

# Automatic reconfiguration of video sensor networks for optimal 3D coverage

Claudio Piciarelli  
University of Udine  
claudio.piciarelli@uniud.it

Christian Micheloni  
University of Udine  
christian.micheloni@uniud.it

Gian Luca Foresti  
University of Udine  
gianluca.foresti@uniud.it

**Abstract**—During the last years, the research in the field of video analytics has focused more and more on video sensor networks. Although single-sensor processing is still an open research field, practical applications nowadays require video analysis systems to explicitly consider multiple sensors at once, since the use of multiple sensors can lead to better algorithms for tracking, object recognition, etc. However, given a network of video sensors, it is not always clear how the network should be configured (in terms of sensor orientations) in order to optimize the system performance. In this work we propose a method to compute a (locally) optimal network configuration maximizing the coverage of a 3D environment, given that a *relevance map* of the environment exists, expressing the coverage priorities for each zone. The proposed method relies on a transformation projecting the observed environment into a new space where the problem can be solved by means of standard techniques such as the Expectation-Maximization algorithm applied to Gaussian Mixture Models.

## I. INTRODUCTION

The use of Pan-Tilt-Zoom (PTZ) cameras in video analytics systems has lately become more and more popular. Active cameras that can be reconfigured in terms of orientation and zoom levels open a totally new area of investigation, leading to new solutions (and new problems) that were not considered in the past, when video analysis systems mainly relied on static sensors. PTZ cameras are nowadays especially used for tracking purposes, so that the camera can simultaneously zoom on a specific target and actively follow it, so that the camera won't lose the target because of its restricted field of view. However, there are many other possible uses of PTZ cameras, one of them is coverage optimization. If the cameras are used to monitor a large environment, it is possibly unrealistic to have each zone observed (covered) by at least one sensor. When deploying the cameras it is thus necessary to find not only the right places where to put them (but this is often constrained by practical issues, such as the presence of a power supply), but also the best orientation. Moreover, this optimal configuration is not necessarily static, as it could depend on dynamically changing coverage requirements.

This paper extends the work of Piciarelli et al. [1] on camera networks reconfiguration. The main idea is to automatically compute the optimal orientation (pan and tilt angles) and magnification (zoom) parameters for each camera of the network, in order to optimize the visual coverage of a given environment. The optimization process relies on the use of a *relevance map*, a function denoting the coverage priority

for each point of the monitored environment. The relevance map is thus used to give more “importance” to the visual coverage of some specific zones, and its definition is of course context-dependent. For example the surveillance system of a parking lot could give more importance to car entry/exit points, meaning that a reconfiguration of the network of sensors should give higher priority on the coverage of those zones. The relevance maps could even be automatically generated, e.g. in order to give more importance to the areas with higher activity (high detection of moving objects). The work proposed in [1] is based on 2D relevance maps, assuming that the cameras are placed high enough to approximate all the zones to be observed as lying on the ground plane. The main novelty of this work is the extension of the reconfiguration technique to full 3D relevance maps for applications where different heights cannot be ignored. An example could be a closed environment (e.g. an office) where the system must primarily focus on the zones where faces can be detected. In this case, the face positions cannot be approximated with their projection on the ground plane and the heights at which heads can be detected are not constant across the environment (e.g. areas where people walk and working areas where people sit).

The paper is structured as follows: section II discusses some related works on sensor reconfiguration, although few works have been proposed on reconfiguration for coverage optimization. Section III describes the proposed method in full detail, and finally experimental results are shown in section IV.

## II. RELATED WORK

Up to now sensor reconfiguration techniques have been mainly proposed to improve tracking performances. Jain et al. [2] propose a master-slave architecture involving both static and dynamic cameras. Soto et al. [3] developed a system for multi-target tracking with a network of self-reconfiguring PTZ cameras without the need of a central processing unit by using a distributed version of the Kalman filter. Karupiah *et al.* [4] propose two new metrics that, based on the dynamics of the scene, allow to choose the pair of cameras that maximize the detection probability of a moving object. In [5], Park et al. discuss a distributed look-up table based approach to determine the cameras' viewing frustums that allows to select the best cameras for tracking purposes. Qureshi and Terzopoulos [6] propose a proactive control of multiple PTZ cameras through a solution that plans assignment and handoff.

In particular, the authors consider the problem of controlling multiple cameras as a *multibody* planning problem in which a central planner controls the actions of multiple physical agents. In the context of person tracking, their approach computes the relevance of a PTZ camera to an observation task by considering five factors: a) camera-pedestrian distance, b) frontal viewing direction, c) PTZ limits, d) observational range and e) handoff success probability. The planing is then achieved by employing a greedy best-first search to find the optimal sequence of states. Other approaches to sensor reconfiguration for people tracking have been developed by employing game theory. Arslan *et al.* [7], demonstrate that the problem of finding an optimal configuration can be expressed in terms of a game whose solution is expressed by the Nash equilibrium. From this formulation, different approaches [8], [9] solve the camera assignment problem by maximizing a global utility function. Different mechanisms to compute the utilities can be provided as in [8], [9], then a bargaining process is executed on the predictions of person utilities at each step. The cameras with the highest probabilities are used to track the target, thus providing a solution to the handoff problem in a video network. On the other hand, when a PTZ camera is reconfigured to track an object or it is switched on/off to save power, the topology of the network is modified. As consequence, a new configuration is required to provide optimal coverage of the monitored environment. Song *et al.* [9] adopt a uniform distribution of the targets and the coverage resolution utility to negotiate the new network reconfiguration.

Few works address the problem of optimizing the camera coverage of the monitored area according to specific criteria. Angella *et al.* [10] propose a method to maximize the area coverage by using a 3D model of the observed zone, but their work only aims at finding a good initial camera displacement, which cannot be dynamically modified according to the observed data. Mittal and Davis [11], [12] also consider the presence of dynamic occluding objects in order to evaluate the visibility of the scene. Piciarelli *et al.* [1] propose a method to automatically and dynamically reconfigure the camera orientations and zoom levels using an Expectation-Maximization-based approach.

### III. SENSOR RECONFIGURATION

Let  $\omega(\mathbf{x})$ ,  $x \in \mathcal{H} \subset \mathbb{R}^3$  be the relevance map denoting the relevance for each point  $\mathbf{x}$  belonging to the finite volume  $\mathcal{H}$  to be monitored,  $\omega(\mathbf{x}) \geq 0 \forall \mathbf{x} \in \mathcal{H}$ .

Assuming that the network is composed of  $K$  different sensors, let also  $\Theta = \{\phi_1, \theta_1, \zeta_1, \dots, \phi_K, \theta_K, \zeta_K\}$  be the set of parameters denoting the current network configuration, expressed in the form of pan ( $\phi$ ), tilt ( $\theta$ ) and cone-of-view width ( $\zeta$ ) angles for each camera. The position  $\{X_k, Y_k, Z_k\}$  of each camera is fixed and defined a priori.

We define the *observation function*  $\gamma_k$  for each camera  $k$  as

$$\gamma_k(\mathbf{x}; \Theta) = \begin{cases} 1 & \text{if } \mathbf{x} \in \text{cone-of-view of camera } k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The function  $\gamma_k$  thus denotes if a given point is currently observed by a specific camera, according to the current configuration  $\Theta$ .

The *score function*  $f(\mathbf{x}; \Theta, C)$  measures the total coverage for each point  $\mathbf{x}$ :

$$f(\mathbf{x}; \Theta, C) = \left( \sum_{k=1}^K c_k \gamma_k(\mathbf{x}; \Theta) \right)^{\omega(\mathbf{x})} \quad (2)$$

Higher scores are given to points covered by several cameras, as in many surveillance and monitoring applications it is generally advisable to have different views of the same zone. Moreover, the exponent  $\omega(\mathbf{x})$  gives higher scores to points with higher relevance. The values  $c_k \in C$  are weights denoting the “importance” of each camera (in terms of relevance of observed points). These weights are important especially in resource-aware contexts, e.g. when the system must switch off the least important cameras because of limited resources (power, bandwidth, etc.). Parameters  $\Theta$  and  $C$  are the system unknowns to be found in order to maximize the global score function  $\Lambda$ , defined using a product integral as:

$$\Lambda(\Theta, C) = \int_{\mathcal{H}}^{\times} f(\mathbf{x}; \Theta, C)^{d\mathbf{x}} \quad (3)$$

or, in case  $\mathcal{H}$  is a discrete set of points as in many practical applications:

$$\Lambda(\Theta, C) = \prod_{\mathbf{x} \in \mathcal{H}} f(\mathbf{x}; \Theta, C) \quad (4)$$

Maximizing  $\Lambda$  is equivalent to maximize its logarithm, thus our final global score function  $\lambda$  is defined as:

$$\lambda(\Theta, C) = \log(\Lambda(\Theta, C)) = \int_{\mathcal{H}} \log f(\mathbf{x}; \Theta, C) d\mathbf{x} \quad (5)$$

or, in discrete form:

$$\lambda(\Theta, C) = \sum_{\mathbf{x} \in \mathcal{H}} \log f(\mathbf{x}; \Theta, C) \quad (6)$$

The optimization process however is non-trivial, and requires the explicit definition of  $\gamma_k$  in terms of  $\phi_k, \theta_k$  and  $\zeta_k$ . We thus propose a reformulation of the problem that can be solved with standard techniques.

Since cameras can rotate in pan and tilt directions, a spherical coordinate system seems a natural choice to represent the data as seen from a given camera. Moreover, the observation function (1) in a spherical system does not depend on the distance from the camera, thus we can drop the distance dimension and keep only the two angular dimensions. More formally, for each point  $\mathbf{x}$  of coordinates  $(x, y, z)$ , its spherical coordinates  $(\phi_k, \theta_k)$  in a system centered on camera  $k$  are defined as

$$\begin{cases} \phi_k = \arctan \left( \frac{y - Y_k}{x - X_k} \right) \\ \theta_k = \arctan \left( \frac{\sqrt{(x - X_k)^2 + (y - Y_k)^2}}{z - Z_k} \right) \end{cases} \quad (7)$$

where the arctan function also takes in consideration the sign of the numerator and denominator, as done in the `atan2` function defined in many programming and scripting languages. Assuming that the cameras are always located above the volume  $\mathcal{H}$  to be monitored, the ranges are  $\phi_k \in (-\pi, \pi]$ ,  $\theta_k \in [0, \pi/2]$ .

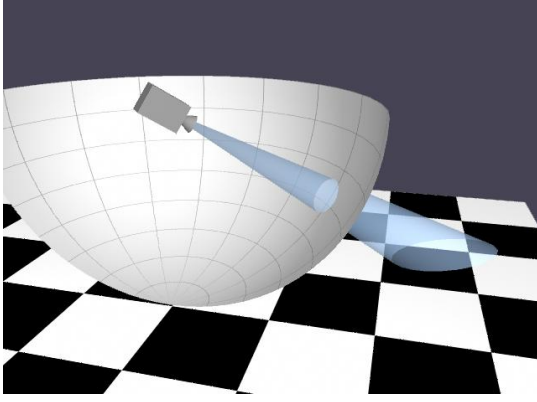


Fig. 1. The intersection of the cone of view of a camera with the unitary sphere centered on the camera forms a circle.

Since we dropped the distance coordinate, the coordinate change (7) maps the monitored volume on the surface of a unitary semisphere centered on the camera. Figure 1 clearly shows that the intersection of the camera's cone of view with the unitary sphere is a circle, which is the spherical equivalent of the observation function  $\gamma_k$ . A second coordinate change is now introduced to project the spherical surface on a plane  $\mathcal{P}_k$ . For this purpose, the *stereographic projection* is used:

$$\begin{cases} u_k = 2 \tan(\theta_k/2) \cos \phi_k \\ v_k = 2 \tan(\theta_k/2) \sin \phi_k \end{cases} \quad (8)$$

Figure 2(a) shows how the stereographic projection works, projecting any point  $P'$  lying on the surface of an unitary sphere to the point  $P''$  on the plane  $z = -1$  from the projection point  $(0, 0, 1)$ . This projection has been widely used in cartography, and it is well known for being a conformal projection (this is, locally preserving angles) such that circles not containing the projection point  $(0, 0, 1)$  are mapped to circles on the plane, as shown in figure 2(b). A combination of equations (7) and (8) thus maps all the points within the cone of view of a given camera  $k$  into a circle in  $\mathcal{P}_k$ , as shown in figure 3.

Let  $M_k$  be the coordinate change described above:

$$M_k(\mathbf{x}) = (2 \tan(\theta_k/2) \cos \phi_k, 2 \tan(\theta_k/2) \sin \phi_k) \quad (9)$$

$$\mathbf{x} \in \mathcal{H}, M_k(\mathbf{x}) \in \mathcal{P}_k$$

with  $\phi$  and  $\theta$  defined as in equation (7). We can now define  $\Gamma_k$  as the equivalent of  $\gamma_k$  in the new coordinate system:

$$\Gamma_k(M_k(\mathbf{x}); \Theta) = \gamma_k(\mathbf{x}; \Theta) \quad (10)$$

and we have shown that  $\Gamma_k$  has a circular shape, since it is the stereographic projection of the intersection of a cone of

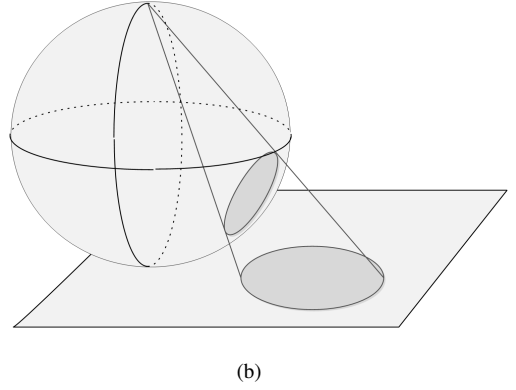
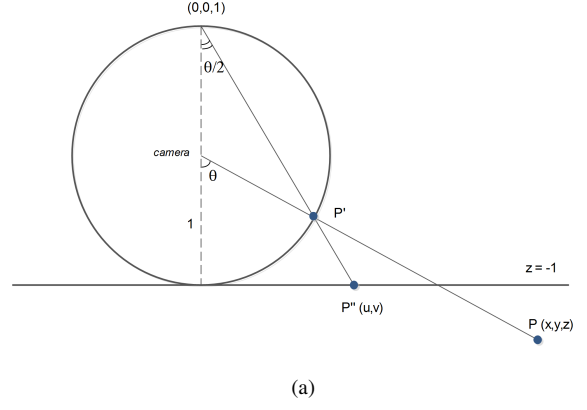


Fig. 2. A stereographic projection maps the surface of a sphere onto a plane. (a) 3D point  $P$  mapped on a 2D plane using a stereographic projection; (b) stereographic projections preserve circles.

view with an unitary sphere centered on the cone vertex.  $\Gamma_k$  can thus be written in the form

$$\Gamma_k(u, v; \Theta) = \begin{cases} 1 & \text{if } (u - \mu_{u,k}^\Theta)^2 + (v - \mu_{v,k}^\Theta)^2 \leq (\sigma_k^\Theta)^2 \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where  $(\mu_{u,k}^\Theta, \mu_{v,k}^\Theta) \in \mathcal{P}_k$  are the coordinates of the center of the circle for camera  $k$  with configuration  $\Theta$  and  $\sigma_k^\Theta$  is its radius.

We now approximate  $\Gamma_k$  with a Gaussian function. This approximation is mainly needed to keep the problem tractable, as typical optimization algorithms can be applied only to continuous functions, but can also be seen as a way to give more importance to the zones at the center of the image rather than on its borders, a property that could be desirable in many practical applications. Since the area where  $\Gamma_k$  is nonzero has a circular shape, an isotropic bivariate Gaussian function is sufficient:

$$\Gamma_k(\mathbf{x}; \Theta) \approx G_k(\mathbf{x}; \Theta) = \frac{1}{2\pi\sigma_k^{\Theta 2}} e^{-\frac{\|\mathbf{x} - \mu_k^\Theta\|^2}{2\sigma_k^{\Theta 2}}} \quad (12)$$

where  $\mu_k^\Theta = (\mu_{u,k}^\Theta, \mu_{v,k}^\Theta)$  is the mean of the Gaussian function and  $\sigma_k^\Theta$  is the standard deviation. Observe that these values depend on the camera  $k$  and the current configuration  $\Theta$ . This

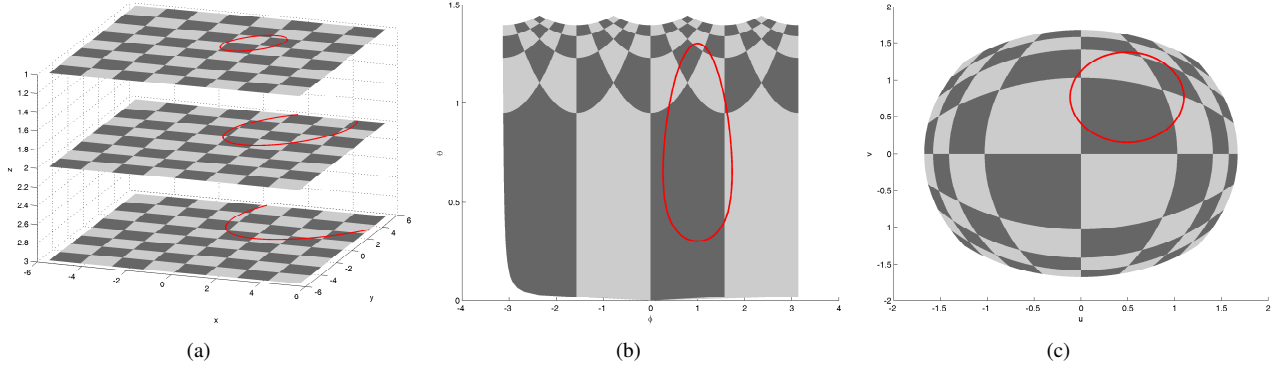


Fig. 3. The proposed coordinate changes. (a) the ellipses resulting from the intersection of the cone of view of a camera with several planes in  $\mathcal{H}$ ; (b) the cone of view is mapped into a 2D region in  $(\phi_k, \theta_k)$  coordinates; (c) the same region has a circular shape in  $\mathcal{P}_k$  after stereographic mapping.

approximation allows to rewrite the score function (eq. 2) as

$$f(\mathbf{x}; \Theta, C) = \left( \sum_{k=1}^K c_k G_k(M_k(\mathbf{x}); \Theta) \right)^{\omega(\mathbf{x})} \quad (13)$$

and by substituting eq. 13 in the discrete global score function (eq. 6) we obtain:

$$\lambda(\Theta, C) = \sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) \log \sum_{k=1}^K c_k G_k(M_k(\mathbf{x}); \Theta) \quad (14)$$

We must now recall that the final goal is to find a camera network configuration  $\hat{\Theta}$  (and a set of weights  $\hat{C}$ ) that maximizes the global score function:

$$\hat{\Theta}, \hat{C} = \underset{\Theta, C}{\operatorname{argmax}} \lambda(\Theta, C) \quad (15)$$

however, the above optimization problem is extremely similar to a well-known parameter estimation algorithm, the *Expectation-Maximization* (EM) algorithm [13]. The only differences with respect to the standard algorithm are the presence of the relevance  $\omega(\mathbf{x})$  and the Gaussian function being computed on  $M_k(\mathbf{x})$  rather than on  $\mathbf{x}$ . As in the original EM formulation, a locally optimal solution of the problem can be found by setting to zero the partial derivatives of  $\lambda(\Theta, C)$  with respect to  $\mu_k^\Theta$ ,  $\sigma_k^\Theta$  and  $c_k$ . Since neither  $\omega(\mathbf{x})$  nor  $M_k(\mathbf{x})$  depend on the integration variables, the procedure is basically the same of standard EM and leads to the following solution:

$$\begin{aligned} \mu_k^\Theta &= \frac{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) p(k|\mathbf{x}) M_k(\mathbf{x})}{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) p(k|\mathbf{x})} \\ \sigma_k^{\Theta^2} &= \frac{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) p(k|\mathbf{x}) \|M_k(\mathbf{x}) - \mu_k^\Theta\|^2}{2 \sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) p(k|\mathbf{x})} \\ c_k &= \frac{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) p(k|\mathbf{x})}{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x})} \end{aligned} \quad (16)$$

where  $p(k|\mathbf{x})$  is defined as

$$p(k|\mathbf{x}) = \frac{c_k \omega(\mathbf{x}) G(M_k(\mathbf{x}); \Theta)}{\sum_{z=1}^K c_z \omega(\mathbf{x}) G(M_z(\mathbf{x}); \Theta)} \quad (17)$$

Equations 16 and 17 cannot be solved independently since they are mutually dependent, however it has been proven

that an iterative process that alternates the computation of 17 (Expectation step) and 16 (Maximization step) will eventually converge to the correct solution. Iteration can also run endlessly in order to automatically take into account changes in dynamically-updated relevance maps.

Once the optimal means  $\mu_k$  have been found, it would be tempting to apply an inverse stereographic mapping to retrieve the corresponding pan and tilt angles for each camera  $k$ . However, it must be noted that the mean of each Gaussian function in  $\mathcal{P}_k$  does *not* coincide with the center of the corresponding circle on the surface of the unitary sphere. This is because the stereographic projection is not isometric (distance preserving). More specifically, the pan angle is preserved, but the tilt angle is not, as shown in the side-view of figure 4, where it is evident that the mean  $\mu$  is different from  $C'$ , projection on  $\mathcal{P}$  of the intersection  $C$  of the optical axis of the camera with the unitary sphere. Inverting the stereographic projection equation (8) it is however easy to compute the minimum and maximum angles  $\theta_1$  and  $\theta_2$  within the cone of view (see again figure 4), defined as:

$$\begin{cases} \theta_1 &= 2 \arctan \left( \frac{\|\mu_k\| + \sigma_k}{2} \right) \\ \theta_2 &= 2 \arctan \left( \frac{\|\mu_k\| - \sigma_k}{2} \right) \end{cases} \quad (18)$$

and the desired tilt angle is the mean of  $\theta_1$  and  $\theta_2$ . These two angles also allow to define the angular width  $\zeta$  of the cone of view.

In other words, an isotropic Gaussian function with mean  $\mu$  and standard deviation  $\sigma$  approximates a circle in  $\mathcal{P}_k$  corresponding to the region observed by camera  $k$  whose configuration is:

$$\begin{cases} \phi_k &= \arctan(\mu_{v,k}/\mu_{u,k}) \\ \theta_k &= (\theta_1 + \theta_2)/2 \\ \zeta_k &= (\theta_1 - \theta_2)/2 \end{cases} \quad (19)$$

Recapping the whole procedure: we have mapped the volume observed by a camera  $k$  into a circle in a new space  $\mathcal{P}_k$  (equation 8). If each circle is approximated with an isotropic Gaussian function, the problem of finding on optimal camera network configuration optimizing the global coverage

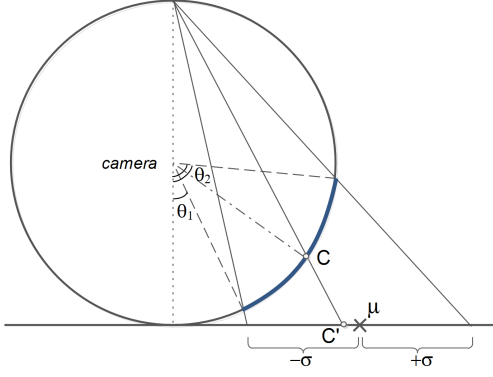


Fig. 4. The back-projection of the mean  $\mu$  on the unitary sphere does not coincide with the center of the circle obtained from the intersection of the cone of view and the unitary sphere.

is reduced to an application of the Expectation-Maximization algorithm (equations 17 and 16). Once the algorithm converges to a solution, the pan, tilt and angular width angles for each camera  $k$  can be obtained from the mean  $\mu_k$  and variance  $\sigma_k$  of the corresponding Gaussian function using equation (19).

#### IV. EXPERIMENTAL RESULTS

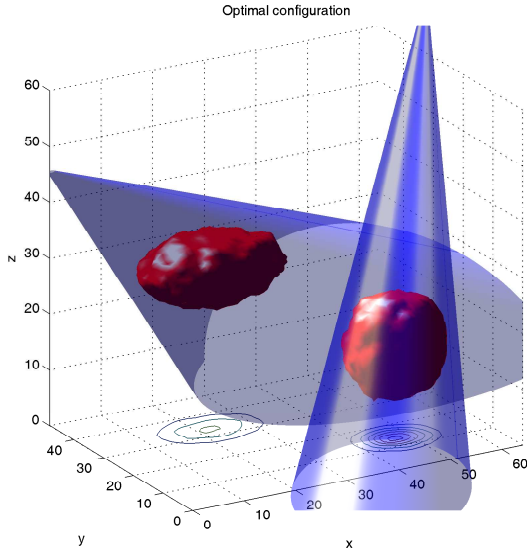


Fig. 5. Optimal 2-camera system configuration for a relevance function defined as a mixture of two trivariate Gaussian functions.

Visualizing the system results can be difficult, since it requires to graphically display the 3D relevance function  $\omega(x, y, z)$ . In order to give a visual example of the proposed method we thus first present how the system runs on a toy dataset shown in figure 5. In this example,  $\omega$  is a bimodal 3D probability density function obtained as a mixture of two Gaussian trivariate functions with centers in  $\mu_1 = (24, 32, 30)$  and  $\mu_2 = (48, 11, 16)$  respectively. The surface displayed in red is the isosurface such that  $\omega(x, y, z) = 0.1$ , a contour

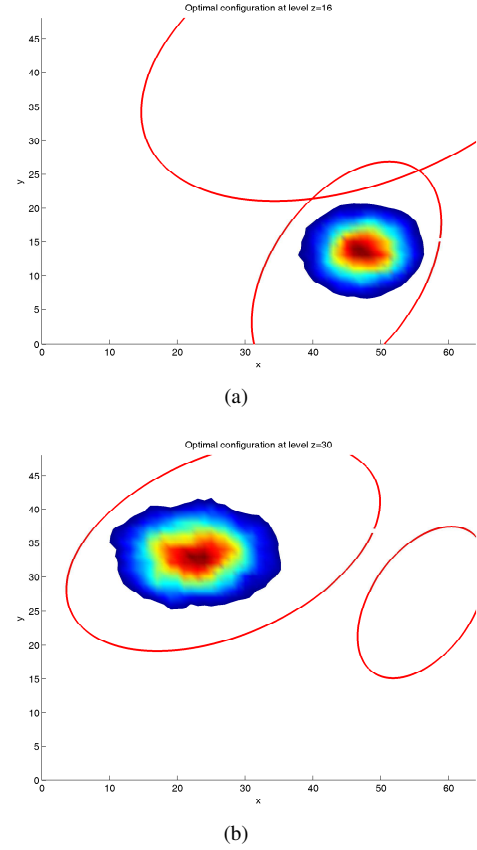


Fig. 6. Slices of the relevance function and camera cone of views at  $z = 16$  (a) and  $z = 30$  (b) for the example shown in figure 5.

projection of  $\omega$  on the plane  $z = 0$  has been drawn in order to give a better visualization of the heights of the two main clusters. Two cameras are placed at coordinates  $(80, 60, 60)$  and  $(-20, 15, 60)$  and initially configured to aim downwards: pan, tilt, and cone-of-view width angles are set to  $\phi = 0^\circ, \theta = 0^\circ, \zeta = 28.6^\circ$  for both cameras. After 12 iterations, the system converges to the configuration  $\phi_1 = -122.00^\circ, \theta_1 = 52.80^\circ, \zeta_1 = 21.00^\circ, \phi_2 = 21.99^\circ, \theta_2 = 21.99^\circ, \zeta_2 = 33.44^\circ$  which is shown in figure 5. Figures 6(a) and 6(b) show two slices of the achieved result respectively at  $z = 16$  and  $z = 30$ , the mean heights of the two main clusters. As it can be seen, at each level the cluster is fully enclosed in one of the two ellipses, representing the intersection of the cone of view with the plane, meaning that the system effectively focused the two cameras on the clusters. Figure 7 shows the global score (equation (6)) at each iteration of the EM algorithm. As it can be seen, the global score increases at each step, until the algorithm converges to a local maximum.

In order to give an intuitive performance measurement, we finally define the *total coverage*  $C$  as the ratio between the relevance of all the points falling within the cone of view of at least one camera versus the total relevance:

$$C = \frac{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x}) \max_k \{\Gamma_k(M_k(\mathbf{x}))\}}{\sum_{\mathbf{x} \in \mathcal{H}} \omega(\mathbf{x})} \quad (20)$$

For example, the final configuration of the experiment shown

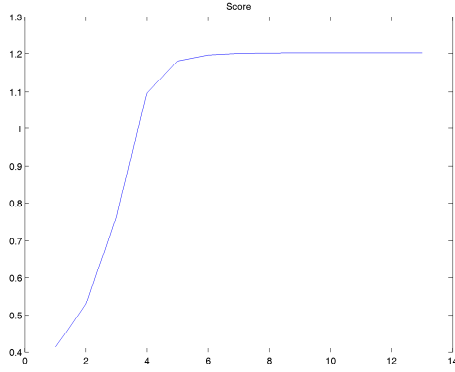


Fig. 7. The global score of the camera configuration increases at each iteration.

in figure 5 has  $C = 0.9529$ , meaning that more than 95% of the relevance falls within the regions observed by at least one camera. Note that  $C$  is not 1 since the isosurface shown in figure 5 is where  $\omega(\mathbf{x}) = 0.1$ , thus there are still some nonzero-relevance points possibly outside the observed regions.

Using the total coverage measure, we finally evaluated the performance of the proposed algorithm on a set of 20 different relevance maps. The maps are created using an extension of the trajectory generator already used in [1]. The trajectory generator creates random sets of 3D trajectories grouped in a limited number of clusters (from 2 to 5). The space where trajectories lie has then been discretized into a grid of  $64 \times 48 \times 30$  cells, and each cells is assigned the number of trajectories crossing it. This way the relevance map is seen as an activity map, denoting how many tracked objects have been detected in a given portion of space (figure 8). We believe this is one of the most direct applications of the proposed system, since in many contexts it is advisable to automatically focus the cameras on the most active zones. The number and position of the cameras is chosen randomly, from a minimum of 2 to a maximum of 5 cameras, always placed outside the map. The measured total coverage for the considered datasets is shown in table I. As it can be seen, the system behaved well in all the cases, with a minimum and maximum total coverage respectively of 0.9304 and 0.9998.

| dataset | total coverage | dataset | total coverage |
|---------|----------------|---------|----------------|
| 1       | 0.9693         | 11      | 0.9987         |
| 2       | 0.9998         | 12      | 0.9545         |
| 3       | 0.9723         | 13      | 0.9785         |
| 4       | 0.9923         | 14      | 0.9614         |
| 5       | 0.9977         | 15      | 0.9735         |
| 6       | 0.9824         | 16      | 0.9788         |
| 7       | 0.9775         | 17      | 0.9817         |
| 8       | 0.9891         | 18      | 0.9937         |
| 9       | 0.9815         | 19      | 0.9798         |
| 10      | 0.9304         | 20      | 0.9615         |

TABLE I

EXPERIMENTAL RESULTS ON 20 DIFFERENT DATA SETS. THE TOTAL COVERAGE IS DEFINED AS THE RATIO BETWEEN THE RELEVANCE OF ALL THE POINTS FALLING WITHIN THE CONE OF VIEW OF AT LEAST ONE CAMERA VERSUS THE TOTAL RELEVANCE.

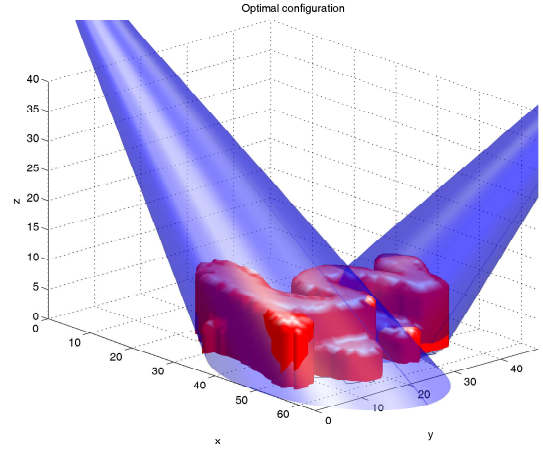


Fig. 8. One of the 20 datasets used for performance measurement.

## ACKNOWLEDGMENT

This work is partially supported by the Interreg IV Italy-Austria project n. 4697 “SRSNet - Intelligent Audio/Video Sensor Networks”.

## REFERENCES

- [1] C. Piciarelli, C. Micheloni, and G. Foresti, “Occlusion-aware multiple camera reconfiguration,” in *4th ACM/IEEE International Conference on Distributed Smart Cameras*, Atlanta, GA, USA, 2010, pp. 88–94.
- [2] A. Jain, D. Kopell, K. Kakligian, and Y. F. Wang, “Using stationary-dynamic camera assemblies for wide-area video surveillance and selective attention,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.
- [3] C. Soto, B. Song, and A. Roy-Chowdhury, “Distributed multi-target tracking in a self-configuring camera network,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009.
- [4] D. Karupiah, R. Grupen, A. Hanson, and E. Riseman, “Smart resource reconfiguration by exploiting dynamics in perceptual tasks,” in *International Conference on Intelligent Robots and Systems*, Edmonton, Canada, Aug 2-6 2005, pp. 1513–1519.
- [5] J. Park, P. C. Bhat, and A. C. Kak, “A look-up table based approach for solving the camera selection problem in large camera networks,” in *Workshop on Distributed Smart Cameras*, Boulder, CO, USA, Oct. 31 2006.
- [6] F. Qureshi and D. Terzopoulos, “Planning ahead for ptz camera assignment and handoff,” in *International conference on Distributed Smart Cameras*, Como, Italy, Aug-Sep 2009, pp. 1–8.
- [7] G. Arslan, J. Marden, and J. Shamma, “Autonomous vehicle-target assignment: A game-theoretical formulation,” *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 129, no. 5, pp. 584–596, 2007.
- [8] Y. Li and B. Bhanu, “Utility-based dynamic camera assignment and hand-off in a video network,” in *IEEE/ACM International Conference on Distributed Smart Cameras*, Stanford, USA, 7-11 Sep. 2008, pp. 1–9.
- [9] B. Song, C. Soto, A. K. Roy-Chowdhury, and J. Farrell, “Decentralized camera network control using game theory,” in *IEEE/ACM International Conference on Distributed Smart Cameras*, Stanford, USA, 7-11 Sep. 2008, pp. 1–8.
- [10] F. Angella, L. Reithler, and F. Gallezio, “Optimal deployment of cameras for video surveillance systems,” in *IEEE International Conference on Advanced Video and Signal based Surveillance*, London, UK, 2007.
- [11] A. Mittal and L. S. Davis, “A general method for sensor planning in multi-sensor systems: extension to random occlusion,” *International Journal of Computing Vision*, vol. 76, pp. 31–52, 2008.
- [12] —, “Visibility analysis and sensor planning in dynamic environments,” in *European Conference on Computer Vision*, Prague, CZ, May 2004.
- [13] T. Moon, “The expectation-maximization algorithm,” *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.