A Sensor Placement Approach for the Monitoring of Indoor Scenes

Pierre David, Vincent Idasiak, and Frédéric Kratz

Laboratoire Vision et Robotique, UPRES EA 2078, 88 Boulevard Lahitolle,
18020 Bourges, France
{pierre.david, vincent.idasiak, frederic.kratz}@ensi-bourges.fr

Abstract. Within the framework of a French project, which aims at developing a new human presence sensor, we intend to design a sensor system simulator. During the establishment of the requirements of that new sensor we raised that the mission of a global scene survey could only be performed by a collection of several systems using very diverse technologies. This article presents the development of a method for the placement of multi-technology and multi-sensor systems. The considered environments are room or set of rooms in office buildings or individual homes. We will explain how we managed to represent the use of different sensors considering their various environments. Then, the way of exploiting these models using genetic algorithms is discussed. Those models are oriented for finding system placement and therefore for helping sensor networks deployment.

Keywords: Simulator, sensor network, genetic algorithm, sensor placement.

1   Introduction

The project in which we are involved is called Capthom. It was set up to design a new low-cost human presence detector characterized by a high reliability. The development of new human presence sensors is currently needed by research projects on energy management (ERGDOM [1]) and on the medical monitoring of the elderly (GERHOME [2], PROSAFE [3], SOPRANO [4]). It is of primary importance in this type of industrial project to be able to present simulations to evaluate the future capacities of the system, both in functional and dysfunctional terms. We thus wish to create a simulation software capable of testing the various research considered for the development of the system. This kind of simulator could also be used in the deployment phase of the systems as a placement tool. The goal of such a use is to find the best way of installing a sensor network considering an established problem composed of detection objectives, detection conditions and a defined scene. During the creation of the tool and of the models the state of art led us to consider mainly camera placement works. We were therefore inspired by this way of modeling and we tried to generalize it to all types of sensors. The study of sensor placement can be very profitable in terms of money savings by limiting the number of needed systems; moreover, an intelligent placement dramatically improves the performances and also

insures a growth in the lifetime of the components. We decided to characterize a sensor system by a special attribute, which we called the efficient zone, that is to say the zone in which a considered sensor is able to catch and interpret its targeted physical flow. We must also be able to manage scenarios of multiple use as well as very different environments. Hence it was necessary to make the creation and characterization of new environments available. We thus defined a representation of the scenes of use and a way of modeling their characteristics. So we defined two models, one for the sensors and the other for their environment, meeting and merging them as a whole model representing the problem of sensor selection and placement. We will describe in the next paragraphs the type of problem that we tried to solve and model, as well as the model that we set up to describe the scene. We will insist on the model of the scene and the elements characterizing the environment. We will then detail a method of modeling the sensors before presenting our way of solving the problem. We will show in this last part the adaptability and the various parameter settings of this method.

2   Definition of Our Problem and First Step in Modeling

Our first work was carried out to optimize and evaluate the placement of the future Capthom sensors. We followed the technique developed in the problems of camera placement. These problems intervene in many fields: photogrammetry, video surveillance, camera management for interfaces with virtual worlds and simulation of cinematographic shot design. Our topic, which is sensor exploitation, seems to be closer to video surveillance. In paragraph 2.1 we will present a review of the problems treated for camera placement. However we will first briefly explain our problem. The main goal that we pursued in this job was to be able to purpose the best composition of sensor network to fulfill desired objectives in a given scene. This problem involves three fundamental notions: the scene, the sensor network and the objectives definition. The scene concerns both the physical geometry of the place and its utilization. The sensor network must be designed with a representation of their efficiency, functioning and behavior. The objectives must also be clearly defined in terms of desired monitoring, efficiency as well as reliability. Our approach should therefore be applied to the design of multi-sensor installations monitoring indoor areas where any kind of physical flows are present.

2.1   Problems of Camera Placement

The basic approaches in this field of research are employed in virtual reality simulators. In the case of [5] and [6] the matter is to provide the user some help for the placement and the control of cameras in 3D environments. The first aims at browsing comfortably in a software tool and at observing a target object. For [6] the objective is to make it possible to simulate the efficiency of the placement of a camera network in a scene representing a real case (a surgical operation). In [5] screenings are computed with an adaptation of the hemi-cube algorithm [7]. The virtual environment proposed in [6] provides the user with a decision-making aid in the placement of a camera network, by providing him information concerning the coverage given by the

cameras and also by putting forward the resolution of the cameras on the various zones of the image. The second class of problem is very close to the field of cinematography and tackles the issue of browsing in virtual environments. The topic of this kind of work is the way by which the visualized scenes are presented on the screen. Some approaches are similar to the realization of storyboards or to the virtual rehearsal of shots for the cinema [8]. Others are directed towards the human machine interfaces [9], [10], [11], [12]. They thus mainly utilize computations that use the internal models of cameras to find the place of the objects observed in the image plan. The position of points in images is also the main interest of work in photogrammetry. Within this framework these techniques are used to set up networks of cameras allowing a very precise measurement of the objects either with very complex geometry or whose size forbid the more classical methods of measurement. Those cases are the measurement of industrial pieces [13], [14], or of buildings of complex architecture [15]. Finally, the third field that uses placement and camera control method is the localization of objects or humans. The objective in the first case is rather close to the concerns seen in photogrammetry. It aims at carrying out a measurement as precise as possible. For video surveillance, some goals are common to the case of cinematography, because one wishes to have some objects in the shot with a sufficient resolution, but the organization of the shot is left completely free [16]. Actually, in this work the coverage provided by the cameras is the priority. All those cases of placement methods and camera control have specificities, nevertheless we note that they are all guided by their final goal. That leads the researchers to set up strategies using constraints optimization. Depending on the fields, those constraints are extremely diverse, but they are generally translated into objectives on the parameters describing the cameras.

2.2   Definition and Model of the Studied Scenes

In our problem we consider that the study is undertaken on a scene. According to the requirements of the Capthom project the scenes are either tertiary buildings or a set of rooms, in which we want to carry out a control of the energy consumption. A scene describes a zone and its environment that one wants to supervise. A scene is thus a set of element describing the physical aspects coupled with the objectives researched in term of collected information. The shapes of the rooms must hence be described as well as the position and shape of the furniture. The disturbances of physical flow must also be indicated, as well as the elements describing the mission of the system. The mission can be defined by a zoning of the scene indicating various priorities for various parts of the spaces. However, the description of the scene should not be limited to the zones to be observed, but it should also take into account some excluded zones. These kinds of areas are for example the space behind a window or behind an opened door. In the future we would like to add the utilization that humans make of the room to conduct very precise scenarios concordant with real use. The obstacles are zones in which a given flow cannot pass through, therefore they can be material like solid objects or immaterial like electromagnetic flows.

We thus have defined several types of zones to specify a scene. There are initially the internal zone or zone to be covered and the one considered as external. This definition may seem to be naive but demonstrates its utility if we want to consider the

space seen behind a door, a window or any other kind of opening. One can then define the zones of material or immaterial obstacle by defining the edges of the latter. This allows to represent the geometry of the scene. A third type of zone used is the disturbing zone, which do not stop flows, but degrade them or modify them. They are utilized to model that certain phenomena which appears in houses, like the displacement of air mass of different temperatures or the radiations emitted by various equipments, can have effects that are far from being negligible for the sensors. To summarize, there are then two principal types of zones, on the one hand the zones to be observed and on the other hand the zones not to be supervised. Using levels of priority between the zones to be observed and to be avoided can graduate this binary definition.

To model those scenes we decided to start from the basis utilized in [16] that we have enriched to fulfill our multiple needs. In our models we chose to represent the scene by a list of points whose value indicates their nature (belonging or not to an obstacle). Each element of this vector corresponds to a point of the scene. The points of the scene are thus numbered in a given order. This vector, noted thereafter **Scene**, is carrying much information and can become a vector of couple of values, this is a modification of the model used in [16] that enrich is power of representation. We describe in this vector all the zones inherent to the mission. The points to be supervised are coded with a 1 in the element that carries their index and with a 0 if they do not represent an objective to be covered. The example in Fig. 1 presents a simple case where the interior of the red zone represents the points to be supervised.
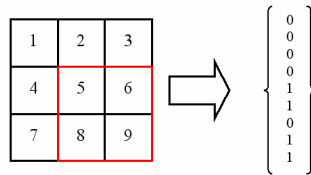
**Fig. 1.** Creation of the description vector of a scene

The construction of the vector representing the scene is done in two stages. The first part is the mathematical description of the scene by primitives: the vertices, the edges and the polygons. Then, the scene is sampled in the form of a vector whose elements carry information on the nature of a point belonging to the scene. We can obviously influence the smoothness of the discretization by increasing the number of points represented in the vector of the scene. The description vector of the scene is built by tests of membership to the polygons. For each point belonging to the discretization we check that it rests with the points to be observed or with those to avoid. This enables us to obtain the digital model of the scene that we will exploit to develop optimization algorithms. To exploit this digital model, we also set up an adapted model of the systems of sensors. We defined a model common to any class of sensor that we numerically adapted to the scene model for evaluating and creating construction solutions for sensor system adapted to particular problems.

2.3 Definition and Model of Sensor Systems

2.3.1 General Model

Before describing a system of sensors as a whole we tried to represent a sensor whatever its nature and the physical flow it measures in a universal way. According to us a sensor is characterized by various parameters that are the zone it covers, the precision of its measurement through this zone, its placement and the flow or perturbations to which it is sensitive. The covered zone can be very variable between various sensors. For a camera, this zone is the field of view, for a contact sensor this zone is reduced to a point. It thus appeared interesting for us to represent a sensor by its efficient zone i.e. the zone in which it can provide information on the flow it measures. Moreover, to represent the precision of the taken measurements this efficient zone must also be a spatial distribution of the reliability and measuring accuracy. For all the types of sensors we can consider that the efficient zone emanates from the sensitive cell of the system. To build these efficient zones and to adapt to various geometries of efficient zones we chose the method of ray tracing in the construction of the detection polygons of the sensors.

We give in Fig. 2 the example of the efficient zone concerning a camera. We can note that this type of efficient zone is quite similar for ultrasonic sensors and pyroelectric infrared systems.
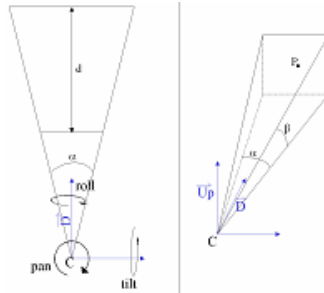

Fig. 2. Standard parameters of a camera, efficient zone

We understand on the preceding figure (Fig. 2) that the efficient zone of a camera-like sensor can be defined by three main parameters for a three-dimensional description. Those parameters are the azimuth α and latitude β angles as well as the depth of field d; concerning this last parameter, two options can be developed. At first sight, we can only consider the most distanced element that we can observe by taking only a depth distance. But we can also consider points that we cannot be analyzed because they are too close from the sensor, by considering two distances: one for the start of the effective zone and another for its end.

This kind of profile could also fit the effective zone of ultrasonic or pyroelectric infrared sensor. Indeed, an ultrasonic transducer is emitting on a zone where an object is detected if it reflects a detectable energy to the ultrasound receiver. The efficient zone for this type of sensor is then defined by the opening angle of the ultrasound emitter (that leads to a conic approximation of the detection polygon), and by the

maximal distance from which an object could reflect a significant energy. For the pyroelectric infrared sensor the basic technology is very different but its efficient zone is very close to the case of the camera and the ultrasonic sensor. The pyroelectric infrared sensors currently used for human motion detection are not localization sensors, like camera or ultrasonic transducer, in the sense that they are not able to give the position of the detected objects. This type of sensor only gives a binary measure of what it monitors, that is the presence or not of an object in movement that emits an infrared radiation. Concretely, the pyroelectric infrared sensors utilized by most industries for human detection [17] are composed of a passive pyroelectric infrared sensor component set up behind a Fresnel lens. This installation is used to create a distributed set of detection lobes across the monitored room. The sensor detects intrusion when a body that emits infrared radiation crosses at least two lobes consecutively. Physically the sensor covers the room with different rays as expressed in Fig. 3. In fact the real efficient zone is not only the ray where the sensor catches infrared signals but actually the whole environment between the first and the last ray. This zone is therefore comparable to a camera detection polygon characterized in the two-dimensional value the azimuth and the depth of field. This zone can be seen as the space in which the sensor detects the phenomenon that it tracks.

By using a ray-tracing algorithm, we then obtained a list of points representing the vertices of the detection polygon for the sensor considered. Indeed, each vertex is the intersection point with the first obstacle met and whose distance with the center of the sensor is lower than the depth of field. An algorithm implemented in ©Matlab enabled us to obtain detection polygons sufficiently precise toward the scene scale considered, Fig. 4 presents a polygon made up of 102 vertices.
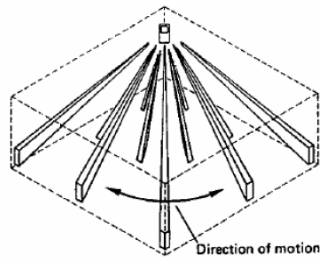

Fig. 3. Infrared rays used by an infrared movement detector (figure from [17])
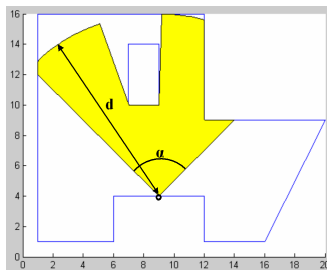
Fig. 4. A detection polygon

2.3.2 Modeling Reliability and Measurement Confidence

This process enabled us to obtain the polygon representing the efficient zone of a sensor. We used this element as a basic to detail the characteristics of the considered sensor. Indeed, by considering the laws giving the precision relative to the distance of a point, we can graduate the detection polygon with the reliability of measurements. These laws, meaning the measurement accuracy, depend on the quality of the sensor and can also be classified by mission. For a camera, we can express that the precision of information relates to the number of pixels per millimeter. However, we can also complement this information by giving the minimal resolution necessary for various applications. We will thus be able to say if the information collected at various spots of the scene is sufficient to carry out face recognition or activity recognition. We used in experimentations an ultrasonic sensor made by Polaroid whose precision decreases linearly. If we choose to represent this sensor considering that it can sweep an angle of 90 degrees with a depth of field of 12 meters, we obtain the following profile (Fig. 5).
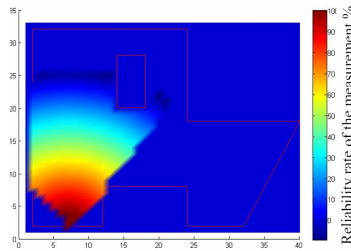
Fig. 5. Example of distribution of spatial measurement reliability

We represented the scene by a vector of points. We identically present the detection polygon in a vector where all the points of the scene are considered. In this way the vectors of the sensors and those of the scene are comparable. It is then easy to find the zones covered by the sensors as well as their characteristics. We chose to represent the sensors by two distinct but complementary vectors. The first collects the points that are in the detection polygon and carries information on the type of zone to which they belong. These zones can be as we presented in the preceding part, the zones to be seen, closed areas or zones of various priority levels. The second vector is used to describe the reliability of measurements at each point of the scene provided by the sensor. We then registered for each point of the scene the measuring accuracy that the sensor offered.

In order to represent a sensor system with those models, we added the descriptive vectors of each element of the system. We then obtain all the points "observed" by the system. For some of them, the addition allows to highlight redundancies in their observation. It is the same for the reliability vector that indicates the combinations of sensor that increase the reliability of measurement.

We now have the models of the scene and we also know how to model very different sensors. We tried to make these models concordant to be able to model the whole placement problem and the evaluation of the feasibility of the scene monitoring.

3   Problem Modeling

The problem that we pose is to be able to propose and assess sensor systems for human presence detection, in a scene proposed by an external user. We considered the following data as the input data of the problem: the shape, the furnishing and the zoning of the scene as well as the positions and orientations possible for the sensors usable in the scene. Therefore we had a finished number of sensors usable and a finished set of position for this. Consequently, we obtained a finished set of solution to the problem. The problem thus consists in finding the best solution in the space suggested. In [16] the procedure is quite similar. Nevertheless their research is only turned towards obtaining solution ensuring 100% of coverage of a room. Moreover, this work is limited to the use of cameras and not of other types of monitoring systems. We thus largely adapted the way of proceeding of this work to adapt to our needs. We largely widened the models employed and increased their representation capacity. Our contribution is to have generalized the models for any type of presence sensor and to put forward parameters much more complex than a simple covering of a room. We think in particular about the addition of the reliability vector characterizing the sensor systems.

The installations in real environment present a lot of specificities that could not be solved by a solution guaranteeing only a total coverage of the room as in [16]. Indeed the real establishments of sensor systems often mean the use of limited means as the presence of specific technical difficulties that are sometimes unforeseeable. Moreover, the priorities of installation cannot be reduced to simple research of zones to cover. To carry out the resolution of such requirements we chose to use genetic algorithms offering the flexibility and the effectiveness that were necessary to our approach. In this part we will explain the mathematical formulation of the problem. Then, we will discuss the methods usable for its resolution.

3.1   Mathematical Formulation

In order to solve the problem of sensor placement we wished to place ourselves in the traditional case of linear programming problem under constraints. I.e. we wished to formulate the problem in the form of formula (**1**). Function **f** is the cost function to minimize, A is the system matrix, b the constraints and x the selectable parameters whose upper and lower bounds are ub and lb.

Minimize $f(x) = c^T x$

$$\text{With respect to} \qquad A \times x > b \qquad\qquad (1)$$

$$ub \geq x \geq lb$$

The construction of the models previously presented is clearly directed to this end. In fact, in this context, a point is seen by the capture system if its value in the detection vector is positive. We also note that this point must be observed if a positive value is allotted to him in the description vector of the scene. Finally, to represent a sensor system we saw that it consisted in adding the detection vectors of each component. This is what we can do by using a vector of selection. This vector is the one that makes it possible to indicate which components of the basic list are used to

constitute the system. Its number of rows is the number of sensors that can be installed and carries a 1 for the element representing a selected component. In formula (**1**) the "b" vector is thus the description vector of the scene named before **Scene**, the "x" vector is the selection one. Matrix "A", of dimension Numbers of Points $\times$ Number of possible sensors, is built by concatenating on the right all the detection vectors of the sensors installables in the same order as the one used for the selection vector. We thus can compare for each point of the scene if it appears in the points seen by the system, i.e. if $A \times x > b$. The process can be schematized in the following way (Fig. 6).
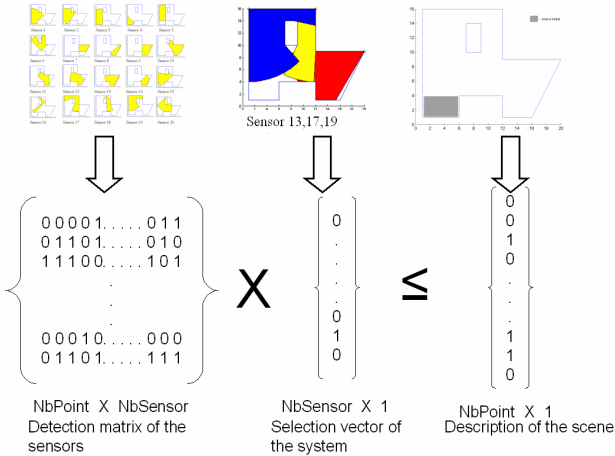
Fig. 6. Mathematical model of the placement problem

In order to pose the linear programming problem we still have to express a cost function **f** to be minimized. This function can be the means to introduce a real cost, related to the price of the material employed, the installation or the energy consumption. It can also represent the quality of the material or another more subjective evaluation. The final function can balance these various aspects to form a multi-criteria function that will be adapted according to the policy of desired installation. This function gives for each possible sensor the total cost of its installation and of its exploitation; the various parameters must be parameterized according to the desired priorities. For example, the priority can be, a low energy consumption that would result in assigning a very high cost to the systems consuming a lot of energy. On the contrary, if one simply wishes to minimize the number of systems used it is enough to assign a unit weight to all the sensors. To solve the optimization problem previously posed, there are many possible ways that we will present now.

3.2 Resolving Method

Many works on camera placement used a large variety of different optimization methods. Our work is inspired from those that have followed the 0-1 canonical model

of optimization [16] to solve the problem formulated in (**1**). But work of Olague proposed the use of genetic algorithms that are multi-cellular [13] or "Parisian" [14]. The genetic algorithms are known for their effectiveness to solve very complex and non-linear problems, whose form of the solution set is badly known, or when the problem is difficult to formalize by traditional methods. This kind of algorithm functions like a research in parallel, which functions on the principles of natural selection, that also allows the avoidance of local optimum. In our case, the variable to be optimized is the selection vector. This vector represents the system. We thus optimized the selection of the sensors. This vector is in a simple case a binary vector, it thus lends itself particularly well to the use of optimization of 0-1 canonical models like [16] done. To solve the problem, we initially used two distinct methods: the Branch and Bound algorithm to have a comparison base with [16] and the genetic algorithms.

3.2.1 Use of the Branch and Bound Algorithm

We have adapted the format of our input data to use this method. We thus wrote all the vectors as binary vectors. This induces a huge simplification of the problem because we can consider only the problem of covering 100% of the scene and avoiding areas. The Branch and Bound method is a generic method of resolution of optimization problems, and more particularly of combinative or discrete optimization. This method makes it possible to solve complex problems by enumerating the solutions in an eligible set. However, it has of a reduction mechanism of exploration by the progressive evaluation of the solutions considered. In the Branch and Bound method we divide the basic problem in a collection of simpler sub-problems covering the totality of the root problem. The Bound function makes it possible to limit exploration, by trying to test the utility to develop or not a branch of the tree representing the total research. This is done when it is feasible to easily calculate the optimums of the sub-problems. If for instance a local minimum is higher than a solution previously calculated the branch is then abandoned. The use of the Branch and Bound method enabled us to validate our models and implementations that were made with ©MATLAB. Moreover, we then succeeded in obtaining results on the discovery of placement solution ensuring 100% of room monitoring with a minimal number of sensors. We tested it on the scene described hereafter for which a version of Branch and Bound coded with ©MATLAB gives the following results (Fig. 7).

The use of this algorithm enabled us to get our first results for the placement of the system with three main constraints which were: to cover the entire room, not to supervise the prohibited points and to be limited for the number of sensors, their characteristics and the place where they could be located. We did this to validate models for the type of scene that we wish to study. However, for the establishment of our simulation and placement software we wish to go further in the type of constraints considered. We also wish to be able to easily have partial solutions if a solution perfectly respecting all the constraints does not exist. For these various reasons, we chose to explore a new way for the optimization of the systems of sensors construction.
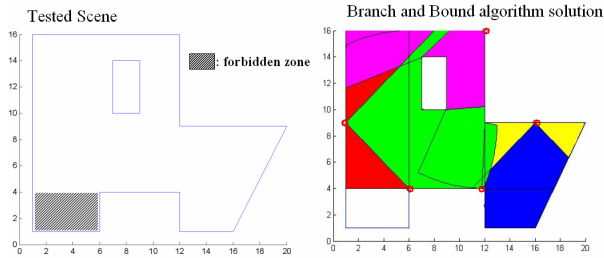
Fig. 7. Branch and Bound algorithm solution

3.2.2   Use of Genetic Algorithm

For our problem, we used a binary genetic algorithm justified by the object of the research, which is the sensor selection vector of the total system. The genetic algorithm and their functioning have been well defined in [18]. Many parameters can influence the results of such research. Nevertheless, we currently choose to focus on one main aspect. We mainly worked on the fitness function to carry out various optimization policies. These policies attempt to exploit the maximum of information that the models can carry. We used a selection function, based on geometrical law, whose probability of selection is proportional to the adaptation of the individual. This has been done to preserve a partly random selection and thus to avoid the local optimum.

The variety of the scenarios under consideration for the system and the variety of the priorities of each user make extremely interesting the possibility of proposing various configurations for the same scene. The real deployment of the solutions and the unpredictability related to the real case force to have multiple solutions to the same problem referring to a given scene. We have been able to fill this need for adaptability in the generation thanks to the use of these algorithms. Indeed, this type of algorithm is shown to be very flexible and effective in the search for solution to mixed multi-criteria problems. To emphasize the effectiveness of this option, we carried out many placement tests. The great flexibility of this method is due to the facility to create fitness functions directing researches towards different solutions. We thus expressed various policies in the creation of these functions and we implemented many tests to justify and highlight the parameters to be varied as well as the values to affect to them. Thereafter, we will present these policies and the results obtained, by pointing out the variations under consideration for the various parameters.

We initially chose policies of fitness function searching an acceptable coverage rate in ratio to the cost or the heaviness of the monitoring installation. We considered four principal parameters to be fixed to obtain results satisfying our expectations. Those consist in the maximum avoidance of the excluded zones, not exceeding the maximum number of sensors desired in a solution (being able to be fixed with the results of the Branch and Bound algorithm) and the balance between the profit in the covered zone provided by a new sensor and its cost, which was never considered in the methods seen in the bibliographic study. For each policy used we will fix a percentage of minimal additional coverage that a new sensor must induce to legitimate its installation. To influence the three goals that we have just described, we built the fitness function around four parameters functioning with the addition of

bonus or handicap. The genetic algorithm that we used searches a maximum value for the result of the fitness function. We describe below the fitness function that we use.

This function takes in input the A matrices (detection matrix of the sensors) and **Scene** (description vector the scene) described previously and respectively representing the visibility of the sensors and the set of points to be observed in the scene. Are also considered in input, the selection matrix X of the sensors (X is the research solution), NbPoint the number of points in the scene, MaxSens the number of sensors used in the solution computed by the Branch and Bound if it exists, NbSens the maximum number of sensors in the scene and Cov the additional value of covering that a new sensor must bring. val is the value that will be given to the fitness function for a given individual X.

The sum function is the sum of every element of a given matrix.

```
val ← 0
for i=0..NbPoint
        if Scene ( i ) > 0
                if (A × x)( i ) ≥ Scene ( i )
                val ← val + 100 / sum(Scene )  // This line is built to work with
        if (A × x)( i ) < 0                     // percentage of coverage.
                val ← val – 100                 // Seeing the forbidden zone cancel
        if sum( x ) > MaxSens                   // the bonus of seeing 100% of the scene.
                val ← val – 100                 // It is the same if too much sensors are used.
val ← val + Cov × ( NbSens – sum( x ) )  // The smaller the number of sensor is, the
                                          // bigger val is.
```

This fitness function gives access to equilibrium between the covered zone and the number of deployed sensors. Indeed, a solution with a more important cover rate is not better noted than a solution with fewer sensors. It is what arises from Fig. 8 that presents the topology of the fitness function presented for a value of the Cov parameter of 20% and by considering the maximum number of desired sensors as 5. The number of prohibited points is set to zero. We can thus observe a step around a number of five sensors that makes it possible to avoid any solution using a too high number of sensors. This is used to support the convergence of the genetic algorithm towards the best solution. We also notice in Fig. 8 that solutions with a less important cover rate are as well noted as a solution with 20% additional coverage and with one supplementary sensor. Besides, they are better noted than a solution with a supplementary sensor and a higher coverage rate of only 10%. This type of policy thus enables us to go further from a research of Branch and Bound type. In fact we are not searching only a single optimum based on a single criterion. Fig. 8 presents the topology of the function if all the cases represented are accessible. For instance, if for each possible number of sensors one can obtain any possible coverage rate. In real cases, the topology is only a part of the complete case. All the points illustrated below will not be accessible, which will give topologies closer to the one presented in Fig. 9.
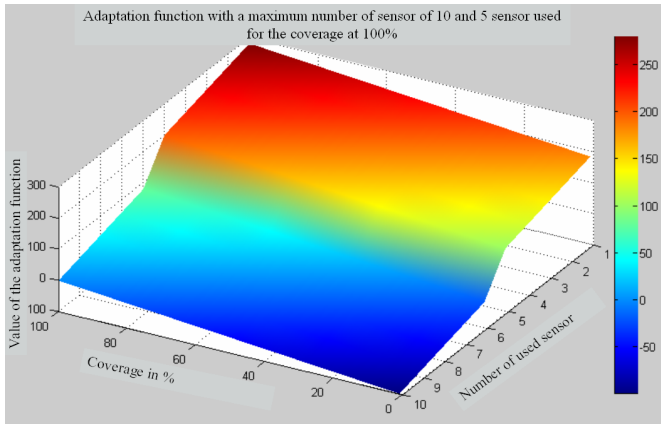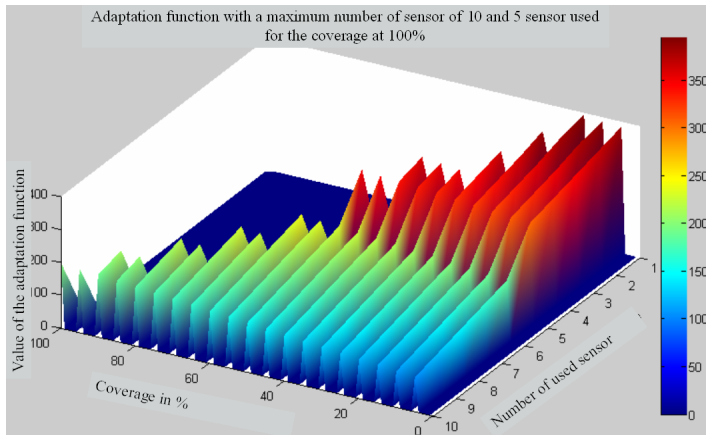
Fig. 8. Fitness function topology

Fig. 9. Fitness function topology in a real case

In Fig. 9 the topology appears to be largely modified compared to the case when all the configurations exist. In this example, there are 20 possible sensors, considered disjoined (their detection polygon does not overlap), the user does not want to use more than 5 sensors and the various percentages of coverage that we could find are 5, 10 or 15%.

3.3   Experiment

Using the genetic algorithm is like moving on the surfaces plotted on the two previous figures and searching to find the highest point of the surface. This function was used to calculate sensor systems usable in the following scene (Fig. 10). Others types of scene, giving encouraging results, were tested but are not presented in this article.
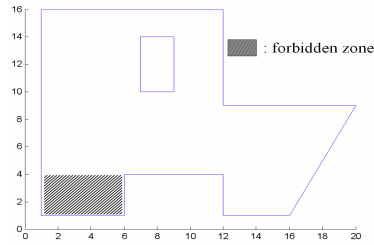
Fig. 10. The studied scene

The allowed positions for the sensor that we could employ in this scene were the following (Fig. 11). Those sensors are all camera-like systems but as explained on section 2.3.1 the polygon shape are quite the same for pyroelectric or ultrasonic component. Moreover the shape of those zones does not affect the efficiency of the method because all the polygons are used as list of points.
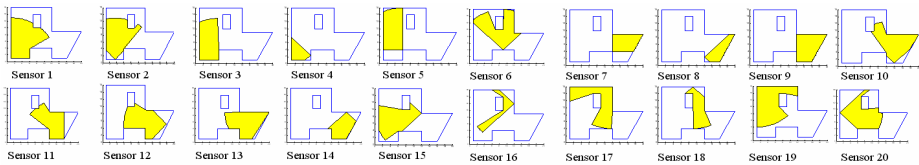
Fig. 11. Set of usable sensors

A solution ensuring 100% of coverage of the scene has been found by using a Branch and Bound algorithm and is presented in Fig. 7.

Various values of the Cov parameter were tested. We thus carried out tests for values of 5, 10, 20, 30, 40 and 50% of additional covers necessary to justify the installation of an additional sensor. The following table synthesizes the obtained results. We show on each line a test carried out with a different value of Cov.

Table 1. Configuration for the scene with coverage-centered policies

Cov	Number of utilized sensors	Rate of coverage in %	Final value of the fitness function	Index of the selected sensors			Obtaining rate of the solution for 10 tries*
5%	3	95.9815	17598	13	17	19	100%
10%	3	95.9815	26598	13	17	19	100%
20%	2	85.7805	44578		13	19	100%
30%	2	85.7805	62578		13	19	100%
40%	1	49.4590	76946			19	100%
50%	0	0	95000		NULL		100%

* For 4000 generations of 100 individuals

The use of this policy enabled us to highlight alternative solutions to the solution comprising 100% of coverage rate. These solutions are more adapted to the problem

of this scene. Indeed we obtain configurations with high cover rates (95,98% and 85,78%) and a number of sensors used less important (3 even 2 compared with the 5 of the first solution). We notice in this table that the addition of sensor n°17 brings 10,2% of additional cover. It is thus added to the configurations when Cov is only set to 5 or 10%, which strictly respects the behavior awaited for our policy. It is the same for sensor 13 that represents a profit lower than 40% of additional coverage rate. The results are presented in a graphic way in the following figure (Fig. 12).

Sensor: 19   Sensors: 13,19   Sensors: 13,17,19
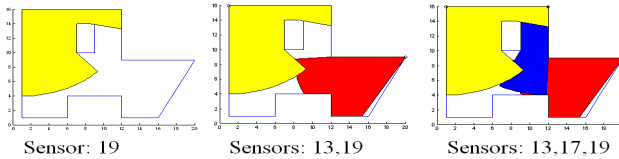
Fig. 12. Solution obtained with genetic algorithms

4   Conclusion

We have presented our first investigation in the establishment of a placement simulator for the Capthom project. The principal characteristics aimed for this system are a huge generalization in the choice of the systems. We searched placement solutions taking into account mixed objectives related to very heterogeneous aspects such as: the number of systems, the redundancy of measurements, the search for particular points or the electric consumption of the system. This has been made through the concordance of the scene and the sensor system models. The models were made up to recreate a classical optimization problem. Then, the problem has been represented in the form of a computable problem with the use of genetic algorithm in the ©Matlab software. We described how to solve this problem by the use of genetic algorithms. We presented a construction of the fitness function by providing an easily reusable and modifiable framework. We indicated the parameters to be modified according to the required objectives. Moreover, this implementation is shown to be very flexible and can be a basis for the design of a final simulator. Indeed, it allowed us to easily represent various phenomena as we showed with the use of a zoning of the scene and the calculation of the distribution of reliability through the scene. We presented results about policies balancing cost and coverage; others have been obtained for the management of redundancies between sensors involving progress in the system reliability. This method is to be utilized in design time. The future developments will have to validate the real application of the Capthom project. More than a simple placement aid simulator, we aim at creating a system model for sensor network but also a software that will validate the use of such systems in well-defined utilization scenarios of the scene. It will be a very interesting point to be able to consider changing environments perhaps by creating a furniture map containing probability of presence of the furnishings.

References

1. Benard, F., Barthel, L., Campo, E., Esteve, D., Raffray, P., Delaunay, P.: Etude d'un Gestionnaire d'Energie de Concept ERGDOM. Rappots de contrat, convention EDF/ADEME n 02.04.134 (2004)
2. Zouba, N., Bremond, F., Thonnat, M., Vu, V.T.: Multi-sensors Analysis for Everyday Activity Monitoring. In: SETIT 2007 (March 25-29, 2007)
3. Chan, M., Campo, E., Esteve, D.: PROSAFE: a multisensory remote monitoring system for the elderly or the handicapped. In: 1st International Conference On Smart homes and health Telematics (ICOST 2003) Paris, pp. 89–95 (2003)
4. European project SOPRANO. Information Society Technologies (IST) public deliverable IST – 2006 – 045212 (2007)
5. Philips, C.B., Badler, N., Granieri, J.: Automatic viewing control for 3D direct manipulation. In: Proceedings of the 1992 Symposium on Interactive 3D Graphics, pp. 71–74 (1992)
6. State, A., Welch, G., Ilie, A.: An Interactive Camera Placement and Visibility Simulator for Image-Based VR Applications Stereoscopic Displays and Virtual Reality Systems. Proceedings of the SPIE 6055, 640–651 (2006)
7. Cohen, M.F., Greenberg, D.P.: The Hemi-cube a radiosity solution for complex environments. Computer Graphics 19(3) (1985)
8. Bares, W.H., Thainimit, S., McDermott, S.: A model for constraint-based camera planning. Smart Graphics. In: Papers from the 2000 AAAI Spring Symposium, Stanford, March 20-22, 2000, pp. 84–91 (2000)
9. Drucker, S., Zeltzer, D.: Intelligent Camera Control in a Virtual Environment. In: Graphics Interface 1994, pp. 190–199 (1994)
10. Drucker, S., Zeltzer, D.: Intelligent Camera Control in Graphical Environment. Ph D. thesis, Massachusetts Institute of Technology, Cambridge (1994)
11. Drucker, S., Zeltzer, D.: CamDroid: A System for Implementing Intelligent Camera Control. In: 1995 Symposium on Interactive 3D Graphics, pp. 139–144 (1995)
12. Bares, W., Lester, J.: Intelligent Multi-Shot Visualization Interfaces for Dynamic 3D Worlds. In: IUI-99, Proceedings of the 1999 International Conference on Intelligent User Interfaces, Los Angeles, pp. 119–126 (1999)
13. Olague, G.: Automated photogrammetric network design using genetic algorithms. Photogramm. Eng. Remote Sensing 68(5), 423–431 (2002)
14. Dunn, E., Olague, G., Lutton, E.: Parisian Camera Placement for Vision Metrology. Pattern Recognition Letters 27, 1209–1219 (2006)
15. Saadat, S.M., Samdzadegan, F., Azizi, A., Hahn, M.: Camera Placement for Network Desig In Vision Metrology based On Fuzzy Inference System. XXth ISPRS Congress, July 12-23, 2004. Istanbul, Turkey (2004)
16. Erdem, U.M., Sclaroff, S.: Automated Camera Layout to Satisfy Task-Specific and Floor Plan-Specific Coverage Requirements. Computer Vision and Image Understanding 103, 156–169 (2006)
17. Gobeau, J.F.: Détecteurs de mouvement à infrarouge passif. Cinquièmes rencontres capteurs Capteurs 2006, Pôle Capteurs automatismes (2006), http://www.bourges.univ-orleans.fr/pole_capteur
18. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)