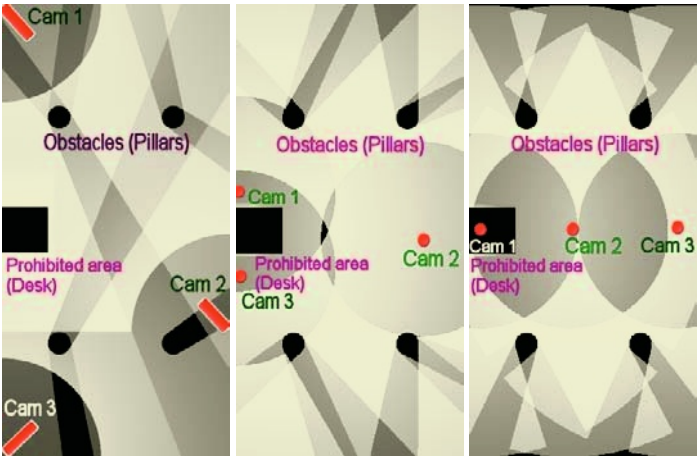
Fig. 3. Illustration of the effect of scene geometry on sensor placement. Optimum configuration when (a): obstacle size is small. (b): obstacle size is big. (c): obstacle size is such that both configurations are equally good.

[0,1] maps onto [0,255], thus creating a gray scale image. Brighter regions represent higher visibility. Note how the visibility decreases as we move away from a camera due to an increase in the distance of occlusion d_i.

Fig. 3 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($<$60cm). Configuration [b] was optimum when the object size was big ($>$60cm). For the object size shown in configuration [c] ($\sim$60cm), 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.

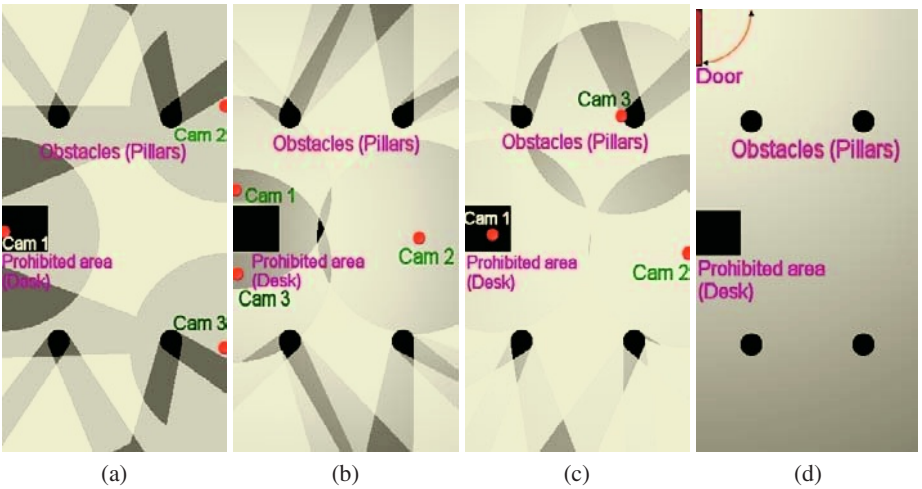
Fig. 4 illustrates how the camera specifications can significantly alter the optimum sensor configuration. Notice that the scene has both obstacles and prohibited areas. With three available cameras, configuration [a] was found to be optimum when the cameras have only 90° field of view but are able to "see" up to 25m. With the same resolution, configuration [b] is optimum if the cameras have a 360° field of view (Omni-Camera). If the resolution is lower so that cameras can "see" only up to 10m, configuration [c] is optimum.

Fig. 5 illustrates the effect of different optimization criteria. With the other assumptions the same as above, configuration [a] was found to be optimum when the worst case analysis was utilized [Eq. 15]. On the other hand, a uniform object density assumption [Eq. 16] 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 such that they can better capture this region.



(a) (b) (c)

Fig. 4. Illustration of the effect of different camera specifications. With a uniform density assumption, the optimum configuration when the cameras have (a): field of view of 90° and resolution up to 25m, (b): 360° field of view (Omni-Camera), and resolution up to 25m, (c): 360° field of view, but resolution only up to 10m.



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

Fig. 5. Illustration of the effect of different optimization criteria. Optimum configuration for: (a): worst-case analysis [Eq. 15], (b): uniform density case [Eq. 16], (c): variable density case [Eq. 16] for the object density shown in (d).

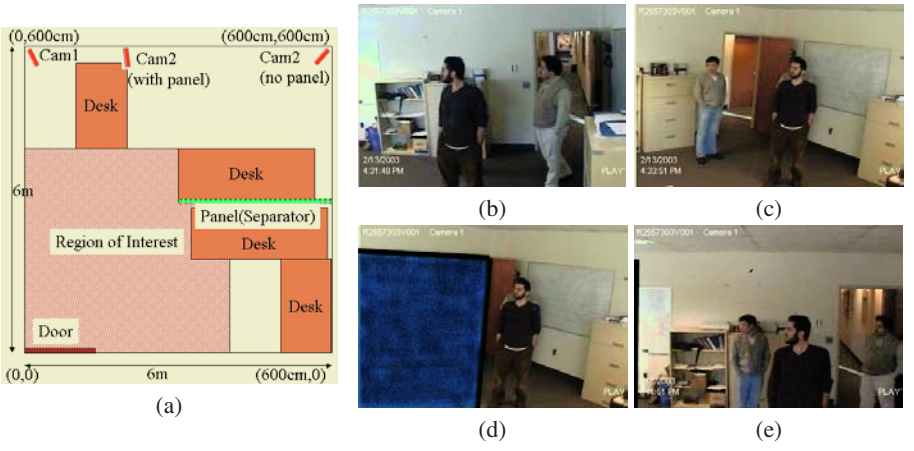


Fig. 6. (a) Plan view of a room used for a real experiment. (b) and (c) are the views from the optimum camera locations when there is no panel (obstacle). Note that, of the three people in the scene, one person is occluded in each view. However, all of them are visible from at least one of the views. 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.

5.2 Analysis of a Real Scene

We now present analysis of sensor placement for a real office room. The structure of the room is illustrated in Fig. 6 (a). We used the following parameters - uniform density $\lambda = 0.25m^{-2}$, object height = 170cm, $r = 23$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 2.5m high. In order to view people's face as they enter the room, we further restricted the cameras to be placed on the wall facing the door. 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 6 (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 6 (d) & (e) show the views from the original and new location of the second camera.

6 Conclusion

We have presented two methods for evaluation of visibility given a certain configuration of sensors in a scene. The first one evaluates the visibility probabilistically assuming a density function for the occluding objects. The second method evaluates worst-case scenarios and is able to provide conditions that would guarantee visibility regardless of

object configuration. Apart from obtaining important performance characterization of multi-sensor systems, such analysis was further used for sensor planning by optimization of an appropriate cost function. The algorithm was tested on several synthetic and real scenes, and in many cases, the configurations obtained were quite interesting and non-intuitive. The method has applications in surveillance and can be utilized for sensor planning in places like museums, shopping malls, subway stations and parking lots. Future work includes specification of more complex cost functions, investigation of more efficient methods for optimization of the cost function and better estimation of visibility probability by considering the effect of interaction between objects.

Acknowledgments. We would like to thank Nikos Paragios for help in improving the presentation of the paper, and Visvanathan Ramesh for helpful discussions on the topic.

References

[1] Darrell, T., Demirdjian, D., Checka, N., Felzenszwalb, P.: Plan-view trajectory estimation with dense stereo background models. In: ICCV, Vancouver, Canada (2001) II: 628–635
[2] Mittal, A., Davis, L.: M_2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. IJCV **51** (2003) 189–203
[3] Khan, S., Javed, O., Rasheed, Z., Shah, M.: Human tracking in multiple cameras. In: ICCV, Vancouver, Canada (2001) I: 331–336
[4] Collins, R., Lipton, A., Fujiyoshi, H., Kanade, T.: Algorithms for cooperative multi-sensor surveillance. Proceedings of the IEEE **89** (2001) 1456–1477
[5] Stauffer, C., Grimson, W.: Learning patterns of activity using real-time tracking. PAMI **22** (2000) 747–757
[6] Kettnaker, V., Zabih, R.: Counting people from multiple cameras. In: ICMCS. (1999) II:253–259
[7] Cai, Q., Aggarwal, J.: Tracking human motion in structured environments using a distributed-camera system. PAMI **21** (1999) 1241–1247
[8] Miura, J., Ikeuchi, K.: Task-oriented generation of visual sensing strategies. In: ICCV, Boston, MA (1995) 1106–1113
[9] Ye, Y., Tsotsos, J.: Sensor planning for 3d object search. CVIU **73** (1999) 145–168
[10] Pito, R.: A solution to the next best view problem for automated surface acquisition. PAMI **21** (1999) 1016–1030
[11] Kutulakos, K., Dyer, C.: Recovering shape by purposive viewpoint adjustment. IJCV **12** (1994) 113–136
[12] Cameron, A., Durrant-Whyte, H.: A bayesian approach to optimal sensor placement. IJRR **9** (1990) 70–88
[13] O'Rourke, J.: Art Gallery Theorems and Algorithms. Oxford University Press (1987)
[14] Cowan, C.K., Kovesi, P.: Automatic sensor placement from vision tast requirements. PAMI **10** (1988) 407–416
[15] Reed, M.K., Allen, P.K.: Constraint-based sensor planning for scene modeling. PAMI **22** (2000) 1460–1467
[16] Tarabanis, K., Tsai, R., Kaul, A.: Computing occlusion-free viewpoints. PAMI **18** (1996) 279–292
[17] Yi, S., Haralick, R., Shapiro, L.: Optimal sensor and light-source positioning for machine vision. CVIU **61** (1995) 122–137

[18] Shang, Y.: Global Search Methods for Solving Nonlinear Optimization Problems. PhD thesis, University of Illinois at Urbana-Champaign (1997)
[19] Duda, R., Hart, P., Stork, D.: Pattern Classification. John Wiley and Sons (2001)
[20] Ingber, L.: Very fast simulated re-annealing. Mathematical Computer Modeling **12** (1989) 967–973