

# Optimal 3D Sensor Placement to Obtain Accurate 3D Point Positions

Gustavo Olague, Roger Mohr

► **To cite this version:**

Gustavo Olague, Roger Mohr. Optimal 3D Sensor Placement to Obtain Accurate 3D Point Positions. Primer Encuentro de Computacion ENC 97: Vision Robotica, Sep 1997, Queretaro, Mexico. pp.116–123. inria-00548349

**HAL Id: inria-00548349**

**<https://hal.inria.fr/inria-00548349>**

Submitted on 22 Dec 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Optimal 3D Sensors Placement to Obtain Accurate 3D Points Positions

Gustavo OLAGUE, Roger MOHR

MOVI-GRAVIR\*

ZIRST - 655 Avenue de l'Europe

38330 Montbonnot Saint Martin, France

tel. +33 4 76 61 52 35

fax. +33 4 76 61 52 10

*Gustavo.Olague@inrialpes.fr*

*Roger.Mohr@inrialpes.fr*

## Abstract

3D measurements can be achieved from several views using the principle of optical triangulation. This paper deals with the problem of where to place cameras in order to obtain a minimal error in the detection. We pose the problem in terms of an optimization design, dividing this in two main components: 1) an analytical part dedicated to the analysis of error propagation from which a criterion is derived, 2) an heuristical part which is going to minimize this criterion. In this way, the approach consists of an uncertainty analysis applied to the reconstruction process from which a covariance matrix is computed. This matrix represents the uncertainty of the detection from which the criteria is derived. Thus, a multicellular genetic algorithm is implemented in order to minimize the criterion. Graphical examples are provided to illustrate the effectiveness and efficiency of the solution.

**Key words:** Computer vision, genetic algorithms, covariance matrix, camera network design, machine learning.

## 1 Introduction

High accurate measures can be obtained through the selection of a well designed convergent camera configuration and a suitable mathematical model of the process. Researchers of the computer vision and photogrammetric communities have recognized the necessity to automate this process. For an example see the work of Fraser [Fraser, 1992] in which he reports highly accurate measures employing a strong photogrammetric network (hand-designed) using 36 photographs taken by 18 well disposed cameras with high convergence angles. In this way, the problem is very complex, because it involves many interrelated aspects: 1) the optical constraints related to the kind of sensor that we are using, 2) the mathematical model of optical triangulation which produces the data fusion and relates the optimal spatial placement of multiple sensors in order to reduce the uncertainties. An heuristic method is also necessary to resolve this highly complex process. This article deals with the problem of how to determine the attitude of several cameras which are observing some targets placed over a 3D object with the goal to achieve high-accurate measures. We present a prototype system called EPOCA (an acronym for “Evolving POSitions of CAMeras”) which we have implemented to test the validity of our approach.

The remaining part of the paper is divided as follows: firstly, we pose the problem specifying several questions which are going to be useful in the general planning of our discussion. Next, we continue with a review of the literature related with sensor planning. After, we describe our approach, dividing this part of our exposition in the following subsections: the reconstruction process, the 3D error estimation, the image error estimation all useful in the development of a criteria, next the optimal camera position method implemented like a multicellular genetic algorithm is described. Finally, some results product of our experiments are presented, followed by a discussion.

### 1.1 Statement of the Problem

The problem that we would like to approach is the automation of the camera network design process in or-

---

MOVI is a joint research project between the CNRS, INPG, INRIA Rhône-Alpes and UJF.

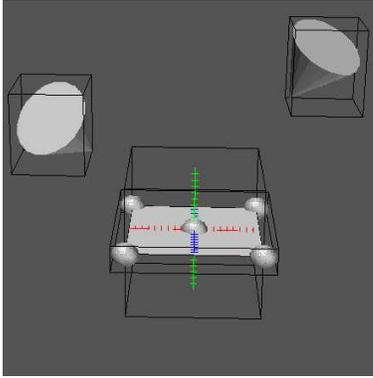


Figure 1: *Two convergent cameras. The cones illustrates the cameras and the ellipsoids represents the uncertainty of the observed targets.*

der to obtain accurate 3D measures. Imaging that we would like to accurately measure some fiducial marks distributed over a planar surface. The idea then, is to place the cameras in order to obtain a minimal error. Figure (1)<sup>1</sup> shows two cameras observing five targets represented by their error ellipsoids which describes the uncertainty in the position. Each time we change the attitude (position and orientation) of the cameras, so does the orientation and size of the ellipsoids. The question we would like to answer is:

- 
- Where are the approximate camera positions in order to obtain the minimal 3D error?

From this question several subproblems are derived like:

1. How can we develop a good criterion to judge our configuration?
  2. Under which conditions our system must work?
  3. Which are the interrelated aspects involved in the development of the system?
  4. What would be a good method to optimize the camera placement?
- 

Table 1: *Some initial questions.*

From these initial questions the choice of a **criterion** combined with an **optimization** process will be the key concepts.

## 1.2 Related Works

The problem to automate the multiple camera placement process in order to attain high accurate measures has received scant attention, though it is of great practical importance. Next, a survey of existing literature is presented starting with works to automate the placement

---

<sup>1</sup>Figures 1 and 6 were generated with software written at the Geometry Center, University of Minnesota.

of a single camera up to the expert system approach of Mason [Mason, 1994].

Between the systems using a single camera approach we have: the HEAVEN system developed by Sakane et al. [Sakane and Sato, 1991]. This is an example of a system developed to find the possible sensor positions using a *generate-and-test strategy* for inspecting an object with a tessellated sphere of pre-given radius. In which the object is unoccluded incorporating an analysis of light source planning. This study was further extended to the VIO system developed by Niepold and Sakane. Another example of generate-and-test approach is ICE [Yi *et al.*, 1995], where camera and light positions are generated and then evaluated based on selected criteria. Overall, it places emphasis on the illumination placement planning component of the problem. For the camera placement the task constraint considered is edge visibility. Some feature edges appear least occluded when the optimal sensor position takes place. The evaluation of this criterion is based on the aspect graph representation of the object.

However, other works have been adopting a *synthesis approach*. Instead of taking the discrete approach the idea is to model the constraints as analytical functions. *The Automatic Sensor and Illuminator Positioning Work of SRI* carried out by Cowan et al. [Cowan and Kovesi, 1988] talks about camera locations which are automatically generated. They satisfy several geometric constraints like: visibility, polyhedral objects in focus, within the sensor field of view, at sufficient resolution and incidence angle, and unoccluded by the object itself or other objects in the working environment. The MVP system developed by Tarabanis et al. [Tarabanis *et al.*, 1991] determines an optimal sensor location and sensor parameters (focal length, focus setting and aperture) for viewing a set of surfaces and avoiding occlusion. MVP is currently being extended to function in dynamic environments [Abrams *et al.*, ]. The placement of illumination sources for inspecting edge features has also been investigated by Cowan et al. While all of these systems provide solutions to the task of viewing an object, they do not address the task of deriving accurate object dimensions. They are also inadequate for complex objects on which not all features will be visible from a *single* viewpoint.

Tarbox and Gottschlich [Tarbox and Gottschlich, 1993] have recognised the need of *multistation solutions* to overcome problems of object occlusion. They have implemented a solution in the IVIS system for an active triangulation sensor. Cowan has also experimented methods to place multiple sensors overcoming the occlusion problem associated with 3D objects. Finally, a work called CONSENS which follows the expert system approach and uses multiple cameras in combination with optical triangulation, was developed by Mason et al. [Mason and Gruen, 1995]. It outlines a way to over-

come these limitations.

Unlike all other approaches, our idea is to pose the problem in terms of an optimization design.

## 2 The approach

This section presents a method to solve the problem of camera placement, which could be divided in two main components. Firstly, we will develop an analytic uncertainty analysis based in error propagation phenomenon which will allow us to express an error criterion to be minimized. Secondly, an evolutionary method like genetic algorithms [Goldberg, 1989] which optimizes this criteria will be presented.

### 2.1 Three dimensional reconstruction

The problem involves a set of  $M$  cameras viewing  $P$  points. We want to know the best camera network in order to reduce the uncertainty of  $P$ . In this way, each camera produces a perspective projection through the well known pin-hole model. Consequently, we can reconstruct the 3D point in the least square sens. We assume here that cameras have no distortion and can be identified with pin-hole cameras which can be modeled as  $M : \mathbb{P}^3 \rightarrow \mathbb{P}^2$ .

$$p_{ij} = M_i P_j \quad (1)$$

This mapping is represented by a matrix  $M_{3 \times 4}$

$$M = K R T \quad (2)$$

where  $K$  is the matrix of intrinsic parameters,  $R$  is a rotation matrix and  $T$  a translation matrix. Equation (1) could be rewritten as

$$AP = b \quad (3)$$

where  $A$  is a  $2i \times 3$  matrix and  $b$  is a  $2i \times 1$  vector. If matrix  $A^t A$  is invertible we can find the least squares solution

$$P = (A^t A)^{-1} A^t b \quad (4)$$

which minimizes  $\|AP - b\|$ .

### 2.2 3D error estimation

Until now, we have studied the function to transform a point in space into an image point

$$P = f(p) \quad (5)$$

given by equation (4) which is going to be useful to develop an analysis of error propagation [Faugeras, 1993, chapter V]. In this way, the key to manipulate geometric uncertainty is to be able to transform the information or probability density function on a feature available in one form (image point) into another form of interest (point in space). This transformation of information could be

grouped into a family of transformations which we approximate to the exact transformation by a first-order relation using Taylor series. Successive moments can be found by equating higher order terms; however, using higher order terms is neither viable nor desirable, as any computational simplicity will be lost. In this way, a linear approximation is to be used in which we assume a Gaussian distribution. Then the mean  $E[P]$  and covariance  $\Lambda P$  are sufficient information to define completely the feature density function. All this is given to us by the following proposition (see [Csurka, 1996]):

**Proposition 1** *Given a random variable  $p \in \mathbb{R}^m$ , of Gaussian distribution, mean  $E[p]$ , and covariance  $\Lambda p$ , and  $P \in \mathbb{R}^n$ , the random vector given by  $P = f(p)$ , where  $f$  is a function of class  $C^1$ , the mean of  $P$  could be approximated to a first-order Taylor expansion by  $f(E[p])$  and its covariance by:*

$$\Lambda P = \frac{\partial f(E[p])}{\partial p} \Lambda p \frac{\partial f(E[p])^t}{\partial p} \quad (6)$$

therefore,  $\Lambda P$  is a symmetric positive definite matrix, which describes the bounds on  $P = f(p)$  in the vicinity of  $E[P] = f(E[p])$  given those of  $p$  in the vicinity of  $E[p]$ . In this way, this proposition is letting us compute the uncertainty of the three-dimensional point knowing the uncertainty in the image points. This is taking us into another direction to give an interpretation of matrix  $\Lambda p$  which describes bounds on the possible values of the coordinates of  $p$ . This carries us to the necessity of an image error model.

### 2.3 Image error estimation

In order to compute  $\Lambda p$ , the covariance matrix of our 2D measured points, we will use proposition 1. Therefore, we must have a function to relate some points of interest (fiducials or targets) taken from several photographs over different angles. This relation will be established through the cross-ratio [Morin, 1993]:

$$f(k) = k(x_1, x_2, x_3, x_4) = \frac{(x_3 - x_1)(x_4 - x_2)}{(x_4 - x_1)(x_3 - x_2)} \quad (7)$$

which is a projective invariant. In this way

$$\Lambda k = \partial f(k) \Lambda p \partial f(k)^t \quad (8)$$

computes the uncertainty of the cross-ratio with respect to the uncertainty of the image points. It could be rewritten like

$$\sigma_p = \sigma_k \frac{\lambda}{\|J_k\|^2} \quad (9)$$

where  $\sigma_k$  represents the cross-ratio standard deviation,  $\lambda$  is the average distance between targets, and  $J_k$  is the jacobian of  $k$ . Now, following equation (9) we can compute the uncertainty in the image points  $\sigma_p$  having the

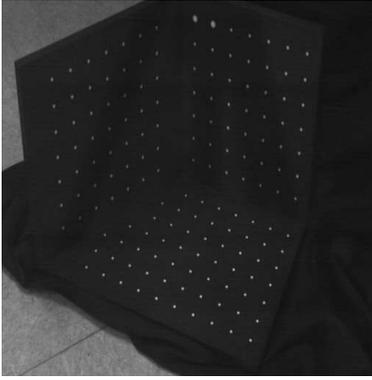


Figure 2: *The calibration grid.*

uncertainty of the cross-ratio  $\sigma_k$ . Thus, assuming that the errors over the images are Gaussian, also considering the uncertainty of the points as identical, and choosing four well-equally distant points to calculate the cross-ratio. We can carry out an experimentation using a target detector [Brand, 1995] over the calibration grid<sup>2</sup> of figure (2). From this we can compute a model of  $\Lambda p$ , useful to the previous section, in order to compute the covariance matrix  $\Lambda P$  of the 3D points.

## 2.4 The criteria

Once we have computed the covariance matrix  $\Lambda P$ , it is necessary to choose a criteria useful to the optimization process. In this sense, we need to select a metric to compare symmetric positive definite matrices. The *comparison* of covariance matrices is interpreted as the requirement the standard deviation of function  $P = f(p)$  to be better when calculated with covariance matrix  $\Lambda P_1$  than with  $\Lambda P_2$ .

$$\Lambda P_1 \leq \Lambda P_2 \quad \sigma_f^{\Lambda P_1} \leq \sigma_f^{\Lambda P_2} \quad (10)$$

The maximum eigenvalue  $\lambda_{max}$  can also be determined from

$$\Lambda P q = \lambda q \quad (11)$$

The squareroot  $\sqrt{\lambda_{max}}$  is related to the maximum standard deviation. Another metric for comparing covariance matrices  $\Lambda P$  and  $\Lambda Q$  (an ideal matrix) can be achieved by using the sum of the squared logarithms of the eigenvalues (see Förstner [Förstner, 1995]):

$$d(\Lambda P, \Lambda Q) = \sqrt{\sum_{i=1}^3 \ln^2 \lambda_i(\Lambda P, \Lambda Q)} \quad (12)$$

However, this metric requires a criterion matrix and due to the methodology, we cannot establish an ideal matrix, because we don't know the answer in advance. In this way, the maximum eigenvalue of  $\Lambda P$  may be replaced by

<sup>2</sup>this grid is composed of retro-reflective targets.

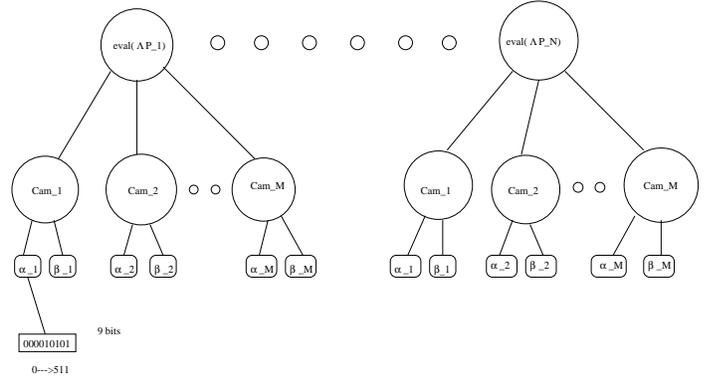


Figure 3: *The multicellular genetic algorithm is represented by a tree structure composed of a main node where the evaluation process is stored and several leaves corresponding to each camera. All cameras are codified in two parameters ( $\alpha, \beta$ ) which corresponds to the cells of an artificial being.*

a less tight norm in order to avoid the rigorous determination of the maximum eigenvalue, e.g. the trace

$$tr(\Lambda P) = \sum_{i=1}^3 \Lambda P_{ii} = \sum_{i=1}^3 \lambda_i(\Lambda P) \quad (13)$$

However, due to the fact that the maximum eigenvalue is in relation to the maximum standard deviation. Instead of the trace of  $\Lambda P$  we propose to use the *maximum element in the diagonal of  $\Lambda P$*  which corresponds to the worst variance of individual parameters

$$\max_{j=1\dots 3} \Lambda P_{jj} \quad (14)$$

This measure is easy to compute and gives a good uniformity of network precision in each coordinate axis of the error ellipsoid represented by  $\Lambda P$ .

## 2.5 The optimization process

Genetic algorithms are stochastic search techniques based on the mechanism of natural selection and natural genetics [Goldberg, 1989]. In this section, we describe a multicellular genetic algorithm (MGA) used to solve the camera position and orientation problem in order to obtain the minimal 3D error. The idea is to create a computational model to optimize a set of convergent camera networks using an evolutionary technique. This lets implement the design as a problem of *optimization*. Therefore, we propose a special implementation of the standard genetic algorithm approach following genetic programming representation [Koza, 1992]<sup>3</sup> of a tree, see figure 3, which for us is known and fixed. This representation establishes a classification of each part of the solution. Therefore, each camera evolves only with cameras of the same class. Our algorithm does not measure

<sup>3</sup>see also [Kinnear, 1994].

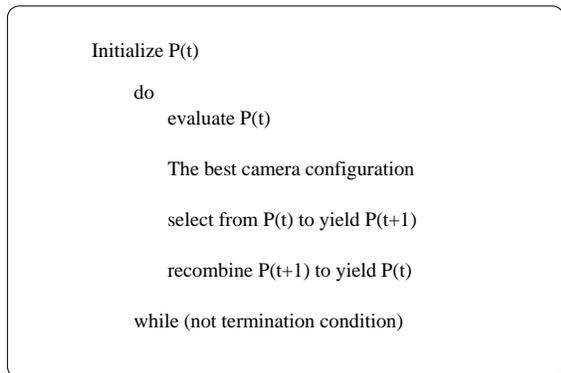


Figure 4: *The multicellular genetic algorithm.*

the fitness of a string particularly. Instead a global evaluation of a well designed criterion using derivatives is implemented. Then, we can see the algorithm as an idealized genetic algorithm IGA [Mitchell, 1996, page 132] where we know ahead of time what schemas make for good fitness.

Together with the tree representation, a model which is going to let us place cameras in orbit forming a sphere is necessary. For this, we propose to use the viewing sphere model. This model provides convergent configurations which gives improved object measurements precision compared to other networks configurations [Fraser, 1996]. In this way, cameras move in the space to achieve the design goals. The direct relationship between the spherical coordinates of the viewing sphere and the components of imaging geometry permits the heuristic search of the genetic algorithm to be exploited. Consequently, the spherical coordinates system  $(\alpha, \beta, r)$  is well suited to the representation of the search space.

Thus, the MGA is represented by a fixed tree structure composed of a main node where the evaluation process (the maximum value in the diagonal of  $\Lambda P$ ) is stored and several leaves corresponding to each camera. All cameras are codified by the two parameters  $(\alpha, \beta)$ , as  $r$  is considered as a constant here.

The multicellular genetic algorithm then proceeds as follows, see figure 4:

1. an initial random population of  $N$  convergent networks is chosen represented by  $(\alpha_n, \beta_n)$  working over a binary string representation.
2. next, we execute the process of evaluation. In this way, we store the corresponding maximum value in the diagonal of  $\Lambda P_n$  for each tree structure. This corresponds to the fitness value which says how well the network is, compared with other solutions in the population  $P(t)$ .
3. now, we pass to the process of selection implemented like a *tournament selection*[Blickle and Thiele, 1995] in which two networks are selected from  $P(t)$  and

are entered into competition selecting the best individual according to his fitness yielding the population  $P(t+1)$ .

4. from this population, we recombine the binary strings  $(\alpha_n, \beta_n)$  for each camera using the following operations:

- crossover, with a probability<sup>4</sup>  $P_c = 0.7$ , this operation was implemented like a *one-cut-point*<sup>5</sup>. Let the two parents be

$$\alpha_x = [\alpha_{x1} \ \alpha_{x2} \ \alpha_{x3} \ \alpha_{x4} \ \alpha_{x5} \ \alpha_{x6} \ \alpha_{x7} \ \alpha_{x8} \ \alpha_{x9}]$$

$$\alpha_y = [\alpha_{y1} \ \alpha_{y2} \ \alpha_{y3} \ \alpha_{y4} \ \alpha_{y5} \ \alpha_{y6} \ \alpha_{y7} \ \alpha_{y8} \ \alpha_{y9}]$$

if they are crossed after the random  $k$ th position = 4, the resulting offsprings are

$$\alpha'_x = [\alpha_{x1} \ \alpha_{x2} \ \alpha_{x3} \ \alpha_{x4} \ \alpha_{y5} \ \alpha_{y6} \ \alpha_{y7} \ \alpha_{y8} \ \alpha_{y9}]$$

$$\alpha'_y = [\alpha_{y1} \ \alpha_{y2} \ \alpha_{y3} \ \alpha_{y4} \ \alpha_{x5} \ \alpha_{x6} \ \alpha_{x7} \ \alpha_{x8} \ \alpha_{x9}]$$

- mutation; with a probability  $P_m = 0.005$ , this operation alters one or more genes. Assume that  $(\alpha_{y5} = 1)$  gene of the chromosome  $\alpha'_x$  is selected for a mutation. Since the gene is 1, it would be flipped into 0.

These operations yields a new population which we copy into  $P(t)$ .

5. Steps 2,3, and 4 are repeated until the criteria is achieved.

Finally, this algorithm minimizes the maximum value in the diagonal of  $\Lambda P$

$$\min_{i=1\dots N} (\max_{j=1\dots 3} \Lambda P_{jj}^i) \quad (15)$$

Thereby, the camera placement  $M_i$  relative to the world coordinate frame can be optimized. Geometrically, each  $\Lambda P_i$  represents an hyper-ellipsoid, which as each sensor placement  $M_i$  changes, so does the orientation and size of the hyper-ellipsoid  $\Lambda P_i$  of each point. Thus, an optimal placement solution is proposed, where the combined uncertainty of all points is minimal.

### 3 Experiments

We have run a series of experiments to test the validity of our approach. Between these, we present some results in figure (6) which shows six cases of configurations designed by EPOCA in which several cameras are looking several targets represented by their error ellipsoids aligned in one, two, or three planes. These configurations are product of our evolutionary system.

<sup>4</sup>For a discussion of the threshold values associated to  $P_c$  and  $P_m$  see [Mitchell, 1996].

<sup>5</sup>due to the classification of the MGA this operation works like a *multiple-cut-point*.

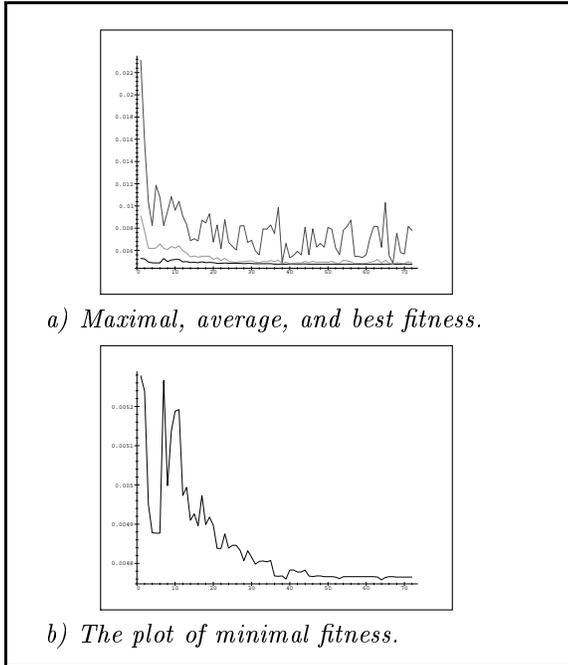


Figure 5: These two plots shows the statistics and convergence time of a single MGA run composed of 4 convergent cameras.

The graphs of figure (5) were carried out with a population of 30 cameras networks which have converged in no more than 150 generations. In general, the population size must be big enough to let the MGA find the good solutions. A small population (less than 10 networks) discourages premature convergence to suboptimal solutions<sup>6</sup> while a large population requires more evaluations per generation to achieve the same fitness values. It is important to say that the problem of convergence is usually related to the length of the string [Mitchell *et al.*, 1993]. In our implementation strings of 300 bits or bigger have been optimized thanks to the tree classification. Then, these graphs show the curves of maximal, average and minimal fitness for one run of the multicellular genetic algorithm. These results correspond to the four cameras network of figure (6b). This experiment has taken a total of 72 generations to converge, discovering the best configuration with a fitness value = .0046 in generation 64. It has taken about 7 seconds CPU time in a UltraSPARC 200 Mhz. Note also that evolution is terminated near generation 40.

Figure (6b) illustrates a solution with four cameras looking on a planar surface. This solution is not the standard solution used by the experts: photogrammetrists usually put the four camera at four corner of a cube whose center contains the targets to be measured. In

<sup>6</sup>This is mainly due to the necessity of niches which are landscapes from which the global or optimal solution is chosen. Consequently, if these zones are less rich, the opportunities to obtain a better design are reduced.

fact our configuration was already pointed out by Fraser [Fraser, 1982]; he noticed that this configuration is not atypical. Our experiments confirm Fraser statement, and even we claim that this configuration is slightly better than the classical one.

## 4 Discussion

In this paper we have posed the problem of camera placement with the goal to achieve accurate 3D measures in terms of an optimization design. We have divided the problem in two main parts. The first one, was dedicated to the analytical part for which a criterion was derived. The criterion we have chosen was the maximum element in the diagonal of the covariance matrix which is based in the error propagation phenomenon. This criterion was chosen in order to speed up the MGA process.

The second one, corresponds to the heuristical part which minimize this criterion. Our idea was to create an artificial system modelling the uncertainty of the reconstruction process. We can see the system like an information processing system, rich in autonomy and creativity, due to the evolutionary technique we have used. In this way, the system is able to evolve its own structure autonomously. This implementation is an example of a system which realizes the dream of artificial intelligence community “How can we program computers specifying what must be done and not how to do it?” [Koza, 1992]. This is an adaptive complex system that is similar to living systems. The system is a dynamic process where the global state *emerges* by coevolution through the behavior and interaction of the components (cells) which are related by coadaptation. It is necessary to say that the behavior of these cells is influenced by the global state. In this way, an extrinsic parallelism together with the intrinsic parallelism is executed. A phenomenon of *niche* is present in each camera representation. These niches evolves with respect to the others mainly due to the rules of the system.

The system we have implemented EPOCA is succesful in producing two and three camera networks designs similar to photogrammetrists. In the case of four cameras a non-standard design was proposed like one which gives slightly better results compared to the more classical networks. Moreover, the system can design networks for several adjoin and opposite planes. All the configurations are good in terms of camera distribution and ray inclination.

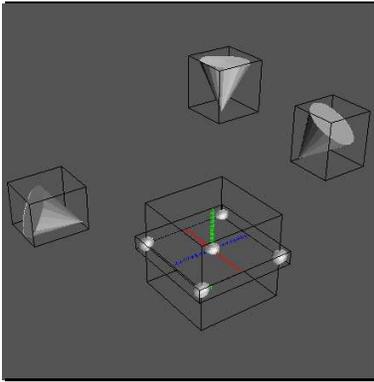
**Acknowledgments** Author Gustavo Olague is grateful for research founding to CONACYT México (grant 66291).

## References

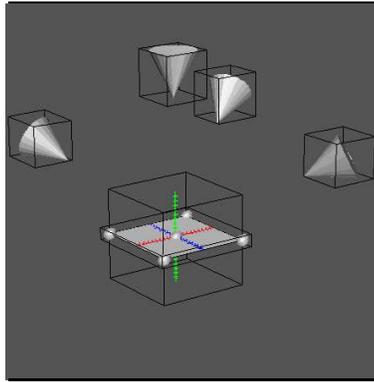
[Abrams *et al.*, ] S. Abrams, P.K. Allen, and K.A. Tarabnis. Dynamic sensor planning. unpublished.

- [Blickle and Thiele, 1995] T. Blickle and L. Thiele. A mathematical analysis of tournament selection. In *Proceedings of the Sixth International Conference on G.A.*, pages 9–16. Univ. of Pittsburgh, July 1995.
- [Brand, 1995] P. Brand. *Reconstruction tridimensionnelle d'une scène à partir d'une caméra en mouvement: de l'influence de la précision*. Thèse de doctorat, Université Claude Bernard, Lyon I, October 1995.  
ftp://ftp.imag.fr/pub/MOVI/theses/brand.ps.gz.
- [Cowan and Kovesi, 1988] C.K. Cowan and P.D. Kovesi. Automatic sensor placement from vision task requirements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):407–416, May 1988.
- [Csurka, 1996] G. Csurka. *Modelisation projective des objets tridimensionnels en vision par ordinateur*. Thèse de doctorat, Université de Nice – Sophia Antipolis, April 1996.
- [Faugeras, 1993] O. Faugeras. *Three-Dimensional Computer Vision - A Geometric Viewpoint*. Artificial intelligence. The MIT Press, Cambridge, MA, USA, Cambridge, MA, 1993.
- [Förstner, 1995] W. Förstner. A metric for comparing symmetric positive definite matrices. unpublished, August 1995.
- [Fraser, 1982] C.S. Fraser. Optimization of precision in close-range photogrammetry. *Photogrammetric Engineering and Remote Sensing*, 48(4):561–570, April 1982.
- [Fraser, 1992] C.S. Fraser. Photogrammetric measurement to one part in a million. *Photogrammetric Engineering and Remote Sensing*, 58(3):305–310, March 1992.
- [Fraser, 1996] C.S. Fraser. Network design. In K.B. Atkinson, editor, *Close Range Photogrammetry and Machine Vision*, chapter 9, pages 256–281. Whittles Publishing, Roseleigh House, Latheronwheel, Caithness, KW5 6DW, Scotland, UK, 1996.
- [Goldberg, 1989] D.E. Goldberg. *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley, 1989.
- [Kinnear, 1994] K.E. Kinnear. *Advances in Genetic Programming*. The MIT Press, Cambridge, MA, USA, 1994.
- [Koza, 1992] J.R. Koza. *Genetic Programming, On the Programming of Computers by Means of Natural Selection*. The MIT Press, Cambridge, MA, USA, 1992.
- [Mason and Gruen, 1995] S.O. Mason and A. Gruen. Automatic sensor placement for accurate dimensional inspection. *Computer Vision and Image Understanding*, 3(61):454–467, 1995.
- [Mason, 1994] S.O. Mason. *Expert System-Based Design of Photogrammetric Networks*. PhD thesis, Institute for Geodesy and Photogrammetry, Swiss Federal Institute of Technology (ETH) CH-8093 Zurich, May 1994.
- [Mitchell et al., 1993] M. Mitchell, S. Forrest, and J.H. Holland. The royal road for genetic algorithms: Fitness landscapes and ga performance. In F.J. Valera and P. Bourguine, editors, *Proceedings of the First European Conference on Artificial Life*, pages 245–254. MIT Press/Bradford Books, 1993.
- [Mitchell, 1996] M. Mitchell. *An Introduction to Genetic Algorithms*. The MIT Press, Cambridge, MA, USA, second printing edition, 1996.
- [Morin, 1993] L. Morin. *Quelques contributions des invariants projectifs à la vision par ordinateur*. Thèse de doctorat, Institut National Polytechnique de Grenoble, January 1993.
- [Sakane and Sato, 1991] S. Sakane and T. Sato. Automatic planning of light source and camera placement for an active photometric stereo system. In *Proceedings of IEEE International Conference on Robotics and Automation, Sacramento, California, USA*, pages 1080–1087. IEEE Robotics and Automation Society, April 1991.
- [Tarabanis et al., 1991] K. Tarabanis, R.Y. Tsai, and P.K. Allen. Automated sensor planning for robotic vision tasks. In *Proceedings of IEEE International Conference on Robotics and Automation, Sacramento, California, USA*, pages 76–82, April 1991.
- [Tarbox and Gottschlich, 1993] G.H. Tarbox and S.N. Gottschlich. Planning for complete sensor coverage in inspection. Technical Report TR-CAT-93-4, NYS Center for Advanced Technology in Automation and Robotics., 1993.
- [Yi et al., 1995] S. Yi, R.M. Haralick, and L.G. Shapiro. Optimal sensor and light source positioning for machine vision. *Computer Vision and Image Understanding*, 61(1):122–137, 1995.

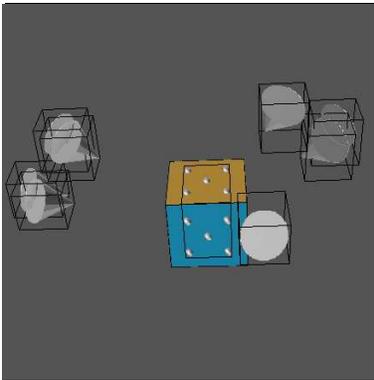
**EPOCA network designs.**



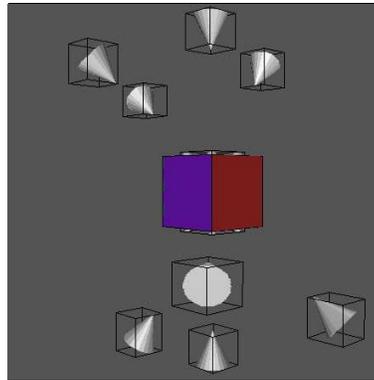
*a) 3 cameras observing a plane.*



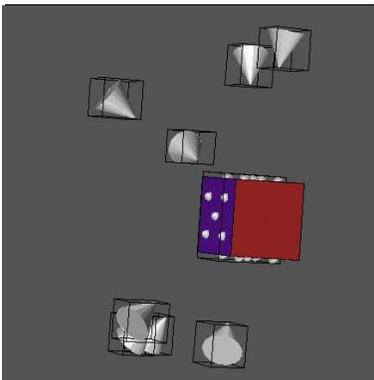
*b) 4 cameras over a surface plane.*



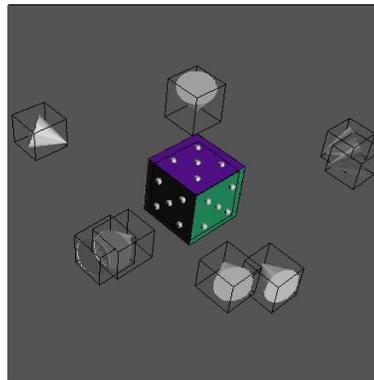
*c) 8 cameras over 2 adjoin planes.*



*d) 8 cameras observing opposite sides.*



*e) 8 cameras observing 3 opposite faces.*



*f) 8 cameras observing 3 adjoin faces.*

**Examples of configurations designed by EPOCA.**

Figure 6: *Between the several designs proposed by EPOCA we have identified b) like one used by Fraser [Fraser, 1982] which is not atypical of an imaging geometry.*