# Perception Planning for an Exploration Task of a 3D Environment *

María Teresa Lozano Albalate † ‡ Michel Devy ‡ José Miguel Sanchiz Martí †
†Universitat Jaume I
12071 Castelló de la Plana, SPAIN
lozano@linx.act.uji.es, sanchiz@uji.es
‡LAAS-CNRS
31077 Toulouse Cedex 04, FRANCE
michel@laas.fr

## Abstract

*The incremental recovery of the 3D model of a closed environment is a complex task as it involves several functions: 3D acquisition, data registration and fusion, construction of a 3D triangular mesh or other representations. The sensor must be placed in several positions so that the resulting model is complete (no unseen areas) and has the required resolution. This paper presents a method devoted to the selection of the next best view for a sensor used for 3D modelling of an environment; the objective is to minimise the number of acquisitions and optimise the quality of the final model. From the current sensor position, an optimal next view is selected by the use of utility functions computed for some possible solutions; the method combines exhaustive search and hill climbing optimisation. Thanks to a simulation, results are presented for a synthetic environment, using a virtual 3D sensor on a 5 degrees of freedom robot.*

## 1. Introduction

This paper deals with the autonomous reconstruction of a 3D model for a closed environment. Many applications require such a model, for example: the quality control by architects, the 3D modelling of historic monuments or museums to propose virtual visits on a web site, etc.

3D modelling is a tedious incremental process, involving complex equipments and methods. At each iteration, data are acquired using a 3D sensor placed in a "good" position. These data are registered and fused with the current model and, if required, an environment representation is generated or updated from the augmented model, typically a triangular mesh for graphical applications, or a set of surface features for metrology applications. A perception planning method must be executed between each acquisition. The problem to be solved is the selection of the next best view and the detection of the end for the modelling task. Several criteria can be used depending on the sensor geometry, the equipment used to move the sensor (a mobile robot, an operator or else) and the application requirements, especially the resolution and the accuracy of the final model.

Other researchers have addressed the perception planning problem for several tasks, using different sensory modalities. In [10] a geometrical algorithm is presented to compute, from the current scene model, the visibility volumes from where occluded areas can be seen. In [8] a method is proposed to select the best view for a sensor moving on a cylindrical surface, and turning around an object to be modelled. [3] describes two algorithms based on the local octree models to determine the next best view around a little object to be modelled. In [4] it is assumed that a polygonal map of a workspace is already known. The problem consists of finding the minimum set of sensors and their respective positions in the scene, inside this map from which the whole workspace boundary is visible. The approach is based on the application of probabilistic methods for sensor placement; the provided sensor positions are not the optimal ones, but they represent a practical solution, with a low computational cost. Nevertheless, this paper does not deal with the modelling task. Other works, [2, 7], deal with the construction of a 2D model using a laser sensor with only a line scanning device. The model is polygonal, and geometrical reasoning is performed to select the next sensor position considering polygon intersections.

Our research is based on the work presented in [5, 9]. The modelling task is assumed to be performed autonomously by a 3D sensor (typically a laser with pan and tilt scanning) mounted on a mobile robot. Our method uses the optimisation strategy and scene representation presented in [9]. Our contribution concerns two extensions of this method: a measure of the model quality is taken into account, and utility functions for a viewpoint are defined with respect to new criteria, designed to encode the interest of this viewpoint. The optimisation method has been modified. Consequently the algorithm can choose the position more accurately to give us a greater amount of information.

* Work carried out as part of EC funded TMR network CAMERA (ERB FMRX-CT97-0127) and partially funded by Spanish MCYT (TAP99-0590-C02-01). Thanks to K.Klein for providing the environment model.

As in [5, 9], in our work there is not any initial knowledge about the environment before recovering it.

In the next section an overview of our system is presented. Section 3 focuses on the utility functions used during the optimisation in order to select the next best view. In Section 4, experimental results using synthetic data are presented. Finally, in Section 5 conclusions are drawn.

## 2. Overview of the approach

This section gives an overview of our method. A voxel map is used as an environment representation. It is also used by the registration algorithm (a variant on the ICP method [6]) to efficiently access the already acquired 3D data ; from the acquired points, another representation could be built [1], but it is out of the scope of this paper. A voxel map is a 3D matrix of voxels, each representing a little cube in the space. The voxel map resolution is an important parameter. In the current implementation it is fixed, and typically set to 15 cm. In future works we intend to begin the modelling task with a weaker resolution (typically 40 cm) and to improve the model quality increasing it in required areas. This means that the observer would do a first tour in the workspace to build an initial coarse model, and could plan other tours until the final model has the required resolution.

For each voxel, the position in the voxel map, and the label are known. Moreover, depending on the voxel label, the number of acquired points (*occupied* and *border* voxels) and the voxel mean normal (*occupied*, *border* and *occplane* voxels) are known. Possible labels are:

- *unmarked*: voxel in an unseen area. Initially the environment is unknown and all voxels are *unmarked*.
- *empty*: voxel in a perceived area, but no points have been acquired at this position.
- *occupied*: voxel on which points have been acquired; such a voxel belongs to a surface that could be an obstacle for the sensor motion.
- *occluded*: voxel in an area that was in the field of view but that has not been seen because an occupied area was detected between the sensor and this voxel;
- *occplane*: voxel that is occluded, but is adjacent to an empty voxel with any of its six faces;
- *border*: voxel that is occupied, but is adjacent to an *unmarked* voxel; such a voxel was at the limit of the field of view during a previous acquisition.

For *occupied* and *border* voxels, the mean normal is computed from all the points currently acquired for each voxel. The *border* voxels have been introduced for two reasons. First, they allow to detect the boundary between known and unknown areas. The next acquisition to be planned must decrease the number of *unmarked* voxels, but must also perceive some *occupied* voxels again, so that a registration can be done between the current model and the next view. This means that the next field of view must cover some *border* voxels. Second, in order to improve the quality of the acquired model, it is better to perceive surfaces with the view ray orthogonal to them; unseen surfaces normals are unknown, but can be estimated using the nearest perceived surfaces. The *border* and *occupied* voxels are useful to estimate the surface normals on *unmarked* voxels. For *occplane* voxels, the normals are also computed using the plane that they form with their neighbours. It is introduced to choose the optimal view to perceive an occluded area.

## 3. Utility function of a view

Our method is based on an optimisation function to search the best sensor position and orientation to acquire a new view reasoning on the current model. A sensor position includes the coordinates of its reference frame origin $(X_s, Y_s, Z_s)$, and the pan and tilt orientation according to the global reference frame $(\theta, \phi)$. These are the sensor extrinsic parameters. The intrinsic parameters are assumed to be constant: sensor minimal and maximal range, solid angle defining the field of view, image size, etc. The utility criteria must quantify the interest of a possible sensor placement. A view point should have different interests: overlap some occupied areas to make easier the registration; solve occlusions (perceive new areas beyond an occluded plane); perceive unseen areas to extend the model; and improve the quality of *occupied* voxels.

The utility criteria for a sensor placement are based on (a) the ratios of *occupied*, *unmarked*, *occplane* and *border* voxels present in the view, and (b) the acquisition quality, that can be expressed using the normal estimations or predictions. We have defined functions, built from different parts, to evaluate the utility of a view. Each part is related to a voxel class; and built from two terms: an overlap term and optionally a quality term. For *occupied*, *occplane* and *unmarked* voxels, the overlap term is the same. As a generic overlap term, we have chosen a cubic equation: $f_{overlap}$=$Ax^3 + Bx^2 + Cx + D$ ($x$: voxel class ratio on a view). This function must have a maximum utility 1 for an optimal percentage $x$=$\alpha$, and a minimal utility if only one class of voxels is perceived $x$=1 or if no voxel of this class is perceived $x$=0. Using the constraints on the curve, the overlap term is:

$$f_{overlap1(x)} = \frac{-2}{\alpha^3 - 3\alpha + 2}x^3 + \frac{3\alpha}{\alpha^3 - 3\alpha + 2}x^2 + \frac{-3\alpha + 2}{\alpha^3 - 3\alpha + 2}$$

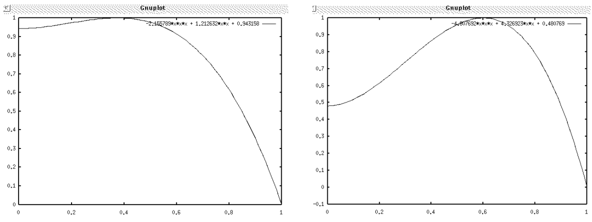

**Figure 1.** $f_{utility1}$ **(**$\alpha$**=0.375, left) and** $f_{utility2}$ **(**$\alpha$**= 0.6, right) used for** *unmarked***/***occplane* **voxels.**

As a generic overlap term for *border* voxels we have selected a quadratic equation: $f_{overlap}$=$Aw^2 + Bw + C$ ($w$: *border* voxel ratio), with constraints: $f_{overlap2}(w)$ must have a maximum of value $f_{overlap2}(w)$=1 for an optimal ratio $w$=$\beta$, and must be $f_{overlap2}(w)$=0 for $w$=$\gamma$. The advantage of using a quadratic function is that it allows a bigger variation in the low values range.

$$f_{overlap2(w)} = \frac{-1}{(\gamma-\beta)^2}w^2 + \frac{2\beta}{(\gamma-\beta)^2}w + \frac{-\beta^2}{(\gamma-\beta)^2} + 1$$

Using these generic terms, we have evaluated two utility functions. The first one, $f_{utility1}$, is obtained using the following ratios: $\alpha$=0.2 for *occupied* voxels, $\alpha$=0.375 for *occplane* (Figure 1), $\alpha$=0.375 for *unmarked* voxels and $\beta$=0.05 for *border* voxels. $\gamma$=0.15 have been chosen because a small border voxel quantity is preferred.

$$\begin{aligned} f_{utility1} \quad = \quad & (-1.420455x^3 + 0.426136x^2 + 0.994318) \\ & +(-2.155789y^3 + 1.212632y^2 + 0.943158) \\ & *\frac{\sum_{i=1,np} cos(\delta_i)}{np} \\ & +(-2.155789z^3 + 1.212632z^2 + 0.943158) \\ & *\frac{\sum_{j=1,no} Cos(\epsilon_j)}{no} \\ & +(-100w^2 + 10w + 0.75) \end{aligned}$$

Where, $x$, $y$, $z$ and $w$ stand respectively for the ratios of *occupied*, *occplane*, *unmarked* and *border* voxels in the view, $\delta_i$ is the angle between the normal vector of *occplane* voxel $i$ and the sensor orientation, $\epsilon_j$ is the angle between the normal vector of *occupied* voxels $j$ and the sensor orientation, $np$ and $no$ are the respective totals of *occplane* and *occupied* voxels in the image.

We evaluated a second utility function; for wich the term involving the *border* voxels has been removed because they represent a very small percentage of the totals. $f_{utility2}$ is obtained using $\alpha$=0.4 for *occupied* voxels, $\alpha$=0.6 for *occplane* voxels and $\alpha$=0.6 for *unmarked* voxels (Figure 1).

The two proposed utility functions allow us to acquire data on some *occupied* voxels. The difference is that $f_{utility1}$ favours acquisition of *occluded* and *unmarked* voxels as perpendicularly as possible, as a whole. In other words, an average solution, but not the best one. On the other hand, with $f_{utility2}$ the algorithm is able to choose a better position in relation with either *occplane* or *unmarked* voxels, depending on which one could give the better result. So, using $f_{utility2}$ it is more likely that the bigger proportion of voxels in the next image will be *occplane* or *unmarked*, but not both at the same time, as it occurs with $f_{utility1}$. Using $f_{utility2}$ the possibility of being in an intermediate position is seriously reduced. Now sensor placement in front of only *occluded* or only *unmarked* areas will be favoured.

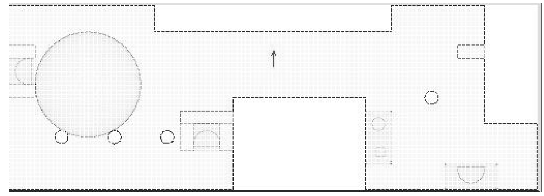

**Figure 2. The CAD model of our synthetic environment.**

## 4. Experimental results

Figure 2 shows a top view of a simulated environment consisting of a triangulated room with columns, tables and a cupola (about 380.000 triangles). Figure 3 shows images representing a sequence of acquisitions. In (a) only the occupied voxels are shown; the sensor perceives a wall, the ground and the ceiling. In (b) some *occplane* voxels are shown (light coloured). In (c) the sensor acquires data on the other side of the environment. (d) shows it from another point of view, and switching off the occlusion plane voxels. In (e) more occupied voxels from other areas are shown. After some more acquisitions the cupola and its surroundings have been perceived (f). Figure 4 shows the evolution of some measures versus view number. Three versions are compared: 1 uses the strategy proposed in [9]; 2.0 uses $f_{utility1}$; and 2.1 uses $f_{utility2}$. Figure 4(a) represents the area filled by the occupied voxel faces that touch an empty voxel. It is shown that version 2.0 grows quicker than version 1. $f_{utility2}$ (2.1) allows to perceive more occupied areas with less images. The results for the last iteration (25 acquisitions here) are the same for the three functions because almost all the environment has already been perceived. Figure 4(b) represents the mean quality of the *occupied* voxels. During the initial iterations it is very low because images are taken from the initial position. This is because there is not enough empty space for the sensor to move, so it only changes its orientation. Later, at iteration 8 and so on, the sensor can move because enough empty space has been imaged. Using our two improvements, the quality grows quicker than version 1 and, in the last iteration, $f_{utility2}$ obtains the best result.

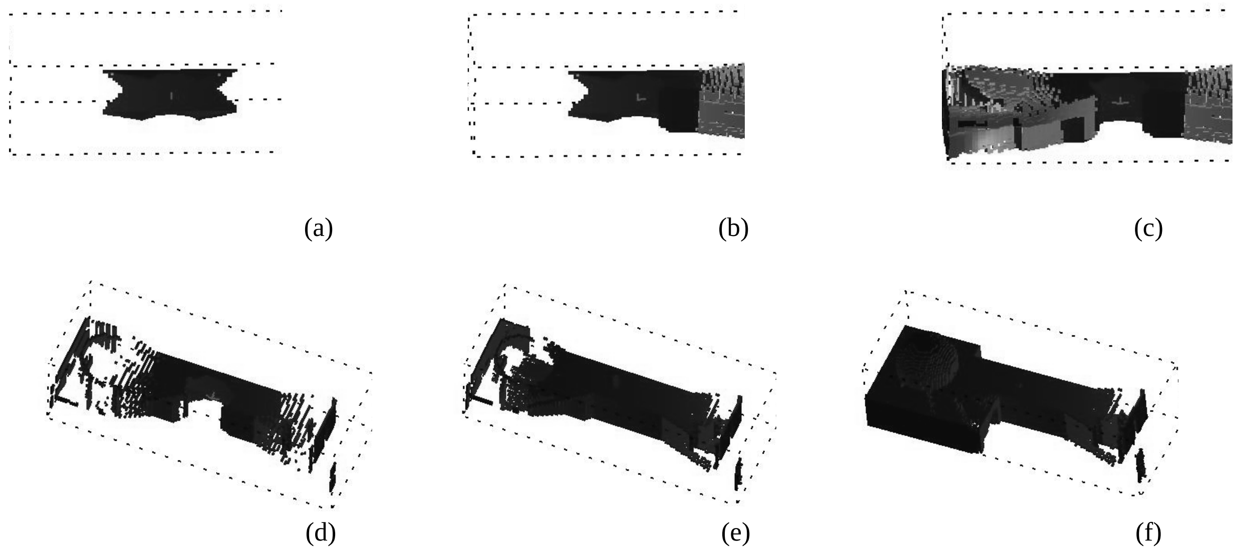


**Figure 3. (a) 1 acquisition, (b) 2 acquisitions, (c) 3 acquisitions, (d) 3 acquisitions without** *occplane* **voxels, (e) last image before any sensor movement, (f) an almost finished model of the left zone.**

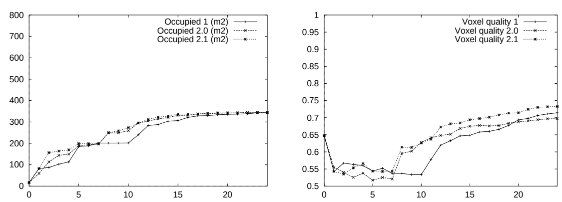


**Figure 4.** *Occupied* **voxels: (a) area, (b) mean quality**

## 5. Conclusions

A perception planning method has been presented. It deals with the next best view selection for autonomous and incremental reconstruction of a 3D model of an environment. It is intended to be performed using a 3D sensor mounted on a 5 degrees of freedom robot. Our contribution concerns the utility criteria used to quantify the interest of a view point. The method has been validated using a virtual 3D sensor and a synthetic environment.

As a future work, we intend to integrate this method with other 3D modelling functions on a robot equipped with an acquisition system, so that experimental validations can be performed. Another issue will be improvements in the optimisation method, so that the next best view selection could be processed one decade less than in the current implementation. Next, we will use a multi-resolution voxel map in order to improve progressively the model quality. Other improvements could be performed using geometric features rather than the voxel map, but it is well known that the extraction of surfacic features is not yet a robust process.

## References


[1] J. Bozier, M. Devy, and A. Sappa. A geometrical approach for the incremental modelling of free form surface by triangular meshes. In *International Symposium on Intelligent Robotic Systems, SIRS*, July 2000.

[2] J. Colly, M. Devy, M.Ghallab, and L. Pampagnin. Sensory control for 3d environment modelling. In *International Advanced Robotics Programme, 1st workshop on multi-sensor fusion and environment modelling*, pages 1–14, 1989.

[3] C. Conolly. The determination of next best views. In *International Conference on Robotics and Automation*, pages 432–435, 1995.

[4] H. González-Banos and J. Latombe. A randomized art-gallery algorithm for sensor placement. In *ACM Symposium on Computational Geometry, SCG*, 2001.

[5] N. Massios and R. Fisher. A best next view selection algorithm incorporating a quality criterion. In *9th British Machine Vision Conference, BMVC*, pages 780–789, 1998.

[6] G. Medioni and Y. Chen. Object modeling by registration of multiple range images. In *IEEE Conference on Robotics and Automation*, Apr. 1991.

[7] R. Murrieta-Cid, B. Tobar-López, C. Esteves-Jaramillo, and D. Pérez-Pérez. Robot motion planning under visibility constraints. In *International Symposium on Intelligent Robotic Systems, SIRS*, July 2001.

[8] R. Pito. A sensor-based solution to the next best view problem. In *13th International Conference on Pattern Recognition*, volume I, pages 941–945, 1996.

[9] J. Sanchiz and R. Fisher. A best-next-view algorithm for 3d scene recovery with 5 degrees of freedom. In *BMVC*, volume I, pages 163–172, 1999.

[10] K. Tarabanis, R. Tsai, and A.Kaul. Computing occlusion-free view-points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(3):279–292, Mar. 1996.