Generalized Multi-Sensor Planning

Anurag Mittal

Dept of Computer Science and Engg**
Indian Institute of Technology Madras
Chennai, India - 600036

Abstract. Vision systems for various tasks are increasingly being deployed. Although significant effort has gone into improving the algorithms for such tasks, there has been relatively little work on determining optimal sensor configurations. This paper addresses this need. We specifically address and enhance the state-of-the-art in the analysis of scenarios where there are dynamically occuring objects capable of occluding each other. The visibility constraints for such scenarios are analyzed in a multi-camera setting. Also analyzed are other static constraints such as image resolution and field-of-view, and algorithmic requirements such as stereo reconstruction, face detection and background appearance. Theoretical analysis with the proper integration of such visibility and static constraints leads to a generic framework for sensor planning, which can then be customized for a particular task. Our analysis can be applied to a variety of applications, especially those involving randomly occuring objects, and include surveillance and industrial automation. Several examples illustrate the wide applicability of the approach.

1 Introduction

Systems utilizing possibly multiple visual sensors have become essential in many applications. Surveillance and Monitoring, industrial automation, transportation and automotive, and medical systems are a few of the important application domains. Existing research has mainly focused on improving the algorithms deployed in these systems, while little focus has been given to the placement of sensors for optimal system performance. Each system also has its own set of requirements. In security systems, for instance, the captured video streams may be inspected either manually, or a more advanced computerized system may be utilized to detect spurious activity automatically. Furthermore, automated people detection and tracking systems may have different objectives. Some systems utilize multiple closely-spaced cameras for the purpose of accurate stereo matching. Others utilize widely separated cameras for maximizing the object visibility in a dense situation [14, 11]. Still others [23, 1, 4, 19], use multiple cameras for the main purpose of increasing the coverage area by utilizing non-overlapping field-of-view cameras. In this paper, we develop a generic formulation that can be customized to find good sensor configurations for any of these systems.

** This work was conducted while the author was with Siemens Corporate Research, Princeton, NJ USA

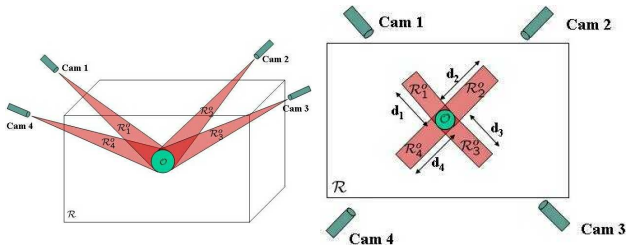
Sensor planning has been researched quite extensively, and there are several different variations depending on the application. A popular set of methods, called next-view planning, attempt to build a model of the scene incrementally by successively sensing the unknown world from effective sensor configurations using the information acquired about the world up to this point [17, 25, 5, 13, 12, 2, 8]. A related set of methods [10] have 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.

Methods that are directly related to ours are those that assume that complete geometric information is available and determine the location of static cameras so as to obtain the best views of a scene. This problem was originally posed in the computational geometry literature as the "art-gallery problem" [18]. The traditional formulation of such problem assumes the simple assumption that two points are called visible if the straight line segment between them lies entirely inside the polygon. Even with such simple definition of visibility, the problem is NP-complete.

Some of the recent work has concentrated on incorporating a few more constraints like incidence angle and range into the problem and obtain an approximate solution to the resultant NP-complete problem via randomized algorithms [7]. Several researchers [6, 20, 24, 13, 26, 22] 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. However, the problem becomes much more complex to be amenable to fast approximation solutions.

In addition to the "static" constraints considered so far, there are additional constraints that arise when dynamic obstacles are present. Such constraints are essential to analyze since system performance is a function of object visibility. In [3], it was proposed to combine visibility and static constraints via a weighted sum of the error due to the two factors. On the other hand, our earlier paper [15] proposed maximization of the visibility while static constraints were analyzed simply as *hard* constraints that would either be satisfied or not at a given location. In this work, we provide a more general approach towards integration of these two types of constraints. We utilize analysis of visibility constraints and determination of multi-camera visibility rates from [15]. Integration of such analysis with a variety of static constraints and application requirements leads to a generic formulation for sensor planning. Customization of the method for a given system allows the method to be utilized for a variety of different tasks and applications.

The paper is organized as follows. Section 2 briefly reviews prior work on estimating the probability of visibility of an object at a given location in a scene for a certain configuration of sensors. Section 3 describes the integration of static constraints with probabilistic visibility constraints. Maximization of the thus obtained quality measure over an entire region of interest will be considered in section 4. Section 5 concludes the paper with planning experiments for some synthetic and real scenes.



(a) (b)

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

2 Visibility Analysis

In this section, we briefly review and generalize some visibility analysis results from [15] that are pertinent to this work.

Since the particular application domain might contain either two or three dimensions, we consider the general case of an m dimensional space. Let us assume that we have a region $\mathcal{R} \subset \mathbb{R}^m$ of *content* A observed by n sensors [Fig. 1]. Here, we use the term "content" in a general sense, such that it is the area of $\mathcal{R}$ if $m = 2$, and is the 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. Then, it is useful to compute the following probabilities:

$$\{P(\mathcal{E}_i), i = 1..n\}, \quad \{P(\mathcal{E}_i \cap \mathcal{E}_j), i, j = 1..n\}, \ \ldots \ , \ P(\bigcap_i \mathcal{E}_i) \tag{1}$$

In order to compute these probabilities, we first note that there exists a region of occlusion $\mathcal{R}_i^o$ for each sensor i such that the presence of another object in $\mathcal{R}_i^o$ would cause $\mathcal{O}$ to not be visible from i [1] [Fig. 1]. Now, let us assume that objects are located randomly in the scene with object density λ. Since λ is a function of the location and may also be influenced by the presence of nearby objects, let $\lambda(\mathbf{x_c}|\mathbf{x_O})$ be the density at location $\mathbf{x_c}$ given that visibility is being calculated at location $\mathbf{x_O}$. Then, it can be shown [16] that the probability that object $\mathcal{O}$ is visible from all of the sensors in a specified set $(i_1, i_2 \ldots i_m)$ is[2]:

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

where

$$a = \frac{1}{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\, \mathrm{d}\mathbf{x_c}}, \quad b = \frac{A_{ob} \cdot \lambda_{avg}}{\int_{\mathcal{R}^o_{(i_1,\ldots i_m)}} \lambda(\mathbf{x_c}|\mathbf{x_0})\, \mathrm{d}\mathbf{x_c}} \tag{3}$$

[1] Note that this region of occlusion is dependent on the application-specific definition of visibility. For instance, one may require that all of the object be visible, or one may require visibility of only the object center.

[2] Note that this is a better approximation than the one given in our earlier paper[15]

Here, λ_{avg} is the average object density in the region, A_{ob} is the content of an occluding object, and $\mathcal{R}^o_{(i_1,\ldots i_m)}$ is the combined region of occlusion 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. $\mathcal{R}^o_{(i_1,\ldots i_m)} = \bigcup_{p=1}^{m} \mathcal{R}^o_{i_p}$.

It may be noted that a is the effect on the probability due to the presence of an object, and b is a *correction* to such effect due to the finite object size.

3 Static Constraints and the Capture Quality

Several stationary factors affect the quality of the data acquired by a 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.

3.1 "Static" Constraints

Some of the static constraints affecting the view of the camera are described next. Many of these constraints may be considered in either of two ways: *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 the quality of the acquired data.

1. FIELD OF VIEW: Cameras have a limited field of view, and a constraint can be specified terms of a maximum angle from a central camera direction.
2. OBSTACLES: Fixed *high* obstacles like pillars block the view of a camera, and such constraint can be verified for a given object location.
3. PROHIBITED AREAS: There might also exist prohibited areas like desks or counterswhere people are not able to walk. 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. IMAGE 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.
5. ALGORITHMIC CONSTRAINTS: There are several algorithmic constraints that may exist. Such constraints may also be more complex involving inter-relationships between the views of several cameras. Stereo matching across two or more cameras is an example of such a constraint and involves a complex integration of several factors including image resolution, the maximum distortion of a view that can occur from one view to the other and the triangulation error.
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 such as object recognition. When the vertical viewing angle is considered, this constraint translates into a constraint on the minimum distance from the sensor that an object must be. The horizontal viewing angle can also be considered similarly by consideration of the angle between the object orientation and the camera direction.

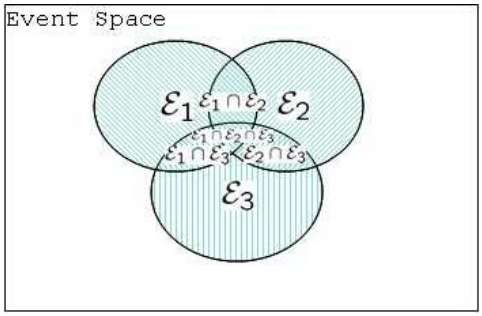


Fig. 2. The event space may be partitioned into disjoint event sets. Here, *only* $\mathcal{E}_i$, for instance, would only include event space that is not common with other events.

3.2 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 quantity is a random variable that depends on the occurrence of events $\mathcal{E}_i$. The event space may be partitioned into the following disjoint sets [Fig. 2]:

$$\begin{aligned} \text{No}\mathcal{E}_i \text{ occurs, with quality:} \quad & 0 \\ \text{Only}\mathcal{E}_i \text{ occurs, with quality:} \quad & q(\mathcal{E}_i) \\ \text{Only}\mathcal{E}_i \cap \mathcal{E}_j \text{occurs, with quality:} \quad & q(\mathcal{E}_i \cap \mathcal{E}_j) \qquad \ldots \\ \bigcap_i \mathcal{E}_i \text{ occurs, with quality:} \quad & q(\bigcap_i \mathcal{E}_i) \end{aligned}$$

Such separation allows one to specify the quality measure for each of such events separately. More specifically, such quality function needs to be specified for all of such events. In other words, one needs to specify for all possible sets, the quality measure $q_l(i_1, \ldots, i_m, \theta)$ that 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 given m-tuple i.e. the event $(\bigcap_{i \in (i_1, \ldots i_m)} \mathcal{E}_i)$ occurs.

To give some insight into such specification of the quality function, one can consider the case of stereo matching. In such an application, since visibility from at least two sensors would be required for matching, the capture quality $\{q_l(i, \theta)\}, i = 1 \ldots n$ would be zero. For the terms involving two sensors, several competing requirements need to be considered. Under some simplifying assumptions, the error in the recovered depth due to image quantization may be approximated as being 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 the 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 size of the object view, which might further decrease the accuracy of stereo matching. Thus, in the perpendicular direction, 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 quality function that peaks for some given distance and tapers off in either direction can be considered. Thus, for any given task requirement, 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.

Computation of probabilities of these disjoint events along with the specification of the capture quality associated with such events yields a probability function for the capture quality at a particular location (Fig. 5 illustrates an example where the function for a typical scene is averaged over the entire region of interest.). Given such a probability function, one can consider several integration measures of which the mean will be considered in this paper for simplicity purposes. The *mean* capture quality at a particular location for a particular object orientation θ may be written as:

$$q(\theta) = \sum_{\forall i} q(\mathcal{E}_i, \theta) P(\text{Only } \mathcal{E}_i) + \sum_{i<j} q(\mathcal{E}_i \cap \mathcal{E}_j, \theta) P(\text{Only } \mathcal{E}_i \cap \mathcal{E}_j) + \cdots + q(\bigcap_i \mathcal{E}_i, \theta) P(\text{Only } \bigcap_i \mathcal{E}_i)$$

The probabilities $P(\text{Only } \bigcap_i \mathcal{E}_i)$ may be rewritten using the $P(\bigcap_i \mathcal{E}_i)$ terms that we had calculated earlier.

3.3 Integration of Quality across Space

The analysis presented so far yields a function $q_{\mathbf{s}}(\mathbf{x}, \theta)$, that refers to the capture quality of an object with orientation θ at location $\mathbf{x}$ given that the sensors 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 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. Such optimization will yield the optimum sensor parameters according to the specified cost function.

Several cost functions may be considered. One may define a cost function that maximizes the minimum quality in the region. Another cost function, and perhaps the most reasonable one in many situations, is to define the cost as the negative of the average capture quality in a given region of interest:

$$C(\mathbf{s}) = -\int_{\mathcal{R}_i} \int_0^{2\pi} \lambda(\mathbf{x}, \theta) q_{\mathbf{s}}(\mathbf{x}, \theta) \, \mathrm{d}\theta \, \mathrm{d}\mathbf{x} \tag{4}$$

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

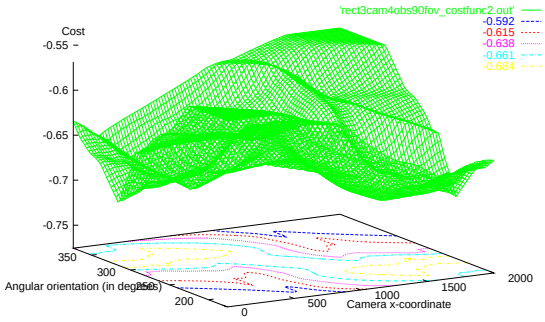


Fig. 3. The Cost Function for the scene in Figure [4 (b)] where, for illustration purposes, only the x-coordinate and direction of the second camera have been varied.

incorporate information about the object orientations into the density function. Since the orientation does not affect the occluding characteristics of an object, such parameter was integrated (and eliminated) for the visibility analysis presented previously.

4 Minimization of the Cost Function

The cost function defined by Equation 4 (as also other suitable ones) is highly complex and due to the variegated nature of the constraints, it is not possible to obtain a single method that optimizes such function in a very efficient manner. Furthermore, even for simple scenarios, it can be shown that the problem is NP-complete and not amenable to fast polynomial time solutions. Figure 3 illustrates the cost function for the scene shown in Figure 4 (b) 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 the generality and characteristics of the cost function, we propose to use a general method that is able to find the global minima of complex cost functions. Simulated Annealing and Genetic Algorithms are two classes of algorithms that may be considered[21]. For our experiments, we utilized a highly sophisticated simulated re-annealing software *ASA* developed by L. Ingber [9].

Using this algorithm, we were able to obtain extremely good sensor configurations in a reasonable amount of time (5min to 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 Experiments

We will now demonstrate how the generic method developed so far may be customized for different task requirements. For simplicity, we will consider the specific $2.5D$ case of objects moving on a ground plane and sensors placed at some known heights H_i from this plane. The objects are also assumed to have the same horizontal profile at each height, such that the area of their profile onto the ground is A_{ob}. Examples of such objects include cylinders, cubes, cuboids, and square prisms.

We will also assume that we require only visibility of the center line of the object and only up to a length h from its top. Then, assuming that the average "radius" of the object is r, the region of occlusion 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:

$$d_i = (D_i - d_i)\mu_i = D_i \frac{\mu_i}{\mu_i + 1}, \quad \text{where} \quad \mu_i = \frac{h}{H_i} \tag{5}$$

Then, one may approximate the area of the region of occlusion R_i^o as $A_i^o \approx d_i(2r)$. These models enable one to reason about the particular application of people detection and tracking for objects moving on a plane. Using these assumptions, we first consider some synthetic examples.

5.1 Synthetic Examples

In the synthetic examples we consider, we use the following assumptions. The objects occur randomly with object density $\lambda = 1m^{-2}$, object height = 150cm, object radius r=15cm, minimum visibility height h=50cm and maximum visibility angle $\alpha_{max} = 45^\circ$. The sensors are mounted H = 2.5m above the ground. The maps shown are capture quality maps scaled such that [0,1] maps onto [0,255]. First, we consider a rectangular room of size 10mX20m.

The first two examples [Fig. 4 a & b.] assume a simple quality function such that visibility from *any* direction is considered equally valid (i.e. the parameter θ is neglected) and fixed thresholds are put on the visibility distance from the camera based on camera resolution ($maxdist_{res}$) and maximum viewing angle α_{max} ($mindist_{view}$):

$$q_{\mathbf{x}}(\mathcal{E}_i, \theta) = \begin{cases} 1 \text{ if } mindist_{view} < d(\mathbf{x}, cam) < maxdist_{res} \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

Furthermore, for multiple sensor terms, the quality is defined simply as the quality of the sensor having the best view:

$$q(\bigcap_{i \in (i_1, \ldots i_m)} \mathcal{E}_i, \theta) = \max_{i \in (i_1, \ldots i_m)} q(\mathcal{E}_i, \theta) \tag{7}$$

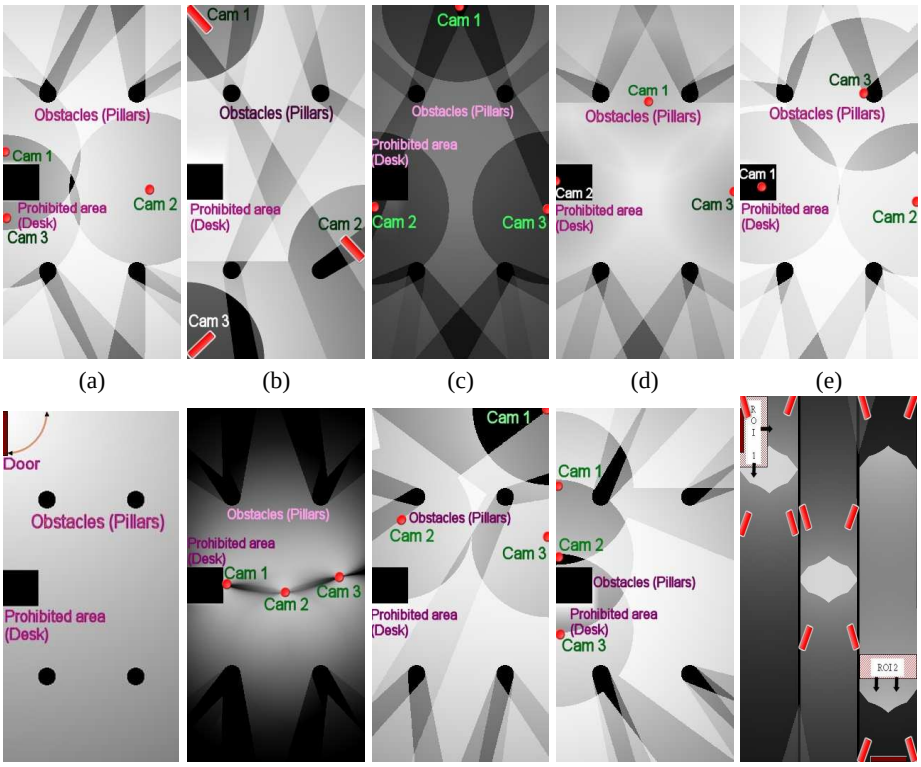


Fig. 4. Optimum configuration using: (a): a non-directional object visibility constraint and a uniform object density. [Eq. 4], (b): field of view restricted to 90°, (c): directional object visibility [Eq. 8], (d): directional object visibility, and a soft constraint on resolution and viewing angle [Eqs. 9 & 10], (e): non-directional object visibility, and using variable densities shown in Fig. (f), (g): stereo requirement, non-directional object visibility and uniform densities, (h): algorithmic constraint of no visibility with the top wall as background, (i): no visibility with the left wall as background, (j): Sensor Planning in a large "Museum", where several constraints are to be satisfied simultaneously.

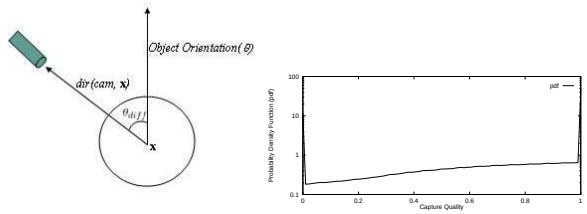


Fig. 5. (a): Computation of the viewing angle θ_{diff}, (b): The probability density function for the capture quality for Fig. 4 [d]. Note the unusually high values for zero and one due to the possibilities of complete object occlusion and perfect capture in certain conditions.

Using $mindist = 5m$ and $maxdist = 25m$, if the sensors have a field of view of 360° (omni-camera), configuration [a] was found optimum, while a field of view of 90° resulted in configuration [b]. The omni-camera is used for the rest of the examples in this scene.

Assuming that one requires visibility from *all* directions, one may alter the quality function as:

$$q_{\mathbf{x}}(\mathcal{E}_i, \theta) = \begin{cases} 1 \text{ if } \theta_{diff} < \theta^{max} \\ \quad \& \; d_{view}^{min} < d(\mathbf{x}, cam) < d_{res}^{max} \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

where θ^{max} is the maximum angular orientation at which the observation of the object is still considered meaningful, and $\theta_{diff} = abs(\theta - dir(cam, \mathbf{x}))$ such that $dir(cam, \mathbf{x})$ is the angular direction of the camera from the point of view of $\mathbf{x}$ [Fig. 5 (a)]. Assuming that $\theta^{max} = 90^\circ$, we obtain the sensor configuration shown in [c]. 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. Furthermore, one may allow a soft constraint on the viewing orientation. Using one such possible quality function:

$$q_{\mathbf{x}}(\mathcal{E}_i, \theta) = H(\theta_{diff}) * \begin{cases} 1 & \text{if } d_{view}^{min} < d(\mathbf{x}, cam) < d_{res}^{max} \\ \frac{d(\mathbf{x}, cam)}{d_{view}^{min}} & \text{if } d(\mathbf{x}, cam) < d_{view}^{min} \\ exp\left(-\frac{d(\mathbf{x}, cam) - d_{res}^{max}}{d_{res}^{max}}\right) & \text{if } d(\mathbf{x}, cam) > d_{res}^{max} \end{cases} \tag{9}$$

where

$$H(\theta_{diff}) = \begin{cases} 1 & \text{if } \theta_{diff} < \theta^{min} \\ \frac{\theta_{diff} - \theta^{min}}{\theta^{max} - \theta^{min}} & \text{if } \theta^{min} < \theta_{diff} < \theta^{max} \\ 0 & \text{if } \theta_{diff} > \theta^{max} \end{cases} \tag{10}$$

and using $\theta^{min} = \pi/2$ and $\theta^{max} = \pi$, we obtained sensor configuration [d]. Note that camera one moves inwards compared to configuration [c] 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. [5 (b)]. 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.

Relaxing the assumption of uniform density, if variable density is assumed such that the density is highest near the door and decreases linearly with the distance from it[f], configuration [e] was found to be the best. Note that, compared to [a], the cameras move closer to the door in order to better capture the region with higher object density.

Next, we consider a stereo assumption such that 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:

$$e_{tr} \propto \sqrt{d_1^2 + d_2^2 + d_1 d_2 \cos(\alpha)} / \sin(\alpha) \tag{11}$$

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:

$$e_m \propto d_1/cos(\alpha/2) + d_2/cos(\alpha/2) \tag{12}$$

Considering a quality function that uses a weighted average of the two errors: $q = -(w_1 e_{tr} + w_2 e_m)$, configuration [g] was found to the best. Note that all the three cameras come closer to each other in order to be able to do 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 [h]. The same constraint with the left wall yields configuration [i]. 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. 4 [j], 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. Additionally, 3D object localization is to be performed via stereo reconstruction for all parts of the scene. Note how the sensor placement varies in the three sections due to different combination of tasks.

5.2 Real Scenes

In real scenes, we first consider sensor planning in a small controlled environment [Fig. 6]. 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. 7 (a)]. 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 human to try to position the cameras manually and the experiments were conducted with this configuration as well. Results of this experiment are presented in Fig.s [6(a),(b),(c),(d) & 7]. In the next experiment, we 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 and the middle pillar/obstruction, thus making detection in front of them difficult. This condition was then integrated into the quality function. The results of this experiment are shown in Fig.s [6 (e),(f),(g),(h) & 7].

Next, we consider camera placement in the lobby of a building where we estimated the person densities over a period of time via a common background subtraction method [23] and a subsequent "foot finding" algorithm. This information was then fed back into the sensor planner to optimize for different objectives as shown in Fig. [8].

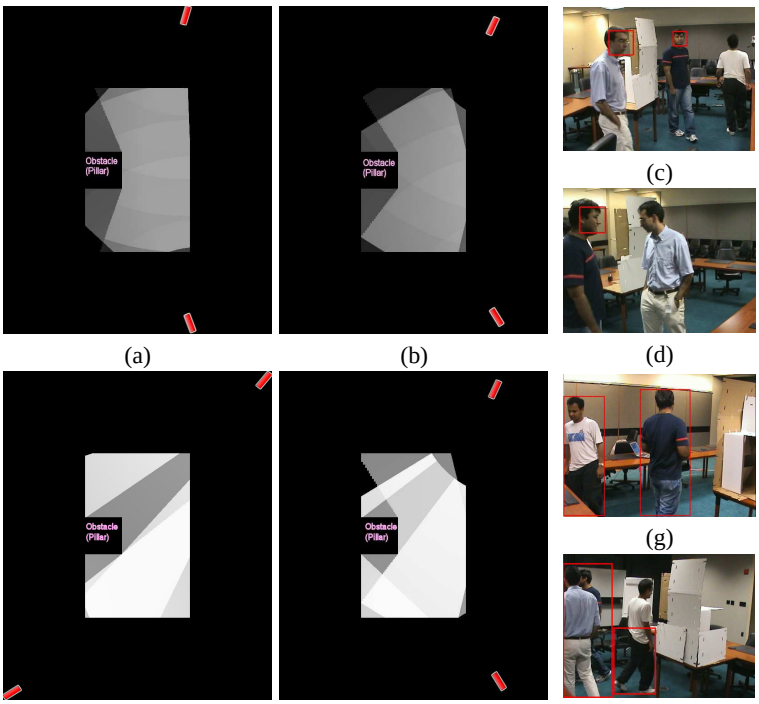


Fig. 6. (a): Configuration of two cameras for optimum face detection. (b): Configuration selected by a human operator. (c): An image from one of the cameras in (a). (d): An image from one of the cameras in (b). Note that one of the faces is not detected because of a large viewing angle. (e): Configuration of two cameras for person detection using background subtraction, where the top wall matches the color of people 33% of the time. (f): Configuration selected by a human operator. (g): An image from one of the cameras in (e). (h): An image from one of the cameras in (f). Note how the top portion of one person is not detected due to similarity with the background.

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

(a)

	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%

(b)

Fig. 7. (a): Face Detection rates for different distances from the cameras. Additionally, detection rates reduced by about 30% from frontal to the side view. This information was used by the sensor planner in the quality function. (b): Detection rates predicted by the algorithm compared with the actual rates obtained from experimental data.

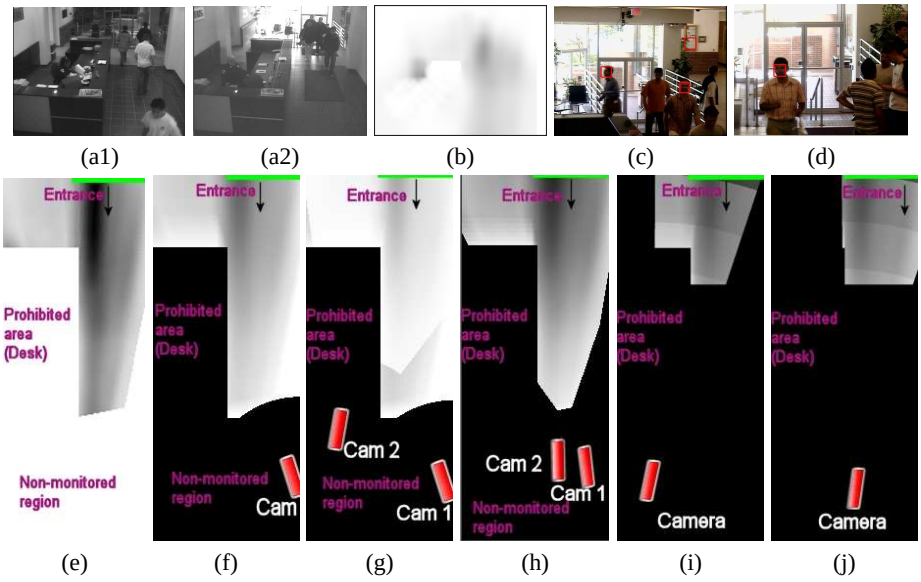


Fig. 8. Sensor placement in a lobby. (a): Two views from an original camera location at different times of the day. (b): Density map obtained via background subtraction (darker represents higher object density). (e): Mapping of the density map onto a plan view of the scene. (f): Optimal object visibility using one camera (72% visibility predicted, 78% obtained). (g): Optimal sensor placement using two cameras (91% visibility predicted, 93% obtained). (h): Optimal sensor placement using two cameras and a stereo requirement. (i): 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 (c). (j): Optimization of face detection when the position of the camera cannot be changed (but the direction and zoom can) (33 % detection predicted, 35 % obtained). An example of face detection using this setting is shown in (d).

6 Conclusion

We have considered analysis of scenes that may contain dynamic objects occluding each other. Multi-view visibility analysis for such scenes was integrated with user-defined quality criteria based possibly on several static constraints such as image resolution, stereo matching and field of view. Apart from obtaining important performance characteristics of multi-sensor systems, such analysis was further utilized for obtaining optimal sensor configurations. The algorithm can be customized for optimum sensor placement for a variety of existing multi-sensor systems and has applications in several fields, including surveillance where it can be utilized in places such as 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 the visibility probability by considering the effect of interaction between objects.

References

[1] Q. Cai and J.K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *PAMI*, 21(11):1241–1247, November 1999.

[2] A. Cameron and H.F. Durrant-Whyte. A bayesian approach to optimal sensor placement. *IJRR*, 9(5):70–88, 1990.

[3] Xing Chen and James Davis. Camera placement considering occlusion for robust motion capture. Technical Report CS-TR-2000-07, Stanford University, December 2000.

[4] R.T. Collins, A.J. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multisensor surveillance. *Proceedings of the IEEE*, 89(10):1456–1477, October 2001.

[5] D.J. Cook, P. Gmytrasiewicz, and L.B. Holder. Decision-theoretic cooperative sensor planning. *PAMI*, 18(10):1013–1023, October 1996.

[6] C. K. Cowan and P.D. Kovesi. Automatic sensor placement from vision tast requirements. *PAMI*, 10(3):407–416, May 1988.

[7] H. González-Banos and J.C. Latombe. A randomized art-gallery algorithm for sensor placement. In *SCG*, Medford, MA, June 2001.

[8] G. Hager and M. Mintz. Computational methods for task-directed sensor data fusion and sensor planning. *IJRR*, 10(4):285–313, August 1991.

[9] L. Ingber. Very fast simulated re-annealing. *Mathematical Computer Modeling*, 12:967–973, 1989.

[10] S.B. Kang, S.M. Seitz, and P.P. Sloan. Visual tunnel analysis for visibility prediction and camera planning. In *CVPR*, pages II: 195–202, Hilton Head, SC, June 2000.

[11] S. Khan and M. Shah. Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. *PAMI*, 25(10):1355–1360, October 2003.

[12] K.N. Kutulakos and C.R. Dyer. Recovering shape by purposive viewpoint adjustment. *IJCV*, 12(2-3):113–136, April 1994.

[13] J. Maver and R.K. Bajcsy. Occlusions as a guide for planning the next view. *PAMI*, 15(5):417–433, May 1993.

[14] A. Mittal and L.S. Davis. M_2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 51(3):189–203, February 2003.

[15] A. Mittal and L.S. Davis. Visibility analysis and sensor planning in dynamic environments. In *ECCV*, page III: 543 ff., Prague, Czech Republic, May 2004.

[16] A. Mittal and L.S. Davis. A general method for sensor planning in multi-sensor systems: Extension to random occlusion. *Submitted to IJCV*, 2005.

[17] J. Miura and K. Ikeuchi. Task-oriented generation of visual sensing strategies. In *ICCV*, pages 1106–1113, Boston, MA, 1995.

[18] Joseph O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, August 1987.

[19] A. Rahimi, B. Dunagan, and T.J. Darrell. Simultaneous calibration and tracking with a network of non-overlapping sensors. In *CVPR*, pages I: 187–194, 2004.

[20] M. K. Reed and P. K. Allen. Constraint-based sensor planning for scene modeling. *PAMI*, 22(12):1460–1467, December 2000.

[21] Yi Shang. *Global Search Methods for Solving Nonlinear Optimization Problems*. PhD thesis, University of Illinois at Urbana-Champaign, 1997.

[22] J. Spletzer and C.J. Taylor. A framework for sensor planning and control with applications to vision guided multi-robot systems. In *CVPR*, Kauai, Hawaii, 2001.

[23] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *PAMI*, 22(8):747–757, August 2000.

[24] K. Tarabanis, R.Y. Tsai, and A. Kaul. Computing occlusion-free viewpoints. *PAMI*, 18(3):279–292, March 1996.

[25] Y. Ye and J.K. Tsotsos. Sensor planning for 3d object search. *CVIU*, 73(2):145–168, February 1999.

[26] S.K. Yi, R.M. Haralick, and L.G. Shapiro. Optimal sensor and light-source positioning for machine vision. *CVIU*, 61(1):122–137, January 1995.