Visual Data Fusion : Application to Objects Localization and Exploration

Grégory Flandin, François Chaumette

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Visual Data Fusion: Application to Objects Localization and Exploration

Grégory Flandin — François Chaumette

N° 4168

Avril 2001

THÈME 3

ISSN 0249-6399 ISRN INRIA/RR--4168--FR+ENG

Visual Data Fusion: Application to Objects Localization and Exploration

Grégory Flandin, François Chaumette

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Vista

Rapport de recherche n° 4168 — Avril 2001 — 39 pages

Abstract: Visual sensors provide exclusively uncertain and partial knowledge of a scene. In this report, we present a suitable scene knowledge representation that makes integration and fusion of new, uncertain and partial sensor measures possible. It is based on a mixture of stochastic and set membership models. We consider that, for a large class of applications, an approximated representation is sufficient to build a preliminary map of the scene. Our approximation mainly results in ellipsoidal calculus by means of a normal assumption for stochastic laws and ellipsoidal over or inner bounding for uniform laws. With these approximations, we coarsely model objects by their including ellipsoid. Then we build an efficient estimation process integrating visual data online in order to refine the location and approximated shape of the objects. Based on this estimation scheme, we perform online and optimal exploratory motions for the camera.

Key-words: active vision, 3D modeling, reconstruction, data fusion, autonomous exploration

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 Rennes Cedex (France)
Téléphone : +33 2 99 84 71 00 — Télécopie : +33 2 99 84 71 71

Fusion d'informations visuelles: application à la localisation et l'exploration

Résumé : Les capteurs visuels fournissent une connaissance partielle et incertaine de la scène observée. Dans ce rapport, nous présentons un modèle de représentation de la connaissance qui permet la comparaison et la fusion de données incertaines et partielles. Il est basé sur un mélange de modèles stochastique et à erreur bornée. En considérant une hypothèse gaussienne des incertitudes et une approximation ellipsoïdale des supports de l'erreur, nous modélisons les objets par leur ellipsoïde englobant. Nous développons ensuite un processus d'estimation efficace intégrant les données visuelles. Celui-ci permet d'estimer en ligne la position et la forme approchée des objets. Enfin, nous élaborons un processus d'exploration optimale en temps réel.

Mots-clés : vision active, modélisation 3D, reconstruction, fusion de données, exploration autonome

1 Overview

Whatever the application, a main issue for automated visual systems is to model the environment: it must have a suitable knowledge representation in order to perform efficiently the assigned task. In the context of robot vision, most papers deal with 3D reconstruction and focus on modeling accuracy. Classically, this is done either considering geometric objects (in that case, techniques are based on primitive reconstruction [7, 16]) or representing the shapes thanks to meshes [3] or using an exhausting voxel representation of the scene (see [21] for a recent survey of methods involving sample representations), eventually reducing the complexity by means of hierarchical techniques like octrees [10, 19, 22]. But, for several kinds of applications such as path planning, obstacle avoidance or exploration, only a preliminary 3D map of the scene is sufficient. Moreover, very fast algorithms are requested to perform online computation. This motivated our work about coarse model estimation. This notion was previously studied by Marr [17] and Ferrie [8]. Coarse models are strongly related to the assigned task so that we need to introduce an other concept concerning the kind of object we treat of. It is not restrictive at all to consider that any scene can be partitioned in coherent groups or parts of objects. The term coherent should be understood in the sense that the group or part can be significantly described by a single including volume according to the assigned task. Typically, a set of close objects is coherent when observed from a sufficiently far distance but is incoherent when observed closely enough to treat each object as a coherent one. Even an isolated object may appear incoherent when constituted of inhomogeneous parts. Each part should be described individually when necessary. By misuse of language we will simply call "object" each coherent group or part of physical objects. This paper is dedicated to the description, the estimation and the refinement of objects' including volume. This volume is defined by an ellipsoidal envelope. Thus, if describing a strongly concave physical object by an ellipsoid appears obviously incoherent for the assigned task, this object will need to be partitioned in coherent parts. However, this point is out the scope of this paper. Besides, the image processing comes down to the extraction of an ellipse including the segmented mask of the object in the image. As a consequence, our technique can theoretically apply to any kind of object provided that it can be segmented. In practice, to deal with general scenes with no constraint on the object aspect, we make the only assumption that there is a depth discontinuity at the frontier of the objects so that a motion segmentation algorithm will give the mask of the objects.

The first part of our work focuses on the description of the including volume of an object (center and envelope). The method we developed stems for the class of state estimation techniques. Typically, the problem of parameter and state estimation is approached assuming a probabilistic description of uncertainty. In order to be compared and fused, observations are expressed in a common parameter space using uncertain geometry [2, 5, 6]. But in cases where either we do not know the associated distribution or it is not intrinsically stochastic, an interesting alternative approach is to consider unknown but bounded errors. This approach, also termed set membership error description, has been pioneered by the work of Witsenhausen and Schweppe [26, 20]. But, in this method, the observation update needs the calculus of sets intersection. A computationally inexpensive way to solve the problem

is to assume that error is bounded by known ellipsoids [15]. Mixing probability and set membership theories in a unified stochastic framework, we will take advantage of both representations in order to model the center and envelope of objects (see also [11] for other work about combining statistical and set theoretic theories). This model is all the more interesting that it enables, for each point of the scene, the calculation of its probability to belong to a given object.

Once a suitable model is available, a common issue is to wonder which movements of the camera will optimally build or refine this model. In a general case, this is referred to optimal sensor planning [23]. When a coarse or fine reconstruction of the scene or objects is in view, we will speak about exploration. It is said autonomous when the scene is totally or partially unknown. In this context, previous works have adopted different points of view. In [4], Connolly describes two algorithms based on the determination of next best views. The views are represented by range images of the scene and the best one tends to eliminate the largest unseen volume. In [25], Whaite and Ferrie model the scene by superquadrics. The exploration strategy is based on uncertainty minimization and the sensor is a laser range finder. Kutulakos, Dyer and Lumelsky [12] exploit the notion of the occlusion boundary that are the points separating the visible from the occluded parts of an object. Lacroix and Chatila [13] developed motion and perception strategies in unknown outdoor environments by means of either a laser range finder or stereo cameras. A search algorithm provides an optimal path among a graph. This path is analyzed afterwards to deduce the perception tasks to perform. Let us mention the recent work of Arbel and Ferrie [1] about view point selection. Even if it deals with the quite different problem of object recognition, the approach is interesting. It is based on the off-line calculation of an entropy map which is related to ambiguity of recognition. Marchand and Chaumette [16] use active motion of a single camera to explore geometrical objects such as polygons and cylinders. The viewpoint selection is achieved minimizing a cost function. In our case where location is modeled by a gaussian distribution and shape by an ellipsoid, the exploration concept must be seen as a way to improve localization (see Figure 1). The exploration is optimal when the convergence rate of the estimated location is the best we can do. The strategy we develop thus consists in reducing uncertainty of the distribution associated with the observed object using visual data. The gaussian modeling of uncertainty and a linearization of the visual acquisition process allow us to build analytical solutions to optimal exploration.

In Section 2, we precisely describe the model of an object as a mixture of stochastic and set membership models. This model is seen as a probability density called set distribution. In Section 3, we define rules that makes propagation of a set distribution possible. These rules are applied to the propagation of visual data. Multiple images of a same object can then be compared and fused. In Section 4, we describe an estimation process for static objects which is based on camera motion. In the context of exploration, the camera motion has to be defined. With this aim in view, an optimality criterion and two associated exploratory control laws are examined in Section 5.

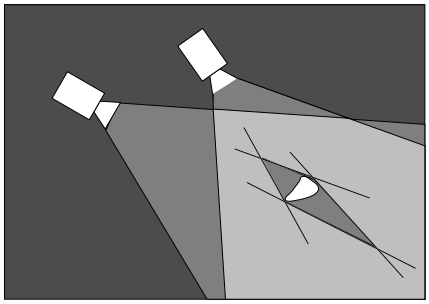


Figure 1: Exploration for coarse recontruction tends to reduce the uncertainty on the localization and shape

2 Modeling

For every object $\mathcal{O}$ belonging to a scene $\mathcal{S}$ and for every point $x \in \mathcal{S}$, we aim at calculating the probability that $x \in \mathcal{O}$ denoted $\mathcal{P}(x \in \mathcal{O})$. If we consider the 3D coordinates of a point $c \in \mathcal{O}$ as a random vector whose distribution is, for every $x \in \mathcal{S}$, $\mathcal{P}(c = x)$ denoted $\mathcal{P}_c(x)$, from this distribution, we can infer $\mathcal{P}(x \in \mathcal{O})$ since:

$$\mathcal{P}_c(x) = \mathcal{P}_c(x|x \in \mathcal{O}).\mathcal{P}(x \in \mathcal{O}) + \mathcal{P}_c(x|x \notin \mathcal{O}).\mathcal{P}(x \notin \mathcal{O})$$

$\mathcal{P}_c(x|x \notin \mathcal{O})$ is the probability that a point $c \in \mathcal{O}$ is at x knowing that $x \notin \mathcal{O}$, it is obviously null. $\mathcal{P}_c(x|x \in \mathcal{O})$ is the probability that a point $c \in \mathcal{O}$ is at x knowing that $x \in \mathcal{O}$. We naturally model it by a uniform law whose value can be calculated after normalization. Indeed, for every $\mathcal{O}$:

$$\int_{\mathcal{S}} \mathcal{P}_c(x|x \in \mathcal{O})dx = \mathcal{P}_c(x|x \in \mathcal{O})Volume(\mathcal{O}) = 1$$

$$\Rightarrow \mathcal{P}_c(x|x \in \mathcal{O}) = \frac{1}{Volume(\mathcal{O})}$$

As a consequence:

$$\mathcal{P}_c(x) = \frac{\mathcal{P}(x \in \mathcal{O})}{Volume(\mathcal{O})} \tag{1}$$

Thus, modeling $\mathcal{S}$ comes down to finding for each $\mathcal{O}$ a suitable distribution to model the density function of $\mathcal{P}_c(x)$. To do so, we break down c into the sum of a mean vector $\overline{c}$ and two independent random vectors (see Figure 2-a):

$$c = \overline{c} + p + e \tag{2}$$

From now on, we will distinguish between what we call uncertainty (modeled by p) and error (modeled by e). Uncertainty belongs to the class of random models whereas error belongs to the class of set membership models (e is uniformly distributed on a bounded set denoted $\mathcal{V}$).

In (2) p represents the uncertainty on the location of the object and the bounds on the error e define its volume. We now recall the assumptions made for future developments and whose motivations are given in introduction:

1. we first assume that p follows a zero mean normal distribution whose variance is P. This is a quite good approximation of most of the uncertainty sources such that camera localization and calibration or image processing. It also makes the propagation of the laws easier. To deal with partial inobservability, we will use the information matrix $\Sigma = P^{-1}$ instead of P. An infinity variance along the inobservability axis is then replaced by a null information. The associated distribution is:
$$\mathcal{N}(0,\Sigma) \rightsquigarrow \frac{\sqrt{det\Sigma}}{(2\pi)^{3/2}} e^{-\frac{1}{2}x^T\Sigma x}$$
2. The mathematical representation of the volume $\mathcal{V}$ needs to be simple otherwise its manipulation becomes prohibitive. We decide to approximate $\mathcal{V}$ by ellipsoids and remind that an ellipsoid is completely defined by its quadratic form: its center and a positive definite matrix E. In the sequel and by misuse of language, an ellipsoid will be denoted by its matrix E.

From previous assumptions, the global distribution associated with an object is completely defined by $\overline{c}$, Σ and E. More precisely, it is the distribution of the sum of independent variables, that is the convolution product of a uniform distribution $\mathcal{U}_E$ on $\mathcal{V}$ by a normal one $\mathcal{N}(\overline{c},\Sigma)$:
$$\mathcal{P}_{\overline{c}+p+e}(x) = \int_{\mathbf{e}} \mathcal{P}_{\overline{c}+p}(x-\mathbf{e})\mathcal{P}_e(\mathbf{e}) = (\mathcal{P}_{\overline{c}+p} * \mathcal{P}_e)(x)$$
We call this distribution a set distribution and we denote
$$\mathcal{E}(\overline{c},\Sigma,E) = \mathcal{N}(\overline{c},\Sigma) * \mathcal{U}_E$$
The corresponding shape is represented on Figure 2-b in the two dimensional case. According to equation (1), $\mathcal{P}(x \in \mathcal{O})$ is related to $\mathcal{E}(\overline{c},\Sigma,E)$ by a simple scale factor so that our work will focus on the estimation and refinement of $\overline{c}$, Σ and E.

In order to apply this model to the particular case of visual data, we identify three stages in the chain of visual observation (see Figure 3):

1. In the image, the measure is a 2D set distribution $\mathcal{E}^i(\overline{c^i},\Sigma^i,E^i)$ where $\overline{c^i}$ and E^i represent the center and matrix of the smallest outer ellipsoid including the projection of the object in the image. This projection must be extracted by segmentation algorithms (see Section 6). Besides Σ^i must account for all sources of uncertainty such that intrinsic parameters or image processing(see Section 3.2.1).
2. The process transforming the 2D visual data in a 3D observation is called back-projection. The associated 3D set distribution is denoted $\mathcal{E}^c(\overline{c^c},\Sigma^c,E^c)$. This leads us to distinguish between the **measure** related to the 2D information in the image and the **observation** which is the associated back-projection.

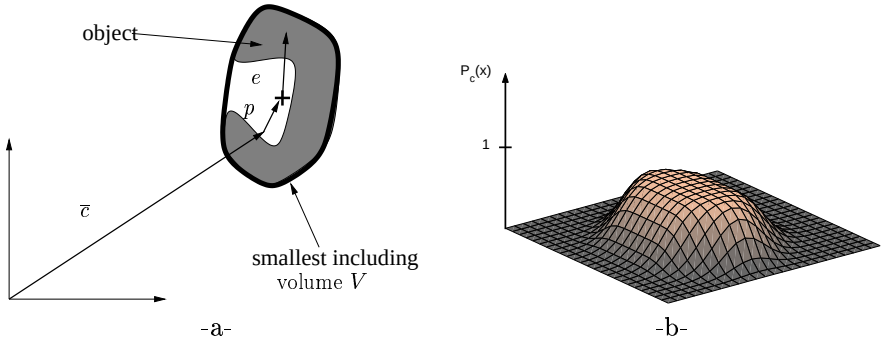


Figure 2: a- modeling definitions, b- set distribution in the two dimensional case

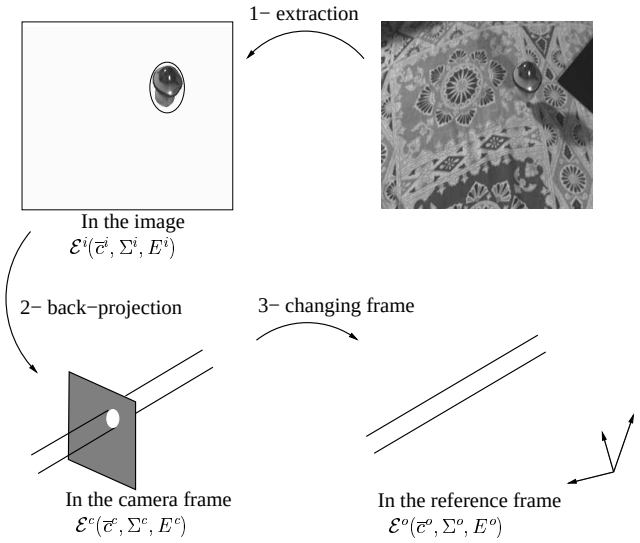


Figure 3: Successive transformations of visual measurements. 1-The projection of the object is extracted and the associated 2D set distribution is calculated. 2-The 2D set distribution is back-projected in the 3D space. 3- The 3D set distribution is expressed in the reference frame.

3. At last, we must express every observation in a common frame called the reference frame ($\mathcal{R}$). The associated observation is denoted $\mathcal{E}^o(\overline{c^o}, \Sigma^o, E^o)$.

Thanks to these successive transformations, every visual measurements can be compared and fused. In order to define the influence of such transformations on a set distribution, the next section is dedicated to the description of several propagating rules.

3 Propagating Rules

In Section 3.1, we define general rules applying to any kind of transformation. In Section 3.2, we specialize them to the case of visual transformations.

3.1 General Transformations

Rule 1 (Transformation of a set distribution)
Let c be a random vector following a set distribution $\mathcal{E}(\overline{c}, \Sigma, E)$. As a first order approximation, if we denote $J = \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}}$ the jacobian of T^{-1}, the transformed random vector $c' = T(c)$ follows a set distribution $\mathcal{E}'(\overline{c'}, \Sigma', E')$ where

$$\left\{ \begin{array}{lcl} \overline{c'} & = & T(\overline{c}) \\ \Sigma' & = & J^T \quad \Sigma \quad J \\ E' & = & J^T \quad E \quad J \end{array} \right.$$

In this rule, T^{-1} denotes the inverse transformation of T which implicitly requires T to be a diffeomorphism. The proof is then achieved approximating T with:

$$c' = T(c) \approx T(\overline{c}) + \frac{\partial T}{\partial c}\Big|_{\overline{c}} (c - \overline{c}) = T(\overline{c}) + \frac{\partial T}{\partial c}\Big|_{\overline{c}} p + \frac{\partial T}{\partial c}\Big|_{\overline{c}} e$$

Thus we show:

$$\overline{c'} = E[c'] = E[T(\overline{c}) + \frac{\partial T}{\partial c}\Big|_{\overline{c}} (c - \overline{c})] = T(\overline{c})$$

since expectation is a linear operator. Furthermore, $p' = \frac{\partial T}{\partial c}\big|_{\overline{c}} p$ follows a normal distribution whose variance is:

$$\begin{array}{rcl} P' & = & E[p'p'^T] \\ & = & E[\frac{\partial T}{\partial c}|_{\overline{c}} \quad pp^T \quad \frac{\partial T}{\partial c}|_{\overline{c}}^T] \\ & = & \frac{\partial T}{\partial c}|_{\overline{c}} \quad E[pp^T] \quad \frac{\partial T}{\partial c}|_{\overline{c}}^T \\ & = & \frac{\partial T}{\partial c}\big|_{\overline{c}} \quad P \quad \frac{\partial T}{\partial c}\big|_{\overline{c}}^T \end{array}$$

Thus, since T is a diffeomorphism:

$$\Sigma' = \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}}^T \quad \Sigma \quad \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}}$$

Finally, $e' = \frac{\partial T}{\partial c}\big|_{\overline{c}} e$ follows a uniform distribution on an ellipsoidal set whose matrix is E'. The ellipsoidal support of e is defined by:

$$\begin{array}{rcl} e^T E e = 1 & \Leftrightarrow & [\frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}} e']^T \quad E \quad [\frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}} e'] = 1 \\ & \Leftrightarrow & e'^T \ \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}}^T \quad E \quad \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}} e' = 1 \\ & \Rightarrow & E' = \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}}^T \quad E \quad \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c}} \end{array}$$

When T depends on external parameters p_e ($c' = T(c, p_e)$ where p_e is $\mathcal{N}(\overline{p_e}, P_e)$), we can linearize the transformation around $\overline{c}$ and $\overline{p_e}$ to take into account other sources of uncertainty such as the location of the camera or camera calibration errors. We obtain:

$$\begin{aligned} c' = T(c, p_e) &\approx T(\overline{c}, \overline{p_e}) + \frac{\partial T}{\partial c}\Big|_{\overline{c},\overline{p_e}}(c - \overline{c}) + \frac{\partial T}{\partial p_e}\Big|_{\overline{c},\overline{p_e}}(p_e - \overline{p_e}) \\ &= T(\overline{c}, \overline{p_e}) + \frac{\partial T}{\partial c}\Big|_{\overline{c},\overline{p_e}} p + \frac{\partial T}{\partial p_e}\Big|_{\overline{c},\overline{p_e}}(p_e - \overline{p_e}) + \frac{\partial T}{\partial c}\Big|_{\overline{c},\overline{p_e}} e \end{aligned}$$

In that case, e' remains unchanged but $p' = \frac{\partial T}{\partial c}\Big|_{\overline{c},\overline{p_e}} p + \frac{\partial T}{\partial p_e}\Big|_{\overline{c},\overline{p_e}}(p_e - \overline{p_e})$ follows a normal distribution whose variance is:

$$P' = \frac{\partial T}{\partial c}\Big|_{\overline{c},\overline{p_e}} P \frac{\partial T}{\partial c}\Big|^T_{\overline{c},\overline{p_e}} + \frac{\partial T}{\partial p_e}\Big|_{\overline{c},\overline{p_e}} P_e \frac{\partial T}{\partial p_e}\Big|^T_{\overline{c},\overline{p_e}} \tag{3}$$

thanks to independance of p and p_e. Thus, using the formula

$$(A + B)^{-1} = A^{-1}(A^{-1} + B^{-1})^{-1}B^{-1}$$

we show that:

$$\Sigma' = J_{p_e}^T {P_e}^{-1} J_{p_e} (J_{p_e}^T {P_e}^{-1} J_{p_e} + J_c^T \Sigma^c J_c)^{-1} J_c^T \Sigma^c J_c$$

$$\text{where} \quad J_c = \frac{\partial T^{-1}}{\partial c}\Big|_{\overline{c},\overline{p_e}} \quad \text{and} \quad J_{p_e} = \frac{\partial T^{-1}}{\partial p_e}\Big|_{\overline{c},\overline{p_e}}$$

Then Rule 1 becomes

Rule 2 (Transformation of a set distribution with uncertainty)
Let c be a random vector following a set distribution $\mathcal{E}(\overline{c}, \Sigma, E)$. Let p_e be $\mathcal{N}(\overline{p_e}, P_e)$. As a first order approximation, the transformed random vector $c' = T(c, p_e)$ follows a set distribution $\mathcal{E}'(\overline{c'}, \Sigma', E')$ where

$$\left\{ \begin{array}{lcl} \overline{c'} & = & T(\overline{c}, \overline{p_e}) \\ \Sigma' & = & J_{p_e}^T {P_e}^{-1} J_{p_e} (J_{p_e}^T {P_e}^{-1} J_{p_e} + J_c^T \Sigma^c J_c)^{-1} J_c^T \Sigma^c J_c \\ E' & = & J_c^T \quad E \quad J_c \end{array} \right.$$

Both previous rules require T to be a diffeomorphism. In particular, the definition set and the image set must have the same dimension. When it is not the case (by instance for perspective projection from 3D points to 2D image points), we need a rule dedicated to projection on a subspace:

Rule 3 (Projection of a set distribution on a subspace)
Let $c = (c1, c2)^T$ be a random vector following a set distribution

$$\mathcal{E}\left(\left(\begin{array}{c} \overline{c_1} \\ \overline{c_2} \end{array} \right), \left(\begin{array}{cc} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_{22} \end{array} \right), \left(\begin{array}{cc} E_{11} & E_{12} \\ E_{12}^T & E_{22} \end{array} \right)\right)$$

the projected random vector $c1$ *follows a set distribution* $\mathcal{E}'(\overline{c'}, \Sigma', E')$ *where*

$$\begin{cases} \overline{c'} & = & \overline{c_1} \\ \Sigma' & = & \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T \\ E' & = & E_{11} - E_{12}E_{22}^{-1}E_{12}^T \end{cases}$$

Proof:

The proof for Σ' directly comes from the formula of partitioned inverse:

$$\Sigma^{-1} = \begin{pmatrix} (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)^{-1} & -(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)^{-1}\Sigma_{12}\Sigma_{22}^{-1} \\ -(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)^{-1}\Sigma_{12}\Sigma_{22}^{-1} & \Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{12}^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)^{-1}\Sigma_{12}\Sigma_{22}^{-1} \end{pmatrix}$$

This is the covariance of the gaussian vector (p_1, p_2) thus the covariance of p_1 is its upper left term:

$$P' = (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)^{-1}$$

and

$$\Sigma' = (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)$$

The equation of the error bound is defined as follows:

$$\forall e = (e_1, e_2)^T \in E, f(e_1, e_2) = (e_1, e_2)E(e_1, e_2)^T = 1$$

Every point $e' = e_1$ of the projected error bound is such that the gradient of f at point (e', e_2) is parallel to the subspace of e_1. The gradient is defined by

$$\left.\frac{\partial f}{\partial (e_1, e_2)}\right|_{(e_1, e_2)} = 2 \begin{pmatrix} E_{11} & E_{12} \\ E_{12}^T & E_{22} \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}$$

thus every projected point satisfies

$$E_{12}^T e' + E_{22} e_2 = 0 \tag{4}$$

Besides

$$\begin{array}{rcl} (e', e_2)E(e', e_2)^T = 1 & \Leftrightarrow & e'^T E_{11} e' + e_2^T(E_{22}e_2 + E_{12}^T e') + e'^T E_{12} e_2 = 1 \\ & \Leftrightarrow & e'^T E_{11} e' - e'^T E_{12} E_{22}^{-1} E_{12}^T e' = 1 \text{ thanks to equation (4)} \\ & \Leftrightarrow & e'^T (E_{11} e' - E_{12} E_{22} E_{12}^T) e' = 1 \end{array}$$

which ends proof of Rule 3 □

We now specialize the previous rules to the case of visual data.

3.2 Specialized rules

3.2.1 Measure in the image

First of all, the projection of each object in the image must be extracted. We will see in Section 6 how we achieve this task in practice. Then $(\overline{c^i}, E^i)$ represents the center and matrix of the smallest outer ellipse in the image. They can be extracted thanks to algorithms like the one proposed in [24]. Σ^i must account for all sources of uncertainty that may occur in the calculus of this ellipse: errors on camera intrinsic parameters estimation and inaccuracy of image processing. To do so, we first remind the pixel/meter transformation. If we denote $c^i = (X^i, Y^i)^T$ the coordinates in meter, they are related to the coordinates in pixels $c^p = (X^p, Y^p)^T$ thanks to the following relation:

$$\begin{cases} X^i = l_x(X^p - X_c) \\ Y^i = l_y(Y^p - Y_c) \end{cases}$$

where l_x and l_y are the width and height of a pixel at a one meter focal length, (X_c, Y_c) are the coordinates (in pixels) of the principal point. If we denote $p_{int} = (X_c, Y_c, l_x, l_y)^T$ the vector of intrinsic parameters, the previous relation comes to:

$$c^i = T(c^p, p_{int})$$

if we assume that c^p is $\mathcal{N}(\overline{c^p}, P^p)$ and p_{int} is $\mathcal{N}(\overline{p_{int}}, P_{int})$ then thanks to equation (3) we can write:

$$P^i = J_{c^p} P^p J_{c^p}^T + J_{p_{int}} P_{int} J_{p_{int}}^T$$

where

$$\begin{cases} J_{c^p} = \frac{\partial T}{\partial c^p}\Big|_{\overline{c^p}, \overline{p_{int}}} = \begin{pmatrix} \overline{l_x} & 0 \\ 0 & \overline{l_y} \end{pmatrix} \\ J_{p_{int}} = \frac{\partial T}{\partial p_{int}}\Big|_{\overline{c^p}, \overline{p_{int}}} = \begin{pmatrix} -\overline{l_x} & 0 & \overline{X^p} - \overline{X_c} & 0 \\ 0 & -\overline{l_y} & 0 & \overline{Y^p} - \overline{Y_c} \end{pmatrix} \end{cases}$$

The gaussian assumption we make for intrinsic parameters can appear not very realistic since it looks nearer to a bias. But we will attach importance to the choice of a sufficiently large P_{int} so that the gaussian model can take small bias into account. We will model P_{int} and P_p by diagonal matrices, considering that the uncertainties are not correlated. Since P_p is related to image processing uncertainty, an estimate is not easy to derive. We will fix its diagonal values to sufficiently large variances.

3.2.2 Back-projection

In that part, we implicitly consider that the projected center of an ellipsoid is the center of the projected ellipse. This is theoretically wrong but the difference is very small more especially as the ellipsoid is centered in the image. This will be the case in practice thanks to a visual servoing control scheme (see Section 5). Back-projection strongly depends on the camera configuration. We study the monocular and binocular configurations.

Monocular configuration In that case, because of partial inobservability, $\mathcal{E}^c$ is degenerated. Let us denote $c^c = (x^c, y^c, z^c)^T$. To account for the inobservability of z^c, we increase the random vector c^i adding artificially the independent measure z^c whose distribution is $\mathcal{E}(\overline{z^c}, 0, 0)$. In other words, we allocate a value for z^c with a null confidence. Besides, c^c is related to (c^i, z^c) according to the following equation:

$$(x^c, y^c, z^c)^T = (z^c X^i, z^c Y^i, z^c)^T$$

Since previous transformation is a diffeomorphism, using Rule 1, we show:

Rule 4 (Back-projection of a set distribution)
The back-projection c^c of c^i follows a set distribution:

$$\left\{ \begin{array}{lcl} \overline{c^c} & = & (\overline{z^c}\,\overline{X^i}, \overline{z^c}\,\overline{Y^i}, \overline{z^c})^T \\ \Sigma^c & = & J^T \begin{pmatrix} \Sigma^i & 0 \\ 0 & 0 \end{pmatrix} J \\ E^c & = & J^T \begin{pmatrix} E^i & 0 \\ 0 & 0 \end{pmatrix} J \end{array} \right.$$

where $J = \left.\frac{\partial T^{-1}}{\partial c}\right|_{\overline{c}} = \begin{pmatrix} 1/\overline{z^c} & 0 & -\overline{X^i}/z^c \\ 0 & 1/\overline{z^c} & -\overline{Y^i}/z^c \\ 0 & 0 & 1 \end{pmatrix}$

In this rule, $\overline{z^c}$ is a priori unknown but we will see in Section 4 that it can be fixed by the previous estimate.

Contrarily, c^i is related to c^c according to:

$$(X^i, Y^i)^T = (x^c/z^c, y^c/z^c)^T$$

which is not a diffeomorphism but, using a composition of a subspace projection and a diffeomorphism (see appendix A), we show the perspective projection rule:

Rule 5 (Perspective projection of a set distribution)
If we denote

$$E^c = \begin{pmatrix} E^c_{11} & E^c_{12} \\ {E^c_{12}}^T & E^c_{22} \end{pmatrix} \text{ and } \Sigma^c = \begin{pmatrix} \Sigma^c_{11} & \Sigma^c_{12} \\ {\Sigma^c_{12}}^T & \Sigma^c_{22} \end{pmatrix}$$

where E^c_{11} and Σ^c_{11} are (2,2), E^c_{12} and Σ^c_{12} are (2,1) and E^c_{22} and Σ^c_{22} are scalar then the perspective projection c^i of c^c follows a set distribution:

$$\left\{ \begin{array}{lcl} \overline{c^i} & = & (\overline{x^c}/\overline{z^c}, \overline{y^c}/\overline{z^c})^T \\ \Sigma^i & = & J^T[\Sigma^c_{11} - \frac{\Sigma^c_{12}{\Sigma^c_{12}}^T}{\Sigma^c_{22}}]J \\ E^i & = & J^T[E^c_{11} - \frac{E^c_{12}{E^c_{12}}^T}{E^c_{22}}]J \end{array} \right. \quad \text{where } J = \begin{pmatrix} \overline{z^c} & 0 \\ 0 & \overline{z^c} \end{pmatrix}$$

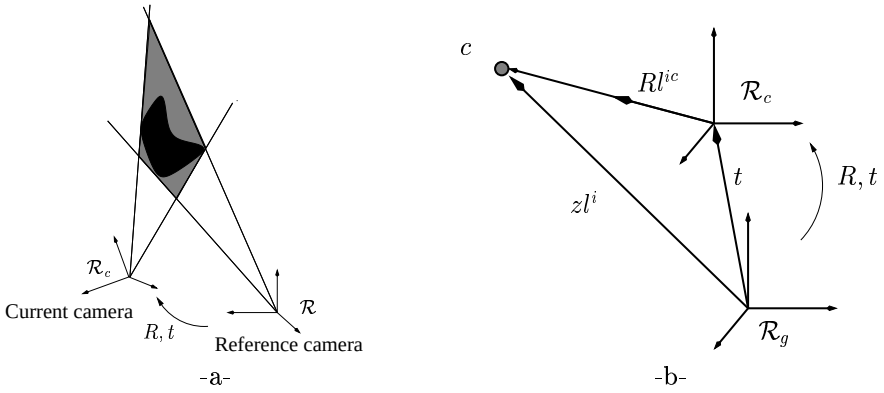


Figure 4: Initialization of the distribution with two images

Binocular configuration This configuration can be achieved either considering a stereo-vision system or using simultaneously an eye-in-hand and an eye-to-hand system. In both cases, two images of the same object are available. We call them the reference image and the current image. A point c is projected on c^i in the reference image and on c^{ic} in the current one. If we denote $l^i = (c^i, 1)$, $l^{ic} = (c^{ic}, 1)$ and (R, t) the displacement between the two cameras (see Figure 4), then:

$$\begin{cases} (c - t) \wedge Rl^{ic} &= 0 \\ c \wedge l^i &= 0 \end{cases}$$

This is a system of six equations. Four of them are independent when the cameras are not aligned with the object. As a consequence we can find a solution $c = T(R, t, c^i, c^{ic})$. For example the one resulting in the least square error (see appendix B). Thus, using Rule 1, we can infer the parameters of the associated set distribution.

3.2.3 Changing the frame

$\mathcal{E}^c$ is expressed in the camera frame $\mathcal{R}_c$. Yet all the observations must be compared in the same frame $\mathcal{R}$ also called the reference frame. The displacement parameter between $\mathcal{R}$ and $\mathcal{R}_c$ are denoted p_e. We model p_e by a gaussian noise: $p_e = \overline{p_e} + \mathcal{N}(0, P_e)$. If c^o is the coordinate vector of c^c expressed in $\mathcal{R}$, we can write:

$$c^o = \underbrace{\begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}}_{D} \begin{pmatrix} c^c \\ 1 \end{pmatrix} = T(c^c, p_e)$$

where D is related to p_e. Using Rule 2, we infer the following rule:

Rule 6 (Changing frame)
If c^o is the expression of c^c in $\mathcal{R}$ then it follows a set distribution whose parameters are:

$$\begin{cases} \overline{c^o} &= R\overline{c^c} + t \\ \Sigma^o &= J_{p_e}^T P_e^{-1} J_{p_e} (J_{p_e}^T P_e^{-1} J_{p_e} \\ &+ J_c^T \Sigma^c J_c)^{-1} J_c^T \Sigma^c J_c \\ E^o &= J_c^T \quad E^c \quad J_c \end{cases}$$

where $J_c = \left.\frac{\partial T^{-1}}{\partial c}\right|_{\overline{c},\overline{p_e}} = R^T$ *and* $J_{p_e} = \left.\frac{\partial T^{-1}}{\partial p_e}\right|_{\overline{c},\overline{p_e}}$.

For a six degrees of freedom robot, p_e accounts for the position of joints and the effector/camera transformation.

4 Estimation process

We now describe how the set distribution of an object can be estimated and refined using camera motion.

4.1 Initialization

At the first step, two images of the same object are available. In the monocular case , they are obtained by two successive positions of the camera. Using the equations of Section 3.2.2 related to the binocular configuration, we can estimate the parameters of the distribution $\mathcal{E}_0$ that will initialize the knowledge model.

4.2 Fusing new images

As shown in Figure 4-a, only two images can not provide a good estimation neither for the object volume nor for its location, especially when the view points are close. In the exploration context, a sequence of several images is available ; we must be able to take them into account in an efficient and robust way. At time k, the known a priori distribution is $\mathcal{E}_k(\overline{c_k}, \Sigma_k, E_k)$. At time $k+1$, the observation likelihood is given by $\mathcal{E}^o_{k+1}(\overline{c^o_{k+1}}, \Sigma^o_{k+1}, E^o_{k+1})$. We estimate independently the uncertainty parameters and the error bounds (see Figure 5):

4.2.1 Uncertainty distribution

If we denote $\overline{c}$ the real center of the object, at time k we a priori know that $p(\overline{c})$ is $\mathcal{N}(\overline{c_k}, \Sigma_k)$ and at time $k+1$, we observe

$$\overline{c^o_{k+1}} = \overline{c} + p^o_{k+1} \quad \text{where} \quad p^o_{k+1} \text{ is } \mathcal{N}(0, \Sigma^o_{k+1})$$

We are looking for the MAP estimate of $\overline{c}$ that is:

$$\overline{c_{k+1}} = \arg\max_{\overline{c}} p(\overline{c}|\overline{c^o_{k+1}})$$

$p(\overline{c}|\overline{c^o_{k+1}})$ can be calculated as follows:

$$\begin{aligned} p(\overline{c}|\overline{c^o_{k+1}}) &= \frac{p(\overline{c^o_{k+1}}|\overline{c})p(\overline{c})}{p(\overline{c^o_{k+1}})} = \frac{p(p^o_{k+1}=\overline{c^o_{k+1}}-\overline{c})p(\overline{c})}{p(\overline{c^o_{k+1}})} \\ &= \frac{\sqrt{\det \Sigma^o_{k+1}}\sqrt{\det \Sigma_k}}{p(\overline{c^o_{k+1}})} e^{-\frac{1}{2}(\overline{c^o_{k+1}}-\overline{c})^T \Sigma^o_{k+1}(\overline{c^o_{k+1}}-\overline{c})} e^{-\frac{1}{2}(\overline{c}-\overline{c_k})^T \Sigma^o_{k+1}(\overline{c}-\overline{c_k})} \end{aligned}$$

This expression is of the following shape:

$$p(\overline{c}|\overline{c^o_{k+1}}) = Ce^{f(\overline{c})}$$

where C is a normalization constant and

$$f(\overline{c}) = e^{-\frac{1}{2}((\overline{c^o_{k+1}}-\overline{c})^T \Sigma^o_{k+1}(\overline{c^o_{k+1}}-\overline{c})+(\overline{c}-\overline{c_k})^T \Sigma_k(\overline{c}-\overline{c_k}))}$$

Thus $p(\overline{c}|\overline{c^o_{k+1}})$ is maximal at $\overline{c_{k+1}}$ if

$$\begin{aligned} \frac{\partial C e^{f(\overline{c})}}{\partial \overline{c}}\Big|_{\overline{c_{k+1}}} = 0 \quad &\Rightarrow \quad \frac{\partial f(\overline{c})}{\partial \overline{c}}\Big|_{\overline{c_{k+1}}} = 0 \\ &\Rightarrow \quad -\Sigma^o_{k+1}(\overline{c^o_{k+1}} - \overline{c_{k+1}}) + \Sigma_k(\overline{c_{k+1}} - \overline{c_k}) = 0 \\ &\Rightarrow \quad (\Sigma^o_{k+1} + \Sigma_k)\overline{c_{k+1}} = \Sigma^o_{k+1}\overline{c^o_{k+1}} + \Sigma_k\overline{c_k} \end{aligned}$$

And finally:

$$\overline{c_{k+1}} = (\Sigma_k + \Sigma^o_{k+1})^{-1}(\Sigma_k\overline{c_k} + \Sigma^o_{k+1}\overline{c^o_{k+1}})$$

This is a mean of previous knowledge and new observation respectively weighted by the confidence (inverse of covariance) we have in them. Besides this estimation leads to an error $(\overline{c_{k+1}} - \overline{c})$ whose variance is:

$$\begin{aligned} Var(\overline{c_{k+1}} - \overline{c}) &= Var[(\Sigma_k + \Sigma^o_{k+1})^{-1}(\Sigma_k\overline{c_k} + \Sigma^o_{k+1}\overline{c^o_{k+1}}) - \overline{c}] \\ &= Var[(\Sigma_k + \Sigma^o_{k+1})^{-1}(\Sigma^o_{k+1}(\overline{c^o_{k+1}} - \overline{c}) + \Sigma_k(\overline{c_k} - \overline{c}))] \\ &= (\Sigma_k + \Sigma^o_{k+1})^{-1} \end{aligned}$$

since $Var[\overline{c_k} - \overline{c}] = \Sigma_k^{-1}$, $Var[\overline{c^o_{k+1}} - \overline{c}] = {\Sigma^o_{k+1}}^{-1}$ and both stochastic vectors are independent. All that shows that

$$\Sigma_{k+1} = \Sigma_k + \Sigma^o_{k+1}$$

4.2.2 Error bounds

The new bound on the error is given by the intersection between two ellipsoids (E_k and E^o_{k+1}) supposed to be centered at the origin. This intersection is not an ellipsoid itself. We thus need to approximate it. Two types of approximation can be performed: an outer or an inner approximation (see Figure 5-b).

4.2.2.a Outer approximation E^+_{k+1} for error bounds

Let us consider the family of ellipsoids generated by

$$(1-\lambda)E_k + \lambda E^o_{k+1} \qquad \lambda \in \Re$$

This is a family of outer ellipsoids. We have to choose λ in such a way that the ellipsoid is minimal in one sense. Classically, we can maximize either the determinant (which minimizes the volume) or the trace of this ellipsoid (see [15]). In the case of a volume criterion,the optimal value of λ has an analytical solution (see appendix C) :

$$\widehat{\lambda} = \begin{cases} \frac{-det(B)tr(AB^{-1}) - \sqrt{det(B)^2 tr(AB^{-1})^2 - det(B)det(A)tr(BA^{-1})}}{det(B)} & \text{if } det(B) > 0 \\ \frac{-det(B)tr(AB^{-1}) + \sqrt{det(B)^2 tr(AB^{-1})^2 - det(B)det(A)tr(BA^{-1})}}{det(B)} & \text{if } det(B) < 0 \end{cases}$$

where $A = E_k$ and $B = E^o_{k+1} - E_k$. This is the solution we use.

Remark: In practice, $\lambda \in [0,1]$.

- when $\lambda = 0$, the ellipsoid $\mathcal{E}^o_{k+1}$ contains the whole information. It means $E^o_{k+1} \subset E_k$.
- if $\lambda = 1$ the measure does not convey any information. This is the case either when the camera comes back on its own trajectory or when the scene has already been completely reconstructed.

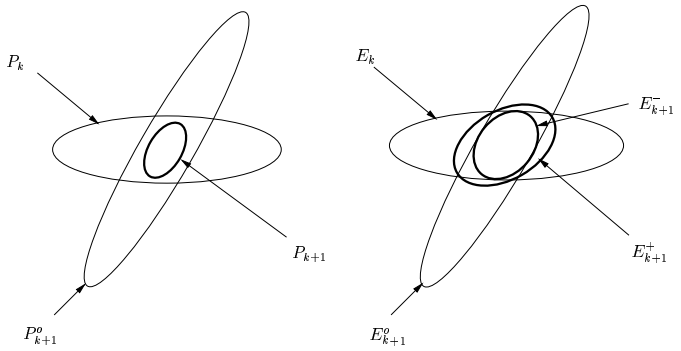
4.2.2.b Inner approximation E^-_{k+1} for error bounds

The largest inner ellipsoid can be easily calculated. Thanks to an affinity $\mathcal{A}$, we transform the work space in a parameter space for which one of the ellipsoids (e.g. $\mathcal{A}(E_k)$) is spherical. Then, for each eigenvector of $\mathcal{A}(E^o_{k+1})$ associated with the eigenvalue e_i ($i = 1..3$), we just affect to $\mathcal{A}(E^-_{k+1})$ the eigenvalue $max(1, e_i)$. Thanks to $\mathcal{A}^{-1}$, we come back to E^-_{k+1} expressed in the work space.

Comments:
Because it is very pessimistic, the use of E^+ is more robust to measurement errors than the use of E^- but the convergence rate of E^+ is very low, depending on the sample rate. The use of a medium approximation $E^- \subset E \subset E^+$ is worth considering. For future experiments, we choose a simple weighted mean between E^+ and E^-.

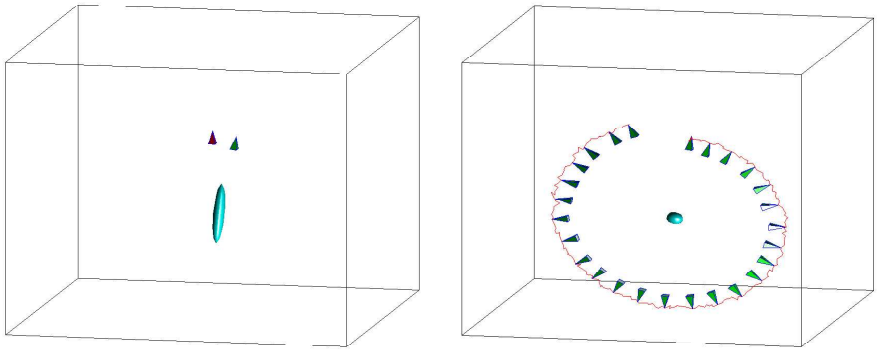
4.3 Simulation

The previous estimation process has been simulated in order to analyse its convergence rate and precision. The results we present were obtained in the context of a single camera. Simulations were computed as follows: after the initialization stage (see Figure 6-a), the virtual camera is moving with a constant speed (3cm per iteration) along a circular trajectory (see Figure 6-b). At the center of this trajectory is placed the object: a virtual sphere



-a- -b-

Figure 5: Fusing a new measure: -a- of uncertainty, -b- of error bounds



-a- -b-

Figure 6: Simulation conditions: a- initialization, b- circular trajectory

with known position ($X = -4.7cm$, $Y = 4.7cm$ and $Z = 63cm$) and radius (4 cm). The uncertainty on the camera location is a normal unbiased additive noise with a standard deviation outweighing 10 cm for translation and 5 deg for rotations. The covariance on center estimation and intrinsic parameters where chosen as follows:

$$P^p = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} \quad P_{int} = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 10^{-7} & 0 \\ 0 & 0 & 0 & 10^{-7} \end{pmatrix}$$

Successive observations refine the estimated location of the sphere and its volume over 200 iterations without any stopping criterion. Figure 7 shows convergence of the estimated

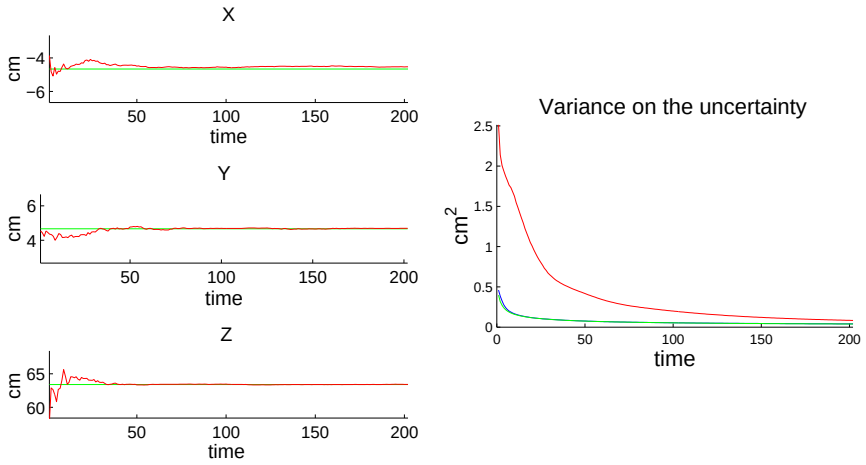


Figure 7: Estimation of the sphere location and uncertainty

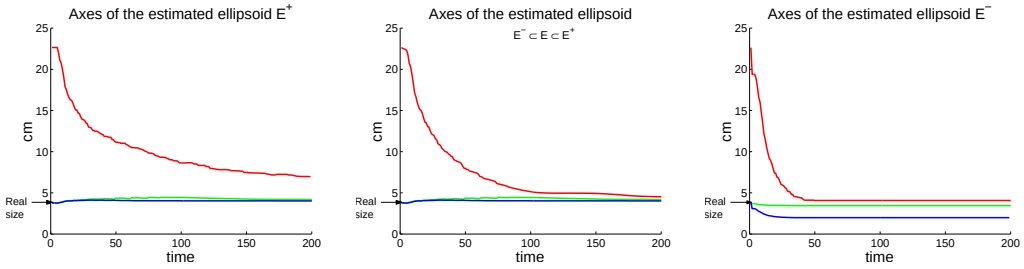


Figure 8: Estimation of the sphere axes

location to the real position, ensuring final accuracy. On the figure describing the variance on the uncertainty, we notice that the axis initially parallel to the optical axis is badly estimated at the initialization.

We have simulated the convergence of the axes for E^+, E^- and $E = 0.98E^+ + 0.02E^-$. We note that the final accuracy and convergence rate strongly depend on the used combination of inner and outer estimation (see Figure 8). The outer approximation is converging very slowly whereas the inner approximation under-estimates the including volume of the object. This issue is worth considering and should be studied in further analysis. We intuitively believe that a compromise between accuracy and convergence speed has to be performed.

5 Exploration process

We now want to identify a control law that automatically generates exploratory movements of the camera. The principle of this command is to minimize the uncertainty of the predicted a posteriori knowledge for the next iteration. We could imagine a command based on the reduction of the estimated including ellipsoid. But this strategy seems not very judicious since the real shape of the object is a priori unknown.

5.1 Predicted a posteriori information

At time k, we have deduced, from the estimation process, the knowledge $\mathcal{E}_k(\overline{c_k}, \Sigma_k, E_k)$. For notational convenience, it is expressed in the current camera frame instead of the so called reference frame. If, at time $k+1$, the predicted camera motion is (R,t), we can deduce the corresponding predicted a priori information, the predicted observation and finally the predicted a posteriori information.

Since the object is known to be static with assurance, the predicted a priori information is simply the propagation of Σ_k through a changing frame (R,t). If we assume that the motion is perfectly known, thanks to Rule 6, the associated information is $R^T\Sigma_k R$. In the absence of real measurement, the predicted observation is the propagation of the predicted a priori knowledge through projection and back-projection. If we denote

$$R^T\Sigma_k R = \begin{pmatrix} A & B \\ B^T & c \end{pmatrix}$$

where A is (2,2), B is (2,1) and c a scalar then thanks to Rule 5, the predicted measure information is

$$\widehat{\Sigma^i_{k+1}} = J_0^T A - \frac{BB^T}{c} J_0 \qquad \text{where } J_0 = \begin{pmatrix} z_{k+1} & 0 \\ 0 & z_{k+1} \end{pmatrix}$$

Thanks to Rule 4, $\widehat{\Sigma^i_{k+1}}$ corresponds to the following 3D information:

$$\widehat{\Sigma^o_{k+1}} = J_1^T \begin{pmatrix} J_0^T(A - \frac{BB^T}{c})J_0 & 0 \\ 0 & 0 \end{pmatrix} J_1 \qquad \text{where } J_1 = \begin{pmatrix} 1/z_{k+1} & 0 & -X^i_{k+1}/z_{k+1} \\ 0 & 1/z_{k+1} & -Y^i_{k+1}/z_{k+1} \\ 0 & 0 & 1 \end{pmatrix}$$

X^i_{k+1} and Y^i_{k+1} are the predicted coordinates of $\overline{c^i}$ at time $k+1$. In practice, we use a visual servoing control scheme such that the explored object is centered in the image $(\forall k, X^i_k = Y^i_k = 0)$. This is a first step to impose the visibility of the object during the exploration process. In practice, a tracking rotational motion is calculated as the one presented in [9].

Then, combining the predicted observation with the predicted a priori knowledge (see Section 4.2), we deduce the predicted a posteriori knowledge in the camera frame at time $k+1$:

$$\widehat{\Sigma_{k+1}} = R^T\Sigma_k R + J_1^T \begin{pmatrix} J_0^T(A - \frac{BB^T}{c})J_0 & 0 \\ 0 & 0 \end{pmatrix} J_1 = \begin{pmatrix} 2A - \frac{BB^T}{c} & B \\ B^T & c \end{pmatrix} \qquad (5)$$

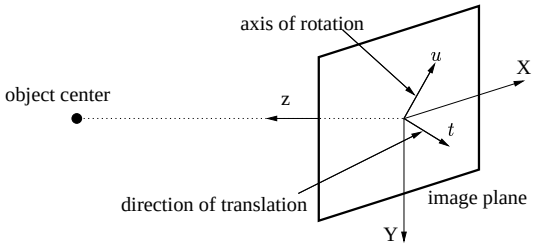


Figure 9: Definition of the exploratory control law

5.2 Exploratory control law

Motion parameters (R, t) must be calculated in such a way that $\widehat{\Sigma_{k+1}}$ is maximal in one sense. In order to introduce the idea of isotropy concerning the whole view point directions, we will attach importance to the sphericity of $\widehat{\Sigma_{k+1}}$.

We can show, in equation (5), that the depth z_{k+1} from the camera to the object does not influence the predicted information matrix. This is due to the linear approximation we made in Rule 1. As a first consequence, the optimal translation can be calculated in the image plane so that we can use the remaining degree of freedom to regulate the projected surface:

$$V_{zk} = -\frac{\lambda z_k}{2S_k}(S_k - S^*)$$

where S_k is the current object surface while S^* is the desired one (for example the initial one). An other consequence is that the direction of translational motion t is related to the axis of rotation by the equality $t = z \wedge u$ where z is the unit vector normal to the image plane (see Figure 9). As a consequence, we can define the exploratory control law either using u or using t. We now examine and compare two types of exploratory motions.

5.2.1 Locally optimal exploration

In that part the camera motion locally optimizes the increase of Σ_k and the criterion is the trace of $\widehat{\Sigma_{k+1}}$. At time $k+1$, the camera will have rotated with an angle $\alpha \geq 0$ around the unit vector $u = (u_x, u_y, 0)$. The optimal motion is defined by

$$(u_x, u_y) = argmax \quad tr[\widehat{\Sigma_{k+1}}]$$

We show in appendix D that (u_x, u_y) is the solution of a linear system which is easily computed. We also noticed that the study is correct if and only if the center of the camera is not located on the direction pointed by an eigenvector of Σ_k. Simulations will confirm that, in such a situation, the camera is in a local minimum (see appendix D for demonstration) . Besides, when Σ_k is spherical (i.e. when each eigenvalue equals the other) $tr[\widehat{\Sigma_{k+1}}]$ is constant. In that case (u_x, u_y) can be randomly chosen.

5.2.2 Best view point exploration

Now, instead of locally optimizing $\widehat{\Sigma_{k+1}}$, the best view point motion tends to reach the next best view point: the one which leads to the "biggest spherical" $\widehat{\Sigma_{k+1}}$, that is proportional to the identity matrix. Judging from equation (5), the best view point is the point at which:

$$\widehat{\Sigma_{k+1}} = \begin{pmatrix} 2A - \frac{BB^T}{c} & B \\ B^T & c \end{pmatrix} \text{ is diagonal } \Leftrightarrow B = 0 \text{ and } A \text{ diagonal}$$

$$\Leftrightarrow R^T \Sigma_k R \text{ diagonal}$$

$$\Leftrightarrow R \text{ is composed of eigenvectors of } \Sigma_k \text{ and } c \text{ is the biggest eigenvalue of } \Sigma_k$$

Both previous results show that the third column vector of R is the eigenvector of Σ_k associated with its maximal eigenvalue. Since the object is centered in the image, the center of the camera must be located on the direction pointed by this eigenvector. In other words, the next best view point is located on the eigen vector of Σ_k associated with the biggest eigenvalue that is the most informative direction. The motion vector $(V_x, V_y)_k$ must be directed to the intersection between the image plane and the biggest information axis.

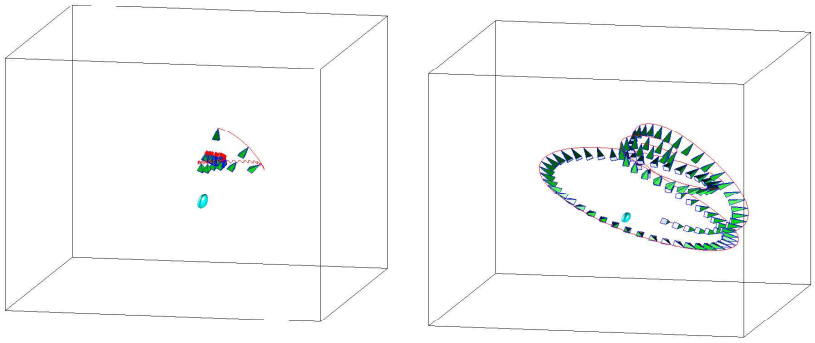
5.3 Simulation

Exploratory motions have been simulated so that we can analyze the associated trajectory. The simulation conditions are the same as described in Section 4.3 except that the trajectory is no longer circular but controlled by either the local control law or the best view point one. The local exploration (see Figure 10-a) leads to a local minimum. This is due to the biggest slope maximization induced by this technique. We can go out of such points by slightly off-centering the object in the image. Applying the criterion

$$Q = \frac{\text{Initial volume - Final volume}}{\text{Initial volume - Real volume}}$$

the local exploration resulted in a 67% reduction of the volume for a 200 iterations simulation. The best view point exploration seems to overpass such local minima (see Figure 10-b) and leads to very intuitive trajectories: the camera is spiraling around the object, from the top to its middle plane. For this simulation, the volume reduction was about 99.5%. The gain induced by the best view point exploration can be seen on Figure 11 where we compare the convergence of the axes when no exploration strategy is used (circular trajectory) to the case of the best view point strategy. In the second case, both the convergence rate and the final accuracy are better. Thanks to its simplicity and local minima avoidance, the best view point strategy appears to be a better solution to exploration.

A suitable stopping criterion is the trace of the covariance matrix which is related to the uncertainty of the model. The best view point exploration ran until the criterion reached an arbitrarily fixed to 0.002 threshold. The corresponding graph is given on Figure 12-b. As we can see on Figure 12-a, because of the local minimum, the criterion for the local strategy does not reach zero. The simulation has thus been arbitrarily stopped after 200 iterations.



-a- -b-

Figure 10: Simulated trajectory for -a- the locally optimal exploration, -b- the best view point exploration

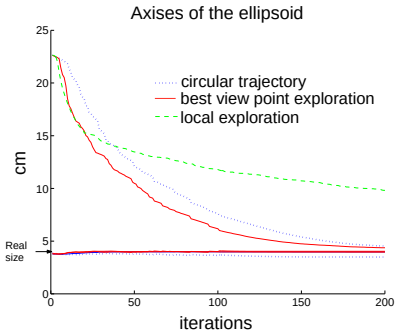


Figure 11: Comparison between circular trajectory, best view point strategy and locally optimal strategy.

6 Experimentation

In order to validate the previous study in real situation, we need to extract the mask of the object we explore. In order to deal with general scenes, we want to impose no constraint on the object aspect (color, texture, grey level, ...). With this aim in view, we make the only assumption (not very restrictive in most situations) that there is a depth discontinuity at the frontier of the objects. Then for every translational motion of the camera, the projected motion of each object is distinguishable from the other. A motion segmentation algorithm will give the mask of the objects. For real time constraints, this algorithm must be fast and robust. We chose the parametric motion estimation algorithm imagined by Odobez and

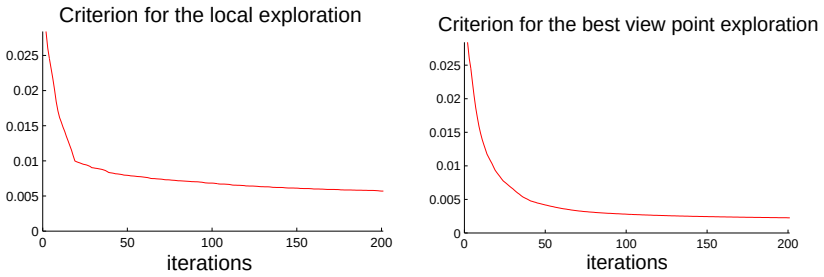


Figure 12: Criterion for -a- the local strategy and -b- the best view point one.

Bouthemy [18]. It furnishes a map of points whose motion is not consistent with dominant motion. In our situation, it corresponds to the mask of the objects.

We implemented the exploration process on a six degrees of freedom robot. The covariance models P^p and P_{int} were chosen identical to the one in section 4.3. Besides, the uncertainty model on robot joints measure is a normal unbiased additive noise whose standard deviation is outweighing 10 cm for translation and 5 deg for rotations.

The Figures 13, 14, 15 and 16 deal with reconstruction of a single object (the mushroom) using the best view point exploration. Figure 13 is the sequence of acquired images where we see that the algorithm maintains the object at the center of the image. The successive segmentation masks are given in Figure 14. Since the camera is not moving for the initialization, the two first segmentation of the object are coarsely approximated by hand. The ellipses corresponding to the measure E^i are superimposed in the images. Figure 15 is the sequence of the estimated including ellipsoid. To show how it matches the real object, these ellipsoids are projected in the final image in Figure 16.

The speed of the algorithm (about 150ms per loop including motion segmentation) allows us to estimate the location and volume of several objects in real time. Figures 17 and 18 show an example with two different objects: a mushroom and a stick. At the initialization, two images of the scene are acquired (see Figures 17-a and 17-b). As previously, the segmentation of the objects are coarsely approximated by hand for the initialization. The associated estimated ellipsoids including the two shapes is given on Figure 17-c. Figure 17-d shows the projection of these first estimation in the final image. It convinces us of the need to refine this estimation. In a second step, the camera is autonomously exploring the objects. We fixed arbitrarily the exploration period to 20 seconds and the strategy is based on the exploration of one of the two objects (the mushroom). Both the locally optimal and the best view point exploration have been tested. The locally optimal trajectory (see Figure 18-a) does not encounter a local minimum thanks to noise inherent to experimentation. The best view point trajectory (see Figure 18-b) is quite similar to the simulated one even if the experimental time is much shorter because of robot joint limits. The final estimated including ellipsoid (see Figure 18-c) has been projected in the final image (see Figure 18-d)

to show the efficiency of the algorithm. The objective to estimate a coarse including volume for the objects is reached.

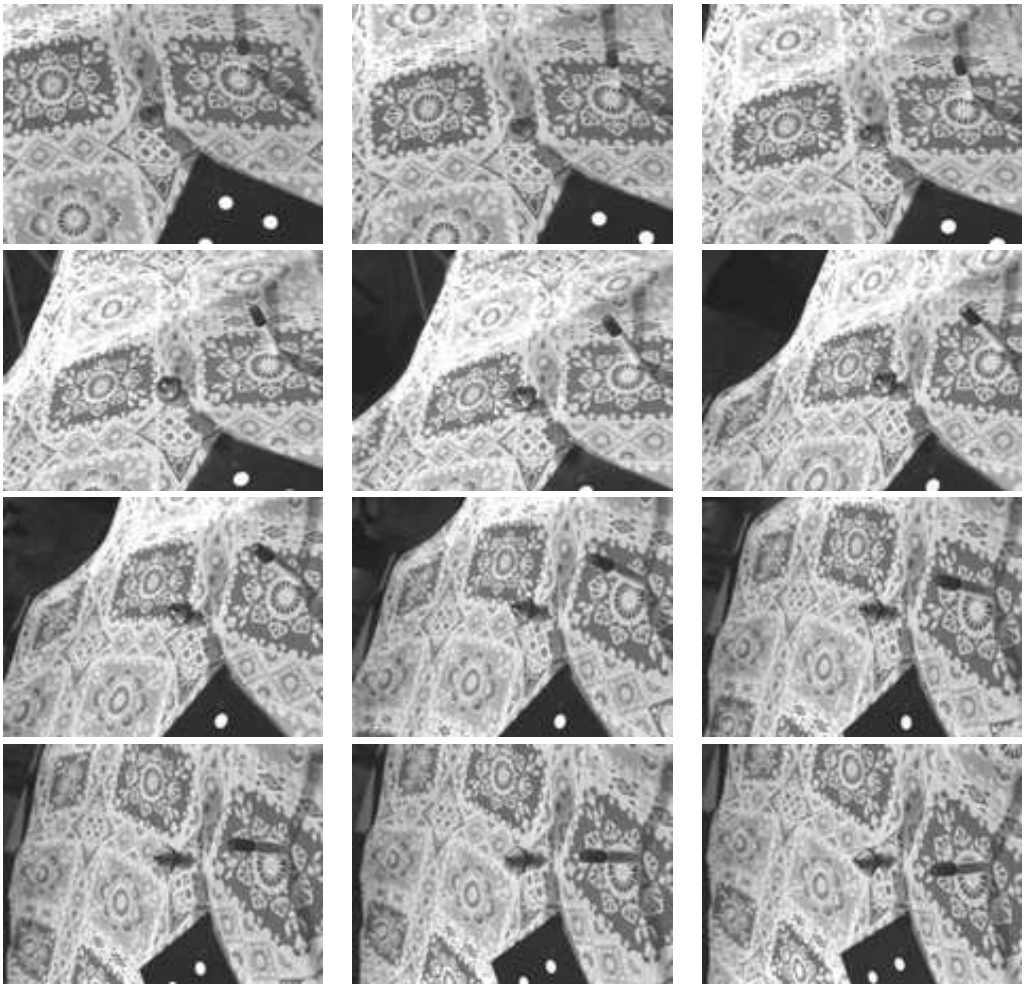


Figure 13: Acquired sequence

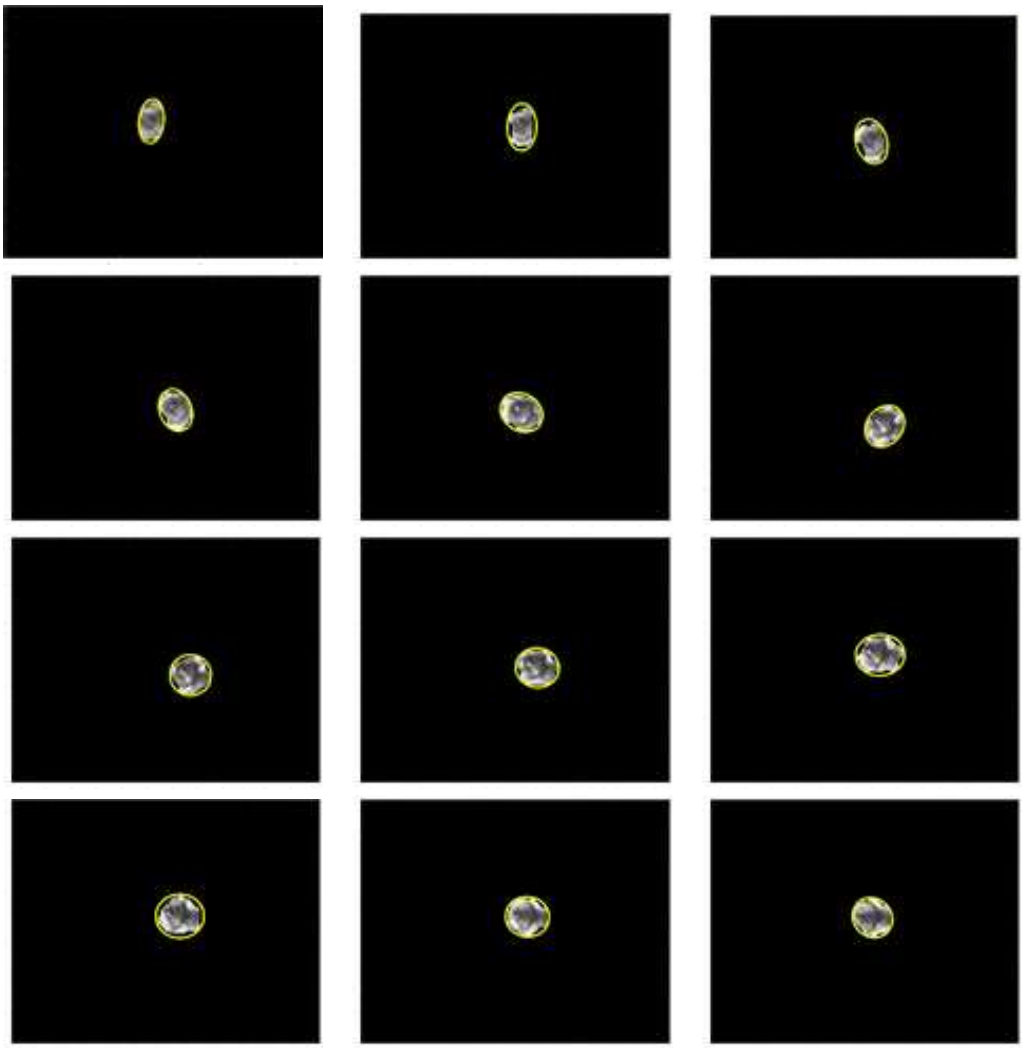


Figure 14: Extracted mask of the object

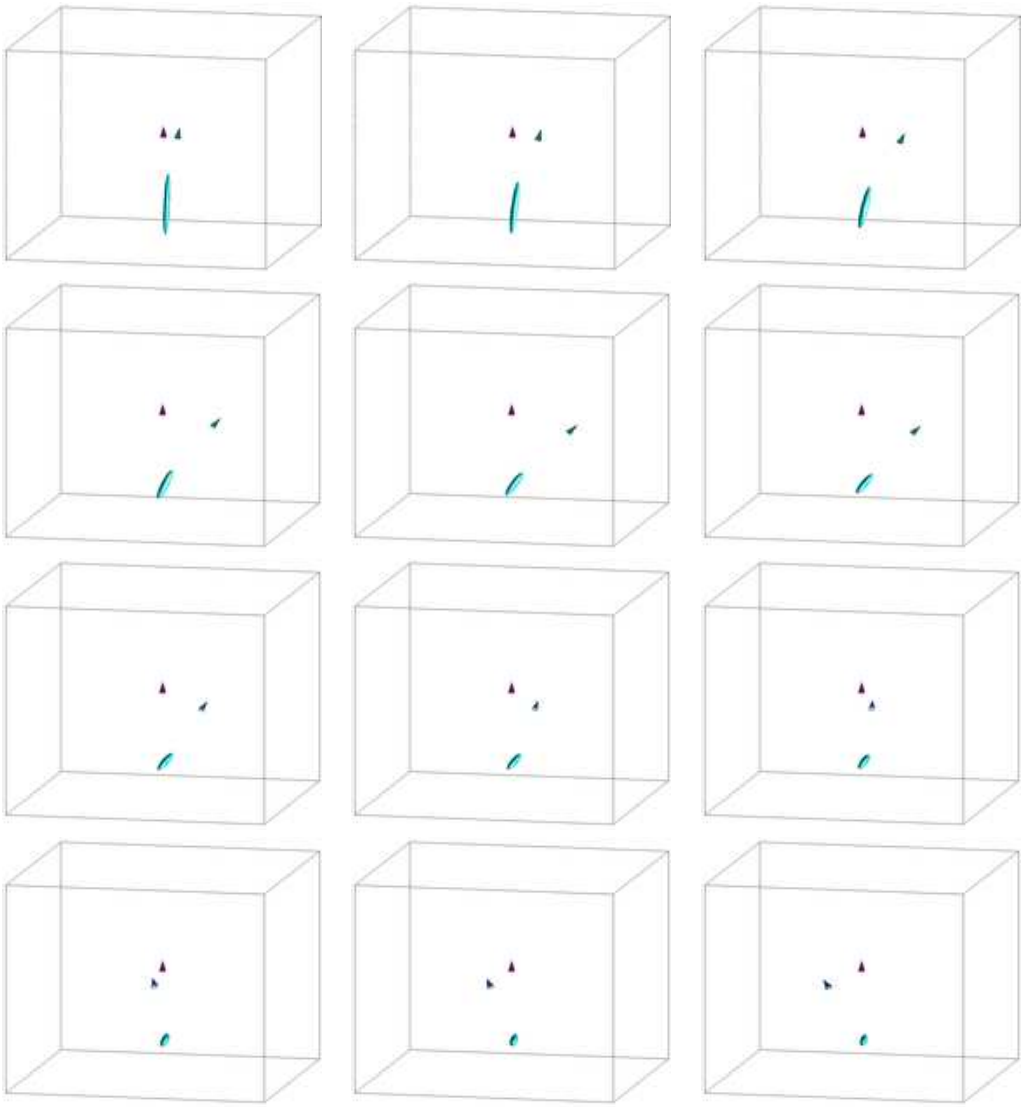


Figure 15: 3D reconstruction of shapes

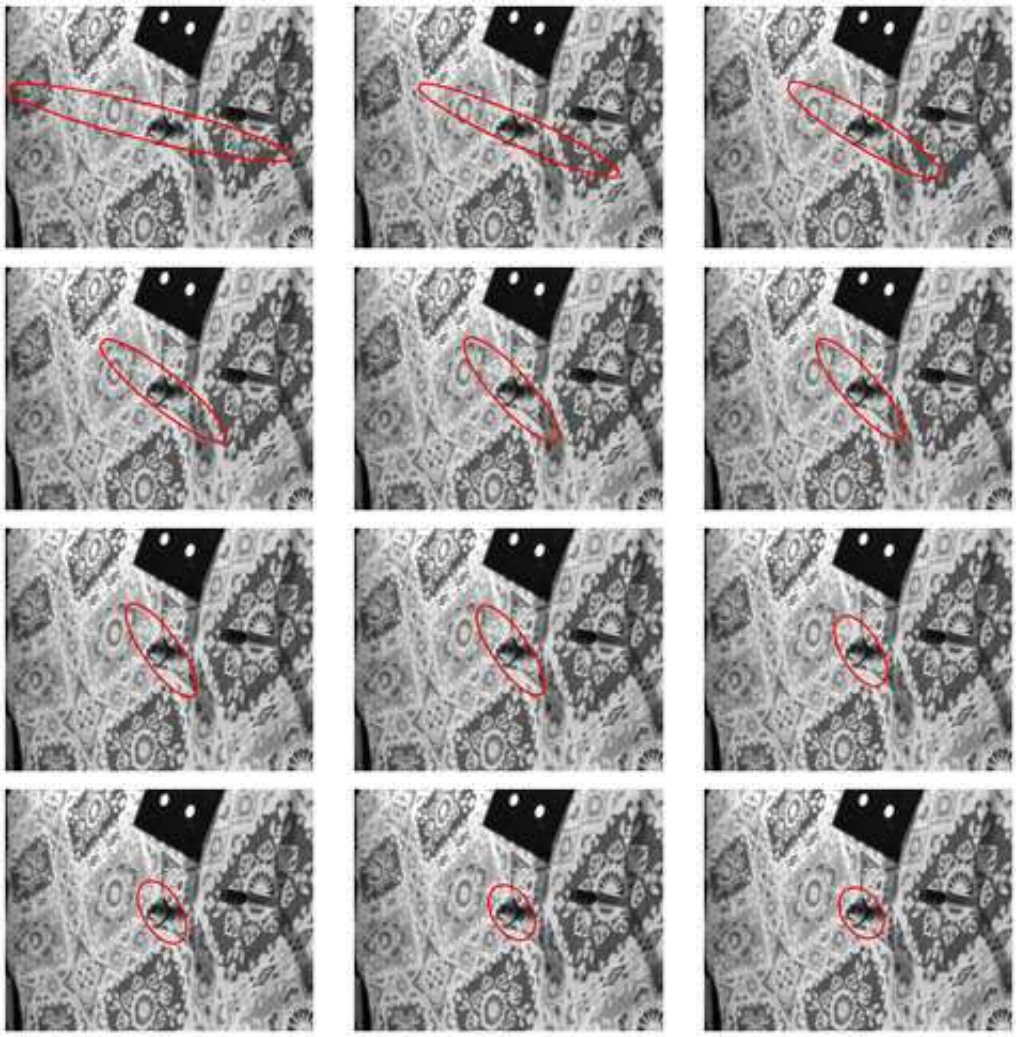


Figure 16: Projected reconstruction

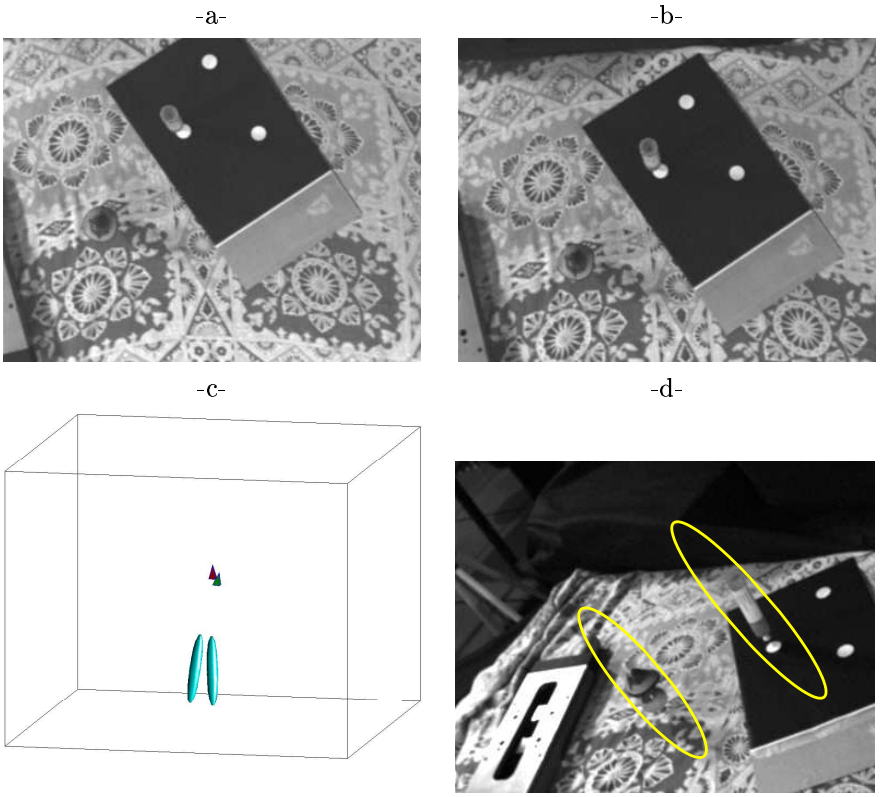


Figure 17: Two objects initialization: -a- and -b- the two first images initializing the exploration process -c- associated estimation of the two objects -d- projection of the estimation in the final image.

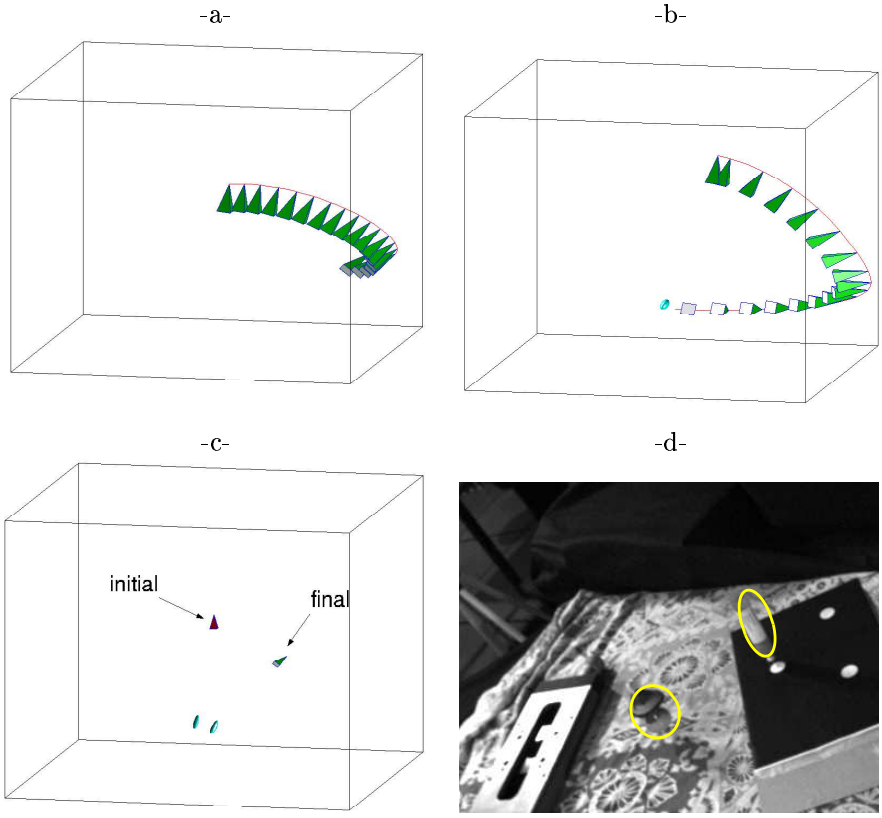


Figure 18: Two objects reconstruction: -a- locally optimal trajectory -b- best view point trajectory -c- final reconstruction -d- projection of the final reconstruction in the final image.

7 Conclusion

We have defined a model representing each detected object of a scene as an approximated probabilistic law allowing us to calculate for every point of a scene its probability to belong to an object. This model is computationally cheap because it only requires a 3D vector and two 3D symmetric matrices to represent the center, the uncertainty and the volume. Several propagating rules have been infered from stochastic geometry resulting in an estimation scheme which is fast and robust. Based on this estimation process, we defined and compared two exploration processes which proved to be optimal in one sense.

Our approach stems for the class of coarse model estimation techniques. It differs from previous work in several points. First the model we use is a mixture of stochastic and set membership models. Its description in a unified probabilistic framework makes it robust to different sources of uncertainty and furnishes a general frame to coarse estimation and fusion. It allowed us to develop very general rules for model propagation. As a consequence, the method can apply to any kind of sensor. Second we derived an analytical solution to optimal exploration so that the calculation of an exploration trajectory which reduces the current uncertainty on the model is solvable on-line. Even if it is obviously not based on accurate reconstruction, the answer we give to which camera motion will improve the coarse model is well-posed. At last, we claim that a main interest of our work is its applicability to real-time processes thanks to the fastness of the algorithms due to adequacy of the model to the assigned task.

As we previously said, the problem of strongly concave objects is worth considering and constitutes an interesting outlook. Judging from our study, reconstruction of such objects is really feasible at the only condition that we are able to partition the object in coherent parts and to match them along the sequence.

At last, the model we defined and the associated tools we developed constitute a good basis to build higher level tasks. We focused on the exploration of objects appearing entirely in the field of view of the camera. Our future work will be dedicated to the research of all the objects of a scene.

Acknowledgments. This study was partly supported by INRIA LARA project and by Brittany County Council.

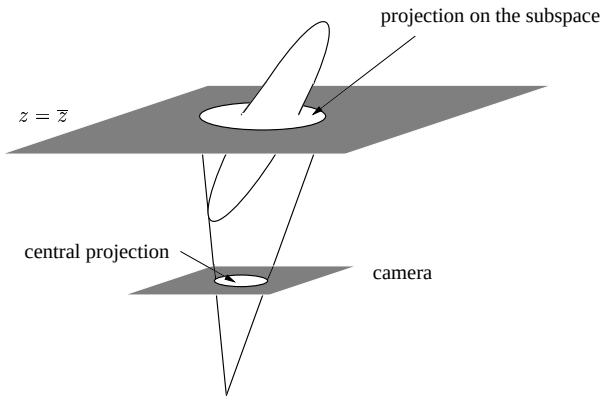


Figure 19: Two steps projection

A Perspective projection of a set distribution

This appendix is dedicated to the proof of Rule 5. Perspective projection is obviously not a diffeomorphism. But, on Figure 19, we see that it can be approximated by the composition of a projection on the subspace $z = \overline{z}$ and a central projection on the camera plane. This is just an approximation since it does not coincide with the exact projection of the limb surface. But this is a quite good approximation whose accuracy depends on the relative depth of the ellipsoid compared with its distance to the camera. As expressed in Rule 5, $c^c = (x^c, y^c, z^c)^T$ is a random vector following a set distribution

$$\mathcal{E}(\begin{pmatrix} \overline{x^c} \\ \overline{y^c} \\ \overline{z^c} \end{pmatrix}, \begin{pmatrix} \Sigma^c_{11} & \Sigma^c_{12} \\ {\Sigma^c_{12}}^T & \Sigma^c_{22} \end{pmatrix}, \begin{pmatrix} E^c_{11} & E^c_{12} \\ {E^c_{12}}^T & E^c_{22} \end{pmatrix})$$

Thanks to Rule 3, the projection c' on the subspace $z^c = \overline{z^c}$ is a random vector following a set distribution $\mathcal{E}'(\overline{c'}, \Sigma', E')$ where

$$\begin{cases} \overline{c'} &= (\overline{x^c}, \overline{y^c}) \\ \Sigma' &= \Sigma^c_{11} - \Sigma^c_{12}{\Sigma^c_{22}}^{-1}{\Sigma^c_{12}}^T \\ E' &= E^c_{11} - E^c_{12}{E^c_{22}}^{-1}{E^c_{12}}^T \end{cases}$$

In a second step, $c^i = T(c') = (x'/\overline{z^c}, y'/\overline{z^c})$ which is a simple diffeomorphism. Applying Rule 1, we show that c^i follows a set distribution

$$\begin{cases} \overline{c^i} &= (x'/\overline{z^c}, y'/\overline{z^c})^T = (\overline{x^c}/\overline{z^c}, \overline{y^c}/\overline{z^c})^T \\ \Sigma^i &= J^T \Sigma' J \\ E^i &= J^T E' J \end{cases} \quad \text{where } J = \begin{pmatrix} \overline{z_c} & 0 \\ 0 & \overline{z_c} \end{pmatrix}$$

replacing Σ' and E' by their expression ends the proof.

B Binocular configuration

Let us remind that the system relating c to (c^{ic}, c^i) is

$$\left\{ \begin{array}{lcl} (c-t) \wedge Rl^{ic} & = & 0 \\ c \wedge l^i & = & 0 \end{array} \right. \tag{6}$$

where $l^i = (c^i, 1)$ and $l^{ic} = (c^{ic}, 1)$. (6) is a system of six equations. Four of them are independent when the cameras are not aligned with the object. As a consequence we can find a solution. The system can be writen as follows:

$$\begin{pmatrix} A \\ B \end{pmatrix} c = \begin{pmatrix} A \\ 0 \end{pmatrix} t$$

where[1] $A = S(Rl^{ic})$ and $B = S(l^i)$. The least square resolution of this system gives the following optimal solution:

$$c = (A^T A + B^T B)^{-1} A^T A t$$

This solution can be numerically computed. But we also need to derive the jacobian $J = \frac{\partial c}{\partial (c^{ic}, c^i)}$. Since the formal expression of $(A^T A + B^T B)^{-1} A^T A t$ is rather complicated, we develop an efficient way to compute J numerically.

Let us denote $Y = (A^T A + B^T B)$, $z = A^T A t$ and $x = (c^{ic}, c^i)$. We have to calculate:

$$\begin{array}{rcl} \frac{\partial c}{\partial x} & = & \frac{\partial Y^{-1} z}{\partial x} \\ & = & Y^{-1} \frac{\partial z}{\partial x} + \left[\frac{\partial Y^{-1}}{\partial x_1} z | \frac{\partial Y^{-1}}{\partial x_2} z | \frac{\partial Y^{-1}}{\partial x_3} z | \frac{\partial Y^{-1}}{\partial x_4} z \right] \end{array}$$

where $(x_i)_{i \in \{1,2,3,4\}}$ are the components of x. In this expression, Y^{-1} and $\frac{\partial z}{\partial x}$ can be computed numerically. More precisely, we have:

$$\frac{\partial z}{\partial x} = \left[\begin{array}{cccc} 2(r_{ic}R_{3,1} + q_{ic}R_{2,1})tx & 2(r_{ic}R_{3,2} + q_{ic}R_{2,2})tx & & \\ -(p_{ic}R_{2,1} + q_{ic}R_{1,1})ty & -(p_{ic}R_{2,2} + q_{ic}R_{1,2})ty & 0 & 0 \\ -(p_{ic}R_{3,1} + r_{ic}R_{1,1})tz & -(p_{ic}R_{3,2} + r_{ic}R_{1,2})tz & & \\ 2(r_{ic}R_{3,1} + p_{ic}R_{1,1})ty & 2(r_{ic}R_{3,2} + p_{ic}R_{1,2})ty & & \\ -(p_{ic}R_{2,1} + q_{ic}R_{1,1})tx & -(p_{ic}R_{2,2} + q_{ic}R_{1,2})tx & 0 & 0 \\ -(q_{ic}R_{3,1} + r_{ic}R_{2,1})tz & -(q_{ic}R_{3,2} + r_{ic}R_{2,2})tz & & \\ 2(q_{ic}R_{2,1} + p_{ic}R_{1,1})tz & 2(q_{ic}R_{2,2} + p_{ic}R_{1,2})tz & & \\ -(p_{ic}R_{3,1} + r_{ic}R_{1,1})tx & -(p_{ic}R_{3,2} + r_{ic}R_{1,2})tx & 0 & 0 \\ -(q_{ic}R_{3,1} + r_{ic}R_{2,1})ty & -(q_{ic}R_{3,2} + r_{ic}R_{2,2})ty & & \end{array} \right]$$

[1] $S(a)$ is the skew symmetric matrix associated with vector a

where

$$\begin{cases} r_{ic} &= R_{3,1}x_{ic} + R_{3,2}y_{ic} + R_{3,3} \\ q_{ic} &= R_{2,1}x_{ic} + R_{2,2}y_{ic} + R_{2,3} \\ p_{ic} &= R_{1,1}x_{ic} + R_{1,2}y_{ic} + R_{1,3} \end{cases}$$

The matrices $\frac{\partial Y^{-1}}{\partial x_i}$ are, after a matrix reorganization, composed by the columns of $\frac{\partial vect(Y^{-1})}{\partial x}$. Where $vect(A)$ stacks the columns of A in a column vector. It can be computed using the following relation [14]:

$$\frac{\partial vect(Y^{-1})}{\partial x} = -(Y^{-T} \otimes Y^{-1}) \frac{\partial vect(Y)}{\partial x}$$

where $\otimes$ symbolizes the Kronecker product and

$$\frac{\partial vect(Y)}{\partial x} = \begin{bmatrix} 2(r_{ic}R_{3,1} + q_{ic}R_{2,1}) & 2(r_{ic}R_{3,2} + q_{ic}R_{2,2}) & 0 & 2y_i \\ -(p_{ic}R_{2,1} + q_{ic}R_{1,1}) & -(p_{ic}R_{2,2} + q_{ic}R_{1,2}) & -y_i & -x_i \\ -(p_{ic}R_{3,1} + r_{ic}R_{1,1}) & -(p_{ic}R_{3,2} + r_{ic}R_{1,2}) & -1 & 0 \\ -(p_{ic}R_{2,1} + q_{ic}R_{1,1}) & -(p_{ic}R_{2,2} + q_{ic}R_{1,2}) & -y_i & -x_i \\ 2(r_{ic}R_{3,1} + p_{ic}R_{1,1}) & 2(r_{ic}R_{3,2} + p_{ic}R_{1,2}) & 2x_i & 0 \\ -(q_{ic}R_{3,1} + r_{ic}R_{2,1}) & -(q_{ic}R_{3,2} + r_{ic}R_{2,2}) & 0 & -1 \\ -(p_{ic}R_{3,1} + r_{ic}R_{1,1}) & -(p_{ic}R_{3,2} + r_{ic}R_{1,2}) & -1 & 0 \\ -(q_{ic}R_{3,1} + r_{ic}R_{2,1}) & -(q_{ic}R_{3,2} + r_{ic}R_{2,2}) & 0 & -1 \\ 2(q_{ic}R_{2,1} + p_{ic}R_{1,1}) & 2(q_{ic}R_{2,2} + p_{ic}R_{1,2}) & 2x_i & 2y_i \end{bmatrix}$$

C Outer approximation of an ellipsoid intersection

We remind that the problem is to find the smallest ellipsoid in the family generated by

$$(1-\lambda)E_k + \lambda E^o_{k+1} \qquad \lambda \in \Re$$

In other words we are looking for

$$\widehat{\lambda} = argmax \quad det(A + \lambda B)$$

where $A = E_k$ and $B = E^o_{k+1} - E_k$ are supposed to be regular[2] . $det(A + \lambda B)$ is clearly a third order polynomial in λ:

$$P(\lambda) = det(A + \lambda B) = a\lambda^3 + b\lambda^2 + c\lambda + d$$

We just need to identify a, b, c and d:

- $\lim_{\lambda \to 0} P(\lambda) = d = det(A)$
- $\lim_{\lambda \to +\infty} \frac{1}{\lambda^3} P(\lambda) = a = \lim_{\lambda \to +\infty} det(\frac{1}{\lambda}A + B) = det(B)$

[2] A and B will be regular in practice

- If we write
$$P(\lambda)det(-B^{-1}) = det(-AB^{-1} - \lambda\mathbb{I})$$
we recognize the characteristic polynomial of $-AB^{-1}$. The coefficient of λ^2 is thus $tr(-AB^{-1})$. Thus
$$b.det(-B^{-1}) = tr(-AB^{-1}) \Rightarrow b = det(B)tr(AB^{-1})$$

- With the same idea,
$$P(\lambda)\frac{det(-A^{-1})}{\lambda^3} = det(-BA^{-1} - \frac{1}{\lambda}\mathbb{I})$$
is the characteristic polynomial of $-BA^{-1}$. The coefficient of $\frac{1}{\lambda^2}$ is thus $tr(-BA^{-1})$. Thus
$$c.det(-A^{-1}) = tr(-BA^{-1}) \Rightarrow c = det(A)tr(BA^{-1})$$

Finally, we can write

$$P(\lambda) = det(B)\lambda^3 + det(B)tr(AB^{-1})\lambda^2 + det(A)tr(BA^{-1})\lambda + det(A)$$

We remind that we are looking for the maximum of $P(\lambda)$. It is given by:

$$\left\{ \begin{array}{lcl} P'(\lambda) & = & 3a\lambda^2 + 2b\lambda + c = 0 \\ P''(\lambda) & = & 6a\lambda + 2b \leq 0 \end{array} \right.$$

these equations can be interpreted as following:

- if $a > 0$, $\lambda < \frac{-b}{3a}$ is the smallest solution of $P'(\lambda) = 0$
- if $a < 0$, $\lambda > \frac{-b}{3a}$ is the biggest solution of $P'(\lambda) = 0$

In other words and to conclude:

$$\widehat{\lambda} = \left\{ \begin{array}{ll} \frac{-det(B)tr(AB^{-1}) - \sqrt{det(B)^2tr(AB^{-1})^2 - det(B)det(A)tr(BA^{-1})}}{det(B)} & \text{if } det(B) > 0 \\ \frac{-det(B)tr(AB^{-1}) + \sqrt{det(B)^2tr(AB^{-1})^2 - det(B)det(A)tr(BA^{-1})}}{det(B)} & \text{if } det(B) < 0 \end{array} \right.$$

D Locally optimal exploration

In this appendix, we express motion which locally optimizes the increase of Σ_k. The criterion is the trace of $\widehat{\Sigma_{k+1}}$. At time $k+1$, the camera will have rotated with an angle $\alpha \geq 0$ around the unit vector $u = (u_x, u_y, 0)$. At a first order approximation ($\alpha << 1$), the associated matrix is:

$$R \simeq I + \alpha \begin{pmatrix} 0 & 0 & u_y \\ 0 & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix}$$

If we note v_1, v_2 and v_3 the eigenvectors of Σ_k and v_x, v_y and v_z the expression of the camera frame in the (v_1, v_2, v_3) base, we remind that

$$(v_1 \quad v_2 \quad v_3) = \begin{pmatrix} v_{1x} & v_{2x} & v_{3x} \\ v_{1y} & v_{2y} & v_{3y} \\ v_{1z} & v_{2z} & v_{3z} \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

If we denote V this matrix of eigen vectors then we can decompose

$$R^T \Sigma_k R = R^T V \Delta V^T R$$

where Δ is a diagonal matrix. Applying decomposition of Section 5.1, we find

$$A = \begin{bmatrix} (v_x - \alpha u_y v_z)\Delta(v_x^T - \alpha u_y v_z^T) & (v_x - \alpha u_y v_z)\Delta(v_y^T + \alpha u_x v_z^T) \\ (v_x - \alpha u_y v_z)\Delta(v_y^T + \alpha u_x v_z^T) & (v_y + \alpha u_x v_z)\Delta(v_y^T + \alpha u_x v_z^T) \end{bmatrix}$$

$$B = \begin{bmatrix} (v_x - \alpha u_y v_z)\Delta(\alpha u_y v_x^T - \alpha u_x v_y^T + v_z^T) \\ (v_y + \alpha u_x v_z)\Delta(\alpha u_y v_x^T - \alpha u_x v_y^T + v_z^T) \end{bmatrix}$$

$$c = (\alpha u_y v_x - \alpha u_x v_y + v_z)\Delta(\alpha u_y v_x^T - \alpha u_x v_y^T + v_z^T)$$

so that, according to equation (5), the problem is to find

$$(u_x, u_y) = argmax \quad tr(2A - \frac{BB^T}{c} + c)$$

At a first order approximation ($\alpha << 1$) and denoting $\beta = u_y v_x - u_x v_y$, we can write:

$$c^{-1} \approx \frac{1}{v_z \Delta v_z^T} - 2\alpha \frac{v_z \Delta \beta^T}{(v_z \Delta v_z^T)^2}$$

$$c \approx v_z \Delta v_z^T + 2\alpha v_z \Delta \beta^T$$

$$\frac{BB^T}{c} \approx \begin{bmatrix} \frac{(v_x \Delta v_z^T)^2}{v_z \Delta v_z^T} + 2\alpha \frac{v_x \Delta v_z^T}{v_z \Delta v_z^T}[v_x \Delta \beta^T - u_y v_z \Delta v_z^T] - 2\alpha \frac{(v_x \Delta v_z^T)^2}{(v_z \Delta v_z^T)^2} v_z \Delta \beta^T & K_1 \\ K_1^T & \frac{(v_y \Delta v_z^T)^2}{v_z \Delta v_z^T} + 2\alpha \frac{v_y \Delta v_z^T}{v_z \Delta v_z^T}[v_y \Delta \beta^T + u_x v_z \Delta v_z^T] - 2\alpha \frac{(v_y \Delta v_z^T)^2}{(v_z \Delta v_z^T)^2} v_z \Delta \beta^T \end{bmatrix}$$

$$A \approx \begin{bmatrix} v_x \Delta v_x^T - 2\alpha u_y v_x \Delta v_z^T & K_2 \\ K_2^T & v_y \Delta v_y^T + 2\alpha u_x v_y \Delta v_z^T \end{bmatrix}$$

In previous equations, K_1 and K_2 are terms whose analytical expression is useless in the determination of $trace(\widehat{\Sigma_{k+1}})$. Then, after simplification, we deduce that

$$trace(\widehat{\Sigma_{k+1}}) \approx K_5 + 2\alpha[\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T} v_z - v_x) + \frac{v_y \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_y \Delta v_z^T}{v_z \Delta v_z^T} v_z - v_y)]\Delta \beta^T$$

where K_5 is a constant term. Previous expression is of the following shape:

$$trace(\widehat{\Sigma_{k+1}}) \approx constant + 2\alpha\gamma\Delta\beta^T$$

If v_z is an eigen value of Δ then $\gamma = 0$ since $\Delta v_z \propto v_z$ which implies $v_y \Delta v_z = v_y \Delta v_z = 0$ since V is orthogonal. In that case $trace(\widehat{\Sigma_{k+1}})$ does not depend on u which means that any direction can be chosen. When v_z is not an eigen value of Δ, we have to maximize the outer product of $(u_x v_y - u_y v_x)$ and $\Delta\gamma^T$. Let us remark that $\Delta\gamma^T$ belongs to the plane (v_x, v_y):

$$v_z \Delta\gamma^T = \frac{v_x \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T} v_z \Delta v_z^T - v_z \Delta v_x^T) + \frac{v_y \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_y \Delta v_z^T}{v_z \Delta v_z^T} v_z \Delta v_z^T - v_z \Delta v_y^T) = 0$$

It means that $(u_x v_y - u_y v_x)$ must be collinear to $\Delta\gamma^T$. That is:

$$\left\{ \begin{array}{rcrcl} u_x & = & v_y \Delta\gamma^T & = & \frac{v_x \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T} v_y \Delta v_z^T - v_y \Delta v_x^T) + \frac{v_y \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_y \Delta v_z^T}{v_z \Delta v_z^T} v_y \Delta v_z^T - v_y \Delta v_y^T) \\ u_y & = & -v_x \Delta\gamma^T & = & -\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_x \Delta v_z^T}{v_z \Delta v_z^T} v_x \Delta v_z^T - v_x \Delta v_x^T) - \frac{v_y \Delta v_z^T}{v_z \Delta v_z^T}(\frac{v_y \Delta v_z^T}{v_z \Delta v_z^T} v_x \Delta v_z^T - v_x \Delta v_y^T) \end{array} \right.$$

Let us note that for this particular control law, the points where v_z is an eigenvector of Δ result in a null command u. Such points are thus local minima. Besides v_z is the optical axis of the camera in the (v_1, v_2, v_3) base. When v_z is an eigenvector of Δ, it means that the optical axis is parallel to an axis of the ellipsoid Σ_k. Since the ellipsoid is centered in the image, it also implies that the camera is located on this axis. Local minima are thus points located on the direction pointed by the axes of Σ_k.

References

[1] T. Arbel and F. Ferrie. Viewpoint selection by navigation through entropy maps. In *Proceedings of the 7th IEEE Int. Conf. on Computer Vision (ICCV-99)*, volume I, pages 248–254, Los Alamitos, CA, September 1999. IEEE.

[2] Nicholas Ayache. *Artificial Vision for Mobile Robots.* The MIT Press, Cambridge, MA, 1991.

[3] J. D. Boissonat. Representing 2D and 3D shapes with the delaunay triangulation. In *Seventh International Conference on Pattern Recognition (Montreal, Canada, July 30-August 2, 1984)*, IEEE Publ. 84CH2046-1, pages 745–748. IEEE, IEEE, 1984.

[4] C. Connolly. The determination of next best views. In *Proc. IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 432–435, St Louis, Missouri, March 1995.

[5] H. F. Durrant-Whyte. *Integration, Coordination, and Control of Multi-Sensor Robot Systems.* Kluwer Academic Publishers, Boston, 1987.

[6] H. F. Durrant-Whyte. Uncertain geometry in robotics. *IEEE Journal of Robotics and Automation*, 4(1):23–31, 1988.

[7] O. D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and line matches. In *First International Conference on Computer Vision, (London, England, June 8–11, 1987)*, pages 25–34, Washington, DC., 1987. IEEE Computer Society Press.

[8] F. P. Ferrie and M. D. Levine. Deriving coarse 3D models of objects. In *CVPR'88 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Ann Arbor, MI*, pages 345–353, Washington, DC., June 1988. Computer Society Press.

[9] G. Flandin, F. Chaumette, and E. Marchand. Eye-in-hand / eye-to-hand cooperation for visual servoing. In *IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, Avril 2000.

[10] B. Garcia and P. Brunet. 3d reconstruction with projective octrees and epipolar geometry. In *IEEE Int. Conf. on Computer Vision*, pages 1067–1072, January 1998.

[11] U. D. Hanebeck, J. Horn, and G. Schmidt. On combining set theoretic and bayesian estimation. In *IEEE Int. Conf. on Robotics and Automation*, 1996.

[12] Kiriakos N. Kutulakos, Charles R. Dyer, and Vladimir J. Lumelsky. Provable strategies for vision-guided exploration in three dimensions. In *Proc. 1994 IEEE Int. Conf. Robotics and Automation*, pages 1365–1372, Los Alamitos, CA, 1994.

[13] Simon Lacroix and R. Chatila. *Motion and perception strategies for outdoor mobile robot navigation in unknown environments.* Lecture Notes in Control and Information Sciences, 223. Springer-Verlag, New York, 1997.

[14] Helmut Lütkepohl. *Handbook of Matrices.* John Wiley & Sons, Ltd., 1996.

[15] D. G. Maksarov and J. P. Norton. State bounding with ellipsoidal set description of the uncertainty. *Int. Journal on Control*, 65(5):847–866, 1996.

[16] Éric Marchand and François Chaumette. Active vision for complete scene reconstruction and exploration. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(1):65–72, January 1999.

[17] D. Marr and K. Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proc. Royal Soc. London Bulletin*, B200:269–294, 1977.

[18] J.M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, 1995.

[19] Michael Potmesil. Generating octree models of 3D objects from their silhouettes in a sequence of images. *Computer Vision, Graphics, and Image Processing*, 40(1):1–29, October 1987.

[20] F. C. Schweppe. Recursive state estimation: unknown but bounded errors and system inputs. *IEEE Trans. on Automatic Control*, AC-13:22–28, 1968.

[21] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer. A survey of methods for volumetric scene reconstruction from photographs. Technical report, Center for Signal and Image Processing, Georgia Institute of Technology, 2001.

[22] R. Szeliski and P. Golland. Rapid octree construction from image sequences. *Computer Vision, Graphics and Image Processing: Image Understanding*, 58(1):23–32, July 1993.

[23] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai. A survey of sensor planning in computer vision. *IEEE Trans. on Robotics and Automation*, 11(1):86–104, February 1995.

[24] E. Welzl. Smallest enclosing disks (balls and ellipsoids). *Lecture Notes in Computer Science*, 555:359–370, 1991.

[25] P. Whaite and F. P. Ferrie. Autonomous exploration: Driven by uncertainty. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 339–346, Los Alamitos, CA, USA, June 1994. IEEE Computer Society Press.

[26] H. S. Witsenhausen. Sets of possible states of linear systems given perturbed observations. *IEEE Transactions on Automatic Control*, AC-13:556–558, 1968.

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399