

Active Visual 3D Perception

Éric Marchand and François Chaumette

IRISA - INRIA Rennes- Université de Rennes I

F-35042 Rennes Cedex, France

fax: (+33) 99.38.38.32, E-mail: {marchand, chaumette}@irisa.fr

Abstract

This paper deals with the 3D structure estimation of a set of geometrical primitives in an active vision context. Our method is based on the structure from controlled motion approach which consists in constraining the camera motion in order to obtain an optimal estimation of the 3D structure of a geometrical primitive. Since this approach involves to focus on the considered primitive, we present a method for connecting up many estimations. We have developed perceptual strategies able to perform a succession of optimal estimation. A scene exploration process centered on current visual 2D information and the structure of the previously studied primitives is presented. It allows us to have an optimal estimation of each primitive of a scene composed of cylinders, segments and polygons.

1 Introduction

Many applications in robotics involve a good knowledge of the robot environment. For such applications, the aim of this paper is to obtain a complete and precise description of a scene using the visual data provided by a camera mounted on the end effector of a robot arm. A recent expansion of computer vision and image analysis is related to the estimation of 3D structure from image sequences [1][6][10][11][24][23]. The approach we have chosen to get an accurate three-dimensional geometric description of a scene is based on the active vision paradigm and consists in controlling the camera motion. The idea of using active schemes to address vision issues has been recently introduced [2][4]. Active vision is defined in [4] as an intelligent data acquisition process. Since the major shortcomings which limit the performance of vision systems are their sensitivity to noise and their low accuracy, the aim of active vision is generally to elaborate control strategies for adaptively setting camera parameters (position, velocity, . . .) in order to improve the knowledge of the environment [2]. Here, the purpose of active vision is handled at two levels : a **local aspect** where active vision is used

to constrain the camera motion in order to improve the quality of the reconstruction results and a **global aspect** which is used to explore the unobserved areas.

The measure of the camera motion, which is often necessary for the 3D structure estimation, characterizes a domain of research called dynamic vision. Approaches for 3D structure recovery may be divided into two main classes : the discrete approach, where images are acquired at distant time instants [6][11][24] and the continuous approach, where images are considered at video rate [1][10][23]. The method used in this paper is a continuous approach which stems on the camera velocity and on the motion of the considered primitive in the image. More precisely, we use a “structure from controlled motion” method which consists in constraining the camera motion in order to obtain a precise and robust estimation of 3D geometrical primitives such as points, straight lines and cylinders [5]. Such constraints are ensured using *the visual servoing approach* [9] where a vision system is considered as a specific sensor dedicated to a task and included in a control servo loop.

As far as the **global aspect** of our reconstruction scheme is concerned, active vision is used to determine the location of the next camera position in order to obtain a complete model of the scene. Previous works have been done in order to answer the “*where to look next*” question. Differences can be done if an *a priori* knowledge about the scene is available or not. If the complete geometrical description about the scene is known, many approaches about automatic sensor placement are described in [8][21]. The problem is different if no *a priori* information about the scene is available *i.e.*, if the sensor is in an unknown environment. It raises the problem of autonomous exploration [7][16][22][25]. Our concern is to deal with the problem of recovering the 3D spatial structure of a whole scene without any knowledge on the localization and the dimension of the different geometrical primitives of the scene (assumed to be composed of polygons, cylinders and segments). Since the proposed structure estimation method involves to successively focus on each primitive of the scene, developing perception strategies to get the spatial organization of complex scenes is thus necessary. Integrating knowledge on 3D data previously gathered, and current 2D informations into an explo-

This work was partly supported by the MESR (French Ministry of the University and Research) within project VIA (*Vision Intentionnelle et Action*) and under contribution to student grant.

ration process allows us to determine the next primitive to be estimated.

The remainder of this paper is organized as follows: Section 2 is devoted to the local aspect of our reconstruction scheme and describes the structure from motion framework based on an active vision paradigm. Section 3 is devoted to the global aspect and deals with the development of perception strategies. An exploration strategy based on a partial 3D model of the scene and 2D visual features extracted from the images are proposed.

2 Active Camera Control for Primitives Reconstruction

The work presented in this section is concerned with the analysis of a sequence of images acquired by a moving camera to get a precise and robust description of geometrical primitives [5]. The camera motion will be performed using the visual servoing approach [9].

2.1 Modeling Visual Sensing

Let us model a camera by a perspective projection. Without loss of generality, the camera focal length is assumed to be equal to 1, so that any point with coordinates $\underline{x} = (x, y, z)^T$ is projected on the image plane as a point with coordinates $\underline{X} = (X, Y, 1)^T$ with:

$$\underline{X} = \frac{1}{z} \underline{x} \quad (1)$$

Let us consider a geometrical primitive \mathcal{P}_s of the scene; its configuration is specified by an equation of the type:

$$h(\underline{x}, \underline{p}) = 0, \forall \underline{x} \in \mathcal{P}_s \quad (2)$$

where h defines the kind of the primitive and the value of parameter vector \underline{p} stands for its corresponding configuration. Using the perspective projection equation (1), we can define from (2) the two following functions [9]:

$$\begin{cases} g(\underline{X}, \underline{P}) = 0, \forall \underline{X} \in \mathcal{P}_i \\ 1/z = \mu(\underline{X}, \underline{p}_0) \end{cases} \quad (3)$$

where:

- \mathcal{P}_i denotes the projection in the image plane of \mathcal{P}_s
- g defines the kind of the image primitive and the value of parameter vector \underline{P} its configuration.
- function μ gives, for any point of \mathcal{P}_i with coordinates \underline{X} , the depth of the point of \mathcal{P}_s the projection of which results in point \underline{X} .
- parameters \underline{p}_0 describe the configuration of μ and are function of parameters \underline{p} .

More precisely, for planar primitives (a circle for example), the function μ represents the plane in which the primitive lies. For volumetric primitives (sphere, cylinder, torus, ...), function g represents the projection in the image of the primitive limbs and function μ defines the 3D surface in which the limbs lie (see Fig. 1). Function μ is therefore called the limb surface.

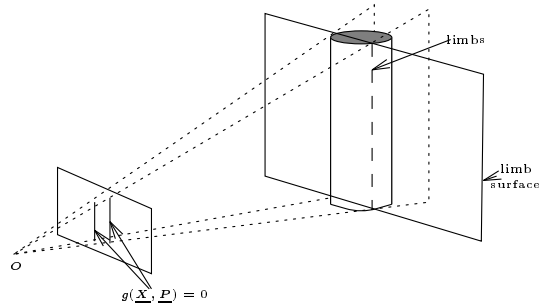


Figure 1: Projection of the primitive in the image (g) and limb surface (μ) in the case of a cylinder

Let $T_c = (V, \Omega)^T$ be the camera kinematic screw where V and Ω represent its translational and rotational components. The time variation of \underline{P} , which links the motion of the primitive in the image to the camera motion T_c , can be explicitly derived [9] and we get:

$$\dot{\underline{P}} = L_{\underline{P}}^T(\underline{P}, \underline{p}_0) T_c \quad (4)$$

where $L_{\underline{P}}^T(\underline{P}, \underline{p}_0)$, called the interaction matrix related to \underline{P} , fully characterizes the interaction between the camera and the considered primitive. A systematic method for computing the interaction matrix of any set of visual features corresponding to geometrical primitives (lines, spheres, cylinders, ...) is proposed in [9].

2.2 Structure From Motion

As previously stated, a geometrical primitive is defined by an equation $h(\underline{x}, \underline{p}) = 0$. Using the relation between the time variation of \underline{P} in the image sequence and the camera velocity T_c , we are able to compute the value of the parameters \underline{p} of the considered primitive [5].

First, from the resolution of a linear system derived from relation (4), we obtain the parameters \underline{p}_0 which represent the position of the limb surface:

$$\underline{p}_0 = \underline{p}_0(T_c, \underline{P}, \dot{\underline{P}}) \quad (5)$$

Then, knowing the position of the primitive in the image described by (3) and using geometrical constraints related to the considered primitive, we can estimate the parameters \underline{p} which fully define its 3D configuration:

$$\underline{p} = \underline{p}(\underline{P}, \underline{p}_0) \quad (6)$$

From a geometric point of view, this continuous approach leads to determine the intersection between the limb surface and a generalized cone, defined by its vertex located at the optical center and by the image of the primitive.

This approach has been applied to the most representative primitives (*i.e.*, point, straight line, circle, sphere and cylinder) [5]. Let us note that in the case of a cylinder, this method can be applied using the projection of only one limb in the image (however, more precise results are obtained using the projection of the two limbs in the image). Such a method based on one single limb will be

used to determine in Section 3.1 the nature of the observed primitive (cylinder or straight line).

2.3 Structure From Controlled Motion

When no particular strategy concerning camera motion is defined, important errors on the 3D structure estimation can be observed. This is due to the fact that the quality of the estimation is very sensitive to the nature of the successive camera motions [10]. An active vision paradigm is thus necessary to improve the accuracy of the estimation results by generating adequate camera motions.

As seen on equation (5), the 3D structure estimation method described in the previous section is based on the measurement of $\dot{\underline{P}}$ the temporal derivative of \underline{P} . However, the exact value of $\dot{\underline{P}}$ is generally unreachable, and the image measurements only supply $\Delta\underline{P}$, the ‘‘displacement’’ of \underline{P} between two successive images. Using $\Delta\underline{P}/\Delta t$ instead of $\dot{\underline{P}}$ generally induces errors in the 3D reconstruction. A sufficient and general condition that suppresses the discretization errors is to constrain the camera motion such that [5]:

$$\dot{\underline{P}} = 0, \text{ and } \dot{\underline{p}}_0 = 0, \forall t \quad (7)$$

These constraints mean that a **fixation task** is required. More precisely, the primitive must constantly appear at the same position in the image while the camera is moving.

Furthermore, the effects of the measurement errors on the estimation depend on the position of the primitive in the image. Therefore, the camera motion has to be constrained in order to minimize the effects of these measurement errors. Such a minimization is obtained by a **focusing task** that consists in constantly observing the primitive at a particular position in the image. We will see afterwards that the visual servoing approach is able to realize such focusing and fixating tasks.

2.4 Length Estimation

Furthermore, in order to determine the length of the primitive, its vertices have to be observed in the image, which generally implies a complementary camera motion. For accuracy issues, this motion is performed in the direction of the primitive axis, at a constant range, and until one of the two endpoints of the primitive appears at the image center. Once the camera has reached its desired position, the 3D position of the corresponding end point is computed as the intersection between the primitive axis and the camera optical axis. A motion in the opposite direction is then generated to determine the position of the other endpoint. Such a camera motion, based on visual data, can again be performed using the visual servoing approach.

2.5 Generating Camera Motion using Visual Servoing

The *image-based visual servoing* consists in specifying a task as the regulation in the image of a set of visual features [9][12][13]. Embedding visual servoing in the task function approach [18] allows us to take advantage of general results helpful for the analysis and the synthesis of

efficient closed loop control schemes. We define a vision-based task as [9]:

$$\underline{\epsilon} = W^+ C (\underline{P} - \underline{P}_d) + (\mathbb{I}_6 - W^+ W) \underline{\epsilon}_2 \quad (8)$$

where :

- \underline{P}_d is the desired value of the selected visual features (such as the primitive appears at the position in the image specified by the focusing task);
- \underline{P} is their current value, measured from the image at each iteration of the control law;
- C is called combination matrix and is defined as $C = W L_{\underline{P}}^{T+}(\underline{P}, \hat{\underline{P}}_0)$, where W is defined as a full rank matrix such that $\text{Ker } W = \text{Ker } L_{\underline{P}}^T$, and where parameters $\hat{\underline{P}}_0$, involved in the pseudo inverse of the interaction matrix $L_{\underline{P}}^T$, are estimated on-line using our 3D structure estimation method.
- $\underline{\epsilon}_2$ is a secondary task which allows the camera to move along a desired trajectory in order to realize the fixation task that suppresses the discretization error or the length estimation task.
- W^+ and $\mathbb{I}_6 - W^+ W$ are two projection operators which guarantee that the camera motion due to the secondary task is compatible with the regulation of \underline{P} to \underline{P}_d .

A general control scheme aimed at minimizing the task function $\underline{\epsilon}$ is described in [18]. We here only give the simplified control law presented in [9], which computes the camera velocity T_c given as input to the robot controller and makes $\underline{\epsilon}$ exponentially decrease:

$$T_c = -\lambda \underline{\epsilon} - (\mathbb{I}_6 - W^+ W) \frac{\partial \underline{\epsilon}_2}{\partial t} \quad (9)$$

where $\lambda > 0$ is a proportional coefficient involved in the exponential convergence of $\underline{\epsilon}$ and the term $(\mathbb{I}_6 - W^+ W) \frac{\partial \underline{\epsilon}_2}{\partial t}$ is tied to the generation of a non zero camera motion when the vision-based task is realized.

2.6 Results: the case of a Cylinder

The whole application presented in this paper has been implemented on an experimental testbed composed of a CCD camera mounted on the end effector of a six degrees of freedom cartesian robot.

Optimal estimation We first use the presented 3D reconstruction method to estimate the parameters of a cylinder (see Fig 2). More details about this derivation can be found in [5]. In order to obtain a non-biased and robust estimation, the cylinder must always appear centered ($\rho_1 = -\rho_2$) and horizontal ($\theta_1 = \theta_2 = \frac{\pi}{2}$) or vertical ($\theta_1 = \theta_2 = 0$) in the image sequence during the camera motion (which here consists in turning around the cylinder). Fig. 2.a represents the initial image acquired by the camera and the selected cylinder. Fig. 2.b contains the image acquired by the camera after the convergence of the visual servoing task.

Fig. 3 describes the evolution of the estimation of the parameters of the cylinder displayed in Fig.2. Fig. 3.a

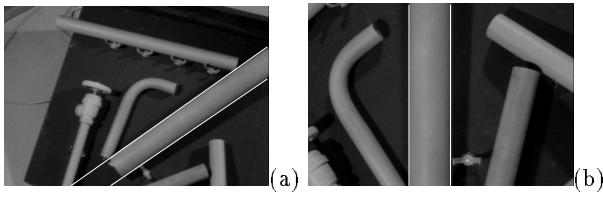


Figure 2: Position of the cylinder in the image before and after the focusing task

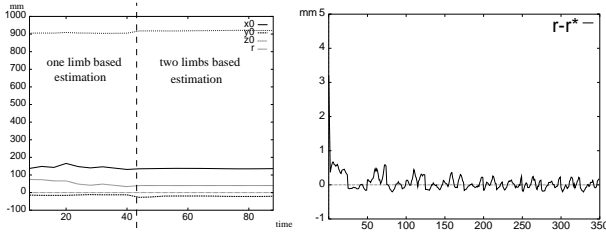


Figure 3: Estimation of the parameters of a cylinder in the camera frame

shows its radius r and the coordinates x_0, y_0, z_0 of a point of its axis. It is divided into two parts: the estimation based on one limb and, then, the estimation based on the two limbs of the cylinder. This second estimation is far better than the first one. Let us note that the cylinder radius \hat{r} is determined with an accuracy less than 0.5 mm whereas the camera is one meter away from the cylinder (and even less than 0.1 mm with good lighting conditions). Fig. 3.b reports the error between the true value of the radius and its estimated value (*i.e.*, $r_i - r^*$) using the two limbs-based estimation. As far as depth \hat{z}_0 is concerned, the standard deviation σ_{z_0} is less than 2.5 mm (that is 0.25%).

Stability and robustness The experiment of the cylinder structure estimation has been carried out fifty times from different initial camera locations in order to ensure that the estimation of the cylinder parameters is really stable and robust. The result of the experiment is depicted

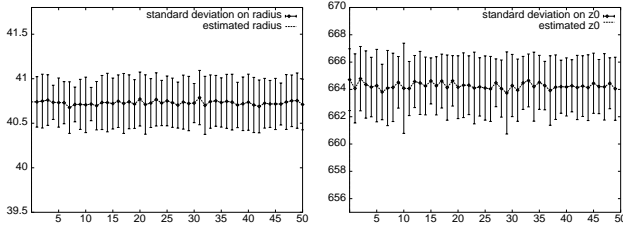


Figure 4: Stability tests (a) estimated radius (in mm) (b) estimated z_0 (in mm)

on Fig. 4 where the radius \hat{r} and depth z_0 are reported. For each one of the 50 experiments, this graph shows on its right the value of the final estimated radius \hat{r} , and, on its left the estimated depth \hat{z}_0 . Their standard deviation $\hat{\sigma}_{\hat{r}}$ and $\hat{\sigma}_{\hat{z}_0}$ are also reported. Each time, the measured error $\hat{r} - r^*$ is less than 0.1 mm and the standard deviation of all the estimations (*i.e.*, $\sigma_{\hat{r}}$) is around 0.02 mm

(resp. $\sigma_{\hat{z}_0} = 0.23$ mm). These results underline the fact that our estimation algorithm is particularly robust, stable and accurate.

3 Perception Strategies

We are now interested in investigating the problem of recovering a precise description of a 3D scene containing several objects using the visual reconstruction scheme presented above. As already stated, this scheme involves focusing on and fixating at the considered primitive in the scene. This can be done on only one primitive at a time, hence reconstructions have to be performed in sequence for each primitive of the scene. For doing that, a database containing 2D visual data is first created. With this database, a process selects a primitive, and after the recognition process which will be describe afterwards, an optimal estimation of its 3D structure is performed. After each estimation of a primitive, an exploration process is required to determine the next selection.

3.1 A Maximum Likelihood Ratio Test for Primitive Recognition

The only information we initially have on the considered scene is composed by the set of 2D segments observed by the camera at its initial position. We assume that these segments correspond to the projection in the image of either a limb of a cylinder, either a 3D segment. Since the structure estimation method is specific to each kind of primitives, a preliminary recognition process is required. In order to obtain a robust criterion, we have developed the following method:

To determine the nature of the observed primitive, we first assume that it is a cylinder, and a one limb-based estimation is performed. When this estimation is done, two competing hypotheses can be acting, respectively:

- H_0 : the observed primitive is a straight line. This hypothesis implies that we should find a radius r close to 0 ;
- H_1 : the observed primitive is a cylinder. This hypothesis implies that we should find $r = r_1$ with $r_1 > 0$;

A maximum likelihood ratio test is used to determine which one of these two hypotheses is the right one. Let us denote L_0 and L_1 the likelihood functions associated with hypothesis H_0 and H_1 . Assuming that the cylinder radius follows a Gaussian law of mean r and variance σ^2 , we obtain after N estimations $r_i, i = 1 \dots N$:

$$L_0 = \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{N}{2}} e^{-\sum_{i=1}^N \frac{r_i^2}{2\sigma^2}} \text{ and } L_1 = \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{N}{2}} e^{-\sum_{i=1}^N \frac{(r_i - r_1)^2}{2\sigma^2}} \quad (10)$$

The likelihood ratio ξ is given by $\xi = \log \frac{L_1}{L_0}$. Substituting for expressions given in (10) in this equation leads to:

$$\xi = -\frac{1}{2\sigma^2} \left(\sum_{i=1}^N (r_i - r_1)^2 - \sum_{i=1}^N r_i^2 \right) \quad (11)$$

The resulting criterion for determining the nature of the primitive can be stated as follows:

$$\max_{r_1} \xi \geq \zeta$$

where ζ is a predetermined threshold. The optimal parameter \hat{r}_1 must satisfy the relation $\frac{\partial \xi}{\partial r_1} = 0$, which leads to $\hat{r}_1 = \bar{r}$ where $\bar{r} = \frac{1}{N} \sum_{i=1}^N r_i$ is the mean value of the estimated r_i . Using this relation, ξ can finally be expressed in the simple form:

$$\xi = \frac{N\bar{r}^2}{2\sigma^2} \quad (12)$$

Clearly, hypothesis H_1 (cylinder) is selected versus hypothesis H_0 (segment) if the obtained value for likelihood ratio ξ is greater than ζ (this threshold can be easily determined by experiment). Indeed, when the primitive is a segment, the reconstruction process using one limb gives a low radius, with a very high variance (leading to a small value of ξ). On the other hand, when the primitive is a cylinder, the estimated radius is close to its real value and its variance is small (leading to a high value of ξ).

3.2 Basic step in locating objects

We assume that the scene is only composed of polyhedral objects and cylinders, so that the contours of all the objects projected in the image plane form a set of segments. A fundamental step in the scene reconstruction process is to build 2D databases composed of these segments. The database is simply obtained by extracting the edges in the image with a Shen Castan filter, and applying a Hough transform on the edge image which computes the equation of the different segments. We denote these database ω_{ϕ_t} , where ϕ_t is the corresponding camera location.

The image processing algorithm we use during the optimal estimation allows us to track a limited number of segment at video rate. Thus, for real time issue, we cannot create a database at each iteration of the estimation process. So, databases are created after each optimal reconstruction. They are used for the selection of the next considered segment and for local exploration.

An other 2D database denoted Ω_{Φ} will be used. This database contains all the unestimated segment previously observed and the camera positions ϕ_t from which they have been observed. In fact, $\Omega_{\Phi_t} = \{(\mathcal{S}_i, \phi_k), i = 1 \dots N, k \in [0, t]\}$ where $\mathcal{S}_i = (\rho_i, \theta_i)$ represents a 2D segment, $\Phi_t = \bigcup_i \phi_t$ and N is the number of untreated segments.

3.3 Visual Strategies for Incremental Exploration

Guided by visual events and the estimated partial map of the environment, we can construct simple camera control strategies, which move the camera towards new viewpoints. This exploration process can be handled at two levels:

- When a segment corresponding to a new primitive appears in the field of view of the camera, or has been previously observed from an other viewpoint, we do

not need to compute explicitly new viewpoints. This level is called **local exploration**.

- When all the 2D segments previously observed have been treated, a more complex strategy has to be implemented in order to focus on a part of the 3D space which has not been already observed. We called this kind of exploration **global exploration**.

The first step in an exploration algorithm is to determine which primitives, observed from a certain viewpoint, have been previously estimated, and which segments correspond to a non estimated 3D primitives.

3.3.1 Primitive Projection and Matching Algorithm.

One of the aspect raised by scene exploration is the matching between 3D information included into the 3D map of the scene, and the 2D features present in the 2D current database. In our particular case, due to the nature of the considered primitives (3D segments and cylinders), the projection of the 3D map into the image plane is composed of 2D segments. After a projection of the 3D map into the image using the perspective projection equation, a simple matching algorithm is performed in order to determine which segments of the 2D database have been previously estimated. We have here:

- to compute, from the parameters \underline{p} of a primitive \mathcal{P}_s in 3D space, the parameters \underline{p} of its projection in the image plane.
- to use a matching algorithm which finds, in a given 2D database, the 2D segments corresponding to the projection in the image of the segment and cylinders limbs of the 3D map.

The result of the matching algorithm is shown on Fig. 5. Fig. 5a shows an image acquired by the camera, with the 2D database superimposed. Fig. 5b shows the current state of the 3D map (in that case, only the cylinder and one 3D segment have been estimated). Fig 5c shows the result of the projection, depicted by the grey lines, and the matching algorithm (the dashed lines correspond to the projection in the image of the primitives previously estimated).

3.3.2 Local Exploration algorithm.

The exploration algorithm described in this section deals with an incremental exploration strategy. The approach we use in a first time is local. Indeed, in order to choose the next primitive to be estimated, we use only current visual informations, and the 3D map previously collected.

0) Initialization. We consider that the camera is located in ϕ_0 . A local database ω_{ϕ_0} is acquired. We do not have any information on the parameters of the corresponding 3D primitives. Therefore the 3D map of the scene is initially empty. Thus, initially, $\Omega_{\Phi} = \omega_{\phi_0} = \{(\mathcal{S}_i, \phi_0), i = 1 \dots n\}$ and $\Phi = \phi_0$. We extract from Ω_{Φ} a segment \mathcal{S} to be estimated. In fact, we choose the segment \mathcal{S} which is the nearest from the image position corresponding to the

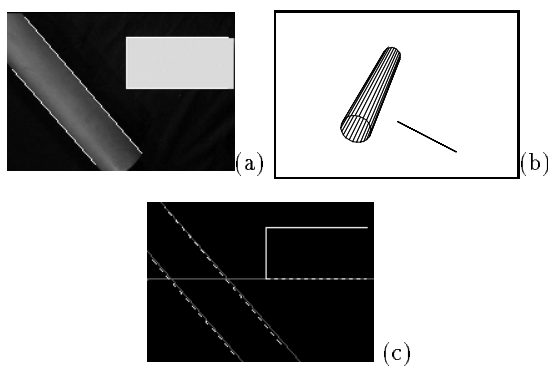


Figure 5: Projection of a 3D map on a 2D database and Matching (a) image and 2D database superimposed (b) 3D view of the current reconstructed map (c) 3D map projection

focusing task (horizontal or vertical and centered in the image, *i.e.* $\theta = \frac{k\pi}{2}$ and $\rho = 0$).

1) Optimal estimation and 3D map creation. Based on \mathcal{S} an estimation is performed, including the recognition process and the optimal estimation. The obtained parameters \hat{p} of the primitive are introduced into the 3D global map of the scene (see Fig. 5b).

2) Local and global 2D database generation. After the optimal estimation, because of the camera motion implied by this process, the camera is located in ϕ'_t . A new local database $\omega_{\phi'_t}$ corresponding to this position is constructed. Then, from the 3D map and this new database, using the matching algorithm, all the matched segments are suppressed from the local database $\omega_{\phi'_t}$ (see Fig 5c) which is merged with the global 2D database Ω_Φ (thus, $\Phi = \cup_{i=1}^t \phi_i \cup \phi'_t$, and Ω_Φ contains all the segments which have been observed from all the previous viewpoints and which have not been estimated yet).

3) Segment selection. If one (or more) unestimated segment is in the current database $\omega_{\phi'_t}$, the new camera position is chosen as $\phi_{t+1} = \phi'_t$ and a new segment \mathcal{S} is chosen. An **optimal estimation** (step 1) based on this segment is then performed. In the case where several unestimated segments are in the current database a choice has to be performed in order to select the next chosen segment. Using the 2D database ω_ϕ and the current 3D map, a neighboring graph is computed where the nodes are the junctions between segments and the vertices represent the state of the segment *i.e.*, treated (T) or untreated (U) (see Fig 6b). Using this graph, we look for an unestimated segment connex to the last estimated one. If such a segment exists, it is selected (see Fig. 6b). Otherwise, we choose the untreated segment the nearest from optimal position for its optimal 3D estimation. We iterate the steps **estimation, database creation and selection** until one of the segments present in the current database $\omega_{\phi'_t}$ has not been estimated.

Backtracking. If all the segments of $\omega_{\phi'_t}$ have been con-

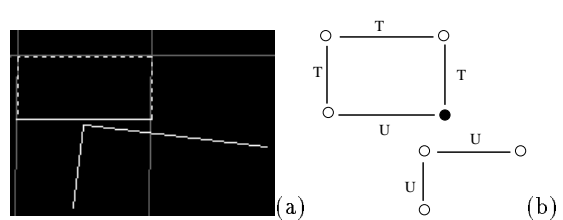


Figure 6: (a) 2D database (b) neighboring graph

sidered and if one or more of the 2D segments previously observed have not been estimated (*i.e.*, $\omega_{\phi'_t}$ empty and Ω_Φ not empty), we look in the global database for the couple (\mathcal{S}, ϕ_k) , for which the distance between the current camera location ϕ'_t and the location ϕ_k (from which the segment \mathcal{S} has been observed) is minimal. Then, the camera moves to the position ϕ_k (thus, $\phi_{t+1} = \phi_k$). An optimal estimation (step 1) is then performed. Finally, if Ω_Φ is empty (*i.e.*, all the 2D segments observed from any previous camera positions have been treated), a new viewpoint must be found. A **global exploration** is thus necessary.

An important feature of this algorithm is its ability to easily determine the next primitive to be estimated without any knowledge or assumption on the nature of the scene (localization or spatial relation between objects). Furthermore, we do not have to define complex planning strategies. Our strategy can be defined as an on-line strategy VS an off-line strategy where a plan must be previously computed. Since this exploration strategy is local, it allows the camera to minimize its displacement between two successive viewpoints. Let us however note that some objects or some complex scenes can not be completely recovered using our method. If the composition of simple primitives, such as polygons, has been studied (see experimental results in Section 3.5), more complex combinations raise new problems. An object can be occluded by another one (or by itself) or may not be observed from the different viewpoints. Furthermore, the end effector of a robot cannot move at any 3D location due to mechanical constraints such as the robot joint limits. The local strategy presented here does not solve all these problems, even if it reduces most of them (*e.g.*, the viewpoints computing problem which is solved in part by the incremental strategy).

3.3.3 Exploration Probes.

To cope with these problems, global probing strategies have to be developed. Once all the 3D parameters of the primitives observed during the local exploration process have been computed, a global exploration process must be done in order to collect more data. This problem is raised by the fact that we must make sure that the whole scene has been reconstructed. Most of previous works dealing with sensor placement assume that a complete model of the scene is known [8][21]. Simple strategies have been presented by Wixson [25] where simulation results for the exploration of a 2D world using 1D sensor are described. Some of these strategies can be extended to 3D world

(this is the case for the planetarium algorithm [7] or the occlusion-based strategy [16]). Future work will be devoted to determine viewpoints able to bring a new 2D database on which will be performed the complete estimation process. Such viewpoints will be determined using the previously estimated 3D map environment and the part of the 3D space which has not been already observed. Constraints of minimizing the number of viewpoints and the distance between two viewpoints will also be considered.

3.4 A Hierarchical Parallel Automaton as Controller

We present in this section the specification of the proposed sequencing which is stated in terms of a hierarchical parallel automaton [15]. This kind of complex robotics strategy involves the use of several subsystems (such as the different tasks described in the previous section). Achieving the complete operation requires a dynamic scheduling of these elementary subsystems. An object oriented approach, based the ORCAD system, has been presented to model such a controller in [20]. Other approaches formalize reactive behaviors of vision-guided robot with Discrete Event Systems (DES) [3][14].

The integration of our method is based on the definition of a hierarchical parallel automaton. Using automata

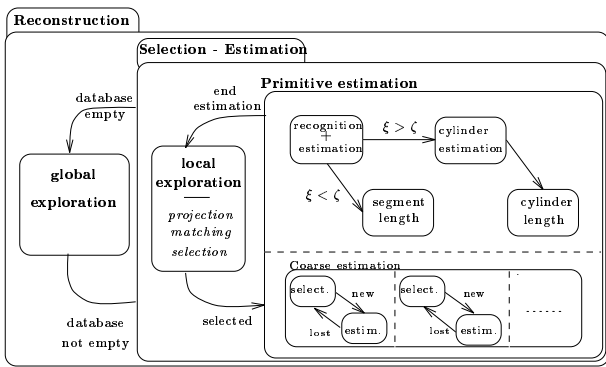


Figure 7: Hierarchical parallel automaton for the application.

remains in the DES framework and enables us to use the same formal tools that the DES. Furthermore, it allows us to specify combinations of tasks. Indeed, we can combine the effects of several tasks executed in parallel (for example, a coarse estimation of some primitives performed in parallel with the optimal estimation of another primitive). The automaton is able to connect up the different stages of the reconstruction process: selection, focusing, optimal estimation of the selected primitive and concurrently, coarse estimations. Each state of our automaton is associated with a certain task such as the creation or the update of the databases, the structure estimation process, the camera motion control using visual servoing, etc (see Fig. 7). The transitions between the states are discrete events and are function of the image data, the value of the estimated parameters of the primitives, and the state of the database.

3.5 Results: scene reconstruction

The example reported here (see Fig. 8.a) deals with a scene composed of a cylinder (whose radius is 40.6 mm), a triangle, and two rectangular polygons. The cylinder and the small rectangular polygon lie in the same plane, the triangle and the other polygon lie in two other different planes. In Fig. 8.b is displayed the initial image acquired by the camera. Note that the whole scene is not in the camera field of view for that position.

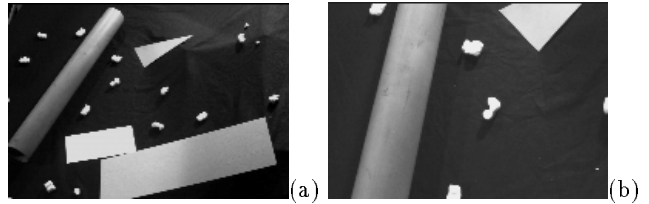


Figure 8: (a) External view of the scene (b) Scene observed from the initial position of the robot.

The parameters of the cylinder have been estimated with the same accuracy that in the experiment described above. Fig. 9 shows 3D views of the reconstructed scene.

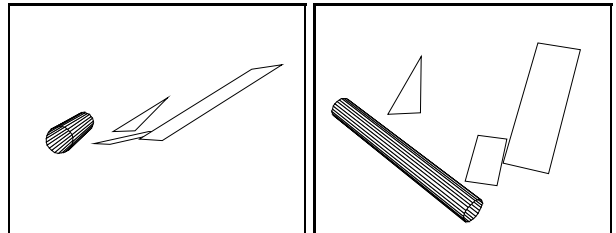


Figure 9: View of the reconstructed scene

The results of the reconstruction of another scene are depicted on Fig. 10. In that case the main difficulty is to make the correct choice on the nature of the different primitives (especially in the case of the plinth on the left of the image which look like a cylinder in the image). Each time, the right choice has been done. This allows us to demonstrate the robustness of the maximum likelihood ratio test we have presented in Section 3.1.

When all the 3D segments of the scene have been reconstructed, it is interesting to group these segments into polygons. In a first time, we look for closed strings of coplanar 3D segments. At the end of this process, we have a set of strings of coplanar segments corresponding to polygons and some isolated segments or unclosed strings. Next step is devoted to transform the closed strings of segments in a list of coplanar points: the vertices of the polygon. The equation of the carrier plane can easily be computed by solving a linear system using a least squares method. Then, the segments are projected on this plane. The coordinates of the vertices are obtained with computing the intersection of two neighbor segments. This position is obtained with an accuracy of 1 mm.

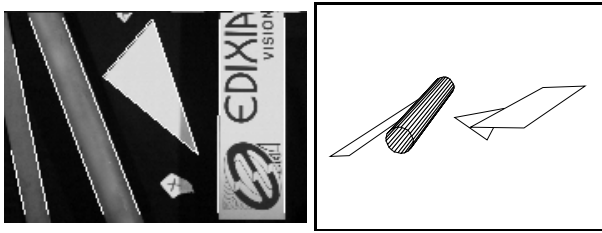


Figure 10: Another scene observed from the initial position of the robot and a view of the reconstructed scene

4 Conclusion

In this paper, we have proposed a method for 3D environment perception using a sequence of images acquired by a mobile camera. We have described a reconstruction process which provides an accurate estimation of the parameters of a geometrical primitive. As this method is based on peculiar camera motions, perceptual strategies able to appropriately perform a succession of such optimal individual primitive reconstruction have been proposed in order to to recover the complete spatial structure of complex scene. Finally, experiments carried out on a robotic cell have proved the validity of our approach (very accurate, stable and robust results, simple but efficient local exploration algorithms) but have also shown its limitations: the constraints on the camera motion, which are necessary to obtain precise results, imply the sequencing of visual estimation and we cannot perform several optimal estimations in parallel. Current work is devoted to the development of global exploration strategies in order to ensure the completeness of the reconstruction of a whole scene.

References

- [1] G. Adiv. – Inherent ambiguities in recovering 3D motion and structure from a noisy flow field. – *IEEE Trans. on PAMI*, 11(5):477–489, May 1989.
- [2] Y. Aloimonos. – Purposive and qualitative active vision. – *ICPR*, pages 346–360, New Jersey, 1990.
- [3] Y. Aloimonos, E. Rivin, and L. Huang. – Designing Visual Systems: Purposive Navigation. – In *Active Perception*, Aloimonos Y. Editor, pages 47–102. Lawrence Erlbaum Assoc., publishers, Hillsdale, NJ, 1993.
- [4] R. Bajcsy. – Active perception. – *Proc. of the IEEE*, 76(8):996–1005, August 1988.
- [5] F. Chaumette, S. Boukir, P. Bouthemy, and D. Juvin. – Optimal estimation of 3D structures using visual servoing. – *CVPR'94*, pages 347–354, Seattle, June 1994.
- [6] C. Chien and J.K. Aggarwal. – Model construction and shape recognition from occluding contour. – *IEEE Trans. on PAMI*, 11(4):372–389, February 1989.
- [7] C. Connolly. – The determination of next best views. – *IEEE Int. Conf. on Robotics and Automation*, pages 432–435, St Louis, Missouri, March 1985.
- [8] C.K. Cowan and P.D. Kovesi. – Automatic sensor placement from vision task requirements. – *IEEE Trans. on PAMI*, 10(3):407–416, May 1988.
- [9] B. Espiau, F. Chaumette, and P. Rives. – A new approach to visual servoing in robotics. – *IEEE Trans. on Robotics and Automation*, 8(3):313–326, June 1992.
- [10] B. Espiau and P. Rives. – Closed-loop recursive estimation of 3D features for a mobile vision system. – *IEEE Int. Conf. on Robotics and Automation*, pages 1436–1443, Raleigh, North Carolina, April 1987.
- [11] O. Faugeras. – *Three-dimensionnal computer vision: a geometric viewpoint*. – MIT press, 1993.
- [12] J.T Feddema, C.S.G. Lee, and O.R. Mitchell. – Weighted selection of image features for resolved rate visual feedback control. – *IEEE Trans. on Robotics and Automation*, 7(1):31–47, February 1991.
- [13] K. Hashimoto, editor. – *Visual Servoing : Real Time Control of Robot manipulators based on visual sensory feedback*. – World Scientific Series in Robotics and Automated Systems, Vol 7, World Scientific Press, Singapore, 1993.
- [14] J. Košeká, H. Christensen, and R. Bajcsy. – Discrete Event Modeling of Visually Guided Behaviors. – *newblock International Journal of Computer Vision*, 14(2):179–191, March 1995.
- [15] E. Marchand, F. Chaumette, and E. Rutten. – Real time active visual reconstruction using the synchronous paradigm. – *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'95*, Pittsburgh, USA, August 1995.
- [16] J. Maver and R. Bajcsy. – Occlusions as a guide for planning the next view. – *IEEE Trans. on PAMI*, 15(5):417–433, May 1993.
- [17] R.D. Rimey and C. Brown. – Control of selective perception using bayes nets and decision theory. – *International Journal of Computer Vision*, 12(2/3):173–207, April 1994.
- [18] C. Samson, B. Espiau, and M. Le Borgne. – *Robot Control: the Task Function Approach*. – Clarendon Press, Oxford, England, 1991.
- [19] G. Sandini and M. Tistarelli. – Active tracking strategy for monocular depth inference over multiple frames. – *IEEE Trans. on PAMI*, 12(1):13–27, January 1990.
- [20] D. Simon, B. Espiau, E. Castillo, and K. Kapellos. – Computer-aided design of a generic robot controller handling reactivity and real-time controller issues. – *IEEE Trans. on Control Systems Technology*, 1(4):213–229, December 1993.
- [21] K. Tarabanis, R. Tsai, and P.K. Allen. – The MVP sensor planning system for robotic vision tasks. – *IEEE Trans. on Robotics and Automation*, 11(1):72–85, February 1995.
- [22] B. Triggs and C. Laugier. – Automatic camera placement for robot vision. – *IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japon, May 1995.
- [23] A.M. Waxman, B.K. Parsi, and M. Subbarao. – Closed-form solutions to image flow equations for 3D structure and motion. – *International Journal of Computer Vision*, 1(3):239–258, October 1987.
- [24] J. Weng, T.S. Huang, and N. Ahuja. – Estimation and structure from line matches: Performance obtained and beyond. – *ICPR'90*, pages 168–172, June 1990.
- [25] L.E. Wixson. – Viewpoint selection for visual search. – *CVPR'94*, pages 800–805, Seattle, USA, June 1994.