Pareto Optimal Sensing Strategies for an Active Vision System

Enrique Dunn   Gustavo Olague
CICESE Research Center
EvoVisión Laboratory
22860 Ensenada, México.
Email:{edunn,olague}@cicese.mx

Evelyne Lutton
INRIA - COMPLEX Team
Domaine de Voluceau BP 105
78153 Le Chesnay Cedex - France
Email:Evelyne.Lutton@inria.fr

Marc Schoenauer
Equipe TAO - INRIA Futurs
LRI Bat 490 Université de Paris Sud
91405 Orsay Cedex - France
Email:Marc.Schoenauer@inria.fr

Abstract— We present a multi-objective methodology, based on evolutionary computation, for solving the sensor planning problem for an active vision system. The application of different representation schemes, that allow to consider either fixed or variable size camera networks in a single evolutionary process, is studied. Furthermore, a novel representation of the recombination and mutation operators is brought forth. The developed methodology is incorporated into a 3D simulation environment and experimental results shown. Results validate the flexibility and effectiveness of our approach and offer new research alternatives in the field of sensor planning.

I. INTRODUCTION

Active vision studies artificial perception scenarios where the sensor is a controllable element within the system infrastructure. This capacity provides greater flexibility and robustness to the vision system [1]. However, the need for suitable control strategies arises in this context. The complexity and the diversity of vision tasks makes it difficult to develop a general sensor planner [2]. In fact, planning for specific tasks, such as *object search*, has been found to be NP-Hard [3]. In the field photogrammetry, it has long been acknowledged that project planning for sensing tasks is a complex problem that generally is solved by empirical means [4].

Many studies on the geometrical aspects involved in viewpoint selection can be found in the literature [5]. The incorporation of a physical mechanism that controls the positioning of the sensor augments the difficulty of the planning [6]. The resulting problem is a complex relationship between vision task goals and restrictions, environmental constraints, infrastructure characteristics and overall system performance requirements. It is difficult to mathematically model and deal with these aspects. A common practice is to develop a suitable parameterization of the problem and state it in optimization terms. Nevertheless, the mathematical modeling of different qualitative measures involved in the vision task can vary greatly among authors. Some recent works address the problem using the evolutionary computation approach [7],[8],[9]. However, the fact that planning for such a system is essentially a multi-objective (MO) task has been ignored. To solve the problem, some researchers have used an aggregate function approach, where different task objectives are combined into a single criterion. In this way, the MO problem is transformed into a single objective one. Another option is the use of a

decoupled approach [10], where different sequential stages of the problem are identified and the solution of one stage is the input to the subsequent ones. This can lead to bias toward the objectives considered at earlier stages.

This work introduces multi-objective methodologies to the sensor planning problem in active vision systems. By addressing the conflicting objectives inherent to active sensing under the MO framework, a novel approach to sensor planning is presented [11]. The use of evolutionary computation techniques allows for flexibility in the problem representation as well as robustness against the complex numerical and combinatorial problems involved in our planning.

This document is organized as follows. First, the characterization of an optimal sensing strategy is described and the mathematical models used in our optimization are presented. Then, we detail the evolutionary computation approach to our planning. In this respect, a special genome representation is described and a novel encapsulation of the reproduction genetic operators is introduced. Afterward, experimental results illustrate the performance of our sensor planner. Discussion and conclusions are presented to end the paper.

II. AN OPTIMAL SENSING STRATEGY

A valid sensing strategy is one that fulfills all the task defined goals and complies with all the related constraints. However, if one aims at obtaining an optimal strategy, then a criterion for discriminating among different solutions is needed. When several criteria are to be considered and they are in conflict with each other, then we have a MO problem. In such a case there is no single optimal solution, instead we have a set of optimal solutions that represent the different trade-offs between the objectives. In this context, the concept of optimality is based on the Pareto dominance relationship among solutions. A solution A dominates a solution B if for each of the considered objectives solution A is not "worse" than B and there exists at least one strict inequality. Hence, an optimal solution is one which is not dominated by any other. The set of all non-dominated solutions form the Pareto optimal set and their corresponding objective values form the Pareto Front in the function space of a MO problem.

In our active vision system, see Figure 1, the objectives to optimize are the reconstruction accuracy and the required
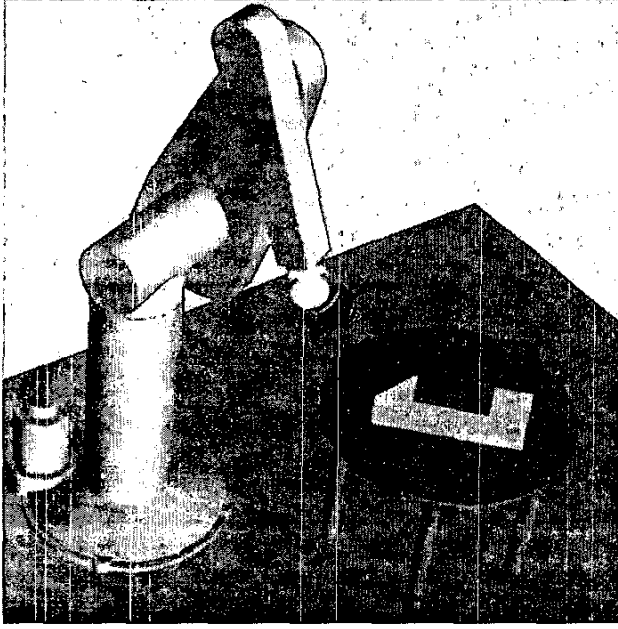
Fig. 1. An active vision system. A manipulator robot equipped with a digital camera on it's end effector has the goal of measuring the object on the table. The MO problem is to obtain optimal reconstruction while minimizing motion.

manipulator motion. We consider these two objective to be in conflict due to the complexity of the camera placement problem, in which larger spatially distributed networks with sufficient redundancy are preferred. In the following subsections the modeling of each of our objectives as well as some preliminary discussion on their interaction is presented.

A. Accurate Reconstruction

In order to estimate the 3D reconstruction error as a function of the disposition of multiple cameras, we will use an approach based on the error propagation phenomena as presented in [12]. Under the pinhole camera model, each 3D point $P_i = (X_i, Y_i, Z_i)^t$ is projected into an image point $p_{ij} = (u_{ij}, v_{ij})^t$ through a projection matrix M_j. This relationship is expressed by the collinearity equations

$$
u_{ij} = \frac{m_{11}^j X_i + m_{12}^j Y_i + m_{13}^j Z_i + m_{14}^j}{m_{31}^j X_i + m_{32}^j Y_i + m_{33}^j Z_i + m_{34}^j}
$$
$$
v_{ij} = \frac{m_{21}^j X_i + m_{22}^j Y_i + m_{23}^j Z_i + m_{24}^j}{m_{31}^j X_i + m_{32}^j Y_i + m_{33}^j Z_i + m_{34}^j}
$$
(1)

Optical triangulation can be obtained solving a least squares system obtained from the projections of a 3D point in several images. In this way, given knowledge of each projection matrix M_j, we have a model of the form $P = f(p)$. We consider the image measurements p as a random variable of Gaussian distribution, mean $E[p]$ and covariance Λp. Given the random vector $P = f(p)$, where f is a function of class C^1, the mean of P can be approximated to a first-order Taylor expansion by

$f(E[p])$ and its covariance by

$$
\Lambda P = \frac{\partial f(E[p])}{\partial p} \Lambda p \frac{\partial f(E[p])^t}{\partial p}
$$
(2)

We consider the image uncertainty Λp as a function of the incidence angle between the viewing direction of a camera and the normal vector of a point's surface. Once such analytical expression is obtained, expression 2 can be evaluated, see [12] for details. The selected criteria, which characterizes the uncertainty of the 3D reconstruction, is the maximum element in the diagonal of the 3×3 covariance matrix ΛP,

$$
f_1(\vec{x}) = \max_{i=1...3} \Lambda P_{ii}.
$$
(3)

B. Efficient Motion

Camera placement, is determined by the motion of the manipulator. In turn, manipulator motion is controlled by its joint values. In order to estimate the cost of camera displacement during task execution, we propose to add the distance traveled by the manipulator for each consecutive pair of images. In this work, obstacle avoidance and robot dynamics are not considered. Hence, the robot is capable of moving freely among any two kinematically attainable configurations. A direct approach consists in calculating the Euclidean distance between each sensing position

$$
D(V_i, V_j) = \|V_i - V_j\|^2 \quad where \quad V_i, V_j \in \mathbf{R}^3.
$$

In this way, given a set of viewpoints, we have a 3D instance of the traveling salesman problem. Another way to evaluate displacement is to measure the difference in the joint values of the manipulator among two subsequent sensing actions. For a manipulator with r joints the inverse kinematic problem is solved by a function $Q(V_i) = \mathbf{q}_i$ where $\mathbf{q}_i \in \mathbf{R}^r$. Therefore, the Euclidean distance must be calculated in an r-dimensional space. However, depending on the robot configuration, not all joints may contribute in the same manner to the motion of the robot. This is the case for many revolute joint manipulators as the one considered in this work. Hence, a weighting value, w_i, is associated to each of the joints and incorporated into the distance measure. These values are represented by a weight vector $\mathbf{w} \in \mathbf{R}^r$, yielding

$$
D(V_i, V_j) = \|diag(\mathbf{w})[Q(V_i) - Q(V_j)]\|^2,
$$
(4)

where $diag(\mathbf{w})$ is a $r \times r$ diagonal matrix with $diag(\mathbf{w})_{kk} = w_k$. Note that the values contained in $\mathbf{w}$ may include other considerations on the behavior of the manipulator, such as power consumption or manipulator wear.

Once a distance function $D(V_i, V_j)$ is defined, we can express the total motion cost for a robot tour consisting of n viewpoints as

$$
f_2(\vec{x}) = \sum_{i=1}^{n-1} D(V_i, V_{i+1}).
$$
(5)

458

III. EVOLUTIONARY BASED OPTIMIZATION

A. Problem Variables and Representation

The parameterization of our sensing strategies will be in terms of the camera placement. In this way, the goal is to find an ordered set of viewing positions which determines robot motion. A viewing sphere model is adopted in order to reduce the search space. Hence, camera position is specified using polar coordinates $[\alpha, \beta]$. In this way, for a network of n camera positions, the real-coded genotype utilized is given by

$$\vec{x} \in \mathbf{R}^{2n} \ where \ \alpha_i = x_{2i-1}, \beta_i = x_{2i} \ for \ i = 1,\ldots,n. \quad (6)$$

Since the proper size of the camera network for a given object is not easy to be determined, we adopt a representation that allows for different size networks to be considered on the same evolutionary process. To achieve this, the principals presented in the *Structured Genetic Algorithm* proposed in [13] are incorporated into our approach. Hence, an additional binary section of the form

$$\vec{x}^b \in \mathbf{B}^n \quad where \quad x_i^b \in [0, 1] \quad for \quad i = 1,\ldots,n, \quad (7)$$

is added to the genotype. The value of each bit x_i^b determines the inclusion of a camera position into a network specification. Thus, our extended genotype is of the form $X = \langle \vec{x}^b, \vec{x} \rangle$.

Under such representation, each camera C_i is described by a set of values $[x_i^b, x_{2i-1}, x_{2i}]$. Moreover, the genotype represents a set of homogeneous *elements*, each codified by multiple *parameters*, which together form a single solution to the problem. This approach is based on the *Multicellular GA* proposed in [5]. Internally we can represent our genotype as a tree based structure with the following characteristics:

1) The root node stores the evaluation function values.
2) Next, there are n nodes which describe each element of the solution, i.e. viewpoint specification.
3) Finally, each node generates leaf nodes where the parameterization values are stored, i.e. polar coordinates.

This structure, see Figure 2, allows the use of recombination operators either at the *element* or *parameter* level. The inclusion of a "control" gene x_i^b in the camera parameterization permits the use of a fixed size tree structure. Thus, the need to control the increase in tree size, common in genetic programming approaches, is avoided.

B. Genotype to Phenotype Transformation

The phenotypic interpretation of our composite genotype consists in a sequence of viewpoints to be visited by the manipulator. Moreover, this sequence is a permutation from an unordered subset of the complete camera placement search space. The genotype decoding provides the unordered subset and the corresponding permutation is obtained by a deterministic procedure. Moreover, Figure 2 illustrates the process of genotype interpretation. For relatively small networks, of less than 15 cameras, an exhaustive search is carried out. Larger networks require the use of approximate algorithms. In this work a greedy heuristic is utilized in order to reduce the computational cost.
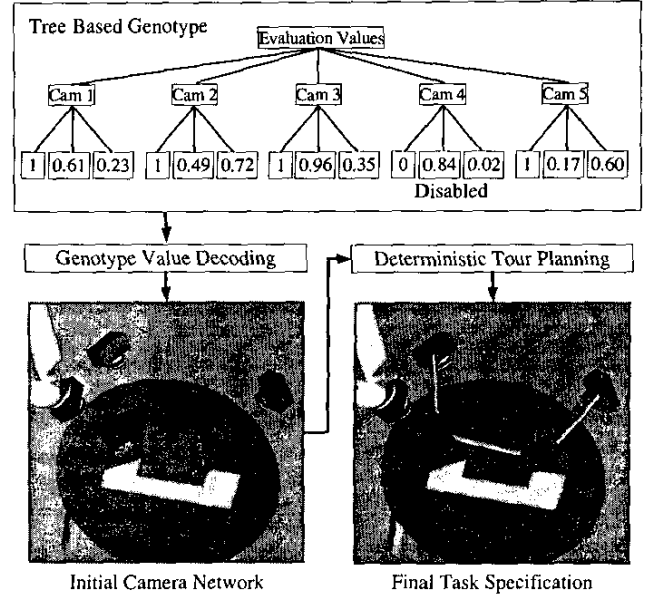

Fig. 2. Genotype to Phenotype Transformation. A genotype representation that allows for up to 5 cameras is depicted. Only 4 are expressed into the decodified camera network robot's tour, from wich a robot tour is determined.

Constraint satisfaction of a phenotype is another aspect of interest. Camera placement constraints in the form of the scene visibility and kinematic limitations of the manipulator are incorporated by a deterministic repair mechanism. On the other hand, global task constraints, such as reconstruction completeness, are handled using objective function penalization. These policies are enforced during the fitness evaluation, not requiring any modification to the underlying evolutionary algorithm.

C. Recombination and Mutation

In this work a *parameter* level recombination is utilized. Therefore, each parameter value of a viewpoint specification is recombined only with parameters of the same class belonging to another individual. The real-coded camera placement variables, encoded in $\vec{x}$, are recombined using SBX crossover [14]. Mutation is applied to each of the variables using a polynomial distribution perturbation. For the binary control variables, encoded in $\vec{x}^b$, uniform crossover is utilized along with bit-wise mutation.

A novel representation proposed in [15],[16] is used for the real-coded evolutionary operators. This consists in encapsulating both crossover and mutation into a single algebraic affine transformation. Since two real-coded variables Y_1 and Y_2 represent a point in the affine plane, an affine transformation of the form

$$Y_1' = b_{11}Y_1 + b_{12}Y_2 + C_1$$
$$Y_2' = b_{21}Y_1 + b_{22}Y_2 + C_2$$

is applied, where the coefficients are arbitrary real numbers, subject to $|b_{rs}| \neq 0$. This transformation can be extended to include the n variables contained in two different solutions.

459

Accordingly, the generation of new solutions within the evolutionary algorithm can be stated as follows:

$$\begin{bmatrix} \underbrace{\begin{matrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{matrix}}_{Crossover} & \underbrace{\begin{matrix} C_1 \\ C_2 \end{matrix}}_{Mutation} \end{bmatrix}_n \begin{pmatrix} Y_{1_1} & Y_{1_2} & \cdots & Y_{1_n} \\ Y_{2_1} & Y_{2_2} & \cdots & Y_{2_n} \\ 1 & 1 & \cdots & 1 \end{pmatrix} = \begin{pmatrix} Y'_{1_1} & Y'_{1_2} & \cdots & Y'_{1_n} \\ Y'_{2_1} & Y'_{2_2} & \cdots & Y'_{2_n} \end{pmatrix}$$

The advantages of this encapsulation are:

1) Standardized treatment of all transformations
2) Complex transformations are composed from simple transformations by means of matrix multiplication.
3) Simple inversion of the transformation by matrix inversion.
4) Extremely fast, hardware supported matrix operations in high-power graphic workstations

D. Evolutionary Process

The evolution of individuals is controlled by an algorithm based on the NSGA-II presented in [17]. The population is driven toward the Pareto Front making use of multi-objective dominance relations among solutions. Selective pressure is supplied by ranking the solutions into different sets according to Pareto dominance. Diversity preservation is then achieved using an adaptive crowding penalization within each of the ranks. Furthermore, generational elitism is enforced to the combined population of parents and offspring according to the rank of the solutions.

IV. EXPERIMENTATION

Several experiments were carried out for the case of a three-dimensional object under observation by a manipulator robot. For all the experiments a population size of 150 individuals was used. Crossover and mutation probabilities were set to 0.99 and 0.03, respectively. Furthermore, the parameters which control the disruptiveness of the SBX crossover and mutation operators were set to $\eta_x = 2.0$ and $\eta_m = 1.0$ respectively. These values were obtained empirically after extensive trials. Different experimental scenarios are presented in which different combinations of genome representations and objectives are studied.

A. Fixed Size Camera Networks

The simplest planning scenario considered in this work is the one in which the number of sensing actions is given as an input to the optimization task. To solve such a problem we use a genome representation which only includes the real-coded part, i.e. the binary "control" alleles are not considered. For our first experiment a network of eight cameras is studied. The results after an evolutionary process of 150 generations are illustrated on Figure 3. Configurations a) and d) represent solutions on opposite extremes of the obtained Pareto Front. The differences on the geometrical disposition of viewpoints exemplifies the conflicting nature between reconstruction accuracy and motion efficiency. To obtain optimal accuracy a good spatial distribution which favors precise triangulation is

needed. On the other hand, a closely located set of viewpoints will not yield satisfactory results in terms of accuracy. Note that for our optimal solutions, there is a lower bound to the amount of displacement necessary for complete reconstruction, as well as a lower bound on the 3D error obtained. This can be attributed to the complexity of the sensor planning problem for a 3D object. Solutions b) and c) represent different trade-off to the planning problem. The proximity of viewpoints on configuration c) and d) makes it difficult to distinguish visually among viewpoint positions, however all the depicted configurations consist of eight camera positions. The importance of these results is that when a single objective optimization or a decoupled approach is adopted, the variety of alternative solutions may not be attainable or simply overlooked.

The following experiment addresses the behavior of the results for camera networks of different size. The algorithm was executed several times, modifying the size of the desired camera network. Each instance was executed for a period of 250 generations. The obtained non-dominated sets are presented together on Figure 4. The plotted information concentrates on a small region of the function space, where the difference among different non-dominated sets is easier to visualize. Some important characteristics can be observed in the different Pareto Fronts. Larger networks in general provide a better precision and find larger Pareto Fronts. Nevertheless, the non-dominated set obtained from considering all the solutions on Figure 4 is formed by networks of different size. The magnitude of the improvement on precision between subsequent Pareto fronts decreases according to the number of cameras. On the lower right corner of the region of interest the discrepancies on the vertical axis (i.e. motion efficiency) are insignificant for networks of very different size.

B. Variable Size Camera Networks

As mentioned earlier, the choice of the number of viewpoints needed for 3D reconstruction is not evident. For this reason a representation that allows for networks of variable size was developed, see Figure 2. In this experiment a codification that includes up to 20 cameras was utilized. The algorithm was run for 400 generations. This increase in the number of generations is due to the higher dimension of the search space caused by the use of the additional binary variables. Population size and other parameters were the same of previous experiments. Figure 5 illustrates the obtained non-dominated set formed by networks of different size. As expected, larger networks provide greater accuracy and are gathered toward the upper right portion of the illustrated non-dominated set. Accordingly, the most efficient networks in terms of manipulator motion are the ones with less number of cameras. One advantage of using the variable length representation is the attainment of greater diversity in the imaging geometry configurations (i.e. phenotypic diversity) found in the Pareto Front. This can be attributed to the concurrent search at different regions of the augmented search space.

Another aspect of interest is the convergence to the real Pareto Front. Since in our problem there is no prior knowledge
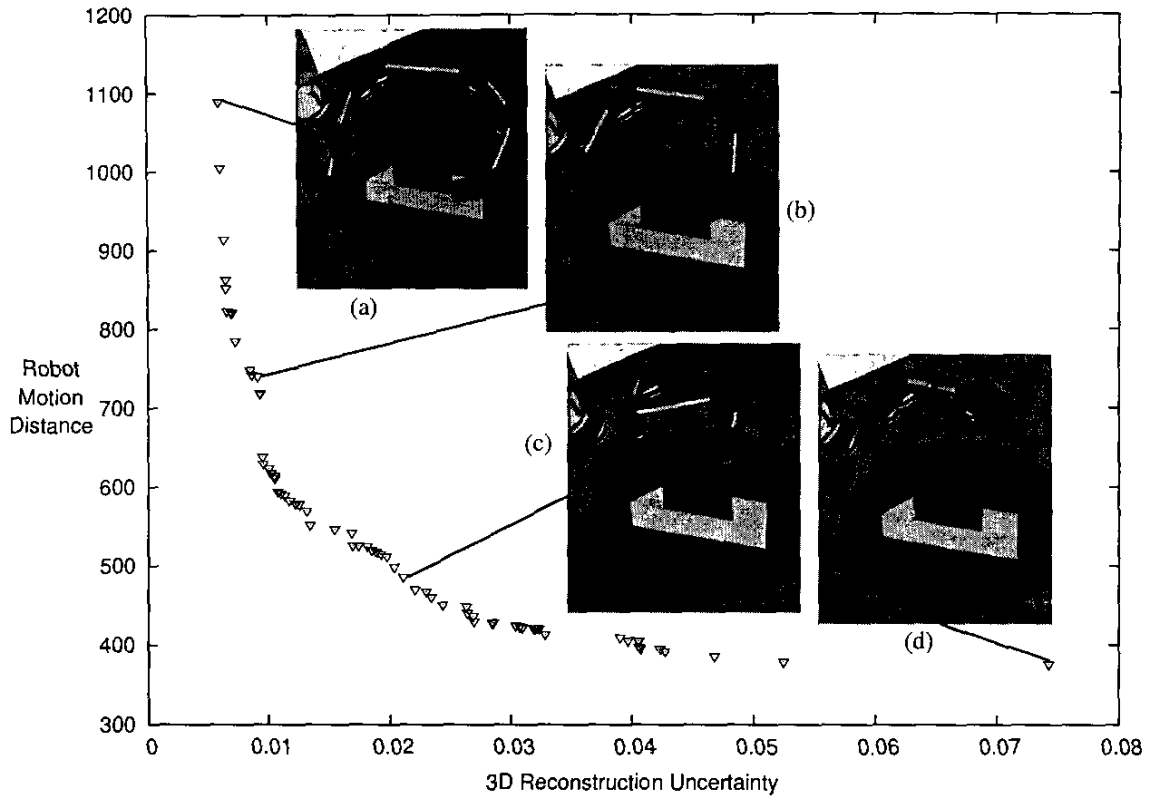
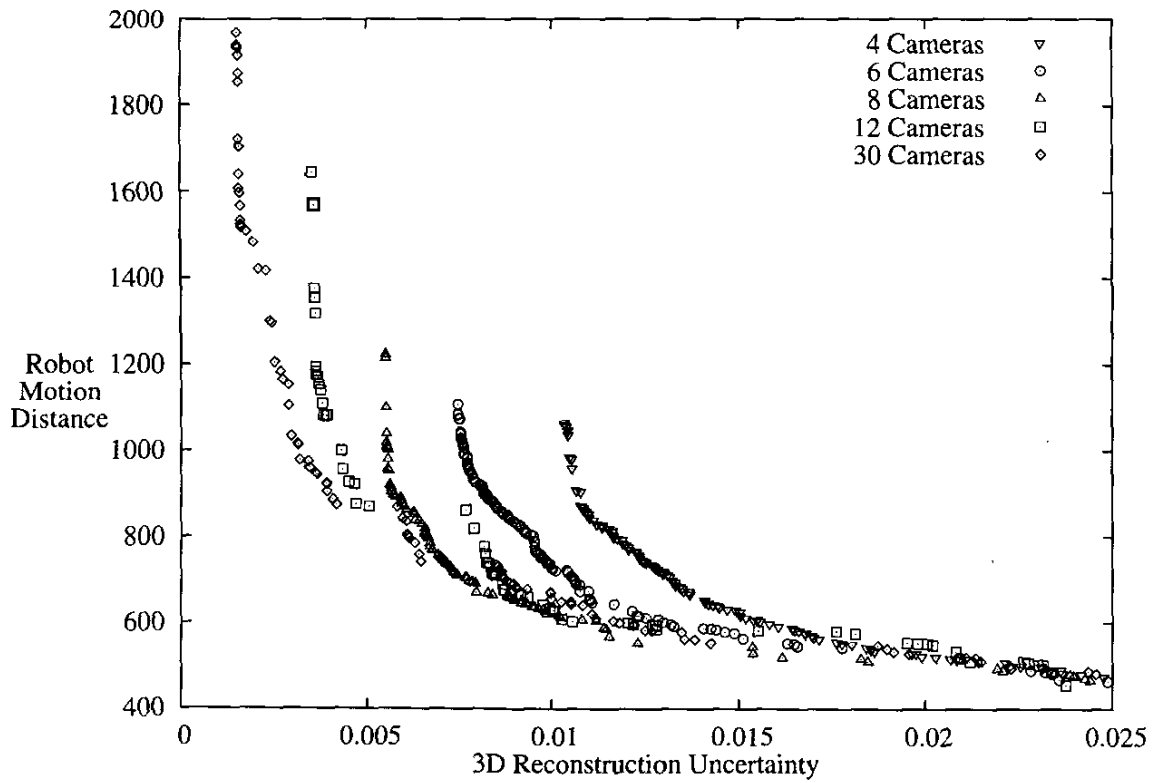Fig. 3. Example of a Pareto Front for the planning of an eight camera network.

Fig. 4. Pareto Fronts obtained for fixed size camera networks. Different algorithm instances were executed varying the size of the genome representation.
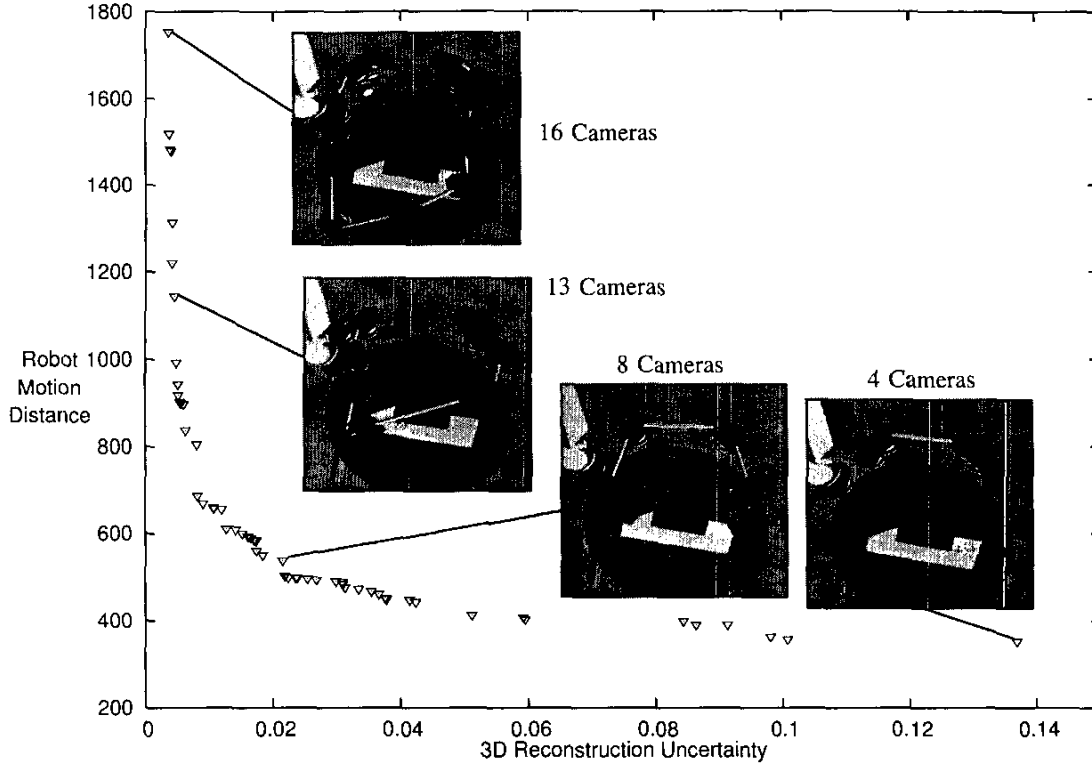
Fig. 5. Pareto Fronts obtained for variable size camera networks.

of the Pareto front, it is difficult to give an appreciation on the convergence properties of our algorithm as well as to compare the results of different instances. In this respect several metrics have been proposed for evaluating the worthiness of a set of solutions against another, see [18]. We utilize a measure similar to the S metric proposed in [19]. The hyper-volume enclosed by the Pareto front and a fixed point in the function space is calculated. This computation is simplified in the case of two objectives. We calculate the area of each triangle formed by two adjacent elements of the non-dominated set and the fixed reference point, see Figure 6. The function values are normalized in such a way that the maximal total area is 1. A performance comparison of the variable size encoding against the fixed size representation was made. 200 generations were run for different instances and the metric was evaluated at each iteration. A fixed size representation of 8 cameras was compared to a variable representation that allows for a maximum of 16 cameras. The result of such an experiment is illustrated in Figure 7. Similar result are obtained for both instances with a slight advantage for the fixed size encoding. Taking into account the difference in the size of the search spaces, such results are encouraging.

V. CONCLUSIONS

A MO approach to planning sensing actions of an active vision system has been presented. The aspects considered in this work were reconstruction accuracy and manipulator
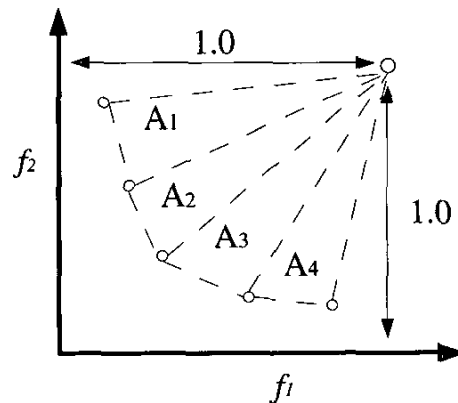

Fig. 6. Metric used to evaluate the convergence of a non-dominated set of solutions.

motion. The criteria for accuracy is based on the image error propagation, while motion efficiency is addressed as a traveling salesman instance. A representation that allows for sensing scenarios of different complexity to be considered during the same evolution process was proposed along with an encapsulation of the crossover and mutation operators into a single transformation. The studied case presents a convex Pareto front with bounded asymptotic behavior on both axes of the objective function plane. Different experimental sce-
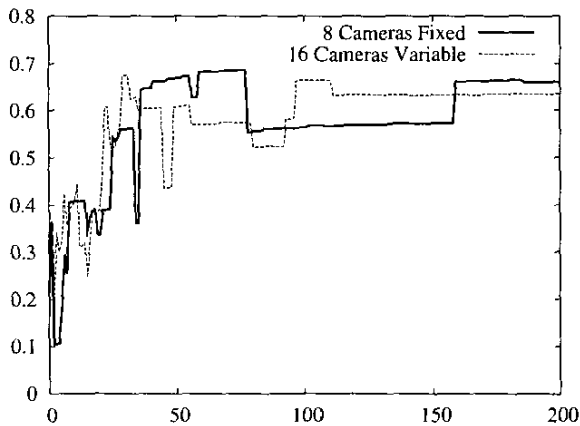
462

Fig. 7. Performance evaluation of fixed vs. variable size representation.

narios were presented and comparison made among different encoding schemes.

Experimental results indicate better performance for an encoding of fixed size networks. However, the determination of the appropriate size must be made empirically. It is in this respect that the variable size representation offers valuable insight into the understanding of the problem. Hence, a viable approach may consist in exploration of the MO problem landscape using variable length representation and in the use of fixed size representation to improve results. Moreover, there are still many opportunities for improvement in the evolutionary process. The incorporation of different recombination and mutation schemes can be studied. The variable encoding can be modified to make a more efficient search. Currently, a NSGA-II based evolutionary engine is utilized but other methodologies could be adapted. Another advantage of using a variable length representation is the possibility of including the size of the network as an additional objective to minimize. This is a viable alternative when considering the high computational cost of rigorous photogrammetric bundle adjustments. These conceptual aspects, along with the implementation of our planner on a real world environment, are to be addressed in future works.

ACKNOWLEDGMENT

REFERENCES

[1] Bajcsy R., 1986. "Passive Perception vs. Active Perception". Proc. IEEE Workshop on Computer Vision.

[2] Tarabanis K.A., Allen P.K., Tsai R.Y. 1995. "A Survey of Sensor Planning in Computer Vision". IEEE Trans. on Robotics and Automation. 11(1):86-104 p.

[3] Ye Y., Tsotsos J. 1999. "Sensor Planning for 3D Object Search". Computer Vision and Image Understanding. 73(2):145-168p.

[4] Fraser C.S., 1996. Network Design. "Close-range Photogrammetry and Machine Vision"(K.B. Atkinson, editor). Whittles Publishing, Scotland, UK. 371 pp.

[5] Olague G. 2002. "Automated Photogrammetric Network Design Using Genetic Algorithms". Photogrammetric Engineering & Remote Sensing, 68(5):423-431. Awarded "2003 First Honorable Mention for the Talbert Abrams Award", by ASPRS.

[6] Triggs B., Laugier C. 1995. "Automatic camera placement for robot vision tasks". IEEE Intl. Conf. on Robotics & Automation.

[7] Sheng W. 2002. "CAD-based Robot Motion Planning for Inspection in Manufacturing". PhD. Thesis. Michigan State University.

[8] Yang C., Marefat M., Ciarallo F. 1998. "Error Analysis and Planning Accuracy for Dimensional Measurement in Active Vision Inspection". IEEE Trans. on Robotics & Automation, 14(3):476-487 p.

[9] Dunn, E., Olague, G. 2003. "Evolutionary Computation for Sensor Planning: The Task Distribution Plan." EURASIP Journal on Applied Signal Processing, Special Issue on Genetic and Evolutionary Computation for Signal Processing and Image Analysis. Volume 2003, No. 8, 1 July 2003. pp. 748-756.

[10] S.Y. Chen and Y.F. Li. 2004. "Automatic Sensor Placement for Model-Based Robot Vision". IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 34, No. 1, pp. 393-408.

[11] Dunn, E. and Olague, G. 2004. "Multi-objective Sensor Planning for Efficient and Accurate Object Reconstruction."6th European Workshop on Evolutionary Computation in Image Analysis and Signal Processing. to appear.

[12] Olague G. Mohr R. 2002. "Optimal Camera Placement for Accurate Reconstruction". Pattern Recognition, 35(4):927-944 p.

[13] Dasgupta D., McGregor R. 1993. "sGA: A Structured Genetic Algorithm". Technical Report No. IKBS-11-93, Department of Computer Science University of Strathclyde.

[14] Deb K. 2001. "Multi-Objective Optimization using Evolutionary Algorithms" John Wiley & Sons, Ltd. Baffins Lane, Chichester, West Sussex, PO19 1UD, England. 497 pages.

[15] Olague, G., Hernndez, B. and Dunn, E. (2003) Accurate L-corner Measurement using USEF Functions and Evolutionary Algorithms. 5th European Workshop on Evolutionary Computation in Image Analysis and Signal Processing. Lecture Notes in Computer Science 2611. pp. 410-421. Springer-Verlag. EvolASP2003.

[16] Olague, G., Hernndez, B. and Dunn, E. 2003. Hybrid Evolutionary Ridge Regression Approach for High-Accurate Corner Extraction. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Madison, Wisconsin, USA, June 16-23, 2003. Vol. 1, pp. 744-749.

[17] Deb K., Agrawal S., Pratab A., Meyarivan T., 2000. "A Fast Elitist Non-Dominated Sorting Algorithm for Multi Objective Optimization: NSGA-II". Proceedings of PPSN VI, Paris France. Springer. Lecture Notes in Computer Science No. 1917.

[18] Knowles J.,Corne D. 2002. "On Metrics for Comparing Nondominated Sets". Proceedings of Congress on Evolutionary Computation 2002.

[19] Zitzler E. 1999. "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications". PhD. Thesis, Swiss Federal Institute of technology (ETH), Zurich, Switzerland.